

Graphical Abstract

Adaptable and Interpretable Framework for Anomaly Detection in Streaming Energy Systems

Marek Wadinger, Michal Kvasnica

Highlights

Adaptable and Interpretable Framework for Anomaly Detection in Streaming Energy Systems

Marek Wadinger, Michal Kvasnica

- Enriches interpretable anomaly detection with adaptive capabilities
- Isolates root cause of anomalies while considering interactions
- Uses self-learning approach on streamed IoT data
- Demonstrates interpretability by providing process limits for signals
- Provides comparable detection accuracy to established general methods

Adaptable and Interpretable Framework for Anomaly Detection in Streaming Energy Systems

Marek Wadinger^{a,*}, Michal Kvasnica^a

^a*Institute of Information Engineering, Automation and Mathematics, Slovak University of Technology in Bratislava, Radlinskeho 9, Bratislava, 812 37, Bratislava, Slovakia*

Abstract

In this paper, we introduce an Adaptable and Interpretable Framework for Anomaly Detection (AID) designed for streaming energy systems utilizing IoT devices. AID leverages adaptive conditional probability distribution modeling to capture the normal operation of dynamic systems and isolate root causes of anomalies. The framework dynamically updates parameter of multivariate Gaussian distribution, allowing it to adapt to non-stationarity. AID interprets anomalies as significant deviations from conditional probability, encompassing interactions as well as both spatial and temporal irregularities by exposing them as features. Additionally, dynamic process limits are drawn to pinpoint root causes. The framework also alerts individual signal as outliers in sampling. Two real-world case studies showcase AID's capabilities. The first study focuses on Battery Energy Storage Systems (BESS), demonstrating AID's effectiveness in capturing system anomalies, providing less conservative signal limits, and leveraging a physical model for temperature anomaly detection. The second case study delves into monitoring temperature profiles of battery modules, where AID successfully identifies hardware faults, emphasizing its importance in energy storage system safety and profitability. A benchmark evaluation on industrial data shows that AID delivers comparable results to other self-learning adaptable anomaly detection methods, with the added advantage of root cause isolation.

Keywords: Anomaly detection, Root cause isolation, Iterative learning control, Statistical learning, IoT

*

Email address: marek.wadinger@stuba.sk (Marek Wadinger)
URL: uiam.sk/~wadinger (Marek Wadinger)

¹ 1. Introduction

² Anomaly detection systems play a critical role in risk-averse systems by
³ identifying abnormal patterns and adapting to novel expected patterns in
⁴ data. These systems are particularly vital in the context of Internet of Things
⁵ (IoT) devices that continuously stream high-fidelity data to control units.

⁶ In this rapidly evolving field with long-spanning roots, Chandola et al.
⁷ conducted an influential review of prior research efforts across diverse appli-
⁸ cation domains (Chandola et al. (2009)). Recent studies have underscored
⁹ the need for holistic and tunable anomaly detection methods accessible to
¹⁰ operators (Laptev et al. (2015); Kejariwal (2015); Cook et al. (2020)).

¹¹ Cook et al. denote substantial aspects that pose challenges to anomaly
¹² detection in IoT, including the temporal, spatial, and external context of
¹³ measurements, multivariate characteristics, noise, and nonstationarity (Cook
¹⁴ et al. (2020)). To address these complexities, Zhang et al. have successfully
¹⁵ employed spatially distributed sensors and time-relative modulation. Their
¹⁶ approach has proven effective, particularly in the context of complex non-
¹⁷ linear systems, offering potential solutions to some of the challenges posed
¹⁸ by IoT data (Zhang et al. (2024)). Huang et al., on the other hand, tackled
¹⁹ the problems of detecting global outliers, local outliers, and outlier clusters
²⁰ simultaneously. Their proposed approach, based on density estimation, relies
²¹ on the notion that density distributions should exhibit minimal variations in
²² local areas. To achieve this, they introduce a novel turning ratio metric,
²³ which reduces reliance on hyperparameters and enhances anomaly detection
²⁴ (Huang et al. (2023)).

²⁵ Additionally, feature engineering techniques play a crucial role in cap-
²⁶ turing contextual properties and enhancing anomaly detection performance
²⁷ (Fan et al. (2019)). However, it's worth noting that feature engineering may
²⁸ introduce categorical variables and significantly increase dimensionality of
²⁹ the data, requiring specific methods for handling large data, sizeable data
³⁰ storage, and substantial computational resources (Talagala et al. (2021)).
³¹ Recently, Li et al. introduced an attribute-weighted outlier detection algo-
³² rithm, designed for high-dimensional datasets with mixtures of categorical
³³ and numerical data. Their approach assigns different weights to individ-
³⁴ ual attributes based on their importance in anomaly detection and uses
³⁵ these weights to calculate distances between data points. Notably, Li et

al. demonstrated the superior performance of their algorithm compared to state-of-the-art methods (Li and Liu (2024)). Another strategy for handling high-dimensional data involves using deep learning methods with synthetic normal data to enhance the detection of outliers with subtle deviations, as proposed in Du et al. (2024).

Nevertheless, the presence of nonstationarity, often stemming from concept drift (a shift in data patterns due to changes in statistical distribution) and change points (permanent alterations in system state), presents a substantial challenge (Salehi and Rashidi (2018)). In practical scenarios, those changes tend to be unpredictable in both their spatial and temporal aspects. Consequently, they require systems with solid outlier rejection capabilities of intelligent tracking algorithms (Barbosa Roa et al. (2019)). This underscores the critical importance of an anomaly detection method's ability to adapt to evolving data structures, especially in long-term deployments. Nevertheless, as (Tartakovsky et al. (2013)) remarked, the immediate detection is not a feasible option unless there is a high tolerance for false alarms.

The former scalability problem now introduces a significant latency in detector adaptation (Wu et al. (2021)). Incremental learning methods allowed adaptation while restraining the storage of the whole dataset. The supervised operator-in-the-loop solution offered by Pannu et al. showed the detector's adaptation to data labeled on the flight (Pannu et al. (2012)). Others approached the problem as sequential processing of bounded data buffers in univariate signals (Ahmad et al. (2017)) and multivariate systems (Bosman et al. (2015)).

1.1. Related Work

Recent advances in anomaly detection have broadened its scope to include root cause identification governed by the development of explanatory methods capable of diagnosing and tracking faults across the system. Studies can be split into two groups of distinct approaches. The first group approaches explainability as the importance of individual features (Carletti et al. (2019); Nguyen et al. (2019); Amarasinghe et al. (2018)). Those studies allow an explanation of novelty by considering features independently. The second group uses statistical learning creating models explainable via probability. For instance, integration of variational Bayesian inference probabilistic graph neural network allowed Zhang et al. to model the posterior distribution of sensor dependency for gas leakage localization on unlabeled

72 data (Zhang et al. (2023)). Yang et al. recently proposed a Bayesian net-
73 work (BN) for fault detection and diagnosis. In this BN, individual nodes
74 of the network represent normally distributed variables, whereas the multi-
75 ple regression model defines weights and relationships. Using the predefined
76 structure of the BN, the authors propose an offline training with online de-
77 tection and diagnosis (Yang et al. (2022)).

78 Given the infrequent occurrence of anomalies and their potential absence
79 in training data, the incorporation of synthetic data or feature extraction for
80 various detected events emerges to assist diagnosis of the system. Brito et al.
81 designed synthetic faults based on expert knowledge and introduced them
82 into a transfer learning classifier to exploit faults in rotating machinery, with
83 a subsequent explanation layer (Brito et al. (2023)). Conversely, We et al.
84 leveraged feature selection to expose various types of abnormal behavior. The
85 team presents competitive performance while using change in relationships
86 to provide causal inference (Wu et al. (2024)).

87 However, it is crucial to underscore that offline training, as previously em-
88 phasized, is inherently inadequate when it comes to adapting to anticipated
89 novel patterns, rendering it unsuitable for sustained, long-term operation on
90 IoT devices.

91 This paper emphasizes the importance of combining adaptability in in-
92 terpretable anomaly detection and proposes a method that addresses this
93 challenge. Here we report the discovery and characterization of an adaptive
94 anomaly detection method for streaming IoT data. The ability to diag-
95 nose multivariate data while providing root cause isolation, inherent in the
96 univariate case, extends our previous contribution to the field as presented
97 in (Wadinger and Kvasnica (2023)). The proposed algorithm represents a
98 general method for a broad range of safety-critical systems where anomaly
99 diagnosis and identification are paramount.

100 *1.2. Novelty of proposed approach*

101 The idea of using statistical outlier detection is well-established. We high-
102 lighting impactful contributions of (Yamanishi and Takeuchi (2002)) and (Ya-
103 manishi et al. (2004)). The authors propose a method for detecting anomalies
104 in a time series. The method is based on the assumption that the continuous
105 data is generated by a mixture of Gaussian distributions, while discrete data
106 is modeled as histogram density. The authors solve the problem of change
107 point detection as well. However, the adaptation system is unaware of such
108 changes, making the moving window the only source of adaptation. Our

109 self-supervised approach offers intelligent adaptation w.r.t. detected change
110 points. Moreover, the author of the study does not attempt to isolate the
111 root cause of the anomaly. We do so by computing the conditional proba-
112 bility of each measurement given the rest of the measurements and drawing
113 limits defining the normal event probability threshold.

114 A limited number of studies have focused on adaptation and interpretabil-
115 ity within the framework of anomaly detection. Two recent contributions
116 in this area are (Steenwinckel (2018)) and (Steenwinckel et al. (2021)). In
117 (Steenwinckel (2018)), the authors emphasize the importance of combining
118 prior knowledge with a data-driven approach to achieve interpretability, par-
119 ticularly concerning root cause isolation. They propose a novel approach
120 that involves extracting features based on knowledge graph pattern extrac-
121 tion and integrating them into the anomaly detection mechanism. This graph
122 is subsequently transformed into a matrix, and adaptive region-of-interest ex-
123 traction is performed using reinforcement learning techniques. To enhance
124 interpretability, a Generative Adversarial Network (GAN) reconstructs a new
125 graphical representation based on selected vectors. However, it's important
126 to note that the validation of this idealized approach is pending further in-
127 vestigation. Lately, (Steenwinckel et al. (2021)) introduced a comprehen-
128 sive framework for adaptive anomaly detection and root cause analysis in
129 data streams. While the adaptation process is driven by user feedback, the
130 specific mechanism remains undisclosed. The authors present an interpreta-
131 tion of their method through a user dashboard, featuring visualizations of
132 raw data. This dashboard is capable of distinguishing between track-related
133 problems and train-related issues, based on whether multiple trains at the
134 same geographical location approach the anomaly. Meanwhile, our attempts
135 aim to develop a self-supervised method capable of learning without human
136 supervision which is often limited in time and poses significant delays in
137 adaptation, while interpretation offers straightforward statistical reasoning
138 and root cause isolation.

139 *1.3. Validation*

140 Two case studies show that our proposed method, based on dynamic joint
141 normal distribution, has the capacity to explain novelties, isolate the root
142 cause of anomalies, and allow adaptation to change points, advancing recently
143 developed anomaly detection techniques for long-term deployment and cross-
144 domain usage. We observe similar detection performance, albeit with lower
145 scalability, when comparing our approach to well-established unsupervised

146 anomaly detection methods in streamed data which create a bedrock for
147 many state-of-the-art contributions, such as One-Class SVM (Amer et al.
148 (2013); Liu et al. (2014); Krawczyk and Woźniak (2015); Miao et al. (2019);
149 Gözüaçık and Can (2021)), and Half-Space Trees (Wetzig et al. (2019); Lyu
150 et al. (2020)).

151 *1.4. Broader Impact*

152 Potential applications of the proposed method are in the field of energy
153 storage systems, where the ability to detect anomalies and isolate their root
154 cause, whilst adapting to changes in operation and environment, is crucial
155 for the safety of the system. The proposed method is suitable for the existing
156 infrastructure of the system, allowing detection and diagnosis of the system
157 based on existing data streams. The dynamic process limits allow opera-
158 tional metrics monitoring, making potential early detection and prevention
159 easier. Using adaptable methods without interpretability, on the other hand,
160 may pose safety risks and lower total financial benefits, as the triggered false
161 alarms may need to be thoroughly analyzed, resulting in prolonged down-
162 times.

163 The main contribution of the proposed solution to the developed body of
164 research is that it:

- 165 • Enriches interpretable anomaly detection with adaptive capabilities
- 166 • Isolates root cause of anomalies while considering interactions
- 167 • Uses self-learning approach on streamed IoT data
- 168 • Demonstrates interpretability by providing process limits for signals
- 169 • Utilizes existing IT infrastructure

170 Combines interpretability with adaptation to change points in single anomaly
171 detection system • Isolates root cause of anomalies considering relationships
172 between fea- tures • Demonstrates interpretability by providing process lim-
173 its for each fea- ture • Provides comparable detection accuracy to established
174 general methods

175 *1.5. Paper Organization*

176 The rest of the paper is structured as follows: We begin with the prob-
177 lem and motivation in **Section 1**, providing context. Next, in **Section 2**,
178 we lay the theoretical groundwork. Our proposed adaptive anomaly de-
179 tected method is detailed in **Section 3**. We then demonstrate real-world
180 applications in **Section 4**. Finally, we conclude the paper in **Section 5**,
181 summarizing findings and discussing future research directions.

182 **2. Preliminaries**

183 In this section, we present the fundamental ideas that form the basis
184 of the developed approach. Subsection 2.1 explains Welford’s online algo-
185 rithm, which can adjust distribution to changes in real-time. Subsection 2.2
186 proposes a two-pass implementation that can reverse the impact of expired
187 samples. The math behind distribution modeling in Subsection 2.3 estab-
188 lishes the foundation for the Gaussian anomaly detection model discussed in
189 Subsection 2.5, followed by conditional probability computation in Subsec-
190 tion 2.4. The last subsection of the preliminaries is devoted to the definition
191 of anomalies.

192 *2.1. Welford’s Online Algorithm*

193 Welford introduced a numerically stable online algorithm for calculating
194 mean and variance in a single pass. The algorithm allows the processing
195 of IoT device measurements without the need to store their values Welford
196 (1962).

197 Given measurement x_i where $i = 1, \dots, n$ is a sample index in sample
198 population n , the corrected sum of squares S_n is defined as

$$S_n = \sum_{i=1}^n (x_i - \bar{x}_n)^2, \quad (1)$$

199 with the running mean \bar{x}_n defined as previous mean \bar{x}_{n-1} weighted by pro-
200 portion of previously seen population $n - 1$ corrected by current sample as

$$\bar{x}_n = \frac{n-1}{n} \bar{x}_{n-1} + \frac{1}{n} x_n = \bar{x}_{n-1} + \frac{x_n - \bar{x}_{n-1}}{n}. \quad (2)$$

201 Throughout this paper, we consider the following formulation of an update
202 to the corrected sum of squares:

$$S_n = S_{n-1} + (x_n - \bar{x}_{n-1})(x_n - \bar{x}_n), \quad (3)$$

203 as it is less prone to numerical instability due to catastrophic cancellation.

204 Finally, the corresponding unbiased variance is

$$s_n^2 = \frac{S_n}{n - 1}. \quad (4)$$

205 This implementation of the Welford method requires the storage of three
206 scalars: \bar{x}_{n-1} ; n ; S_n .

207 2.2. Inverse Welford's Algorithm

208 Based on (2), it is clear that the influence of the latest sample over the
209 running mean decreases as the population n grows. For this reason, regulating
210 the number of samples used for sample mean and variance computation
211 has crucial importance over adaptation. Given access to the instances used
212 for computation and expiration period $t_e \in \mathbb{N}_0^{n-1}$, reverting the impact of
213 x_{n-t_e} can be written as follows

$$S_{n-1} = S_n - (x_{n-t_e} - \bar{x}_{n-1})(x_{n-t_e} - \bar{x}_n), \quad (5)$$

214 where the reverted mean is given as

$$\bar{x}_{n-1} = \frac{n}{n-1}\bar{x}_n - \frac{1}{n-1}x_{n-t_e} = \bar{x}_n - \frac{x_{n-t_e} - \bar{x}_n}{n-1}. \quad (6)$$

215 Finally, the unbiased variance follows the formula:

$$s_{n-1}^2 = \frac{S_{n-1}}{n-2}. \quad (7)$$

216 2.3. Statistical Model of Multivariate System

217 Multivariate normal distribution generalizes the multivariate systems to
218 the model where the degree to which variables are related is represented by
219 the covariance matrix. Gaussian normal distribution of variables is a reasonable
220 assumption for process measurements, as it is a common distribution
221 that arises from stable physical processes measured with noise. The general
222 notation of multivariate normal distribution is:

$$\mathbf{X} \sim \mathcal{N}_k(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \quad (8)$$

223 where k -dimensional mean vector is denoted as $\boldsymbol{\mu} = (\bar{x}_1, \dots, \bar{x}_k)^T \in \mathbb{R}^k$
224 and $\boldsymbol{\Sigma} \in \mathbb{R}^{k \times k}$ is the $k \times k$ covariance matrix, where k is the index of last
225 random variable.

226 The probability density function (PDF) $f(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ of multivariate normal
 227 distribution is denoted as:

$$f(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{k/2} |\boldsymbol{\Sigma}|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x}-\boldsymbol{\mu})}, \quad (9)$$

228 where \mathbf{x} is a k -dimensional vector of measurements x_i at time i , $|\boldsymbol{\Sigma}|$
 229 denotes the determinant of $\boldsymbol{\Sigma}$, and $\boldsymbol{\Sigma}^{-1}$ is the inverse of $\boldsymbol{\Sigma}$.

230 The cumulative distribution function (CDF) of a multivariate Gaussian
 231 distribution describes the probability that all components of the random ma-
 232 trix \mathbf{X} take on a value less than or equal to a particular point \mathbf{x} in space,
 233 and can be used to evaluate the likelihood of observing a particular set of
 234 measurements or data points. The CDF is often used in statistical applica-
 235 tions to calculate confidence intervals, perform hypothesis tests, and make
 236 predictions based on observed data. In other words, it gives the probability
 237 of observing a random vector that falls within a certain region of space. The
 238 standard notation of CDF is as follows:

$$F(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \int_{-\infty}^{\mathbf{x}} f(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) d\mathbf{x}, \quad (10)$$

239 where $d\mathbf{x}$ denotes the integration over all k dimensions of \mathbf{x} .

240 As the equation (10) cannot be integrated explicitly, an algorithm for
 241 numerical computation was proposed in Genz (2000).

242 Given the PDF, we can also determine the value of \mathbf{x} that corresponds to a
 243 given quantile q using a numerical method for inversion of CDF (ICDF) often
 244 denoted as percent point function (PPF) or $F(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})^{-1}$. An algorithm that
 245 calculates the value of the PPF for univariate normal distribution is reported
 246 below as Algorithm 1.

Algorithm 1 Percent-Point Function for Normal Distribution

Input: quantile q , sample mean \bar{x}_n (2), sample variance s_n^2 (4)

Output: threshold value $\tilde{x}_{q,n}$

Initialisation :

1: $f \leftarrow 10; l \leftarrow -f; r \leftarrow f;$

LOOP Process

2: **while** $F(l; \bar{x}_n, s_n^2) > 0$ **do**

3: $r \leftarrow l;$

4: $l \leftarrow lf;$

5: **end while**

6: **while** $F_X(r) - q < 0$ **do**

7: $l \leftarrow r;$

8: $r \leftarrow rf;$

9: **end while**

10: $\tilde{x}_{q,n} = \arg \min_{x_n} \|F(x_n; \bar{x}_n, s_n^2) - q\|$ s.t. $l \leq x_n \leq r$

11: **return** $\tilde{x}_{q,n} \sqrt{s_n^2 + \bar{x}_n}$

247 The Algorithm 1 for PPF computation is solved using an iterative root-
248 finding algorithm such as Brent's method Brent (1972).

249 *2.4. Conditional Probability Distribution*

250 Considering that we observe particular vector \mathbf{x}_i , we can update probability
251 distributions, calculated according to the rules of conditional probability,
252 of individual measurements within the vector given the rest of the measurements
253 in \mathbf{x}_i . Let's assume multivariate normal distribution (8) and without loss of generality, that the vector \mathbf{x}_i can be partitioned into subset variable
254 x_a , and complement vector \mathbf{x}_b as follows

$$\mathbf{x}_i = \begin{bmatrix} x_a \\ \mathbf{x}_b \end{bmatrix} \text{ with dimensions } \begin{bmatrix} 1 \times 1 \\ (k-1) \times 1 \end{bmatrix}, \quad (11)$$

256 where $a = 1, \dots, k$ and $\mathbf{b} = \{1, 2, \dots, k\}$ where $a \notin \mathbf{b}$. This partitioning
257 allows us to define block-wise mean and covariance as follows:

$$\boldsymbol{\mu} = \begin{bmatrix} \mu_a \\ \boldsymbol{\mu}_b \end{bmatrix} \text{ with dimensions } \begin{bmatrix} 1 \times 1 \\ (k-1) \times 1 \end{bmatrix}, \quad (12)$$

258 and

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma_{aa}^2 & \boldsymbol{\Sigma}_{ab} \\ \boldsymbol{\Sigma}_{ba} & \boldsymbol{\Sigma}_{bb} \end{bmatrix} \text{ with dimensions } \begin{bmatrix} 1 \times 1 & 1 \times (k-1) \\ (k-1) \times 1 & (k-1) \times (k-1) \end{bmatrix}. \quad (13)$$

259 where a and \mathbf{b} represent distinct components within the vector.

260 Subsequently, we can derive the conditional distribution of any subset
261 variable x_a , given the complementary vector \mathbf{x}_b . This conditional distribution
262 conforms to a univariate normal distribution, characterized by:

$$X_a | \mathbf{X}_b \sim \mathcal{N}(\mu_{a|\mathbf{b}}, \sigma_{a|\mathbf{b}}^2). \quad (14)$$

263 where $\mu_{a|\mathbf{b}}$ denotes the conditional mean and $\sigma_{a|\mathbf{b}}^2$ represents the condi-
264 tional variance. These crucial parameters can be computed using the Schur
265 complement.

266 For a general matrix M expressed as:

$$M = \begin{bmatrix} A & B \\ C & D \end{bmatrix}, \quad (15)$$

267 the Schur complement of the block matrix M is denoted as:

$$M | D = A - BD^{-1}C. \quad (16)$$

268 Applying Equation (16), we can calculate the conditional variance $\sigma_{a|\mathbf{b}}^2$
269 using the covariance matrix notation from Equation (13) as follows:

$$\sigma_{a|\mathbf{b}}^2 = \sigma_{aa}^2 - \Sigma_{ab}\Sigma_{bb}^{-1}\Sigma_{ba}, \quad (17)$$

270 while the conditional mean, denoted as $\mu_{a|\mathbf{b}}$, is determined by:

$$\mu_{a|\mathbf{b}} = \mu_a + \Sigma_{ab}\Sigma_{bb}^{-1}(\mathbf{x}_b - \boldsymbol{\mu}_b). \quad (18)$$

271 It is important to note that Σ_{bb}^{-1} in Equations (17) and (18) signifies
272 the inverse of the covariance matrix Σ_{bb} . Thus, the conditional variance
273 $\sigma_{a|\mathbf{b}}^2$ essentially represents the Schur complement of Σ_{bb} within the overall
274 covariance matrix Σ .

275 2.5. Gaussian Anomaly Detection

276 From a viewpoint of statistics, outliers are commonly denoted as values
277 that significantly deviate from the mean. Under the assumption that the
278 spatial and temporal characteristics of a system, observed over a moving
279 window, can be suitably represented as normally distributed features, we
280 assert that any anomaly can be identified as an outlier.

281 From a statistical viewpoint, outliers can be denoted as values that sig-
282 nificantly deviate from the mean. Assuming that the spatial and temporal

characteristics of the system over the moving window can be encoded as normally distributed features, we can claim, that any anomaly may be detected as an outlier.

In empirical fields like machine learning, the three-sigma rule (3σ) provides a framework for characterizing the region of a distribution within which normal values are expected to fall with high confidence. This rule renders approximately 0.265% of values in the distribution as anomalous.

The 3σ rule establishes the probability that any sample x_a of a random vector X lies within a given CDF over a semi-closed interval as the distance from the conditional mean $\mu_{a|\mathbf{b}}$ of 3 conditional variances $\sigma_{a|\mathbf{b}}^2$ and gives an approximate value of q as

$$q = P\{|x_a - \mu_{a|\mathbf{b}}| < 3\sigma_{a|\mathbf{b}}^2\} = 0.99735. \quad (19)$$

Utilizing a probabilistic model of normal behavior, we can determine threshold values x_l and x_u corresponding to the closed interval of the CDF where this probability is established. The inversion of Equation (10) facilitates this calculation, yielding:

$$x_l = F((1 - P\{|x_a - \mu_{a|\mathbf{b}}| < 3\sigma_{a|\mathbf{b}}^2\}); \mu_{a|\mathbf{b}}, \sigma_{a|\mathbf{b}}^2)^{-1}, \quad (20)$$

for the lower limit, and

$$x_u = F((P\{|x_a - \mu_{a|\mathbf{b}}| < 3\sigma_{a|\mathbf{b}}^2\}); \mu_{a|\mathbf{b}}, \sigma_{a|\mathbf{b}}^2)^{-1}, \quad (21)$$

for the upper limit. These lower and upper limits together form vectors \mathbf{x}_l and \mathbf{x}_u , respectively, defining the region of normal system operation. This region is conceptualized as a hypercube in the feature space, with each dimension bounded by the corresponding feature limits, as computed using Equations (20) and (21) for all $a = 1, \dots, k$; $\mathbf{b} = \{1, 2, \dots, k\}$ where $a \notin \mathbf{b}$.

Such threshold is computed for each feature of the system, resulting in a vector of lower and upper limits. The vector of limits is used to define the region of normal operation of the system. The region is defined as a hypercube in the feature space, where each dimension is defined by the limits of the corresponding feature.

The predicted state of the system, denoted as y_i , and the presence of anomalies in signals $\mathbf{y}_{s,i}$ at time i are determined based on the maximum distance of observations from the center of the hypercube. The center of the hypercube corresponds to the vector of conditional means $\mu_{a|\mathbf{b}}$ with respect

313 to other features. The calculation of this distance involves the cumulative
 314 distribution function (CDF) of observations and conditional distributions, as
 315 follows:

$$F(x_a; \mu_{a|\mathbf{b}}, \sigma_{a|\mathbf{b}}^2) : a = 1, \dots, k; \mathbf{b} = \{1, 2, \dots, k\} \text{ where } a \notin \mathbf{b}. \quad (22)$$

316 Subsequently, anomalies in individual inputs are defined as follows:

$$\mathbf{y}_{s,i} = \begin{cases} 0 & \text{if } T \leq (22) \\ 1 & \text{if } T > (22), \end{cases} \quad (23)$$

317 where T represents a threshold that distinguishes between normal signal
 318 measurement ($\mathbf{y}_{s,i} = 0$) and abnormal ($\mathbf{y}_{s,i} = 1$).

319 For the predicted state of the system, the maximum value from $\mathbf{y}_{s,i}$ is
 320 considered, resulting in:

$$y_i = \begin{cases} 1 & \text{if } 1 \in \mathbf{y}_{s,i} \\ 0 & \text{otherwise,} \end{cases} \quad (24)$$

321 defining the discrimination boundary between system operation where
 322 $y_i = 0$ indicates normal system operation, and $y_i = 1$ indicates anomalous
 323 operation.

324 2.6. Anomaly Definition

325 In the realm of data analysis, anomalies are conspicuous deviations from
 326 the anticipated patterns within a dataset. Traditionally, the task of anomaly
 327 detection has relied upon unsupervised methodologies, wherein the identifi-
 328 cation of "outliers" entails the comparison of data points in both temporal
 329 and spatial contexts. This approach, often referred to as point-wise anomaly
 330 detection, classifies a data point as an anomaly when it exhibits significant
 331 dissimilarity from its neighboring data points (Iglesias Vázquez et al. (2023)).

332 The concept of point anomalies, influenced by factors such as temporal
 333 and spatial aspects, can be further categorized into conditional and contex-
 334 tual anomalies (Ruff et al. (2021)).

335 Nevertheless, this conventional method may not be suitable for scenarios
 336 characterized by collective anomalies, where clusters of abnormal data points
 337 coexist. A more pragmatic approach defines anomalies as deviations from
 338 established "normal" patterns, resembling the principles of semi-supervised
 339 learning. Change point detection, in a similar vein, can be regarded as a

340 relative approach that takes into account the varying dynamics of changes,
341 whether they occur gradually or abruptly (Iglesias Vázquez et al. (2023)).

342 It is imperative to recognize that the interpretation of anomalies, outliers,
343 and novelties can vary upon the application. Anomalies typically garner
344 significant attention, while outliers are often treated as undesirable noise
345 and are typically excluded during data preprocessing. Novelties, on the other
346 hand, signify new observations that necessitate model updates to adapt to
347 an evolving environment (Ruff et al. (2021)).

348 Notwithstanding the differences in terminology, methods employed for the
349 identification of data points residing in low-probability regions, irrespective
350 of whether they are referred to as "anomaly detection," "outlier detection,"
351 or "novelty detection," share fundamental similarities (Iglesias Vázquez et al.
352 (2023)).

353 3. Novelty Detection and Interpretation Framework

354 In this section, we propose an adaptive and interpretable detection frame-
355 work for multivariate systems with streaming IoT devices. This approach
356 models the system as a dynamic joint normal distribution, enabling it to ef-
357 fectively adapt to pervasive nonstationary effects on processes. Our method
358 handles various factors, including change points, concept drift, and seasonal
359 effects. Our primary contribution lies in the fusion of an adaptable self-
360 supervised system with root cause identification capabilities. This combi-
361 nation empowers the online statistical model to diagnose anomalies through
362 two distinct avenues. Firstly, it employs conditional probability calculations
363 to assess the system's operating conditions' normality. Secondly, it identifies
364 outliers within individual signal measurements and features based on dy-
365 namic alert-triggering process limits. In the following sections, we describe
366 our proposed methodology across three subsections. The initial subsection
367 delves into the process of initializing the model's parameters. The subsequent
368 section describes online training and adaptation, while the final subsection
369 expounds upon the model's detection and diagnostic capabilities. For a con-
370 cise representation of the proposed method, Algorithm 2 is provided.

371 3.1. Model Parameters Initialization

372 The model initialization is governed by defining two tunable hyperparam-
373 eters of the model: the expiration period (t_e) and the threshold (T). The
374 expiration period determines the window size for time-rolling computations,

375 impacting the proportion of outliers within a given timeframe, and directly
376 influencing the relaxation (with a longer expiration period) or tightening
377 (with a shorter expiration period) of dynamic signal limits. Additionally, we
378 introduce a grace period, which defaults to $3/4t_e$, allowing for model calibra-
379 tion. During this grace period, system anomalies are not flagged to prevent
380 false positives and speed up self-supervised learning in Subsection 3.2. The
381 length of the expiration period inversely correlates with the model’s ability
382 to adapt to sudden changes. The adaptation to shifts in the data-generating
383 process, such as changes in mean or variance, is managed through the adapta-
384 tion period t_a . A longer t_a results in slower adaptation but potentially longer
385 alerts, which can be valuable during extended outlier periods. In most cases,
386 $t_a = 1/4t_e$ offers optimal performance.

387 As a general rule of thumb, expiration period t_e should be determined
388 based on the slowest observed dynamics within the multivariate system. The
389 threshold T defaults to the three-sigma probability of q in (19). Adjusting
390 this threshold can fine-tune the trade-off between precision and recall. A
391 higher threshold boosts recall but may lower precision, while a lower thresh-
392 old enhances precision at the cost of recall. The presence of one non-default
393 easily interpretable hyperparameter facilitates adaptability to various sce-
394 narios. We recommend starting with the default values of other parameters
395 and making adjustments based on real-time model performance.

396 *3.2. Online training*

397 Training in RAID follows an incremental learning approach, processing
398 each new sample upon arrival. Incremental learning allows online parame-
399 ter updates, albeit with a potential computational delay affecting response
400 latency.

401 In the case of a dynamic joint probability distribution, the parameters are
402 μ_i and Σ_i at time instance i . Update of the mean vector μ_i and covariance
403 matrix Σ_i is governed by Welford’s online algorithm using equation (2) and
404 (4) respectively. Samples beyond the expiration period t_e are disregarded
405 during the second pass. The effect of expired samples is reverted using inverse
406 Welford’s algorithm for mean (6) and variance (7), accessing the data in the
407 buffer. For details, refer to Subsection 2.2.

408 It’s worth noting that adaptation relies on two self-supervised methods.
409 Adaptation routine runs if the observation at time instance i is considered
410 normal. Adaptation period t_a allows the model to update the distribution
411 on outliers as well. Given the predicted system anomaly state from (24) as

412 y_i over the window of past observations $\mathbf{y}_i = \{y_{i-t_a}, \dots, y_i\}$, the following test
413 holds when adaptation is performed on outlier:

$$\frac{\sum_{y \in \mathbf{y}_i} y}{n(\mathbf{y}_i)} > 2 * (T - 0.5). \quad (25)$$

414 Here $n(\mathbf{y}_i)$ denotes the dimensionality of \mathbf{y}_i . The logic of the (25) follows the
415 probabilistic approach to anomalies that assumes a number of anomalies are
416 lower or equal to the conditional probability at both tails of the distribution

417 *3.3. Online prediction*

418 In the prediction phase, multiple metrics are evaluated to assess the state
419 of the modeled system.

420 Firstly, we calculate the parameters of the conditional distribution con-
421 cerning the dynamic multivariate Gaussian distribution. These calculations
422 are performed for the process observation vector \mathbf{x}_i at time instance i . Specif-
423 ically, we compute the conditional mean using (18) and the conditional vari-
424 ance using (17). These computations yield univariate conditional distri-
425 butions for individual signals and features. These conditional distributions play
426 a crucial role in assessing the abnormality of signals and features concerning
427 other observed values. This assessment relies on the strength of relation-
428 ships defined by the covariance matrix of the dynamic multivariate Gaussian
429 distribution. Consequently, our approach inherently considers the interac-
430 tions between input signals and features. The determination of anomalous
431 behavior is governed by (23).

432 Any anomaly detected within one of the features triggers an alert at the
433 system level. The decision regarding the overall system's anomalous behavior
434 is guided by (24). Nevertheless, individual determinations of anomalies serve
435 as a diagnostic tool for isolating the root cause of anomalies.

436 To assist operators in their assessments, we establish a hypercube defined
437 by lower and upper threshold values, denoted as \mathbf{x}_l and \mathbf{x}_u , respectively.
438 These thresholds are derived from (20) and (21), incorporating updated
439 model parameters. Lower and upper thresholds play a pivotal role as dy-
440 namic process limits. They replace the conservative process limits provided
441 in sensor documentation, accounting for factors such as aging and actual
442 environmental conditions that influence sensor operation.

443 Our framework anticipates unexpected novel behavior, including signal
444 loss. This anticipation involves calculating the cumulative distribution func-
445 tion (CDF) over the univariate normal distribution of sampling, focusing

446 on the differences between subsequent timestamps. We operate under the
447 assumption that, over the long term, the distribution of sampling times re-
448 mains stable. As a result, we employ a one-pass update mechanism utiliz-
449 ing (2) and (4). To proactively detect subtle changes in sampling patterns,
450 self-supervised learning is employed, leveraging anomalies weighted by the
451 deviation from $(1 - F(x_i; \mu, \sigma^2))$ for training.

452 The system is vigilant in identifying change points. When the adaptation
453 test specified in (25) is satisfied, change points are flagged and isolated. This
454 initiation of change points triggers updates to the model, ensuring it adapts
455 to evolving data patterns effectively.

Algorithm 2 Online Detection and Identification Workflow

Input: expiration period t_e

Output: system anomaly y_i , signal anomalies $\mathbf{y}_{s,i}$, sampling anomaly $y_{t,i}$, change-point $y_{c,i}$, lower thresholds $\mathbf{x}_{l,i}$, upper thresholds $\mathbf{x}_{u,i}$,

Initialisation :

- 1: $i \leftarrow 1; n \leftarrow 1; T \leftarrow (19); \boldsymbol{\mu} \leftarrow \mathbf{x}_0; \boldsymbol{\Sigma} \leftarrow \mathbf{1}_{k \times k}; \mu_t \leftarrow 0; \sigma_t^2 \leftarrow 1;$
 - 2: compute $F(\mathbf{x}_0; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ using algorithm in Genz (2000);
LOOP Process
 - 3: **loop**
 - 4: $\mathbf{x}_i, t_i \leftarrow \text{RECEIVE}();$
 - 5: $\mathbf{y}_{s,i} \leftarrow \text{PREDICT}(\mathbf{x}_i, T)$ using (23);
 - 6: $y_i \leftarrow \text{PREDICT}(\mathbf{y}_{s,i})$ using (24);
 - 7: $\mathbf{x}_{l,i}, \mathbf{x}_{u,i} \leftarrow \text{GET}(T, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ using Algorithm 1;
 - 8: $y_{t,i} \leftarrow \text{PREDICT}(t_i - t_{i-1})$ using (23);
 - 9: $\mu_t, \sigma_t^2 \leftarrow \text{UPDATE}(t_i - t_{i-1}, \mu_t, \sigma_t^2)$ using (2), (4);
 - 10: **if** not (24) **or** (25) **then**
 - 11: $\boldsymbol{\mu}, \boldsymbol{\Sigma} \leftarrow \text{UPDATE}(\mathbf{x}_i, \boldsymbol{\mu}, \boldsymbol{\Sigma}, n)$ using (2), (4);
 - 12: **if** (25) **then**
 - 13: $y_{c,i} \leftarrow 1;$
 - 14: **else**
 - 15: $y_{c,i} \leftarrow 0;$
 - 16: **end if**
 - 17: $n \leftarrow n + 1;$
 - 18: **for** \mathbf{x}_{i-t_e} **do**
 - 19: $\boldsymbol{\mu}, \boldsymbol{\Sigma} \leftarrow \text{REVERT}(\mathbf{x}_{i-t_e}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, n)$ using (6), (7);
 - 20: $n \leftarrow n - 1;$
 - 21: **end for**
 - 22: **end if**
 - 23: $i \leftarrow i + 1;$
 - 24: **end loop**
-

456 **4. Case Study**

457 This section provides a benchmark and two case studies that showcase
458 the effectiveness and applicability of our proposed approach. In the following
459 Subsections, we investigate the properties and performance of the approach
460 using streamed benchmark system data and signals from IoT devices in a mi-

461 crogrid system. The successful deployment demonstrates that this approach
462 is suitable for existing process automation infrastructure.

463 The case studies were realized using Python 3.10.1 on a machine employ-
464 ing an 8-core Apple M1 CPU and 8 GB RAM.

465 *4.1. Benchmark*

466 In this subsection, we compare the proposed method with adaptive un-
467 supervised detection methods without an interpretability layer. Two of the
468 well-established methods, providing iterative learning capabilities over mul-
469 tivariate time-series data are One-Class Support Vector Machine (OC-SVM)
470 and Half Spaced Trees (HS-Trees). Both methods represent the backbone
471 of multiple state-of-the-art methods for cases of anomaly detection on dy-
472 namic system data, with a brief list of recent applications in Introduction
473 1.3. Comparison is conducted on real benchmarking data, annotated with
474 labels of whether the observation was anomalous or normal. The dataset of
475 Skoltech Anomaly Benchmark (SKAB) Katser and Kozitsin (2020) is used for
476 this purpose, as no established benchmarking multivariate data were found
477 regarding energy storage systems. It represents a combination of experiments
478 with the behavior of rotor imbalance as a subject to various functions intro-
479 duced to control action as well as slow and sudden changes in the amount
480 of water in the circuit. The system is described by 8 features. The data
481 were preprocessed according to best practices for the given method, namely:
482 standard scaling for OC-SVM, normalization for HS-Trees, and no scaling
483 for our proposed method. The optimal quantile threshold value for both
484 reference methods is found using Bayesian Optimization. Due to no further
485 knowledge about the process, the parameters of the proposed method were
486 optimized using Bayesian Optimization as well. Results are provided within
487 Table 1, evaluating F1 score, Recall and Precision. A value of 100% at each
488 metric represents a perfect detection. The latency represents the average
489 computation time per sample of the pipeline including training and data
490 preprocessing.

491 The results in Table 1 suggest, that our algorithm provides slightly better
492 performance than reference methods. Based on the Scoreboard for various
493 algorithms on SKAB’s Kaggle page, our iterative approach performs com-
494 parably to the evaluated batch-trained model. Such a model has all the
495 training data available before prediction unlike ours, evaluating the metrics
496 iteratively on a streamed dataset.

Table 1: Metrics evaluation on SKAB dataset

Metric	AID	OC-SVM	HS-Trees
F1 [%]	48.70	44.42	34.10
Recall [%]	49.90	56.67	32.57
Precision [%]	47.56	36.52	35.77
Avg. Latency [ms]	1.55	0.44	0.21

497 *4.2. Battery Energy Storage System (BESS)*

498 In the first case study, we verify our proposed method on BESS. The
 499 BESS reports measurements of State of Charge (SoC), supply/draw energy
 500 set-points, and inner temperature, at the top, middle, and bottom of the
 501 battery module. Tight battery cell temperature control is needed to optimize
 502 performance and maximize the battery’s lifespan. Identifying anomalous
 503 events and removal of corrupted data might yield significant improvement
 504 in the process control level and increase the reliability and stability of the
 505 system.

506 The default sampling rate of the signal measurement is 1 minute. How-
 507 ever, network communication of the IoT devices is prone to packet dropout,
 508 which results in unexpected non-uniformities in sampling. The data are
 509 normalized to the range $[0, 1]$ to protect the sensitive business value. The
 510 proposed approach is deployed to the existing infrastructure of the system,
 511 allowing real-time detection and diagnosis of the system.

512 The industrial partner provided a physical model of the battery cell tem-
 513 perature, defined as follows:

$$\begin{aligned} T_{\text{bat},i+1} = & T_{\text{bat},i} + T_s(q_{\text{fan}}V_{\text{b,max}}\rho c_p(T_{\text{out}} - T_{\text{bat},i}) + V_{\text{c,max}}q_{\text{circ,fan}}\rho c_p T_{\text{bat},i} \\ & + q_{\text{circ,fan}}(P_{\text{cool}}q_{\text{cool}}P_{\text{heat}}q_{\text{heat}}) + c_{\text{scale}}Q_{\text{bat}} + q_{\text{inner fans}} \\ & - (V_{\text{b,max}}q_{\text{fan}}V_{\text{c,max}}q_{\text{circ,fan}})\rho c_p T_{\text{bat},i})/(m_{\text{bat}}c_{\text{p,b}}) \end{aligned} \quad (26)$$

514 When combined with an averaged measurement of battery cell temper-
 515 ature, we could compute the difference between real and predicted temper-
 516 ature. Such deviation can be useful in detecting unexpected patterns in
 517 temperature. Nevertheless, it may be inaccurate as the physical model is
 518 simplified and does not account for spatial aspects, like temperature gra-
 519 dients as well as different dynamic effects of charging and discharging on

520 temperature. Therefore, the raw measured temperature is used as well. The
521 deviation between demanded power and delivered power was used to aid the
522 identification of the state, as the increased difference might be related to
523 other unexpected and novel patterns.

524 Fig. 1 depicts the operation of the BESS over March 2022. Multiple
525 events of anomalous behavior happened within this period, confirmed by the
526 operators, that are observable through a sudden or significant shift in mea-
527 surements in a given period. As the first step, the detection mechanism was
528 initialized, following the provided guidelines for parameter selection in Sub-
529 section 3.1. The expiration period was set to $t_e = 7$ days, due to the weekly
530 seasonality of human behavior impacting battery usage. The threshold was
531 kept at default value $T = 0.99735$. A grace period, during which the model
532 learns from both normal and anomalous data (though normal are expected),
533 is shortened to 2.5 days to observe the effect of BESS calibration happening
534 on 3rd day from deployment.

535 The deployment and operation of the anomaly detection system were suc-
536 cessful as shown by its adaptation of changepoint on 7th March 2022 that
537 appeared due to the relocation of the battery storage system outdoors. The
538 model was adapted online based on Subsection 3.2. The sudden shift in envi-
539 ronmental conditions, due to the transfer of the system to outside changed the
540 dynamics of the system’s temperature. However, new behavior was adopted
541 by the top-level anomaly isolation system within five days, reducing potential
542 false alerts afterward, by observably shifting the conditional mean to lower
543 temperatures. Perhaps more interesting are the alerted changepoints.

544 Calibration of the BESS, usually observed as deviations of setpoint from
545 real power demand and multiple peaks in temperature was captured as well.

546 The system identified 6 deviations in sampling, denoted by the red bars
547 in Fig. 1. 4 anomalies with shorter duration represented packet loss. The
548 prolonged anomaly was notified during the transfer of the battery pack. The
549 longest dropout observed happened across 20th March up to 21st. Unexpect-
550 edly, the change point detection module triggered an alarm at the end of
551 the loss, resulting in adaptation and a sharp shift in drawn limits for Power
552 Setpoint Deviation. Red dots represent anomalies at the signal level given
553 by equation (23). The dynamic signal limits are surpassed in one or multiple
554 signals during the system’s anomalies. The root cause isolation allows the
555 pairing of anomalies with specific features. Conditional probability, against
556 which the anomalies are evaluated allows consideration of signal relationships
557 within individual limits.

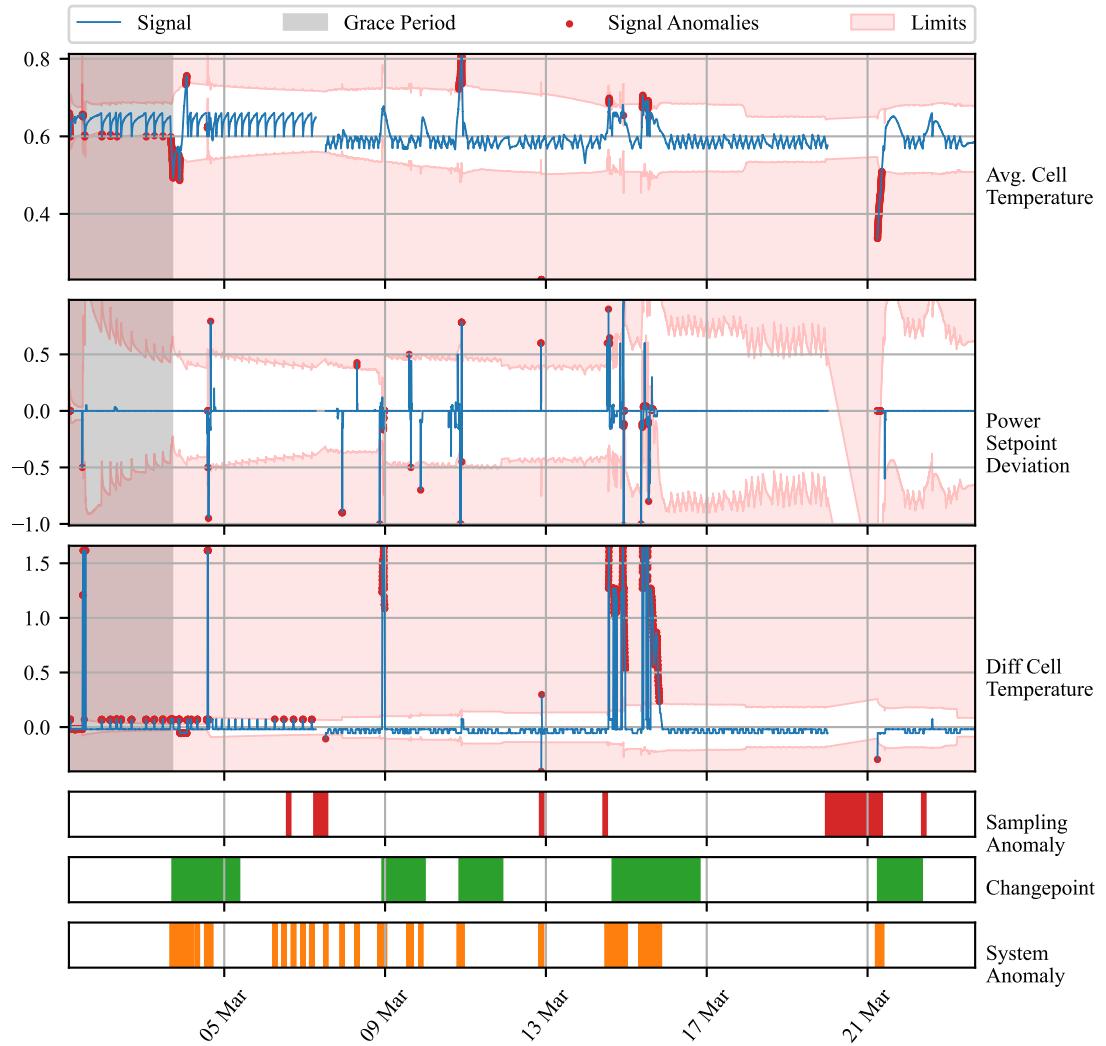


Figure 1: Time Series of BESS measurements (blue line) of process variables. The y-axis renders the values after the normalization of raw inputs. Root causes of anomalies are marked within specific signals as red dots. The light red area represents out-of-limits values for individual signals. Non-uniform sampling is marked as red bars. Green bars represent the times, at which changepoint was detected. All the signal anomalies are depicted as orange bars below the graph.

558 4.3. Kokam Battery Temperature Module(BESS)

559 A second case study is concerned with monitoring temperature profiles of
560 individual modules of battery pack deployed at end user. During the oper-
561 ation, a hardware fault of the cooling fan happened. Our industrial partner
562 was interested in finding out, whether such an event could be captured by
563 an anomaly detection system. The data for 12 modules, each coming with 6
564 channels of measurement were retrieved in 30-second sampling and processed
565 in a streamed manner. We found it informative to compute the deviation
566 of the observed value from the average of all the above-mentioned measure-
567 ments.

568 Our anomaly detection system was, once again, initialized with an ex-
569 piration period of 7 days. The grace period was shortened to 1 day. The
570 threshold value was shifted to a 4 sigma value of 99.977% to minimize the
571 number of alarms.

572 In Figure 2 we observe 5 days of deviations between the observed tem-
573 perature measured by channels of module 9 and the average temperature
574 of all modules. After the grace period, we observe multiple system alarms
575 raised by various channels. Until the noon of 22nd August, they seem to be
576 spread out randomly between individual channels. During the late evening
577 of 22nd, anomalies were reported by both channels 4 and 5 for a prolonged
578 period, followed by an anomalous rise in temperature measured by channel
579 6 early in the morning on 23rd August. The fan fault was observed approx-
580 imately at 5 pm on 23rd August. Our anomaly detection system instantly
581 raised an alarm, notifying us of anomalous behavior reported by channels 1
582 - 3. The prolonged duration of the alarm triggered the changepoint alarm
583 approximately 2 hours later. This resulted in a slightly faster adaptation of
584 the system to the new operation under increased temperature. Surprisingly,
585 the temperature decreased during the next day, notifying us of the fan being
586 in operation, to fail again 30 minutes later after the battery modules were
587 cooled down to the previous setpoint. The anomaly detection system was
588 triggered once again, although adaptation loosened the region of normal op-
589 eration to allow itself to adapt. No significant anomalies in sampling were
590 observed during the period.

591 **5. Conclusion**

592 In this paper, we examine the capacity of adaptive conditional probability
593 distribution to model the normal operation of dynamic systems employing

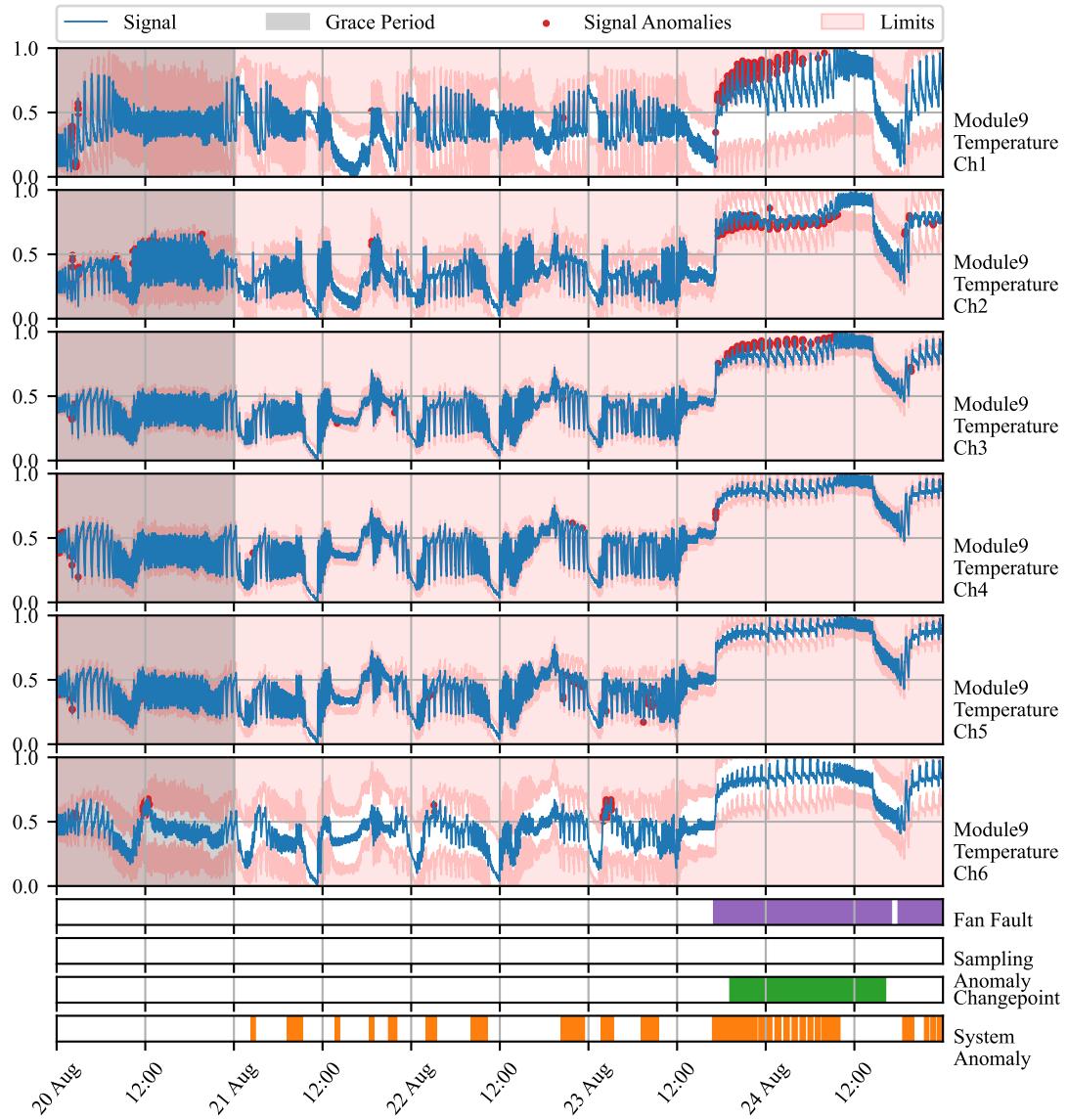


Figure 2: Time Series of BESS measurements (green line) of process variables. Non-uniform ticks on the x-axis mark days of interest (NOTE: some marks are hidden due to the readability). The y-axis renders the normalized process variables. System anomalies are marked as red dots. Non-uniform sampling is detected at blue vertical lines. Yellow vertical lines denote changepoint adaptation

594 streaming IoT devices and isolate the root cause. The dynamics of the sys-
595 tems are elaborated in the model using Welford’s online algorithm with the
596 capacity to update and revert sufficient parameters of underlying multivariate
597 Gaussian distribution in time making it possible to elaborate non-stationarity
598 in the process variables. Moreover, the self-supervision allows protection of
599 the distribution from the effect of outliers and increased speed of adaptation
600 in cases of changes in operation.

601 We assume the Gaussian distribution of measurements over a bounded
602 time frame related to the system dynamics. We consider such an assumption
603 reasonable, with support of multiple trials where the Kolmogorov-Smirnov
604 test did not reject this hypothesis. The statistical model provides the capac-
605 ity for the interpretation of the anomalies as extremely deviating observations
606 from the mean vector. Another assumption held in this study is that any
607 anomaly, spatial or temporal, can be transformed in such a way that makes
608 it an outlier given that we expose such effects as features as shown in case
609 studies.

610 Our approach establishes the system’s operation state at the global anomaly
611 level by considering interactions between input measurements and engineered
612 features and computing distance from mean of conditional probability. At
613 the second level, dynamic process limits based on PPF at threshold probabil-
614 ity, given multivariate distribution parameters, help isolate the root cause of
615 anomalies. This level serves the diagnostic purpose of the model operation.
616 The individual signals contribute to the global anomaly prediction, while
617 the proposed dynamic limits offer less conservative restrictions on individual
618 process operation. In parallel, the detector allows discrimination of signal
619 losses due to packet drops and sensor malfunctioning.

620 The ability to detect and identify anomalies in the system, isolate the
621 root cause of anomaly to specific signal or feature, and identify signal losses
622 is shown in two case studies on real data. Unlike many anomaly detec-
623 tion approaches, the proposed AID method does not require historical data
624 or ground truth information about anomalies, relieving general limitations.
625 Moreover, it combines adaptability and interpretability, which is an area yet
626 to be explored.

627 The benchmark performed on industrial data showed the ability to pro-
628 vide comparable results to other self-learning adaptable anomaly detection
629 methods. This is an important property for our model which allows, in
630 addition, the root cause isolation. The first case study, performed on real
631 operation data of BESS, examined the battery energy storage system and

632 demonstrated the ability to capture system anomalies and provide less con-
633 servative limits to signals. The physical model aided decisions about the
634 normality of the measured temperature of BESS.

635 The second case study exposed the ability to detect anomalies in the
636 temperature profiles of battery modules within the battery pack, consider-
637 ing spatial measurements made by multiple sensors distributed around the
638 module and the average temperature of all the modules within the pack.
639 Hardware fault observed on this deployed device was captured by our model,
640 giving another proof of its importance in energy storage systems monitoring,
641 where tight temperature control plays a significant role in the safety and
642 profitability of the system.

643 Future works on the method will include improvement to the change point
644 detection mechanism, decrease in the latency on high dimensional data, and
645 false positive rate reduction, from which general plug-and-play models suffer.

646 References

647 V. Chandola, A. Banerjee, V. Kumar, Anomaly detection: A survey, ACM
648 Comput. Surv. 41 (2009). URL: <https://doi.org/10.1145/1541880.1541882>. doi:10.1145/1541880.1541882.

650 N. Laptev, S. Amizadeh, I. Flint, Generic and scalable framework for auto-
651 mated time-series anomaly detection, in: Proceedings of the 21th ACM
652 SIGKDD International Conference on Knowledge Discovery and Data Min-
653 ing, KDD '15, Association for Computing Machinery, New York, NY,
654 USA, 2015, pp. 1939–1947. URL: <https://doi.org/10.1145/2783258.2788611>. doi:10.1145/2783258.2788611.

656 A. Kejariwal, Introducing practical and robust anomaly
657 detection in a time series, 2015. URL: https://blog.twitter.com/engineering/en_us/a/2015/introducing-practical-and-robust-anomaly-detection-in-a-time-series.

660 A. A. Cook, G. Misirli, Z. Fan, Anomaly detection for iot time-series data:
661 A survey, IEEE Internet of Things Journal 7 (2020) 6481–6494. doi:10.
662 1109/JIOT.2019.2958185.

663 K. Zhang, J. Chen, C.-G. Lee, S. He, An unsupervised spa-
664 tiotemporal fusion network augmented with random mask and time-
665 relative information modulation for anomaly detection of machines

- 666 with multiple measuring points, Expert Systems with Applications 237 (2024) 121506. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423020080>. doi:<https://doi.org/10.1016/j.eswa.2023.121506>.
- 670 J. Huang, D. Cheng, S. Zhang, A novel outlier detecting algorithm based on the outlier turning points, Expert Systems with Applications 231 (2023) 120799. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423013015>. doi:<https://doi.org/10.1016/j.eswa.2023.120799>.
- 675 C. Fan, Y. Sun, Y. Zhao, M. Song, J. Wang, Deep learning-based feature engineering methods for improved building energy prediction, Applied Energy 240 (2019) 35–45. URL: <https://www.sciencedirect.com/science/article/pii/S0306261919303496>. doi:<https://doi.org/10.1016/j.apenergy.2019.02.052>.
- 680 P. D. Talagala, R. J. Hyndman, K. Smith-Miles, Anomaly detection in high-dimensional data, Journal of Computational and Graphical Statistics 30 (2021) 360–374. URL: <https://doi.org/10.1080/10618600.2020.1807997>. doi:<https://doi.org/10.1080/10618600.2020.1807997>. arXiv:<https://doi.org/10.1080/10618600.2020.1807997>.
- 685 J. Li, Z. Liu, Attribute-weighted outlier detection for mixed data based on parallel mutual information, Expert Systems with Applications 236 (2024) 121304. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423018067>. doi:<https://doi.org/10.1016/j.eswa.2023.121304>.
- 690 X. Du, J. Chen, J. Yu, S. Li, Q. Tan, Generative adversarial nets for unsupervised outlier detection, Expert Systems with Applications 236 (2024) 121161. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423016639>. doi:<https://doi.org/10.1016/j.eswa.2023.121161>.
- 695 M. Salehi, L. Rashidi, A survey on anomaly detection in evolving data: [with application to forest fire risk prediction], SIGKDD Explor. Newsl. 20 (2018) 13–23. URL: <https://doi.org/10.1145/3229329.3229332>. doi:<https://doi.org/10.1145/3229329.3229332>.

- 699 N. Barbosa Roa, L. Travé-Massuyès, V. H. Grisales-Palacio, Dy-
700 clee: Dynamic clustering for tracking evolving environments, Pat-
701 tern Recognition 94 (2019) 162–186. URL: <https://www.sciencedirect.com/science/article/pii/S0031320319301992>. doi:<https://doi.org/10.1016/j.patcog.2019.05.024>.
- 704 A. G. Tartakovsky, A. S. Polunchenko, G. Sokolov, Efficient computer net-
705 work anomaly detection by changepoint detection methods, IEEE Journal
706 of Selected Topics in Signal Processing 7 (2013) 4–11. doi:[10.1109/JSTSP.2012.2233713](https://doi.org/10.1109/JSTSP.2012.2233713).
- 708 H. Wu, J. He, M. Tömösközi, Z. Xiang, F. H. Fitzek, In-network processing
709 for low-latency industrial anomaly detection in softwarized networks, in:
710 2021 IEEE Global Communications Conference (GLOBECOM), 2021, pp.
711 01–07. doi:[10.1109/GLOBECON46510.2021.9685489](https://doi.org/10.1109/GLOBECON46510.2021.9685489).
- 712 H. S. Pannu, J. Liu, S. Fu, Aad: Adaptive anomaly detection system for cloud
713 computing infrastructures, in: 2012 IEEE 31st Symposium on Reliable
714 Distributed Systems, 2012, pp. 396–397. doi:[10.1109/SRDS.2012.3](https://doi.org/10.1109/SRDS.2012.3).
- 715 S. Ahmad, A. Lavin, S. Purdy, Z. Agha, Unsupervised real-time anomaly
716 detection for streaming data, Neurocomputing 262 (2017) 134–
717 147. URL: <https://www.sciencedirect.com/science/article/pii/S0925231217309864>. doi:<https://doi.org/10.1016/j.neucom.2017.04.070>, online Real-Time Learning Strategies for Data Streams.
- 720 H. H. Bosman, G. Iacca, A. Tejada, H. J. Wörtche, A. Liotta, Ensembles
721 of incremental learners to detect anomalies in ad hoc sensor networks,
722 Ad Hoc Networks 35 (2015) 14–36. URL: <https://www.sciencedirect.com/science/article/pii/S1570870515001481>. doi:<https://doi.org/10.1016/j.adhoc.2015.07.013>, special Issue on Big Data Inspired Data
724 Sensing, Processing and Networking Technologies.
- 726 M. Carletti, C. Masiero, A. Beghi, G. A. Susto, Explainable machine learning
727 in industry 4.0: Evaluating feature importance in anomaly detection to
728 enable root cause analysis, in: 2019 IEEE International Conference on
729 Systems, Man and Cybernetics (SMC), 2019, pp. 21–26. doi:[10.1109/SMC.2019.8913901](https://doi.org/10.1109/SMC.2019.8913901).

- 731 Q. P. Nguyen, K. W. Lim, D. M. Divakaran, K. H. Low, M. C. Chan, Gee: A
732 gradient-based explainable variational autoencoder for network anomaly
733 detection, in: 2019 IEEE Conference on Communications and Network
734 Security (CNS), 2019, pp. 91–99. doi:10.1109/CNS.2019.8802833.
- 735 K. Amarasinghe, K. Kenney, M. Manic, Toward explainable deep neural
736 network based anomaly detection, in: 2018 11th International Conference
737 on Human System Interaction (HSI), 2018, pp. 311–317. doi:10.1109/HSI.
738 2018.8430788.
- 739 X. Zhang, J. Shi, X. Huang, F. Xiao, M. Yang, J. Huang,
740 X. Yin, A. Sohail Usmani, G. Chen, Towards deep probabilistic
741 graph neural network for natural gas leak detection and localiza-
742 tion without labeled anomaly data, Expert Systems with Appli-
743 cations 231 (2023) 120542. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423010448>. doi:<https://doi.org/10.1016/j.eswa.2023.120542>.
- 746 W.-T. Yang, M. S. Reis, V. Borodin, M. Juge, A. Roussy, An interpretable
747 unsupervised bayesian network model for fault detection and diagno-
748 sis, Control Engineering Practice 127 (2022) 105304. URL: <https://www.sciencedirect.com/science/article/pii/S0967066122001502>.
749 doi:<https://doi.org/10.1016/j.conengprac.2022.105304>.
- 751 L. C. Brito, G. A. Susto, J. N. Brito, M. A. V. Duarte, Fault diagno-
752 sis using explainable ai: A transfer learning-based approach for rotating
753 machinery exploiting augmented synthetic data, Expert Systems with
754 Applications 232 (2023) 120860. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423013623>. doi:<https://doi.org/10.1016/j.eswa.2023.120860>.
- 757 Z. Wu, X. Yang, X. Wei, P. Yuan, Y. Zhang, J. Bai, A self-supervised
758 anomaly detection algorithm with interpretability, Expert Systems with
759 Applications 237 (2024) 121539. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423020419>. doi:<https://doi.org/10.1016/j.eswa.2023.121539>.
- 762 M. Wadinger, M. Kvasnica, Real-time outlier detection with dynamic pro-
763 cess limits, in: Proceedings of the 2023 24th International Conference on
764 Process Control (PC), 2023. In press.

- 765 K. Yamanishi, J.-i. Takeuchi, A unifying framework for detecting outliers and
766 change points from non-stationary time series data, in: Proceedings of the
767 Eighth ACM SIGKDD International Conference on Knowledge Discovery
768 and Data Mining, KDD '02, Association for Computing Machinery, New
769 York, NY, USA, 2002, pp. 676–681. URL: <https://doi.org/10.1145/775047.775148>. doi:10.1145/775047.775148.
- 771 K. Yamanishi, J.-i. Takeuchi, G. Williams, P. Milne, On-line unsu-
772 pervised outlier detection using finite mixtures with discounting learn-
773 ing algorithms, *Data Mining and Knowledge Discovery* 8 (2004) 275–
774 300. URL: <https://doi.org/10.1023/B:DAMI.0000023676.72185.7c>.
775 doi:10.1023/B:DAMI.0000023676.72185.7c.
- 776 B. Steenwinckel, Adaptive anomaly detection and root cause analysis by fus-
777 ing semantics and machine learning, in: A. Gangemi, A. L. Gentile, A. G.
778 Nuzzolese, S. Rudolph, M. Maleshkova, H. Paulheim, J. Z. Pan, M. Alam
779 (Eds.), *The Semantic Web: ESWC 2018 Satellite Events*, Springer Inter-
780 national Publishing, Cham, 2018, pp. 272–282.
- 781 B. Steenwinckel, D. De Paepe, S. Vanden Hautte, P. Heyvaert, M. Ben-
782 tefrit, P. Moens, A. Dimou, B. Van Den Bossche, F. De Turck, S. Van
783 Hoecke, F. Ongena, Flags: A methodology for adaptive anomaly
784 detection and root cause analysis on sensor data streams by fusing
785 expert knowledge with machine learning, *Future Generation Com-
786 puter Systems* 116 (2021) 30–48. URL: <https://www.sciencedirect.com/science/article/pii/S0167739X20329927>. doi:<https://doi.org/10.1016/j.future.2020.10.015>.
- 789 M. Amer, M. Goldstein, S. Abdennadher, Enhancing one-class support vec-
790 tor machines for unsupervised anomaly detection, in: Proceedings of the
791 ACM SIGKDD Workshop on Outlier Detection and Description, ODD '13,
792 Association for Computing Machinery, New York, NY, USA, 2013, pp.
793 8–15. URL: <https://doi.org/10.1145/2500853.2500857>. doi:10.1145/2500853.2500857.
- 795 B. Liu, Y. Xiao, P. S. Yu, L. Cao, Y. Zhang, Z. Hao, Uncertain one-class
796 learning and concept summarization learning on uncertain data streams,
797 *IEEE Transactions on Knowledge and Data Engineering* 26 (2014) 468–
798 484. doi:10.1109/TKDE.2012.235.

- 799 B. Krawczyk, M. Woźniak, One-class classifiers with incremental
800 learning and forgetting for data streams with concept drift, Soft
801 Computing 19 (2015) 3387–3400. URL: <https://doi.org/10.1007/s00500-014-1492-5>. doi:10.1007/s00500-014-1492-5.
- 803 X. Miao, Y. Liu, H. Zhao, C. Li, Distributed online one-class support vec-
804 tor machine for anomaly detection over networks, IEEE Transactions on
805 Cybernetics 49 (2019) 1475–1488. doi:10.1109/TCYB.2018.2804940.
- 806 Ö. Gözüaçık, F. Can, Concept learning using one-class classifiers for
807 implicit drift detection in evolving data streams, Artificial Intelli-
808 gence Review 54 (2021) 3725–3747. URL: <https://doi.org/10.1007/s10462-020-09939-x>. doi:10.1007/s10462-020-09939-x.
- 810 R. Wetzig, A. Gulenko, F. Schmidt, Unsupervised anomaly alerting for
811 iot-gateway monitoring using adaptive thresholds and half-space trees,
812 in: 2019 Sixth International Conference on Internet of Things: Systems,
813 Management and Security (IOTSMS), 2019, pp. 161–168. doi:10.1109/IOTSMS48152.2019.8939201.
- 815 Y. Lyu, W. Li, Y. Wang, S. Sun, C. Wang, Rmhsforest: Rel-
816 ative mass and half-space tree based forest for anomaly detec-
817 tion, Chinese Journal of Electronics 29 (2020) 1093–1101. URL:
818 <https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/cje.2020.09.010>.
819 doi:<https://doi.org/10.1049/cje.2020.09.010>.
820 arXiv:<https://ietresearch.onlinelibrary.wiley.com/doi/pdf/10.1049/cje.2020.09.010>
- 821 B. P. Welford, Note on a method for calculating corrected
822 sums of squares and products, Technometrics 4 (1962) 419–
823 420. URL: <https://www.tandfonline.com/doi/abs/10.1080/00401706.1962.10490022>.
824 doi:10.1080/00401706.1962.10490022.
825 arXiv:<https://www.tandfonline.com/doi/pdf/10.1080/00401706.1962.10490022>.
- 826 A. Genz, Numerical computation of multivariate normal probabilities, Jour-
827 nal of Computational and Graphical Statistics 1 (2000). doi:10.1080/
828 10618600.1992.10477010.
- 829 R. P. Brent, Algorithms for minimization without derivatives, Prentice-Hall,
830 Englewood Cliffs, N.J, 1972. URL: https://openlibrary.org/books/OL4739237M/Algorithms_for_minimization_without_derivatives.

832 F. Iglesias Vázquez, A. Hartl, T. Zseby, A. Zimek, Anomaly detection in
833 streaming data: A comparison and evaluation study, Expert Systems with
834 Applications 233 (2023) 120994. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423014963>. doi:<https://doi.org/10.1016/j.eswa.2023.120994>.

837 L. Ruff, J. R. Kauffmann, R. A. Vandermeulen, G. Montavon, W. Samek,
838 M. Kloft, T. G. Dietterich, K.-R. Müller, A unifying review of deep and
839 shallow anomaly detection, Proceedings of the IEEE 109 (2021) 756–795.
840 doi:[10.1109/JPROC.2021.3052449](https://doi.org/10.1109/JPROC.2021.3052449).

841 I. D. Katser, V. O. Kozitsin, Skoltech anomaly benchmark (skab),
842 <https://www.kaggle.com/dsv/1693952>, 2020. doi:[10.34740/KAGGLE/DSV/1693952](https://doi.org/10.34740/KAGGLE/DSV/1693952).