

Platform for Anomaly Detection in Time-Series

October 21, 2018

Abstract: The goal of this paper is to present a platform that integrates a number of functionalities necessary in the process of anomaly detection, from data gathering, preprocessing towards various anomaly detection techniques and visualization methods. The purpose of this tool is to allow a developer to test, select and fine tune different algorithms that best fit anomaly detection in a given domain. To demonstrate the utility of the platform, we present a series of experiments done with different methods for anomaly detection on time series and evaluate their results.

Keywords: Anomaly Detection

1 Introduction

We rely on computer and automation systems to manage complex tasks. They are used in many fields and have many applications. Computer systems are used in industrial, academic, commercial and financial applications because of their speed and reliability. Because a defect in such a system can lead to large monetary loss and potentially even loss of life, we need systems that monitor other systems. These systems need to be able to detect faults and anomalous behavior in some way. Algorithms that detect anomalous behavior are therefore necessary and important.

Other applications of anomaly detecting systems include systems that monitor our health. We would like to identify as soon as possible the onset of any disease. By monitoring our health we may be able to spot individuals that have a high risk of contracting some disease and allow medical professionals to act in time to improve their quality of life.

Most anomaly detection algorithms were developed for a given problem or a given range of problems. While creating general algorithms is really hard and may not even be possible, it is best to try many different approaches. The problem we are trying to solve is the lack of a designated platform or tool that can aid in deciding which anomaly detection algorithm works best for a particular problem.

In this paper we propose a platform that enables a user to test a variety of anomaly detection algorithms, with emphasis on time series data. This data has the form $X = \{x_t \in \mathbb{R} : \forall t \geq 0\}$. Furthermore, we will use the following functional definition of what an anomaly is: An anomaly is a data point or set of data, which is significantly different from all other data points or sets. This means that in order to define an anomaly, one must first have a notion of nominal data. The term anomaly is relative and can not be applied to a data-point independently of any dataset.

The following chapters are structured as follows: In Section 2 we describe previous and related work on the subject. Next, in Section 3 we describe the types of anomaly detection methods provided by our platform. We will test these methods in Section 4. This will be followed by a discussion on the future research directions and implications in Section 5.

2 Related Work

Most methods for anomaly detection are developed for a given field or have a specific application. In [1] and [2] the authors developed methods to detect anomalies in airplane data. In [3] similar methods were developed for IoT. In [4] methods for detecting anomalies in Big Data are presented. In [5] methods are used to detect anomalies in industrial machineries.

In [6] a host of algorithms and techniques are described and categorized. The authors found that while some algorithms in different fields are very similar, most algorithms are hard to generalize. They also note, that numerous formulations of anomaly detection problems are not sufficiently explored, i.e. it is not known how well some algorithms perform in a field that was not intended for that algorithm.

We believe that there is a real advantage in a platform that offers a variety of anomaly detection methods to the user. One can test the performance of a number of anomaly detection algorithms for some given test data. By having as many implementations of these algorithms as possible, the user can easily test as many algorithms as she wants with minimal effort and cost.

Efforts to bundle up different anomaly detection algorithms have already begun. In [7] the authors introduced an open source, generic framework for detecting anomalies in large scale time-series data.

In [8] we propose a platform, that offers tools for data visualization, filtering and classification for a variety of data formats, including but not limited to time series data.

3 Time series anomaly detection in the platform

In this section we will present some anomaly detection methods implemented in the platform. We will give a brief description for each one accompanied by an example.

3.1 Outlier Detection Methods

The first type of problem is concerned with classifying each element as anomalous or nominal. We could define a function *label* that labels a data-point $x \in X$ either as anomalous or nominal:

$$label : X \times x \rightarrow \{Nominal, Anomaly\}$$

If it is possible to create a predicate function $p : x \rightarrow \{Nominal, Anomaly\}$ that can accurately detect anomalies, the implementation of the problem becomes trivial: $label(X, x) = p(x)$. Nonetheless, such methods can be used to detect obvious anomalies such as extreme temperature levels or very high blood pressure.

If such a predicate function is not possible, or doesn't meet the correctness requirements, a more complex approach can be used. Since the time series is generated by

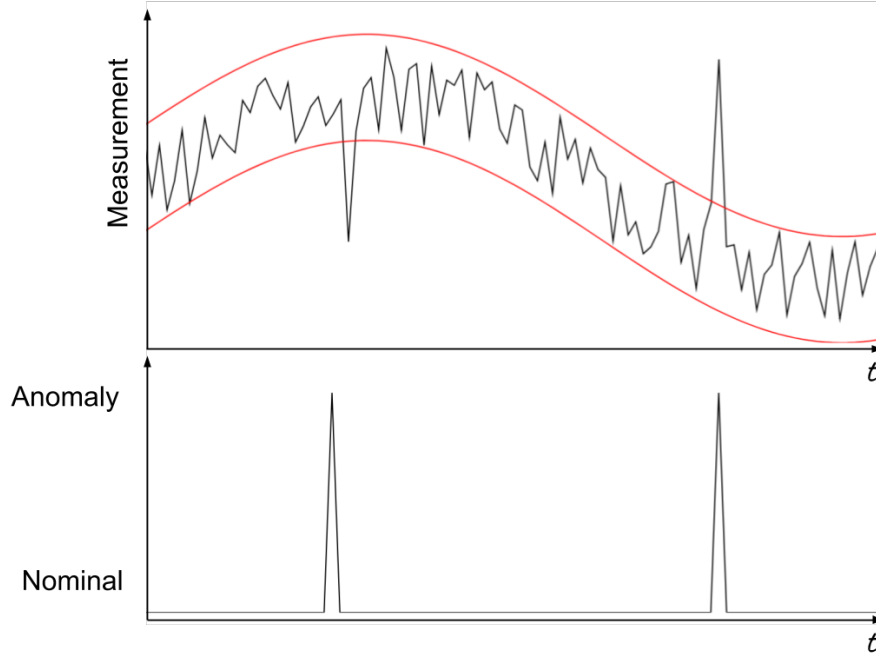


Figure 1: This is a horse.

a generative process, one could hope to be able to accurately describe the underlying process, and create a model of the system:

$$f : t \rightarrow x^*$$

Given such a model, one can label the anomalies based on some threshold given a distance metric $d : x \times x \rightarrow \mathbb{R}$. If the distance between the predicted value x^* and the actual value x is greater than some threshold d_{max} , x can be considered an anomaly:

$$p(x_t) = \begin{cases} \text{Anomaly} & \text{if } d(x, f(t)) \geq d_{max} \\ \text{Nominal} & \text{otherwise} \end{cases}$$

In practice, this is almost never possible. Instead the model will predict the new values based on the old observed values: $f : x^n \rightarrow x^*, n \geq 1$. If $n < |X|$, we use a sliding window approach, where we generate a prediction for the next value based on the actual old values. This is useful if we can model the time series using an autoregressive model.

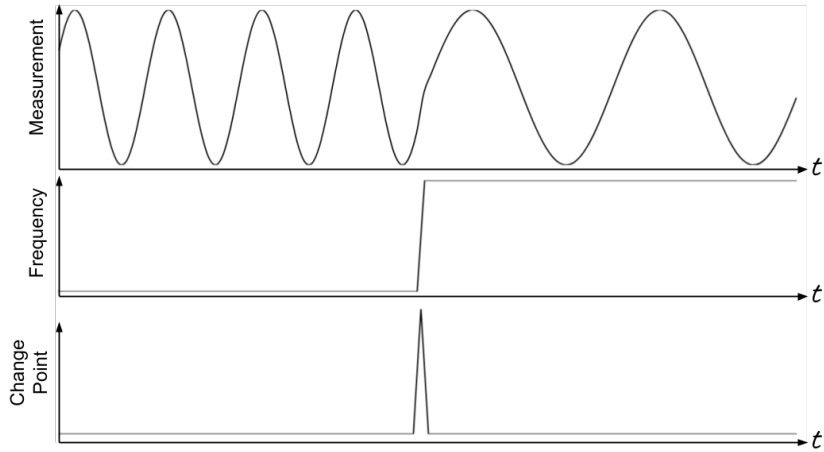


Figure 2: This is a raisin.

3.2 Change Point Detection

3.3 Artificial Intelligence Classification

3.4 Anomalous Time Series Detection

4 Experiments

5 Discussion

Bibliography

References

- [1] G. Silvestri et al. "Fault Detection Using Neural Networks". In: *IEEE World Congress on Computational Intelligence*. 1994. DOI: 10.1109/ICNN.1994.374815.
- [2] S. Basu and M. Meckesheimer. "Automatic outlier detection for time series: an application to sensor data". In: *Knowledge and Information Systems* (2006). DOI: 10.1007/s10115-006-0026-6.
- [3] T. J. Lee et al. "Greenhouse: A Zero-Positive Machine Learning System for Time-Series Anomaly Detection". In: *CoRR* abs/1801.03168 (2018).
- [4] C. K. Maurya, D. Toshniwal, and V. Agarwal. "Anomaly Detection via Distributed Sparse Class-Imbalance Learning". In: *International Conference on Machine Learning*. 2016.
- [5] D. Dasgupta and S. Forrest. "Novelty Detection in Time Series Data using Ideas from Immunology". In: *International Conference on Intelligent Systems*. 1996.

- [6] M. Gupta et al. “Outlier Detection for Temporal Data: A Survey”. In: *IEEE Transactions on Knowledge and Data Engineering* (2014). DOI: 10.1109/TKDE.2013.184.
- [7] N. Laptev, S. Amizadeh, and I. Flint. “Generic and Scalable Framework for Automated Time-series Anomaly Detection”. In: *21th ACM SIGKDD International Conference*. 2015. DOI: 10.1145/2783258.2788611.
- [8] G. Sebestyen et al. “A taxonomy and platform for anomaly detection”. In: *2018 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)* (2018), pp. 1–6. DOI: 10.1109/AQTR.2018.8402710.