Outlier Impact Characterization for Time Series Data

Jianbo Li, [1] **Lecheng Zheng,** [2] **Yada Zhu,** [3] **Jingrui He** [2]
[1] Three Bridges Capital. [2] University of Illinois at Urbana-Champaign, [3] IBM Research
jianboliru@gmail.com, {lecheng4, jingrui}@illinois.edu, yzhu@ibm.us.com

Abstract

For time series data, certain types of outliers are intrinsically more harmful for parameter estimation and future predictions than others, irrespective of their frequency. In this paper, for the first time, we study the characteristics of such outliers through the lens of the influence functional from robust statistics. In particular, we consider the input time series as a contaminated process, with the recurring outliers generated from an unknown contaminating process. Then we leverage the influence functional to understand the impact of the contaminating process on parameter estimation. The influence functional results in a multi-dimensional vector that measures the sensitivity of the predictive model to the contaminating process, which can be challenging to interpret especially for models with a large number of parameters. To this end, we further propose a comprehensive single-valued metric (the **SIF**) to measure outlier impacts on future predictions. It provides a quantitative measure regarding the outlier impacts, which can be used in a variety of scenarios, such as the evaluation of outlier detection methods, the creation of more harmful outliers, etc. The empirical results on multiple real data sets demonstrate the effectivenss of the proposed **SIF** metric.

Introduction

Outlier analysis has been studied for several decades for the sake of data cleaning, fraud detection, gaining insights into the hidden patterns, etc. Numerous models and methods have been proposed to detect outliers either for static data (Aggarwal and Yu 2001; Xiong, Chen, and Schneider 2011; Liu, Huang, and Hu. 2017) or for dynamic data (González and Dasgupta 2003; Rebbapragada et al. 2009; Habler and Shabtai 2017). In the dynamic settings, outliers often exhibit recurring patterns, which can be seen across multiple application domains, such as manufacturing process trace data (Li et al. 2014), medical records (Li et al. 2011; Hauskrecht et al. 2013), sensor data (Subramaniam et al. 2006; Shcherbakov et al. 2016), time-evolving social network data (He, Liu, and Lawrence 2008; Savage et al. 2014), etc. Therefore, it is reasonable to assume that the outliers follow an unknown contaminating process, which contributes to the observed input time series in a probabilistic way.

On the other hand, as machine learning techniques become an indispensable tool in many real applications, there are growing interests to gain insights into the working mechanism of machine learning models. Despite the recent surge of efforts devoted to providing explanations to black-box machine learning models (including outlier detection methods) (Kauffmann, ller, and Montavon 2018; Koh and Liang 2017; Ribeiro, Singh, and Guestrin 2016; Micenková et al. 2013), the vast majority (if not all) of existing techniques focus on static data with feature representations. However, many high-impact application domains (e.g., national security, finance) exhibit the time-evolving nature. The occasional outliers in the time series data can significantly affect the performance of the generated models, rendering the predicted future values not trustworthy. Despite the plethora of outlier detection techniques for time series data, interpretation the detected outliers (especially the recurrent ones) and their underlying generation mechanism is far from solved.

To bridge this gap, in this paper, we tackle the challenge of outlier interpretation in time series data via contamination processes. We start from the influence functional for time series data proposed in (Martin and Yohai 1986), which assumes that the observed input time series is obtained from separate processes for both the core input and the recurring outliers, i.e., the core process and the contaminating process. At each time stamp, with a certain (small) probability, the observed value of the contaminated process comes from the contaminating process, which corresponds to the outliers. In our work, we focus on the generic patchy outliers where the outlying patterns can be present over consecutive time stamps, and aim to study the impact of the contaminating process on both parameter estimation and future value prediction. In particular, we propose a single-valued metric named **SIF** to characterize the impact of the contaminating process on future predictions. Gaining insights on such outlier impact can shed light on not only the relative performance of existing outlier detection techniques but also the type of outliers that a predictive model is robust/sensitive to.

Related Work

Since the pioneering work in (Fox 1972), outlier detection in time series has been well studied over the past decades. Notable distance-based approaches include k nearest neighbor (kNN) (Chandola, Banerjee, and Kumar 2008) and k-

means (Rebbapragada et al. 2009). Exemplary density-based methods include the kNN-CAD and angle-based outlier detection methods (Kriegel, Schubert, and Zimek 2008). Successful deep learning-based approaches include auto-encoder (Han, Kamber, and Pei 2011), deep autoencoding Gaussian mixture model (Zong et al. 2018), and LSTM encoder-decoder (Habler and Shabtai 2017). Due to the lack of labeled data, most outlier detection methods are unsupervised in nature.

Recently, with the increasing complexity of outlier detection techniques, interpretation of outlier detection models and results starts to gain attention from the research community. (Micenková et al. 2013) is based on the maximal separability of outliers and inliers in subspace. (Schwenk and Bach 2014; Kauffmann, Müller, and Montavon 2020) applied the structured one-class Support Vector Machine (SVM) and its neural network reformulation to explain anomalies in terms of input features. In (Liu, Shin, and Hu 2018), the interpretability of an outlier is achieved through outlierness score, attributes that contribute to the abnormality, and contextual description of its neighborhoods by distilling the results of a series of classification tasks. (Cortes 2020) described an outlier detection procedure by evaluating supervised decision tree splits on variables. More generally, to interpret black-box machine learning models, (Ribeiro, Singh, and Guestrin 2016) and (Koh and Liang 2017) proposed approaches to explain individual predictions. These methods are perturbation-based that use data points (Koh and Liang 2017) or features (Ribeiro, Singh, and Guestrin 2016) as a form of perturbation, and measure how the response changes. (Ribeiro, Singh, and Guestrin 2016) used a linear model to approximate model predictions in the local vicinity of a data point and interpreted feature contributions by examining the weights of the sparse linear model. (Koh and Liang 2017) used influence function (Cook and Weisberg 1982) to interpret a training example via the change of parameter estimation and classification loss.

However, existing interpretation techniques cannot be applied to time series data. The key reason lies in the temporal dynamics associated with time series and the low frequency of the recurring outliers. Moreover, although the influence function for time series data originated in robust statistics in the 80s (Martin and Yohai 1986), prior work focuses on exploring the outliers' impact on model parameters. For modern black-box models, such as Recurrent Neural Networks (RNN) (Pineda 1987), the dimensionality of the model parameters can be very high. For instance, a basic RNN has $n^2 + kn + nm$ parameters, where n, k and m denote the dimensionality of the hidden layer, output layer, and input layer, respectively. Therefore, the interpretation in terms of the parameters may not be consumable by humans.

To address these challenges, we propose a single-valued metric, **SIF**, to characterize the impact of the recurring outliers on future predictions. Intuitively, the **SIF** is the partial derivative of the estimated model parameters/predicted values with respect to the degree of contamination. In other words, they measure the sensitivity of the predictive model/predictions to the contaminating process. Therefore, the **SIF** can be used to understand the impact of outliers regardless of the structure and degree of the contaminating processes or the types of predictive models.

Proposed Approach

In this section, we introduce our proposed approach for interpreting recurring outliers in time series data. We start by introducing the influence functional from robust statics (Martin and Yohai 1986) as well as the contaminating process used to model the recurring outliers. Then we present our proposed single-valued metric for characterizing the impact of the contaminating process on future predictions, and discuss its properties in various special cases.

Contaminating Processes

Let y_i^γ denote the observation of the input time series at time stamp i, where $0 \leq \gamma \leq 1$ is a positive parameter controlling the contribution of the contaminating process to the input time series. Following (Martin and Yohai 1986), we assume that y_i^γ has the following definition.

$$y_i^\gamma = (1 - z_i^\gamma)x_i + z_i^\gamma \omega_i \tag{1}$$

where x_i and ω_i denote the observations at time stamp i from the core process (not contaminated by outliers) and the contaminating process (the outliers); z_i^γ denotes the observation at time stamp i from a 0-1 process with parameter γ such that $P(z_i^\gamma = 1) = \gamma + o(\gamma)$. Intuitively, $z_i^\gamma = 1$ indicates that the observed value of the input time series at time stamp i is completely obtained from the contaminating process, and 0 indicates that the observed value is completely obtained from the core process. Furthermore, following the pure replacement model in (Martin and Yohai 1986), we assume that x_i, w_i, and z_i^γ are obtained from mutually independent processes, which are denoted μ_x, μ_w, and μ_z^γ respectively. Without loss of generality, we assume that all these processes are ergodic and stationary (Jürgen Franke 2015).

In general, the 0-1 process z_i^γ captures the characteristics of the observed recurring outliers in the input time series. An example of the 0-1 process corresponds to the so-called *patchy outliers* (Martin and Yohai 1986), where z_i^γ with various values of time stamp i are highly correlated. More specifically, let $\tilde{z}_i^q$ denote *i.i.d.* binomial $B(1, q)$ sequence, and z_i^γ depends on $\tilde{z}_i^q$ in the following way:

$$z_i^\gamma = \begin{cases} 1, & \text{if } \tilde{z}_{i-l}^q = 1 \text{ for some } l = 0, 1, \ldots, k-1 \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

where k is a positive integer for the patch size and $0 \leq q \leq 1$. Notice that when $k = 1$, $z_i^\gamma = \tilde{z}_i^q$, and the patchy outliers are reduced to independent outliers, i.e., z_i^γ is independent of z_j^γ for $i \neq j$. Let $\gamma = kq$. Then it is easy to verify that $P(z_i^\gamma = 1) = kq + o(q)$, which is consistent with the requirement on the contaminating process from Eq. (1).

Influence Functional

Let $\boldsymbol{\theta} \in \mathbb{R}^p$ denote the vector of parameters involved in the predictive model of the input time series data, where p is the number of parameters. It can be estimated by solving the following equation.

$$\int \tilde{\Psi}(\boldsymbol{y}_i^\gamma, \boldsymbol{\theta}) d\mu_y^\gamma = 0 \tag{3}$$

where $\tilde{\Psi}$ denotes a function from $\mathbb{R}^\infty \times \mathbb{R}^p$ to $\mathbb{R}^p$ (e.g., first order derivative of the log-likelihood function), $\boldsymbol{y}_i^\gamma$ denotes the input time series up to time stamp i, and μ_y^γ denotes the process followed by y_i^γ. For the above equation, let $\hat{\boldsymbol{\theta}}^\gamma$ denote its unique root, i.e., the optimal estimate of the model parameters.

Based on the above notation, the influence functional (IF) for time series data is defined by (Martin and Yohai 1986).

$$\text{IF}(\boldsymbol{\theta}, \{\mu_y^\gamma\}) = \lim_{\gamma \to 0} \frac{\hat{\boldsymbol{\theta}}^\gamma - \hat{\boldsymbol{\theta}}^0}{\gamma} = \left. \frac{d\hat{\boldsymbol{\theta}}^\gamma}{d\gamma} \right|_{\gamma=0} \tag{4}$$

From the above definition, it can be seen that the influence functional is a p-dimensional vector measuring the impact of γ on the estimated parameters around $\gamma = 0$, i.e., no outliers observed in the input time series. In other words, the influence functional depends on the intrinsic properties of the core process μ_x and the contaminating process μ_ω, irrespective of the frequency of outliers observed in the input time series. In the next section, we will empirically demonstrate the influence functional associated with various types of the contaminating process for a specific input process. In general, the influence functional can be computed as follows.

Lemma 1 *Under mild conditions, we have*

$$\textit{IF}(\boldsymbol{\theta}, \{\mu_y^\gamma\}) = \lim_{\gamma \to 0} \frac{-E_y(\boldsymbol{C}^{-1}\tilde{\Psi}(\boldsymbol{y}_i^\gamma, \hat{\boldsymbol{\theta}}^0))}{\gamma} \tag{5}$$

where nonsingular $p \times p$ matrix $\boldsymbol{C} = \frac{\partial E_x \tilde{\Psi}(\boldsymbol{x}_i, \hat{\boldsymbol{\theta}}^0)}{\partial \boldsymbol{\theta}}|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}^0}$, $E_y(\cdot)$ and $E_x(\cdot)$ denote the expectation under the observed process μ_y^γ and the core process μ_x respectively, and $\boldsymbol{x}_i$ denotes the core time series up to time stamp i.

Proof Theorem 4.1 in (Martin and Yohai 1986) shows that under mild conditions,

$$\text{IF}(\boldsymbol{\theta}, \{\mu_y^\gamma\}) = \lim_{\gamma \to 0} \frac{E(\mathbf{ICH}(\boldsymbol{y}_1^\gamma))}{\gamma}$$

where $\mathbf{ICH}(\boldsymbol{y}_1^\gamma)$ denotes the Hampel's influence curve (Hampel 1974) with respect to γ. This theorem, together with Eq. (4.2) in (Martin and Yohai 1986) completes the proof.■

Notice that the subscript i in Eq. (5) on the right hand side vanished because of stationarity of $\boldsymbol{y}_i^\gamma$. $\boldsymbol{C} \in \mathbb{R}^{p \times p}$ is essentially the Hessian of the objective function for solving $\boldsymbol{\theta}$, e.g., the log-likelihood function. The inverse of $\boldsymbol{C}$ can be computationally expensive due to the high dimensionality of the parameter space, especially for deep neural networks. To address this problem, following (Koh and Liang 2017), we adopt the implicit Hessian-Vector Products (HVPs) with stochastic estimation (Agarwal, Bullins, and Hazan 2017; Chen, Liu, and Zhang 2018; Li et al. 2018). Following the general method of computing the influence functional as shown in **Lemma 1**, the influence functional can enjoy a closed-form solution for certain classes of the underlying predictive model. Let $f(\boldsymbol{y}_{i-1}^\gamma, \boldsymbol{\theta})$ denote the underlying model for predicting the observed value of input time series y_i^γ at time stamp i. The following lemma demonstrates the influence functional for a simple autoregressive model, i.e., an AR(1) model given by $f(\boldsymbol{y}_i^\gamma, \theta) = \theta y_{i-1}^\gamma$, under patchy outliers with size k and $\gamma = kq$.

Lemma 2 *For AR(1) model with a single parameter θ, the influence functional of patchy outliers with size k can be computed as follows.*

$$\begin{aligned}\textit{IF}(\theta, \{\mu_y^\gamma\}) = \frac{1}{kE_x(x^2)}(&-2E_x(x)E_\omega(\omega) - (k-1)E_\omega(\omega_0\omega_1) \\ &+ \hat{\theta}^0 E_x(x^2) + \hat{\theta}^0 k E_\omega(\omega^2))\end{aligned}$$

where $E_\omega(\cdot)$ denotes the expectation under the contaminating process μ_ω, and $E_\omega(\omega_0\omega_1)$ is the lag 1 autocorrelation of the outliers.

Proof of **Lemma 2** is given in the appendix.

Notice that the analysis can be generalized to AR(n) models, i.e., $f(\boldsymbol{y}_i^\gamma, \boldsymbol{\theta}) = \sum_{j=1}^n \theta_j y_{i-j}^\gamma$ where θ_j is the j^{th} element of $\boldsymbol{\theta}$, and is omitted for brevity here. From this lemma, we have the following observations. First of all, when $k = 1$, i.e., independent outliers, the influence functional is reduced to

$$\text{IF}(\theta, \{\mu_y^\gamma\}) = \frac{-2E_x(x)E_\omega(\omega) + \hat{\theta}^0 E_x(x^2) + \hat{\theta}^0 E_\omega(\omega^2)}{E_x(x^2)}$$

On the other hand, when k goes to infinity (while $\gamma = kp \to 0$), the influence functional is reduced to

$$\text{IF}(\theta, \{\mu_y^\gamma\}) \xrightarrow{k \to \infty} -\frac{E_\omega(\omega_0\omega_1)}{E_x(x^2)} + \frac{\hat{\theta}^0 E_\omega(\omega^2)}{E_x(x^2)} \tag{6}$$

From the above equations, it can be seen that as k increases, the impact of the first-order moment from the contaminating process gradually decreases, and the impact of the second-order moment remains in the influence functional.

Influence on Future Predictions

Notice that the dimensionality of the influence functional increases with the number of parameters p in the model, and can be difficult to comprehend for interpretation purposes, especially for complex models with a large number of parameters. To address this problem, we propose a single-valued metric based on the influence functional to characterize the impact of the contaminating process on future predictions. More specifically, let $g(\gamma, \hat{\boldsymbol{\theta}})$ denote the expected predicted value of $f(\boldsymbol{y}_{i-1}^\gamma, \boldsymbol{\theta})$ with respect to μ_y^γ, i.e., $g(\gamma, \boldsymbol{\theta}) := E_y(f(\boldsymbol{y}_{i-1}^\gamma, \boldsymbol{\theta}))$. Next we propose the following function for measuring the influence of the contaminating process on future predictions.

$$\textbf{SIF}(\boldsymbol{\theta}, \{\mu_y^\gamma\}) := \frac{d}{d\gamma} g(\gamma, \hat{\boldsymbol{\theta}}^\gamma)|_{\gamma=0} \tag{7}$$

Based on the above definition, we have

$$\begin{aligned}\textbf{SIF}(\boldsymbol{\theta}, \{\mu_y^\gamma\}) &= \frac{\partial g(\gamma, \hat{\boldsymbol{\theta}}^\gamma)}{\partial \gamma} + \frac{\partial g(\gamma, \boldsymbol{\theta})'}{\partial \boldsymbol{\theta}} \cdot \left. \frac{d\hat{\boldsymbol{\theta}}^\gamma}{d\gamma} \right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}^\gamma, \gamma=0} \\ &= \frac{\partial g(0, \hat{\boldsymbol{\theta}}^0)}{\partial \gamma} + \frac{\partial g(0, \hat{\boldsymbol{\theta}}^0)'}{\partial \boldsymbol{\theta}} \cdot \text{IF}(\boldsymbol{\theta}, \{\mu_y^0\})\end{aligned} \tag{8}$$

where $\frac{\partial}{\partial \boldsymbol{\theta}} g(\gamma, \boldsymbol{\theta})$ is a p-dimensional vector, and $\frac{\partial}{\partial \boldsymbol{\theta}} g(\gamma, \boldsymbol{\theta})'$ is its transpose.

In general, to compute the **SIF**, in addition to the influence functional, we also need to compute $\frac{\partial}{\partial \gamma} g(0, \hat{\boldsymbol{\theta}}^0)$

and $\frac{\partial}{\partial \boldsymbol{\theta}} g(0, \hat{\boldsymbol{\theta}}^0)$. Notice that $\frac{\partial}{\partial \boldsymbol{\theta}} g(0, \hat{\boldsymbol{\theta}}^0)$ can be re-written as $E_x(\frac{\partial f(\boldsymbol{x}_{i-1}, \hat{\boldsymbol{\theta}}^0)}{\partial \boldsymbol{\theta}})$, where the partial derivation with respect to $\boldsymbol{\theta}$ is easy to implement in auto-grad systems such as TensorFlow, Torch and Theano as illustrated in (Koh and Liang 2017). On the other hand, $\frac{\partial}{\partial \gamma} g(0, \hat{\boldsymbol{\theta}}^0)$ can be calculated based on the following lemma.

Lemma 3 *For patchy outliers with size k, we have*

$$\frac{\partial g(0, \hat{\boldsymbol{\theta}}^0)}{\partial \gamma} = \frac{1}{k} E_{x,w} \left[\sum_{j=1}^{i-1} \left(f(\tilde{\boldsymbol{y}}_{i-1}^{(j)}, \hat{\boldsymbol{\theta}}^0) - f(\boldsymbol{x}_{i-1}, \hat{\boldsymbol{\theta}}^0) \right) \right] \quad (9)$$

where $\tilde{\boldsymbol{y}}_{i-1}^{(j)}$ is a vector with elements defined as follows

$$\tilde{y}_{i-1,s}^{(j)} = \begin{cases} \omega_s, & \text{if } s = j + l \text{ for } l = 0, \cdots, k-1 \\ x_s, & \text{otherwise} \end{cases}$$

Proof For patchy outliers, the expectation over z^γ can be transferred to $\tilde{z}^p$ and then expanded for small p values

$$\begin{aligned} & \left. \frac{\partial E_{z^\gamma} \left[f\left(\mathbf{y}_{i-1}^\gamma, \hat{\boldsymbol{\theta}}^0 \right) \right]}{\partial \gamma} \right|_{\gamma=0} \\ &= \frac{1}{k} \frac{\partial}{\partial p} \left[\begin{array}{l} f(\mathbf{x}_{i-1}, \theta)(1-p)^{i-1} \\ + \sum_{m=1}^{i-1} f(\tilde{\mathbf{y}}_{i-1}^{(m)}, \hat{\boldsymbol{\theta}}^0)(1-p)^{i-2} p + o(p) \end{array} \right]_{p=0} \\ &= \frac{1}{k} \sum_{m=1}^{i-1} \left[f(\tilde{\mathbf{y}}_{i-1}^{(m)}, \hat{\boldsymbol{\theta}}) - f(\mathbf{x}_{i-1}, \theta) \right]. \end{aligned}$$

Substituting the above equation back to the expectation over x, ω yields **Lemma 3**. ■

Note that if the model does not have a long term memory, the summation terms with $m \ll i$ in Eq. (9) become 0s, suggesting the vanishing boundary effects. For AR(n) models, the following lemma shows the closed-form solution for the two partial derivatives in Eq. (8).

Lemma 4 *For AR(n) models, we have*

$$\begin{aligned} \frac{\partial}{\partial \gamma} g(0, \hat{\boldsymbol{\theta}}^0) &= (-E_x(x) + E_\omega(\omega)) \sum_{j=1}^{n} \theta_j \\ \frac{\partial}{\partial \boldsymbol{\theta}} g(0, \hat{\boldsymbol{\theta}}^0) &= E_x([x_{i-1}, \ldots, x_{i-n}]') = E_x(x) \mathbf{1}_n \end{aligned}$$

where $\mathbf{1}_n$ is a $n \times 1$ column vector consisting of all 1s.

Proof For $AR(n)$ models, the predictive model $f(\boldsymbol{y}_{i-1}^\gamma, \boldsymbol{\theta}) = \boldsymbol{\theta} \cdot \boldsymbol{y}_{(i-1):(i-n)}^\gamma$, where $\cdot$ denotes the inner product between two vectors, and $\boldsymbol{y}_{(i-1):(i-n)}^\gamma$ denotes the observed input time series between time stamps $i-1$ and $i-n$. Therefore $g(\gamma, \boldsymbol{\theta}) = \sum_{j=1}^n \theta_j E_y(y_{i-j}^\gamma) = \sum_{j=1}^n \theta_j((1-\gamma)E_x(x) + \gamma E_\omega(\omega))$. The lemma naturally follows by taking the partially derivative of $g(0, \hat{\boldsymbol{\theta}}^0)$ with respect to γ and $\boldsymbol{\theta}$. ■

Based on the above lemma, we can see that if $E_x(x) = E_\omega(\omega)$, i.e., the core process and the contaminating process have the same mean, then $\frac{\partial}{\partial \gamma} g(0, \hat{\boldsymbol{\theta}}^0) = 0$, and the **SIF** is reduced to

$$\textbf{SIF}(\boldsymbol{\theta}, \{\mu_y^\gamma\}) = E_x(x) \mathbf{1}_n' \cdot \text{IF}(\boldsymbol{\theta}, \{\mu_y^0\})$$

From the above equation, it can be seen that in this case, the impact of the contaminating process on future predictions is in proportion to the sum of all the elements in the influence functional.

Outlier Interpretation with SIF

Comparison of Outlier Detection Methods A predictive model could be sensitive to different kinds of outliers. The proposed **SIF** measures the impact of the contaminating process on future predictions with a **specific** predictive model, regardless of the model type. Therefore, it can be used for outlier interpretation and evaluation of existing outlier detection methods. More specifically, given an outlier detection method and the observations that have been *identified as outliers* by this method from the input time series y_i^γ, we first estimate the contaminating process by estimating its moments as required by the computation of the influence functional for AR(n) models (**Lemma 2**), or by estimating the parameters of the contaminating process (e.g., RNN and Gaussian processes). Due to the low frequency of the outliers in general, there will be many missing values in ω_i. For the purpose of estimating the parameters of the contaminating process with patchy outliers enabled by z_i^γ, we first divide the entire time series into multiple sub-series such that each sub-series will consist of one or a few sequences of patchy outliers. Then we can use various filtering techniques such as (Anava, Hazan, and Zeevi 2015; Wang et al. 2005) to estimate the parameters of the contaminating process. The value of γ can be roughly estimated by the percentage of the outliers in the entire input time series. k the patch size can be obtained by solving Eq. (10) in **Lemma** 5 using e.g., Newton's method, where $E_z^\gamma(L)$ can be estimated by the average number of consecutive time stamps of patchy outliers. Note that the root to Eq. (10) might not be unique due to its nonlinearity. A rule of thumb is that reasonable k values should be less than but close to $E_z^\gamma(L)$ when γ is small.

Putting everything together, we estimate both the influence functional and the **SIF** numerically by gradually reducing γ to 0, or probabilistically removing some identified outliers and replacing them by the predicted values from the underlying model of the core process. Following the same procedure, we obtain the **SIF** of various outlier detection methods. Intuitively, larger **SIF** values indicate that the outliers have a higher impact on future predictions, and thus the corresponding detection method is able to identify the more prominent outliers.

Lemma 5 *For generic patchy outliers, the expected number of consecutive time stamps in a patch is given by*

$$E_z^\gamma(L) = k + \frac{k^{k+1} - (\gamma+1)(k-\gamma)^k k}{\gamma(k-\gamma)^k}. \quad (10)$$

Proof Suppose that a patch of outliers start at a time stamp i, i.e. $z_i^\gamma = 1$ and if $i > 0$, then $z_{i-1}^\gamma = 0$. This suggests that

$\tilde{z}_i^q = 1$ based on Eq. (2). Let $A = l$ denote the number of time stamps until the next $\tilde{z}^q = 1$, i.e., $\tilde{z}_{i+j}^q = 0$ for $j = 1, \cdots, l$ and $\tilde{z}_{i+l}^q = 1$. If $l > k$, then the patch length equals k; otherwise, it can be computed by adding l to the expected patch length starting from time stamp $i + l$, which is again $E_z^\gamma(L)$ due to symmetry. Therefore, the expected patch size can be analyzed as an iterative equation given by

$$E_z^\gamma(L) = (1 + E_z^\gamma(L))P(A = 1) + (2 + E_z^\gamma(L))P(A = 2) + \cdots + (k + E_z^\gamma(L))P(A = k) + kP(A > k).$$

Since $\tilde{z}_i^q = 0$ follows *i.i.d* binomial $B(1, q)$, A follows a geometric distribution with parameter q. Solving $E_z^\gamma(L)$ leads to Eq. (10).■

Crafting Adversarial Contaminating Processes (ACP) Recent work shows that the existence of adversarial attacks may be an inherent weakness of machine learning models (Madry et al. 2017). The **SIF** can be applied to craft adversarial contaminating processes (ACP) that can shed light on model vulnerability as well as the type of outliers that a predictive model is robust/sensitive to. Here ACP means identifying the best contamination process that will impact the parameter estimates/predictions to the largest degree without raising suspicion, i.e., the core process and the contaminating process have the same first two moments. To craft ACP, it needs to find the best contaminating time series structure (e.g. AR(n) or RNN) and then estimate the optimal parameters for the data structure. In practice, these two steps are entangled but we split them into two for ease of illustration.

Experimental Results

In this section, we quantitatively evaluate the influence functional (IF) and the **SIF** with respect to the impact of the contaminating process on the parameter estimation and future predictions. We also demonstrate the use of the **SIF** for evaluating outlier detection methods and crafting adversarial contaminating processes.

Analysis of Influence Functional and SIF

The IF/**SIF** is generally applicable to any contaminating processes that are ergodic and stationary, as well as any kinds of predictive models. For the sake of presentation clarity as the dimensionality of the IF increases with the number of parameters, we use an AR(1) model with a single parameter θ as the core process in the following experiments. The IF is approximated by the slope of $\hat{\theta}^\gamma$ with respect to γ. The **SIF** is given by the slope of $g(\gamma, \hat{\boldsymbol{\theta}}^\gamma)$ over a set of γ values passing $(0, g(0, \hat{\boldsymbol{\theta}}^0))$, where $g(\gamma, \hat{\boldsymbol{\theta}}^\gamma)$ is estimated by the average of the first out-of-sample prediction. The parameter γ controls the contribution of the contaminating process to the input time series. We set the coefficient of the core process as 0.7 and $\gamma \in [0, 0.01]$. Notice that we observed similar results with other coefficients and thus, we only present the results with this specific value. The same applies to other experiments in this work. One can verify all the experimental results using our code, to be shared upon paper acceptance. The setting of contaminating processes, patchy size k and predictive models in below experiments are given in Table 1. Following Eq. (1), we obtain the contaminated time series which is used to train the predictive models and estimate $\hat{\theta}^\gamma$ and $g(\gamma, \hat{\boldsymbol{\theta}}^\gamma)$ over a set of γ values. We repeat the simulation 50 times and 1000 times for analysis of the IF and the **SIF**, respectively, and calculate the mean and variance of the parameter estimation. First, we evaluate the IF with respect to k, the parch size of outliers. The results are shown in Fig. 1(a) together with the theoretical IF values for the given predictive model. We observe that: (1) $\hat{\theta}^\gamma$ linearly depends on γ and matches its theoretical value in the studied range; (2) as k increases, the rate of linear dependency (i.e., the IF) decreases as suggested by Eq. (6). Second, we compare different contaminating processes, i.e., *i.i.d.* $N(0, 1)$ and AR(1) with coef. -0.5. Fig. 1(b) shows that the influence of the *auto-correlated contaminating process* is larger than *white noise* for patchy outliers. These studies verify the implication of **Lemma 2**. Third, we compare the **SIF** for predictive models AR(1) and RNN (Goodfellow, Bengio, and Courville 2016). As shown in Fig. 1(c), the **SIF** of the RNN model is larger than that of the AR(1) model, indicating that a simple AR(1) predictive model is more robust than the RNN model.

Table 1: Experiment Setting for Analysis of IF and **SIF**

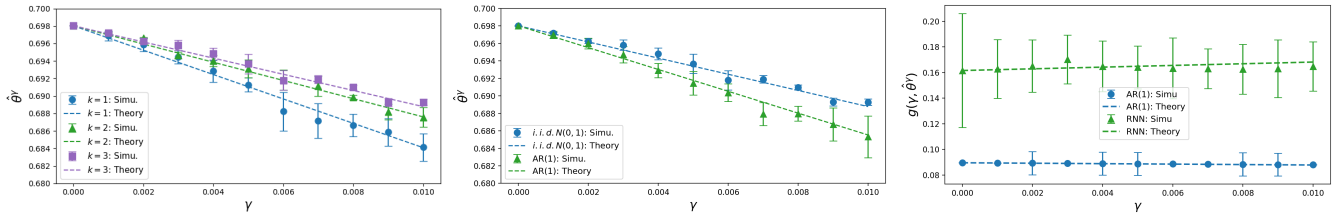
Exp	Contaminating process	Patch size, k	Predictive model
1^{st}	*i.i.d.* $N(0, 1)$	$1, 2, 3$	AR(1)
2^{nd}	*i.i.d.* $N(0, 1)$	3	AR(1)
	AR(1), coef. -0.5		
3^{rd}	*i.i.d.* $N(0, 1)$	3	AR(1), RNN

Evaluation of Outlier Detection Methods

In this section, we demonstrate the evaluation of outlier detection methods for time series data using the **SIF**. In practice, this can facilitate the identification of the most influential outliers and the selection of robust machine learning techniques for time series forecasting. In particular, our proposed technique is *not restricted to any type of outlier detection methods*.

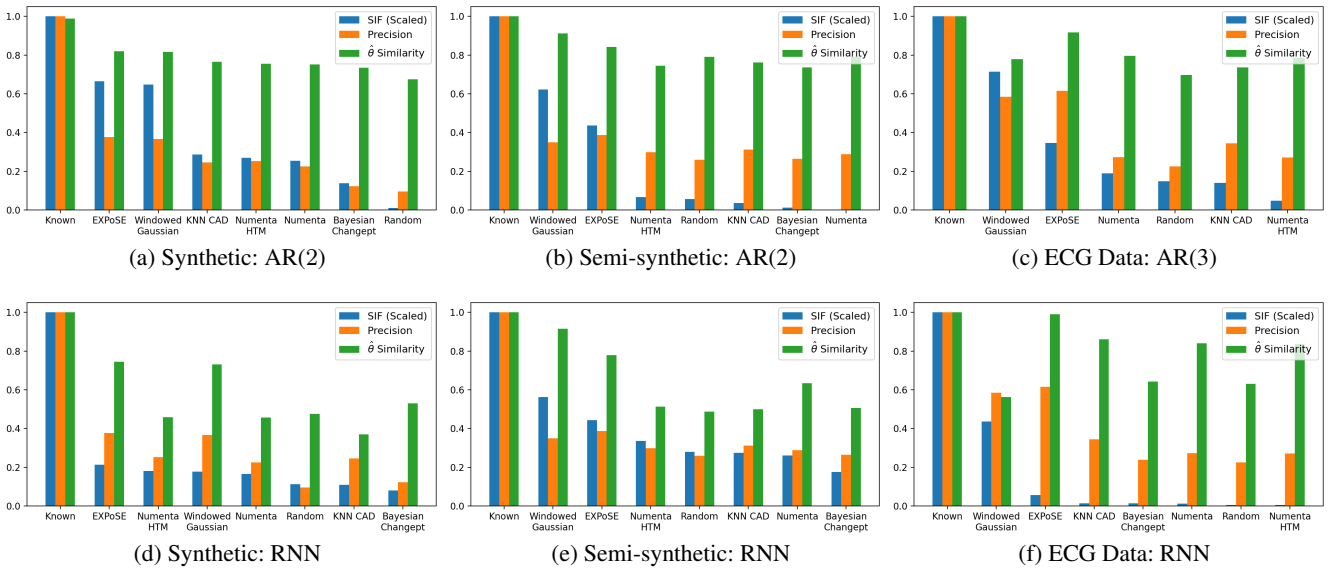
Data The evaluation is based on three data sets: synthetic, semi-synthetic, and electrocardiography (ECG) data, where true outliers are known. Notice that actual outlier labels are not required by using the **SIF** to evaluate outlier detection methods. However, such labels allow us to verify the effectiveness of using the **SIF** to compare various outlier detection methods. The synthetic data is created using an AR(2) model with coefficients $(0.7, -0.3)$ as the core time series, an AR(1) with a coefficient 0.5 as the contaminating process, $k = 5$, and $\gamma = 0.3$. For the semi-synthetic data, a clean time series, *real_35*, is randomly selected from the Yahoo! Webscope* and contaminated in the same way as the synthetic data. The original ECG data is obtained from

*http://labs.yahoo.com/Academic_Relations



(a) IF for different patch sizes (b) IF for different contaminating processes (c) **SIF** for different predictive models

Figure 1: Analysis of IF and **SIF** for core process AR(1) with coef. 0.7, $\gamma \in [0, 0.01]$



(a) Synthetic: AR(2) (b) Semi-synthetic: AR(2) (c) ECG Data: AR(3)

(d) Synthetic: RNN (e) Semi-synthetic: RNN (f) ECG Data: RNN

Figure 2: Ranking of outlier detection methods: sorted based on $|\mathbf{SIF}|$. The *Known* denotes the method using true outliers.

Physionet[†] under the name BIDMC Congestive Heart Failure Database (chfdb), record *chf07*. Following (Van Looveren et al. 2020), the data is processed as follows: 1) each heartbeat is extracted and made equal length via interpolation; 2) normal seasonal signal is removed from the obtained time series using the Holt-Winters' additive method; 3) the residual time series is labeled as normal/abnormal heartbeats based on that provided in (Van Looveren et al. 2020).

Setup We apply all the methods in the wrapped Numenta Anomaly Benchmark (NAB)[‡] tool to identify outliers except those fail to return results for the given data sets. These methods include 1) *Random*: randomly selected outliers; 2) *EXPoSe* (Schneider, Ertel, and Ramos 2016): distance-based; 3) *Windowed Gaussian*: distribution-based; 4) *Bayesian online Changepoint* (Adams and MacKay 2007): distribution-based; 5) *KNN CAD* (Burnaev and Ishimtsev 2016): combination of density- and distance-based; 6) *Numenta*: prediction-based; 7) *Numenta HTM* (Lavin and Ahmad 2015): rule- and prediction-based. For fairness, the thresholds are chosen such that all the methods return the actual number of outliers.

We use three ranking metrics, i.e., the magnitude of **SIF** for a specific predictive model (AR(2) or RNN), **Precision** (**Prec.**) of identified outliers and the **Similarity** (**Sim.**) of model parameters. The **Sim.** is calculated based on $1/(1+d)$ where d denotes the Euclidean distance between the real and the estimated parameters using *cleaned* data based on the results from various outlier detection methods.

Ranking Result The ranking results are presented in Fig. 2 as the normalized bar charts, where the *Known* denotes the method using true outliers and has value 1 or approximately 1. For the $AR(\cdot)$ predictive models, the relative ranking of various outlier detection methods based on the magnitude of the **SIF** matches perfectly to that of **Prec.** and the $\hat{\boldsymbol{\theta}}$ **Sim.** based on the synthetic data and is pretty consistent for the semi-synthetic data and ECG data as well. For the RNN predictive model, the ranking results based on the **SIF** are also largely consistent with **Prec.** and the $\hat{\boldsymbol{\theta}}$ **Sim.** for all the data sets. To further quantify the ranking similarity of different criteria, we report the Kendall's Tau coefficient in

[†]https://physionet.org/content/chfdb/1.0.0/

[‡]https://github.com/numenta/NAB

Table 2: Kendall's Tau Coefficient Summary

Data	Model	SIF &Prec.	SIF &$\hat{\theta}$ Sim.
Synth	AR(2)	0.50	1.00
Synth	RNN	0.43	0.36
Semi-Synth	AR(2)	0.43	0.71
Semi-Synth	RNN	0.21	0.43
ECG	AR(3)	0.43	0.14
ECG	RNN	0.29	0.64

Table 3: Comparison of ACP

ACP	SIF	RMSE
ARMA(2, 2)	0.4263	0.7628 ± 0.0098
Two-Layer LSTM	0.3203	0.7516 ± 0.0169
Two-Layer RNN	0.1184	0.7431 ± 0.0098

Table 2, where all the coefficients are positive. This confirms the ranking consistency between the **SIF** without outlier labels and other metrics utilizing outlier labels.

Crafting Adversarial Contaminating Processes

In this section, we demonstrate using the **SIF** to craft adversarial contaminating processes (ACP) with the most influence on future predictions and provide insights into the type of outliers that a predictive model is robust/sensitive to. Again we use the *real_35* data as the core time series, split it into train and test $(60 : 40)$ sets sequentially, normalize the train set to mean 0 and standard deviation 1, and train an LSTM predictive model.

First, we illustrate that the **SIF** can be applied to select the structure of contaminating time series that a predictive model is most vulnerable to. We set $k = 1$ and generate contaminating processes using autoregressive moving-average (ARMA) model, two layers RNN, and LSTM (Hochreiter and Schmidhuber 1997), which are used to compute the corresponding **SIF** values. To be specific, we randomly select the coefficients for the ARMA(2, 2) model under the constraint of the stationary triangle (Jürgen Franke 2015), and choose the LSTM and RNN models with two layers and 256 hidden states. Notice that the optimal parameters of the selected model are discussed in the next step. In practice, these two steps are entangled but we split them into two for ease of illustration. Given the trained LSTM predictive model, the maximum absolute **SIF** value is obtained from the ARMA(2, 2) model, as shown in Table 3. This implies that in this specific setting and given data structures, the LSTM predictive model is most sensitive to the ARMA type of outliers. To validate this observation, we conduct further experiments. We randomly contaminate the train set with 10% outliers using the same data structures. Given the contaminated train sets, we retrain the LSTM models and obtain their root mean square error (RMSE) on the uncontaminated test data. We fix the contaminating process parameters and repeat the experiment 100 times, the mean and the standard deviation of the RMSEs are reported in Table 3. We observe that the **SIF** values increase with the RMSEs, and the maximum **SIF** and RMSE correspond to the same contaminating process, ARMA.

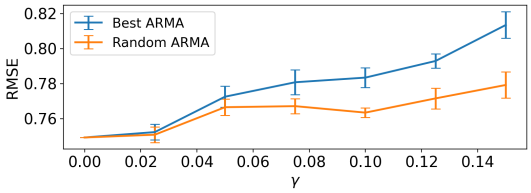


Figure 3: The *Best ARMA* model achieves the largest **SIF** value and RMSE on the uncontaminated test set.

Following the above experiments, we illustrate using the **SIF** values to seek the optimal ARMA parameters that can contaminate the core process with the most adversarial influence on future predictions and raise no suspicion. To this end, we determine the optimal ARMA coefficients by maximizing the $\mathbf{SIF}^2$ under the constraint of the stationary triangle. We solve the constrained optimization problem using SLSQP (Kraft 1988), a standard package available in Python. To avoid suspicion, during each optimization iteration, the generated contaminating time series is scaled to have the same mean and standard deviation as the core process. Finally, we obtain coefficients $[(0.563, 0.437), (5.55, 1.829)]$. To validate this set of ARMA coefficients is indeed lead to the most influential ACP with respect to the LSTM predictive model, we randomly select the coefficients as before and repeat 100 times. For the 100 sets of contaminating process parameters, again we obtain their corresponding RMSEs in the test set and report the mean and standard deviation over a range of γ value in Fig. 3, where the *Best ARMA* indicates the ARMA model with the optimal coefficients. Fig. 3 shows that the RMSE of the *Best ARMA* is consistently larger than that of the randomly selected coefficients on average.

All the experiments in this work are done on a Macbook Pro with Intel(R) Core(TM) i7 CPU and 16 GB memory. The code is written under Python 3.6 with TensorFlow 1.12.

Conclusion

In this paper, we study recurring outliers in time series data and aim to provide a systematic way of measuring the impact of such outliers on time series analysis. To this end, we use the contaminated process to model the input time series. At each timestamp, the observation has a small probability of coming from the contaminating process, i.e., the outliers. Then we introduce the influence functional from robust statistics to quantify the impact of the contaminating process on the parameter estimation. For the sake of outlier interpretation and evaluation of existing outlier detection methods, we further propose a single-valued metric named the **SIF** to characterize the impact of the contaminating process on future predictions, and analyze its properties from various aspects. Notice that the proposed techniques can be naturally extended to multivariate time series analysis. Experimental results demonstrate the proposed approach from various aspects, especially for the use of evaluating existing outlier detection methods and crafting ACP.

Acknowledgements

This work is supported by National Science Foundation under Award No. IIS-1947203 and IIS-2002540, and IBM-ILLINOIS Center for Cognitive Computing Systems Research (C3SR) - a research collaboration as part of the IBM AI Horizons Network. The views and conclusions are those of the authors and should not be interpreted as representing the official policies of the funding agencies or the government.

References

Adams, R. P.; and MacKay, D. J. 2007. Bayesian Online Changepoint Detection. *arXiv:0710.3742* .

Agarwal, N.; Bullins, B.; and Hazan, E. 2017. Second-Order Stochastic Optimization for Machine Learning in Linear Time. *Journal of Machine Learning Research* 18: 116:1–116:40.

Aggarwal, C. C.; and Yu, P. S. 2001. Outlier detection for high dimensional data. *ACM Sigmod Record* 30: 37–46.

Anava, O.; Hazan, E.; and Zeevi, A. 2015. Online Time Series Prediction with Missing Data. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, 2191–2199.

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

Chandola, V.; Banerjee, A.; and Kumar, V. 2008. Understanding categorical similarity measures for outlier detection. Technical Report 08-008, University of Minnesota.

Chen, X. D.; Liu, W.; and Zhang, Y. 2018. First-order Newton-type Estimator for Distributed Estimation and Inference. *ArXiv* abs/1811.11368.

Cook, R. D.; and Weisberg, S. 1982. *Residual and influence in regression*. New York: Chapman and Hall.

Cortes, D. 2020. Explainable outlier detection through decision tree conditioning.

Fox, A. J. 1972. Outliers in Time Series. *Journal of the Royal Statistical Society. Series B* 34(3): 350–363.

González, F. A.; and Dasgupta, D. 2003. Anomaly Detection Using Real-Valued Negative Selection. *Genetic Programming and Evolvable Machines* 4(4): 383–403.

Goodfellow, I.; Bengio, Y.; and Courville, A. 2016. *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

Habler, E.; and Shabtai, A. 2017. Using LSTM Encoder-Decoder Algorithm for Detecting Anomalous ADS-B Messages. *CoRR* abs/1711.10192.

Hampel, F. R. 1974. The Influence Curve and Its Role in Robust Estimation. *Journal of the American Statistical Association* 69(346): 383–393.

Han, J.; Kamber, M.; and Pei, J. 2011. *Data Mining: Concepts and Techniques, 3rd Edition*. Morgan Kaufmann. ISBN 9780123814791.

Hauskrecht, M.; Batal, L.; Valko, M.; Visweswaran, S.; Cooper, G. F.; and Clermont, G. 2013. Outlier detection for patient monitoring and alterting. *Journal of Biomedical Informatics* 46(1): 47–55.

He, J.; Liu, Y.; and Lawrence, R. 2008. Graph-based rare category detection. In *IEEE International Conference on Data Mining*, 833–838.

Hochreiter, S.; and Schmidhuber, J. 1997. Long short-term memory. *Neural computation* 9(8): 1735–1780.

Jürgen Franke, Wolfgang Karl Härdle, C. M. H. 2015. *Statistics of Financial Markets*. Springer.

Kauffmann, J.; ller, K.-R. M.; and Montavon, G. 2018. Towards explaining anomalies: a deep Taylor decomposition of one-class models. *arXiv:1805.06230* .

Kauffmann, J.; Müller, K.-R.; and Montavon, G. 2020. Towards explaining anomalies: A deep Taylor decomposition of one-class models. *Pattern Recognition* 101: 107198. ISSN 0031-3203. doi:10.1016/j.patcog.2020.107198. URL http://dx.doi.org/10.1016/j.patcog.2020.107198.

Koh, P. W.; and Liang, P. 2017. Understanding black-box predictions via influence functions. In *Perceeings of the 34th International Confernce on Machine Learning*.

Kraft, D. 1988. A software package for sequential quadratic programming. *Technical Report DFVLR-FB 88-28, Institut für Dynamik der Flugsysteme, Oberpfaffenhofen* .

Kriegel, H.-P.; Schubert, M.; and Zimek, A. 2008. Angle-based Outlier Detection in High-dimensional Data. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Lavin, A.; and Ahmad, S. 2015. Evaluating Real-Time Anomaly Detection Algorithms – The Numenta Anomaly Benchmark. *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)* doi:10.1109/icmla.2015.141. URL http://dx.doi.org/10.1109/ICMLA.2015.141.

Li, T.; Kyrillidis, A.; Liu, L.; and Caramanis, C. 2018. Approximate Newton-based statistical inference using only stochastic gradients. *ArXiv* abs/1805.08920.

Li, X.; Xue, Y.; Chen, Y.; and Malin, B. 2011. Context-Aware Anomaly Detection for Electronic Medical Record Systems. *Healthsec.2011;* .

Li, Z.; Baseman, R. J.; Zhu, Y.; tipu, F. A.; noam slonim; and Shpigelman, L. 2014. A unified framework for outlier detection in trace data analysis. *IEEE Transactions on Semiconductor Manufacturing* 27(1): 95–103.

Liu, N.; Huang, X.; and Hu., X. 2017. Accelerated Local Anomaly Detection via Resolving Attributed Networks. In *IJCAI*.

Liu, N.; Shin, D.; and Hu, X. 2018. Contextual outlier interpretation. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Interlligence*.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2017. Towards deep learning models resistant to adversial attacks. *ArXiv* 1706.06083v3.

Martin, R. D.; and Yohai, V. J. 1986. Influence Functionals for Time Series. *The Annals of Statistics* 14(3): 781–818.

Micenková, B.; Ng, R. T.; Dang, X.-H.; and Assent, I. 2013. Explaining outliers by subspace separability. In *IEEE 13th International Conference on Data Mining*.

Pineda, F. J. 1987. Generalization of Back propagation to Recurrent and Higher Order Neural Networks. In *Neural Information Processing Systems, Denver, Colorado, USA, 1987*, 602–611.

Rebbapragada, U.; Protopapas, P.; Brodley, C. E.; and Alcock, C. 2009. Finding Anomalous Periodic Time Series. *Machine Learning* 74(3): 281–313.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ISBN 978-1-4503-4232-2.

Savage, D.; Zhang, X.; Yu, X.; Chou, P. L.; and Wang, Q. 2014. Anomaly detection in online social networks. *Social Networks* 39: 62–70.

Schneider, M.; Ertel, W.; and Ramos, F. 2016. Expected similarity estimation for large-scale batch and streaming anomaly detection. *Machine Learning* 105(3): 305–333. ISSN 1573-0565. doi:10.1007/s10994-016-5567-7. URL http://dx.doi.org/10.1007/s10994-016-5567-7.

Schwenk, G.; and Bach, S. 2014. Detecting Behavioral and Structural Anomalies in MediaCloud Applications. *CoRR* abs/1409.8035.

Shcherbakov, M.; Brebels, A.; Shcherbakova, N. L.; Kamaev, V. A.; Gerget, O. M.; and Devyatykh, D. 2016. Outlier detection and classification in sensor data streams for proactive decision support systems. In *Journal of Physics Conference Series*, volume 803, 1–10.

Subramaniam, S.; Palpanas, T.; Papadopoulos, D.; Kalogerakia, V.; and Gunopolos, D. 2006. Online outlier detection in sensor data using non-parametric models. In *VLDB*.

Van Looveren, A.; Vacanti, G.; Klaise, J.; and Coca, A. 2020. Alibi-Detect: Algorithms for outlier and adversarial instance detection, concept drift and metrics. URL https://github.com/SeldonIO/alibi-detect.

Wang, Z.; Yang, F.; Ho, D. W. C.; and Liu, X. 2005. Robust finite-horizon filtering for stochastic systems with missing measurements. *IEEE Signal Process. Lett.* 12(6): 437–440.

Xiong, L.; Chen, X.; and Schneider, J. 2011. Direct robust matrix factorization for anomaly detection. In *ICDM*.

Zong, B.; Song, Q.; Min, M. R.; Cheng, W.; Lumezanu, C.; Cho, D.; and Chen, H. 2018. Deep Autoencoding Gaussian Mixture Model for Unsupervised Anomaly Detection. In *International Conference on Learning Representations*.