An Enhanced Seasonal-Hybrid ESD Technique for Robust Anomaly Detection on Time Series

Rafael G. Vieira[1], **Marcos A. Leone Filho**[2] **and Robinson Semolini**[3]

[1]School of Electrical and Computer Engineering, University of Campinas
400 Albert Einstein – Campinas, SP – Brazil

[2]Venidera Research & Development
700 Marechal Rondom – Campinas, SP – Brazil

[3]Elektro S.A Electricity and Services
321 Ary Antenor de Souza – Campinas, SP – Brazil

giordano@dca.fee.unicamp.br, marcos@venidera.com
robinson.semolini@elektro.com.br

Abstract. *Nowadays, time series data underlies countless research activities. Despite the wide range of techniques to capture and process all this information, issues such as analyzing large amounts of data and detecting unusual behaviors on them still pose a great challenge. In this context, this paper suggests SH-ESD+, a statistical technique that combines the Extreme Studentized Deviate (ESD) test and a decomposition procedure based on Loess to detect anomalies on time series data. The proposed technique employs robust metrics to identify anomalies in a more proper and accurate manner, even in the presence of trend and seasonal spikes. Simulation studies are carried out to evaluate the effectiveness of the SH-ESD+ using the published Numenta Anomaly Benchmark (NAB) collection. Computational results show that the SH-ESD+ performs consistently when compared against state-of-the-art and classic detection techniques.*

1. Introduction

Over the past years, the amount of information available has grown at an ever-increasing pace around the world. Regardless the application domain, companies are collecting data in order to perform better analysis, to make better decisions, and consequently to become more competitive [Witten et al. 2011]. Essentially, such analysis and decision making depend on how data is collected and measured. In this paper, it is considered that available data comes in the form of time series [Box et al. 2015], which represents the most straightforward way for modeling any system which involves temporal measurements.

One notable application area of time series data is anomaly detection. The process of identifying unusual behaviors on data has been a major topic of research during the past decade and finds its significance across many fields, such as business, the stock market, weather and power consumption [Chou and Telaga 2014, Akouemo and Povinelli 2016]. Anomalies may occur in a time series for one of two different reasons: (i) as a function of the inherent variability of the data; and (ii) given to errors on data. Anomalies inserted in the first category are very important, since they may contain valuable and often critical information, which ultimately can support decision-making. On the other hand, anomalies

inserted in the second category are often caused by human mistakes, such as errors in collecting, recording or entering data, and shall not be considered in this paper.

Developing proper anomaly detection techniques became critical for computer networks, given the huge number of datasets and transactions to be analyzed and the unavailability of skilled analysts to discover and understand the appropriate statistical models for further assessment of such data [Agrawal et al. 2017]. There is, therefore, a central question that guides the current research in this paper: "*What is the most appropriate technique to detect anomalies on time series data efficiently and, mostly important, without any sort of human interaction*?". In general, the anomalous behaviors are to be identified in an online manner, or offline on previously recorded data. Online detection puts real-time constraints on the detection system. For example, the technique must process data and outputs a decision in real-time, rather than making some passes through batches of files [Ahmad et al. 2017]. Nonetheless, this paper focuses on offline anomaly detection, i.e., where there are no real-time constraints to be considered.

Recently, authors in [Ahmad and Purdy 2016] have conceived a robust anomaly detection technique, named Seasonal-Hybrid ESD (SH-ESD). The designed technique is composed of a statistical test hypothesis and a time series decomposition method, and are being primarily used to detect both local and global anomalies in a variety of time series. In addition to that, it employs a set of robust metrics, e.g., piecewise approximation and Loess regression [Cleveland 1979, Montgomery and Runger 2013], for improving decomposition accuracy and easing the identification of such anomalies.

This paper suggests a slightly distinct approach, named Enhanced Seasonal Hybrid ESD (SH-ESD+) for detecting anomalies on time series. The SH-ESD+ is intended, but not restricted, to cope with some limitations of SH-ESD and from standard anomaly detection methods. First, it includes robust statistical techniques to minimize the number of false positives and handle effectively with any kind of anomalies on data. It also employs automatic parameter identification procedures to leverage model performance at a low computational cost. The effectiveness of SH-ESD+ is assessed by means of computational experiments using the Numenta Anomaly Benchmark (NAB), which provides a controlled open-source environment for testing anomaly detection algorithms.

The remainder of this paper is organized as follows: Section 2 addresses the central concepts related to anomaly detection on time series and presents some related work. Section 3 introduces the suggested SH-ESD+ technique for detecting anomalies in time series data. Section 4 evaluates the proposed technique using NAB. Finally, Section 5 summarizes the main conclusions of the current research.

2. Related work

The field of anomaly detection was first researched in the context of time series analysis as early as the 19th century. According to [Chandola et al. 2009], an anomaly can be defined as a point in time where the behavior of a given system is unusual and significantly different from previous, normal behavior. There are mainly two types of anomalies that are studied in the literature and are also illustrated in Figure 2:

- **Temporal anomaly** occurs when an individual data point deviates from the considered regular pattern with respect to the rest of data, independent of where it occurs. It is considered the simplest form of an anomaly;

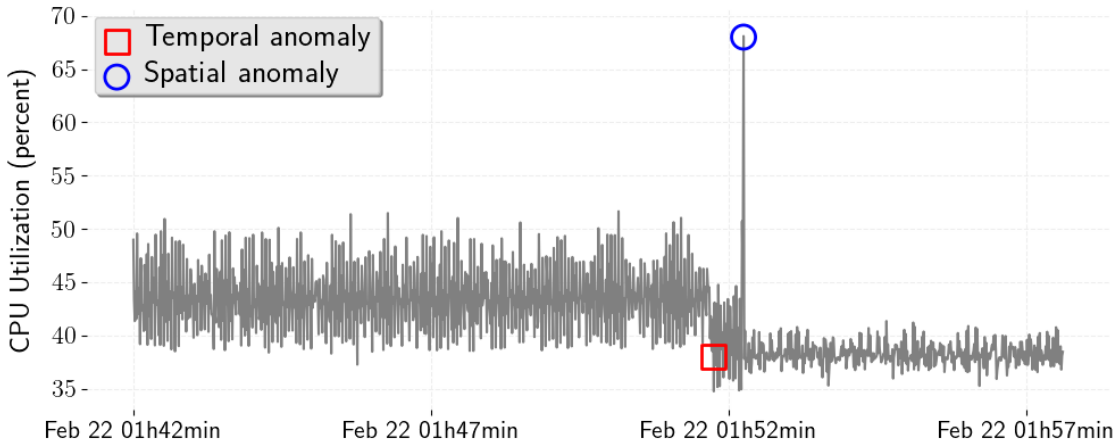


Figure 1. CPU utilization for an Amazon EC2 instance (data from the NAB [Lavin and Ahmad 2015]). The first anomaly represents a temporal change point, while the second anomaly represents a large spatial spike in the data.

- **Spatial/contextual anomaly** appears when a data point or a sequence of data points is considered as an anomaly regarding its local neighborhood (or a specific temporal context), but not otherwise. Temporal anomalies are often subtle and hard to detect. However, they can serve as an early warning for problems with the underlying system [Ahmad et al. 2017].

In recent years, several techniques have been proposed in the literature for detecting anomalies in time series data [Chandola et al. 2009]. Particularly, they may be divided into two large groups, subject to specific learning procedures: unsupervised and supervised ones. The vast majority of anomaly detection techniques are used to be unsupervised, i.e., they are able to model the underlying structure of a given time series without any prior knowledge about such data. Under this category, there are the instance-based techniques [Laxhammar and Falkman 2014], clustering methods [Akoglu et al. 2015], adaptive filtering [Li et al. 2014] and evolving fuzzy systems [Moshtaghi et al. 2015] to name a few. On the other hand, supervised techniques are related to artificial neural networks [Zhou et al. 2016], user-driven systems [Theissler 2017] and most of the statistical methods, such as the Box-Jenkins [Kadri et al. 2016].

The autoregressive integrated moving average (ARIMA) is one of the most popular models to detect temporal anomalies on time series data, given its ability to model trend and seasonality from data [Bianco et al. 2001]. The ARIMA has been also extended to cope with multivariate data and to dynamically determine the period of its seasonal periodicity [Hyndman and Khandakar 2007]. Another widely used approach is known as change point detection [Lu et al. 2004]. The idea is to model a time series into two independent moving windows and detect when there is a significant deviation in some of them. Techniques that use change point detection are often extremely fast to compute and have a low memory overhead. However, the detection performance of those can be sensitive to the size of the windows and thresholds, which may result in many false positives as the data changes, thus requiring frequent updates to the used thresholds.

Over the past decade, the research on anomaly detection has experienced a rapid growth, as many companies have opted to open source components of their infrastructure. Examples of those include: the Skyline project, which provides an open source implementation of a number of statistical techniques for detecting anomalies in streaming data [Stanway 2013]; the EGADS, an open source framework developed by Yahoo for detecting anomalies in large scale time-series data [Laptev et al. 2015]; the S-ESD and SH-ESD statistical learning based techniques released by Twitter, which are used to detect anomalies in the cloud automatically [Ahmad and Purdy 2016]; and the Robust Anomaly Detection (RAD) algorithm of Netflix, which recently was released to the public as a part of the Surus project [Agrawal et al. 2017]. Additional techniques for anomaly detection on time series data include [Burnaev and Ishimtsev 2016, Lavin and Ahmad 2015].

3. The SH-ESD+ technique

This section describes the proposed SH-ESD+ (Enhanced Seasonal Hybrid Extreme Studentized Deviates) technique to automatically detect anomalies on time series data. The core idea behind SH-ESD+ is to use a modified version of STL decomposition (discussed in Section 3.2) to extract the residual component from a previous, transformed input time series, and then to apply an also modified version of the ESD test (addressed in Section 3.3) to detect anomalies in such residual. This process allows SH-ESD+ to detect both global anomalies, that extend the expected seasonal minimum and maximum values, and local anomalies, that would otherwise be masked by such seasonality.

The operation of SH-ESD+ is divided in three steps: data transformation, time series decomposition and residual analysis. Since the proposed technique performs offline anomaly detection only, the input data is composed by a sequence of n univariate data points from a given input time series, represented by $\mathcal{X} = [\mathcal{X}_1, \ldots, \mathcal{X}_n] \in \mathbb{R}^n$. The output, denoted by $\mathcal{A} = [\mathcal{A}_1, \ldots, \mathcal{A}_m] \in \mathbb{R}^m$, is a sequence of $m < n$ anomalies of $\mathcal{X}$.

3.1. Data transformation

Data transformations are common methods that can serve many functions in quantitative analysis of data. While there are many reasons to use transformations, the focus of this paper is on those who improve the normality of data, as parametric statistical tests tend to benefit from normally distributed data. Hence, it was considered using the so-called Box-Cox transformation [Box and Cox 1964], which can be defined by Equation (1):

$$\mathcal{Y}_i = \begin{cases} \dfrac{(\mathcal{X}_i + \lambda_2)^{\lambda_1} - 1}{\lambda_1} & \text{if } \lambda_1 \neq 0, \\ \log(\mathcal{X}_i + \lambda_2) & \text{if } \lambda_1 = 0, \end{cases} \tag{1}$$

where $\mathcal{Y}_i$ is the i-th transformed observation of $\mathcal{X}_i$, $\mathcal{Y} = [\mathcal{Y}_1, \ldots, \mathcal{Y}_n] \in \mathbb{R}^n$ and λ_1, λ_2 are the parameters that define the nature of the transformation. In practice, one can choose λ_2 such that $\mathcal{Y}_i + \lambda_2 > 0$ for any $\mathcal{Y}_i$. The analysis of Box-Cox requires the correct inference on the transformation parameter λ_1. In this paper, this process is carried out by using the Maximum Likelihood method, which is commonly used since it is conceptually easy and the profile likelihood function is simple to compute. The entire procedure for obtaining the estimated value of λ_1 is detailed in [Johansen and Juselius 1990].

3.2. Time series decomposition

Time series data can exhibit a variety of patterns at the same time. One simple method to describe them is called *decomposition* [Box et al. 2015]. Typically, a time series can be decomposed into three components: the trend, the seasonal variation and the residual. The trend is the long-term change in the mean level and often thought of as the underlying growth or decline component in the series. The seasonal part is concerned with the periodic fluctuations in the series over a fixed period. Once the trend and seasonal components have been accounted for, the remaining data is attributed to a set of residuals.

In classical decomposition procedures, the idea is to create separate models for these three components and then combine them, either additively or multiplicatively. However, if anomalies are found in the input data, this procedure can potentially affect normality of the residual component. Therefore, decomposition of $\mathcal{Y}$ is performed in this paper by STL [Cleveland et al. 1990], a robust approach that uses Loess regressions to derive the seasonality. STL consists of two recursive procedures: an inner loop nested inside an outer loop. The inner loop iteratively updates the trend and seasonal components, repeating the process φ times. On the other hand, the outer loop assigns robustness weights to each data point through ϑ passes, which allows for reducing or even eliminating the effects of anomalies on the trend and seasonal components.

The appropriate choice between the additive and multiplicative models is another point to emphasize in order to get a successful time series decomposition. The additive model is usually considered when the magnitude of the seasonal pattern in the data *does not depend* on the magnitude of the data, while the multiplicative model is used otherwise. In order to avoid auto selecting between these two models, one can transform the data until the variation in the series becomes stable over time, and then use an additive model. In this paper, this can be accomplished by taking logarithms of both sides of the model:

$$log(\mathcal{Y}) = log(\mathcal{T}) + log(\mathcal{S}) + log(\mathcal{R}) \tag{2}$$

where $log(\mathcal{Y}_i) = log(\mathcal{T}_i) + log(\mathcal{S}_i) + log(\mathcal{R}_i)$ is the i-th data point of $log(\mathcal{Y})$, $\mathcal{T} = [\mathcal{T}_1, \ldots, \mathcal{T}_n] \in \mathbb{R}^n$, $\mathcal{S} = [\mathcal{S}_1, \ldots, \mathcal{S}_n] \in \mathbb{R}^n$ and $\mathcal{R} = [\mathcal{R}_1, \ldots, \mathcal{R}_n] \in \mathbb{R}^n$ are the trend, seasonal and residual components of $\mathcal{Y}$, respectively. The outline below presents a brief description about the phases involved in STL:

Inner loop – Generate updates for components of trend $\mathcal{T}^{(k+1)}$ and seasonal $\mathcal{S}^{(k+1)}$. Run φ times iteratively for $k = 1, \ldots, \varphi$:

1. *Detrending*. A detrended series, $\mathcal{Y}_{det}^{(k)} = \mathcal{Y} - \mathcal{T}^{(k)}$, is computed. To carry out this procedure on the initial pass through the inner loop, the starting values for the trend component is defined as $\mathcal{T}^{(1)} = [0_1, \ldots, 0_n]$.
2. *Sub-cycle series smoothing*. The detrended series $\mathcal{Y}_{det}^{(k)} \in \mathbb{R}^n$ is broken into v non-overlapping sub-cycle series $\mathcal{C}_1^{(k)}, \ldots, \mathcal{C}_v^{(k)}$ with sizes equals to $\frac{n}{v}$ data points. A sub-cycle series comprises a sequence of values at each position of a seasonal cycle. Then, each sub-cycle series is smoothed by the Loess regression. Finally, the collection of smoothed values for all of the sub-cycle series are recombined to yield a single temporary smoothed seasonal series, defined as $\mathcal{C}^{(k)}$.

3. *Low-pass filter of smoothed seasonal series*. Apply a low-pass filter in $\mathcal{C}^{(k)}$, thus generating the output $\mathcal{L}^{(k)}$. The filter consists of a moving average of length v, followed by another moving average of length v, followed by a moving average of length 3, followed by a Loess regression.
4. *Detrending of smoothed seasonal series*. The seasonal component from $(k+1)$-th step is $\mathcal{S}^{(k+1)} = \mathcal{C}^{(k)} - \mathcal{L}^{(k)}$. The value from $\mathcal{L}^{(k)}$ is subtracted to prevent low-frequency power from entering the seasonal component.
5. *Deseasonalizing*. A deseasonalized series $\mathcal{Y}_{des}^{(k+1)} = \mathcal{Y} - \mathcal{S}^{(k+1)}$ is computed.
6. *Trend smoothing*. The deseasonalized series $\mathcal{Y}_{des}^{(k+1)}$ is smoothed again, thus resulting in the updated trend component $\mathcal{T}^{(k+1)}$. Using the classical STL procedure for trend estimation eases filtering the trend from the raw data, but this is highly susceptible to introduce artificial anomalies in the residual after decomposition. To this end, this paper combines two alternative approaches to extract the trend from a time series: piecewise median [Ahmad and Purdy 2016] and piecewise cubic splines [Poirier 1973]. The first approach is used as a default model for trend estimation, as it preserves edges and prevents impulses from distorting the baseline signal level. The second approach can handle effectively with mean shifts and it is more robust against short-term changes in time series with seasonality. However, it is only employed when the series presents:

    - **Seasonal periodicity** (v). The estimation of v is formulated as a model selection problem that accommodates any periodic signal shape. In context, a range of likely periods is firstly estimated using the fast, robust Welch's periodogram averaging technique [Welch 1967]. Next, a time-domain period estimator based on cross-validated residual errors [Hastie et al. 2009] chooses the best integer period among that range and assigns it to v.
    - **Mean shift** (u). The determination of u is performed by the two-sample Students t-Test [Snedecor and Cochran 1989]. For each data point $\mathcal{Y}_i$, the means of $\frac{n}{10}$ points before and after $\mathcal{Y}_i$ are compared, and the test statistics is computed, having ρ as the critical value at 5% of the t distribution.

Outer loop – Calculate robustness weights. Run ϑ times, for $l = 1, \ldots, \vartheta$:

1. *Residual estimation*. The residual is calculated as $\mathcal{R}^{(l)} = \mathcal{Y} - \mathcal{S}^{(k)} - \mathcal{T}^{(k)}$.
2. *Assign robustness weights*. For each data point $\mathcal{R}_i^{(l)} \in \mathcal{R}^{(l)}$, it is assigned a robustness weight from $\boldsymbol{\omega}^{(l+1)} \in \mathbb{R}^n$ according to:

$$\omega_i^{(l+1)} = B\left(\frac{\mathcal{R}_i^{(l)}}{6 \times \texttt{median}|\mathcal{R}^{(l)}|}\right) \ , \ i = 1, \ldots, n \tag{3}$$

   where $B : \mathbb{R} \to \mathbb{R}$ is the bi-square weight function, which is defined as:

$$B(z) = \begin{cases} (1-z^2)^2 & \text{for } |z| \leq 0 < 1, \\ 0 & \text{for } |z| > 1 \end{cases} \tag{4}$$

   Each weight $\omega_i^{(l+1)}$ is used to make the smoothings at phase 2 and 6 converge closer to the "true" decomposed components in the next iteration.

The actual value for φ (steps of inner loop) should be large enough to reach convergence. In this paper, such value is the number of iterations where the difference between $\mathcal{Y}_{des}$ at steps k and $k-1$ remains greater than a threshold:

$$\varphi = \mathbf{max}(k) \left| \left(\frac{||\mathcal{Y}_{des}^{(k)} - \mathcal{Y}_{des}^{(k-1)}||}{n} \geq 10^{-2} \right) \right| k = 1, 2, \ldots \tag{5}$$

In a similar way, this paper set ϑ (steps of outer loop) as equal to two, since the use of the proposed technique for trend estimation provides robustness in such process.

3.3. Residual analysis

The use of STL allowed the use of a variety of statistical methods to find anomalies in the residual $\mathcal{R}$. This paper adopts the so-called generalized ESD (Extreme Studentized Deviate) test [Rosner 1975] for this purpose. The generalized ESD is used to detect one or more anomalies in a univariate time series that follows an approximately normal distribution. It consists in a generalization of the Grubbs test [Grubbs 1950], where it is not necessary to specify the exact number of anomalies, but an upper bound γ for the suspected number of anomalies to be found.

Given the upper bound, γ, the generalized ESD performs γ separate tests: a test for one anomaly, and so on up to γ anomalies. At each test, the *null* (H_0) and the *alternative* (H_A) hypothesis are: H_0: There are no anomalies in $\mathcal{R}$; and H_A: There are up to γ anomalies in $\mathcal{R}$. In order to reject H_0, the test statistics $\tau^{(k)}$ must be applied:

$$\tau^{(k)} = \max_{i=1}^{n} \left(\frac{|\mathcal{R}_i^{(k)} - \mu_{\mathcal{R}}^{(k)}|}{\sigma_{\mathcal{R}}^{(k)}} \right) \tag{6}$$

where $\mu_{\mathcal{R}}^{(k)}$ and $\sigma_{\mathcal{R}}^{(k)}$ are the average and the standard deviation of $\mathcal{R}^{(k)}$ at $k = 1, 2, \ldots, \gamma$ respectively. Also, the critical value $\Gamma^{(k)}$ must be calculated:

$$\Gamma^{(k)} = \frac{(n-k)t_{p,n-k-1}}{\sqrt{(n-k-1+t_{p,n-k-1}^2)(n-k+1)}} \tag{7}$$

having $t_{\rho,\nu}$ as the ρ percentage point from the t distribution with ν degrees of freedom, $\rho = 1 - \frac{\alpha}{2(n-k+1)}$ and α the level of significance, usually set as $\alpha = 0.05$ (95% confidence) according some extensive experimentation and analysis [Ahmad and Purdy 2016].

The next step aims to remove an $\mathcal{R}_i^{(k)} \in \mathcal{R}^{(k)}$ that maximizes $|\mathcal{R}_i^{(k)} - \mu_{\mathcal{R}}^{(k)}|$ and then recompute Equations (6) and (7) with $n-1$ observations. This process is repeated until γ observations have been removed, resulting in $\tau^{(1)}, \tau^{(2)}, \ldots, \tau^{(\gamma)}$ test statistics, and implicity in $\Gamma^{(1)}, \Gamma^{(2)}, \ldots, \Gamma^{(\gamma)}$ critical values. The number of anomalies is then determined by finding the largest k such that $\tau^{(k)} > \Gamma^{(k)}$.

In this paper, the generalized ESD test was modified in order to avoid the direct specification of such γ. Accordingly, consider that at the end of the k-th iteration, the current vector of residuals $\mathcal{R}^{(k)}$ and also its complement $\mathcal{Q}^{(k)}$ are obtained. The vector

$\mathcal{Q}^{(k)}$ contains the observations that were already removed from $\mathcal{R}^{(k)}$. The number of tests is defined as the maximum value for k such that Condition (8) can be satisfied:

$$\sqrt{2}\,\sigma_{\Upsilon}^{(k)} \geq \sigma_{\Omega}^{(k)} \tag{8}$$

where $\sigma_{\Upsilon}^{(k)}$ and $\sigma_{\Omega}^{(k)}$ are the standard deviations of vectors $\Upsilon^{(k)} = [\Upsilon_1^{(k)}, \ldots, \Upsilon_{n-k}^{(k)}]$ and $\Omega^{(k)} = [\Omega_1^{(k)}, \ldots, \Omega_k^{(k)}]$ respectively. Also, $\Upsilon_i^{(k)} = |\mathcal{R}_i^{(k)} - \mu_{\mathcal{R}}^{(k)}|$ and $\Omega_i^{(k)} = |\mathcal{Q}_i^{(k)} - \mu_{\mathcal{Q}}^{(k)}|$, where $\mu_{\mathcal{R}}^{(k)}$ and $\mu_{\mathcal{Q}}^{(k)}$ are the means of vectors $\mathcal{R}^{(k)}$ and $\mathcal{Q}^{(k)}$ respectively.

The fundamental rationale behind Condition (8) is that anomalies could have a substantial impact on the distribution of data, especially when considering the assumptions of normality and stationarity. A clear indicator of these impacts can be aligned with changes in the mean shifts of such data when some anomaly is removed or included. In context, the mean shifts vectors $\Upsilon^{(k)}$ and $\Omega^{(k)}$ from $\mathcal{R}^{(k)}$ and $\mathcal{Q}^{(k)}$ are computed at each k, assuming absolute values for its components in order to avoid numerical instabilities. Next, the standard deviations $\sigma_{\Upsilon}^{(k)}$ and $\sigma_{\Omega}^{(k)}$ for $\Upsilon^{(k)}$ and $\Omega^{(k)}$ are obtained.

In initial steps, the standard deviation $\sigma_{\Upsilon}^{(k)}$ is used to be greater than $\sigma_{\Omega}^{(k)}$, given the presence of anomalies in such $\mathcal{R}^{(k)}$. As k grows and potential anomalies are removed from $\mathcal{R}^{(k)}$, the value for $\sigma_{\Upsilon}^{(k)}$ becomes smaller and for $\sigma_{\Omega}^{(k)}$ increases. Through extensive experiments and careful observation, one could check that when $\sqrt{2}\sigma_{\Upsilon}^{(k)} < \sigma_{\Omega}^{(k)}$, all anomalies have already been removed from $\mathcal{R}^{(k)}$ and included in $\mathcal{U}^{(k)}$. Naturally, this procedure may yield false positives, as consistent data points could be also included in $\mathcal{U}^{(k)}$. However, such concern can be easily addressed by checking the largest value of k such that $\tau^{(k)} > \Gamma^{(k)}$. Figure 3.3 depict an example on automatic anomaly detection using the modified generalized ESD test. In such example, the proposed technique has achieved a total of 34 tests, where the largest k such that $\tau^{(k)} > \Gamma^{(k)}$ is 32.

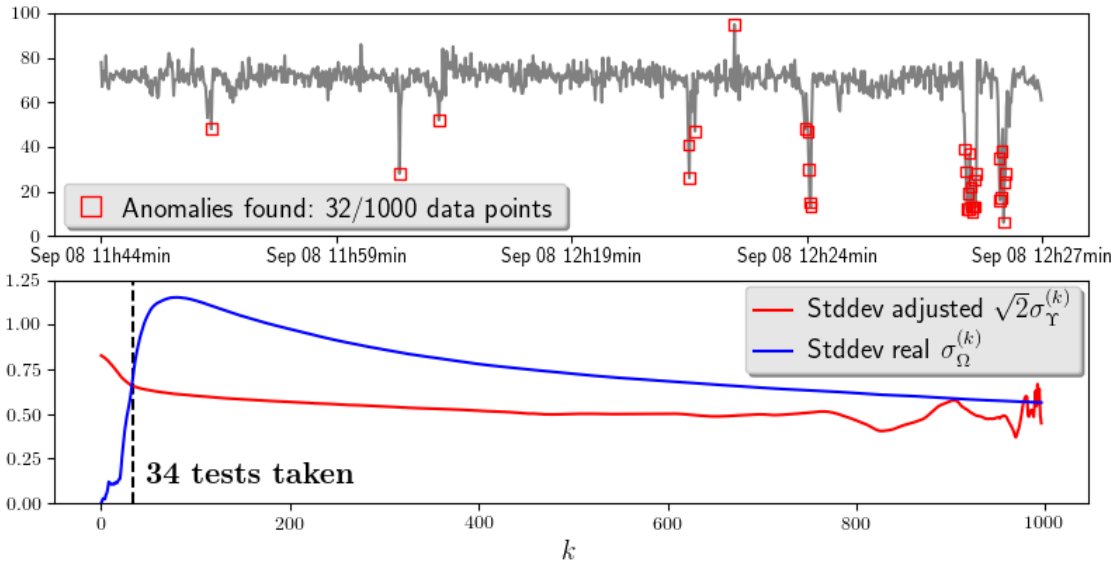


Figure 2. Example of automatic anomaly detection using the modified ESD test. The data points correspond to measurements from real speed traffic volume obtained from the NAB [Lavin and Ahmad 2015].

4. Computational Experiments

This section presents the evaluation results for the proposed technique in the scope of anomaly detection. The evaluation comprises a quantitative analysis of SH-ESD+ and other state-of-the-art anomaly detection models using a real-world benchmark dataset.

4.1. Methodology

In order to provide some quantitative results on anomaly detection, an open source repository called NAB (Numenta Anomaly Benchmark) is considered. NAB attempts to provide a controlled and repeatable environment of tools to test and evaluate the performance of anomaly detection techniques [Lavin and Ahmad 2015]. Also, NAB includes 58 artificial and real-world data sets with over 350.000 records of server metrics, advertisement clicks data and Internet traffic volume. One important aspect of NAB is its scoring methodology for streaming applications. The NAB scoring system quantifies the degree to which the detector under evaluation will be valuable in practical applications. In addition to that, NAB also includes a window around each anomaly and incorporates a time-sensitive scoring mechanism that favors early detection. At last, the so-called "application profiles" define the weighting for the false positives (FP) and false negatives (FN) to illustrate scenarios where fewer missed detections or fewer erroneous detections are more valued.

Since the proposed technique was originally designed to cope with offline anomaly detection, its operation on NAB have considered a single batch input file containing all records for each data set to mimic automation in real-world deployments. The performance of SH-ESD+ was compared to several other approaches in the literature, including: the Numenta Hierarchical Temporal Memory (HTM) model [Ahmad et al. 2017], contextual anomaly detection techniques, such as KNN-CAD [Burnaev and Ishimtsev 2016], the original SH-ESD technique from Twitter [Ahmad and Purdy 2016] and others, such as Etsy's Skyline, EXPoSE and Multinomial Relative Entropy detector [Lavin and Ahmad 2015].

From the standpoint of computational efficiency, the measurements were carried out in a quad-core notebook with a 2.7 GHz Intel processor and 8 GB of RAM using the Linux operating system. The total time to run NAB's complete dataset of 365.558 records was 61 minutes or an average of 10 milliseconds per record.

4.2. Results

The results regarding the performance of such models are presented in Table 1. The first column indicates the standard NAB score for each technique. The two central columns indicate the scores for the reward "low FP" and "low FN" profiles of NAB. Finally, the last column presents the lattency (in miliseconds) for each technique. Latency measures the time taken to process a single data point for anomaly detection. Thus, latency time reported is an average over three runs on 22.695 data records from NAB. A *perfect*, a *null* and a *random* detector were also included in the simulations. The *perfect* one is an idealized detector that makes no mistakes. In the last, the *null* is the worst possible detector. The *random* scores reflect the mean across a range of random seeds.

Overall results show that the HTM-based anomaly detection technique scored the highest value, followed by the proposed SH-ESD+ and KNN-CAD. One can see that most of the techniques performed much better than a random detector, but none of them

Table 1. NAB scoreboard presenting results of each technique considered

Technique	NAB Score	Low FP	Low FN	Lattency (ms)
Perfect	100.0	100.0	100.0	–
Numenta HTM	70.1	63.1	74.3	12.5
SH-ESD+	**59.1**	**53.6**	**68.7**	**2.9**
KNN-CAD	58.0	43.4	64.8	13.9
Relative Entropy	54.6	47.6	58.8	0.06
Twitter SH-ESD	47.1	33.6	53.5	3.4
Etsy Skyline	35.7	27.1	44.5	498.7
EXPoSE	16.4	3.2	26.9	2.8
Random	11.0	1.2	19.5	–
Null	0.0	0.0	0.0	–

has reached close to perfect score, which suggests that there is still significant room for improvement. In addition, Table 1 also demonstrates that values for both application profiles reach closer to high average NAB scores. It occurs mainly because a high score indicates that the underlying model may be able to capture both spatial and temporal anomalies in a more efficient manner, while a low score is often related to techniques that cannot handle appropriately with one of such categories.

Based on a more detailed analysis of the results, it is possible to highlight the strict trade-off between model accuracy and lattency time. In fact, the increase in time complexity is a reflection of the difficulties in capturing the temporal patterns on time series data [Ahmad et al. 2017]. However, the proposed SH-ESD+ performed considerably better than Numenta HTM and KNN-CAD in terms of latency time, and also achieved close proximity considering the accuracy metrics. One reason that lead to such a good result is that SH-ESD+ combines the simplicity of the Twitter's SH-ESD and some straightforward calculations, which can easily provide optimized processes for SH-ESD.

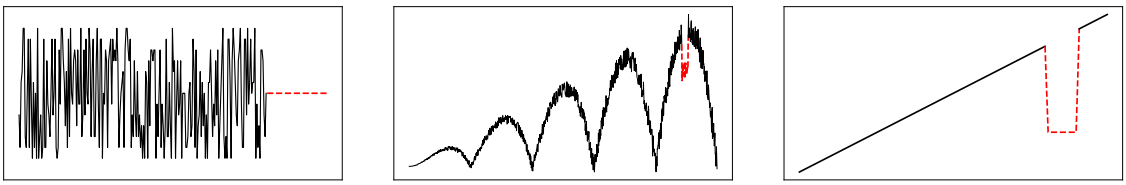
From a different standpoint, Table 2 summarizes some properties of the considered techniques. Categorization is based on their ability to detect spatial and temporal anomalies, to handle concept drifts, and to automatically set and/or to adjust their parameters as data is input. In general, one can see a rough correlation between the number of properties satisfied for each technique in Table 2 and the NAB ranking in Table 1. This is because the ability of each technique to detect anomalies and to handle concept drifts are key to obtaining a good NAB score. Likewise, the level of automation greatly affects the performance of most techniques, as adaption of parameters is critical for any technique to learn continuously and to handle with sustained shifts of data over time.

Another concern that contributes to a better or a worse detection accuracy is the assumptions of each technique regarding the underlying distribution of data. This is one of the main limitations of the Twitter's SH-ESD technique, once it requires a prior definition of two parameters that strongly depend from data distribution, i.e., the length of seasonal periodicity and an upper bound for the number of suspected anomalies [Ahmad and Purdy 2016]. On the other hand, the proposed SH-ESD+ technique employs automatic procedures for identification and tunning with respect to both these parameters.

Table 2. Comparison of properties of each technique implemented on NAB

Technique	Spatial Anomaly	Temporal Anomaly	Concept Drifts	Parameter Automation
Numenta HTM	✓	✓	✓	Update only
SH-ESD+	✓	✓	✓	**Set and update**
KNN-CAD	✓	✓	✓	Update only
Relative Entropy	✓	✓	✓	Update only
Twitter SH-ESD	✓	✓	✓	No automation
Etsy Skyline	✓	×	×	No automation
EXPoSE	✓	✓	✓	Update only

Still considering the relative performance between the Twitter's SH-ESD technique and SH-ESD+, one can see that the proposed technique proved its effectiveness with a higher score for all the considered metrics, as seen in Table 1. Moreover, the SH-ESD+ could deal with several limitations of the SH-ESD at a similar computational effort. For example, SH-ESD is not capable of detecting anomalies in time series which present a flat signal after noisy ones or a noise in growing periodic patterns or disrupted data in flat trends. Figure 3 gives a closer look about the mentioned limitations of SH-ESD.

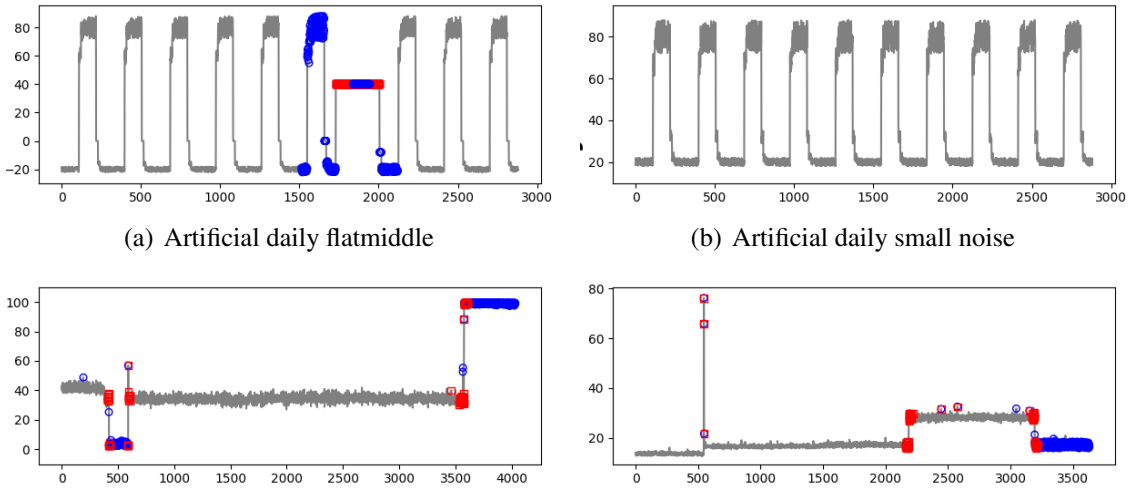


(a) Flat signal (b) Exponential grow (c) Negative trend

Figure 3. Limiations of Twitter's SH-ESD technique for detecting anomalies

In contrast, the proposed technique provides built-in mechanisms for identifying anomalies in the three situations. For example, the detection of anomalies in Figures 3(a) and 3(c) is carried out by the use of piecewise cubic splines, which provides a more accurate estimation of trend component in relation to piecewise median. Also, the anomalies in Figure 3(b) can be easily detected by the proposed technique given its additional mechanism to transform the data until the variation in the series becomes stable over time.

Figure 4 illustrates some comparison between the actual performance of SH-ESD and SH-ESD+ techniques for four different NAB data streams. The selected data streams present diverse characteristics, such as temporal and spatial noise, structural breaks and short-term periodicities. The results strengthens the early discussion about the performance of both models. The SH-ESD+ technique could handle concept drift detections more effectively than SH-ESD. It also has detected subtle temporal anomalies in a more accurate manner, while limiting the number of false positives. For the remaining, both techniques have presented acceptable results in terms of anomaly detection. Therefore, the decision of which one to choose should be made based on application requirements.



(a) Artificial daily flatmiddle

(b) Artificial daily small noise

(c) Amazon EC2 CPU utilization

(d) Amazon RDS CPU utilization

Figure 4. Time series from NAB collection, showing a variety of characteristics. Blue circles represent the anomalies found by SH-ESD while red squares are related to the anomalies detected by the proposed SH-ESD+ technique.

5. Conclusion

This paper addressed the problem of anomaly detection for time series data. When compared to other state-of-the-art anomaly detection algorithms, the proposed technique adds value when considering several mechanisms to cope with anomalies on time series, as summarized in Table 2. It includes robust statistical techniques, like piecewise approximation and Loess smoothing, to minimize the number of false positives and handle effectively with concept drifts. Also, the proposed technique does not need human interaction as it employs advanced parameter identification procedures.

Computational experiments with a real-world benchmark dataset were carried out to assess the performance of the proposed technique regarding its capability of performing accurate anomaly detection. Results have shown that the proposed technique could detect anomalies efficiently and accurately in the experiments, and also presented a considerably lower execution time in relation to some state-of-the-art techniques, and so do Twitter's SH-ESD. Finally, results showed the overall good performance the proposed new version of STL to derive anomalies from a time series and the robustness of ESD to detect them.

Acknowledgment

This research received financial and technical support from Elektro and other associated companies (Aratu and Santa Elina) under the scope of the "ANEEL PD-0385-0063/2015" project: "BID-MONITOR – Big data and data monitoring: the machine intelligence for supporting decision-making under auction-based energy markets", and also received financial and technical support from the AES group and associated companies (CPFL, Brookfield and Global groups) under the scope of the "ANEEL PD-0610-1004/2015" project: "IRIS - Intermittent Renewable Integration: A simulation model of the operation of the Brazilian electrical system to support planning, operation, commercialization and regulation". Both R&D projects are regulated by ANEEL in Brazil.

References

Agrawal, B., Wiktorski, T., and Rong, C. (2017). Adaptive real-time anomaly detection in cloud infrastructures. *Concurrency and Computation: Practice and Experience*, 29(24):1–13.

Ahmad, S., Lavin, A., Purdy, S., and Agha, Z. (2017). Unsupervised real-time anomaly detection for streaming data. *Neurocomputing*, 262(1):134–147.

Ahmad, S. and Purdy, S. (2016). Real-time anomaly detection for streaming analytics. *arXiv preprint arXiv:1607.02480*.

Akoglu, L., Tong, H., and Koutra, D. (2015). Graph based anomaly detection and description: a survey. *Data Mining and Knowledge Discovery*, 29(3):626–688.

Akouemo, H. N. and Povinelli, R. J. (2016). Probabilistic anomaly detection in natural gas time series data. *International Journal of Forecasting*, 32(3):948–956.

Bianco, A. M., Garcia Ben, M., Martinez, E., and Yohai, V. J. (2001). Outlier Detection in Regression Models with ARIMA Errors using Robust Estimates. *Journal of Forecasting*, 20(8):565–579.

Box, G. E., Jenkins, G. M., Reinsel, G. C., and Ljung, G. M. (2015). *Time Series Analysis: Forecasting and Control*. John Wiley & Sons (5th edition).

Box, G. E. P. and Cox, D. R. (1964). An analysis of transformations. *Journal of the Royal Statistical Society*, pages 211–252.

Burnaev, E. and Ishimtsev, V. (2016). Conformalized density-and distance-based anomaly detection in time-series data. *arXiv preprint arXiv:1608.04585 (2016)*.

Chandola, V., Banerjee, A., and Kumar, V. (2009). Anomaly detection: A survey. *ACM Computing Surveys (CSUR)*, 41(3):15–87.

Chou, J. S. and Telaga, A. S. (2014). Real-time detection of anomalous power consumption. *Renewable and Sustainable Energy Reviews*, 33:400–411.

Cleveland, R. B., Cleveland, W. S., McRae, J. E., and Terpenning, I. (1990). STL: A Seasonal-Trend Decomposition Procedure Based on Loess. *Journal of Official Statistics*, 6(1):3–73.

Cleveland, W. S. (1979). Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association*, 74(368):829–836.

Grubbs, F. E. (1950). Sample Criteria for Testing Outlying Observations. *The Annals of Mathematical Statistics*, 21:27–58.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer-Verlag (2nd edition).

Hyndman, R. J. and Khandakar, Y. (2007). *Automatic time series for forecasting: the forecast package for R*. Monash University, Department of Econometrics and Business Statistics.

Johansen, S. and Juselius, K. (1990). Maximum likelihood estimation and inference on cointegrationwith applications to the demand for money. *Oxford Bulletin of Economics and statistics*, 52(2):169–210.

Kadri, F., Harrou, F., Chaabane, S., Sun, Y., and Tahon, C. (2016). Seasonal ARMA-based SPC charts for anomaly detection: Application to emergency department systems. *Neurocomputing*, 173(15):2102–2114.

Laptev, N., Amizadeh, S., and Flint, I. (2015). Generic and scalable framework for automated time-series anomaly detection. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1939–1947.

Lavin, A. and Ahmad, S. (2015). Evaluating Real-Time Anomaly Detection Algorithms – The Numenta Anomaly Benchmark. In *Proceedings of the 14th IEEE International Conference on Machine Learning and Applications*, pages 38–44.

Laxhammar, R. and Falkman, G. (2014). Online learning and sequential anomaly detection in trajectories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(6):1158–1173.

Li, W., Mahadevan, V., and Vasconcelos, N. (2014). Anomaly detection and localization in crowded scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(1):18–32.

Lu, D., Mausel, P., Brondizio, E., and Moran, E. (2004). Change detection techniques. *International Journal of Remote Sensing*, 25(12):2365–2401.

Montgomery, D. C. and Runger, G. C. (2013). *Applied Statistics and Probability for Engineers*. John Wiley & Sons (6th edition).

Moshtaghi, M., Bezdek, J. C., Leckie, C., and Palaniswami, M. (2015). Evolving fuzzy rules for anomaly detection in data streams. *IEEE Transactions on Fuzzy Systems*, 23(3):688–700.

Patcha, A. and Park, J.-M. (2007). An overview of anomaly detection techniques: Existing solutions and latest technological trends. *Computer Networks*, 51(12):3448–3470.

Poirier, D. J. (1973). Piecewise regression using cubic splines. *Journal of the American Statistical Association*, 68(343):515–524.

Rosner, B. (1975). On the Detection of Many Outliers. *Technometrics*, 17(2):221–227.

Snedecor, G. W. and Cochran, W. G. (1989). *Statistical Methods*. Iowa State University Press (8th edition).

Stanway, A. (2013). Etsy skyline. `https://github.com/etsy/skyline`.

Theissler, A. (2017). Detecting known and unknown faults in automotive systems using ensemble-based anomaly detection. *Knowledge-Based Systems*, 123:163–173.

Welch, P. (1967). The use of fast fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms. *IEEE Transactions on Audio and Electroacoustics*, 15(2):70–73.

Witten, I. H., Frank, E., and Hall, M. A. (2011). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann (3rd edition).

Zhou, S., Shen, W., Zeng, D., Fang, M., and Zhang, Z. (2016). Spatial–temporal convolutional neural networks for anomaly detection and localization in crowded scenes. *Signal Processing: Image Communication*, 47:358–368.