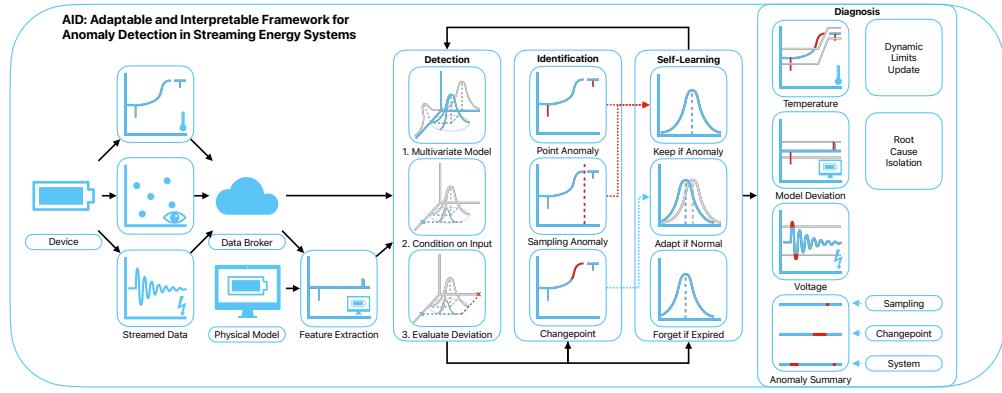


Graphical Abstract

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

- First interpretable anomaly detector with self-supervised adaptation
- Delivers comparable performance to established general methods
- Isolates root cause of anomalies while considering interactions
- Demonstrates interpretability by providing process limits for signals
- Uses self-learning approach on streamed IoT data

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 showcases AID's effectiveness in Battery Energy Storage Systems, capturing anomalies, setting less conservative signal limits, and ability to leverage a physical model. The second study monitors battery module temperatures, where AID identifies hardware faults, emphasizing its relevance to energy storage 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. Our framework is openly accessible on https://github.com/MarekWadinger/online_outlier_detection.

Keywords: Anomaly detection, Root cause isolation, Iterative learning,

*

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 complexity issues, Zhang et al. have suc-
15 cessfully employed spatially distributed sensors and time-relative modula-
16 tion. Their approach has proven effective, particularly in the context of com-
17 plex non-linear systems, offering potential solutions to some of the challenges
18 posed by IoT data (Zhang et al. (2024)). Huang et al., on the other hand,
19 tackled the problems of detecting global outliers, local outliers, and outlier
20 clusters simultaneously. Their proposed approach, based on density estima-
21 tion, relies on the notion that density distributions should exhibit minimal
22 variations in local areas. To achieve this, they introduce a novel turning ra-
23 tio metric, which reduces reliance on hyperparameters and enhances anomaly
24 detection (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 is worth noting that feature engineering
28 may introduce categorical variables and significantly increase dimensionality
29 of 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
36 al. demonstrated the superior performance of their algorithm compared to
37 state-of-the-art methods (Li and Liu (2024)). Another strategy for handling
38 high-dimensional data involves using deep learning methods with synthetic
39 normal data to enhance the detection of outliers with subtle deviations, as
40 proposed in Du et al. (2024).

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

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

60 *1.1. Related Work*

61 Recent advances in anomaly detection have broadened its scope to in-
62 clude root cause identification governed by the development of explanatory
63 methods capable of diagnosing and tracking faults across the system. Stud-
64 ies can be split into two groups of distinct approaches. The first group
65 approaches explainability as the importance of individual features (Carletti
66 et al. (2019); Nguyen et al. (2019); Amarasinghe et al. (2018)). Those stud-
67 ies allow an explanation of novelty by considering features independently.
68 The second group uses statistical learning creating models explainable via
69 probability. For instance, integration of variational Bayesian inference prob-
70 abilistic graph neural network allowed Zhang et al. to model the posterior
71 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 self-
109 supervised approach facilitates intelligent adaptation with respect to detected
110 change points. By leveraging its ability to adapt to changes in operational
111 states, our proposed method operates autonomously when such changes oc-
112 cur. Moreover, Yamanishi et al. (2004) do not attempt to isolate the root
113 cause of the anomaly. Our approach extends statistical outlier detection by
114 incorporating interpretability. This is achieved through the computation of
115 conditional probabilities for each measurement, considering the remainder
116 of the measurements, and establishing limits that define the threshold for
117 normal event probabilities.

118 A limited number of studies have focused on adaptation and interpretabil-
119 ity within the framework of anomaly detection. Two recent contributions
120 in this area are (Steenwinckel (2018)) and (Steenwinckel et al. (2021)). In
121 (Steenwinckel (2018)), the authors emphasize the importance of combining
122 prior knowledge with a data-driven approach to achieve interpretability, par-
123 ticularly concerning root cause isolation. They propose a novel approach
124 that involves extracting features based on knowledge graph pattern extrac-
125 tion and integrating them into the anomaly detection mechanism. This graph
126 is subsequently transformed into a matrix, and adaptive region-of-interest ex-
127 traction is performed using reinforcement learning techniques. To enhance
128 interpretability, a Generative Adversarial Network (GAN) reconstructs a new
129 graphical representation based on selected vectors. However, it is important
130 to note that the validation of this idealized approach is pending further in-
131 vestigation. Lately, (Steenwinckel et al. (2021)) introduced a comprehen-
132 sive framework for adaptive anomaly detection and root cause analysis in
133 data streams. While the adaptation process is driven by user feedback, the
134 specific mechanism remains undisclosed. The authors present an interpreta-
135 tion of their method through a user dashboard, featuring visualizations of
136 raw data. This dashboard is capable of distinguishing between track-related
137 problems and train-related issues, based on whether multiple trains at the
138 same geographical location approach the anomaly. Meanwhile, our efforts are
139 directed towards the development of a self-supervised method that can learn
140 autonomously, reducing the reliance on human supervision, which is often
141 constrained by time limitations and can lead to significant delays in adapta-
142 tion. Our method is distinguished by its straightforward statistical reasoning
143 and the ability to isolate the root cause of anomalies. The interpretability of
144 our method is demonstrated through the establishment of dynamic process
145 limits for each signal, leveraging conditional probabilities derived from the

146 signal and other system measurements and features. This provides operators
147 with a clear understanding of the system’s state and the underlying causes
148 of anomalies.

149 *1.3. Validation*

150 Two case studies show that our proposed method, based on dynamic
151 joint normal distribution, has the capacity to explain novelties, isolate the
152 root cause of anomalies, and allow adaptation to change points, advancing
153 recently developed anomaly detection techniques for long-term deployment
154 and cross-domain usage. We observe similar detection performance, albeit
155 with lower scalability, on benchmark data when comparing our approach to
156 well-established unsupervised anomaly detection methods in streamed data
157 which create a bedrock for many state-of-the-art contributions, such as One-
158 Class SVM (Amer et al. (2013); Liu et al. (2014); Krawczyk and Woźniak
159 (2015); Miao et al. (2019); Gözüaçık and Can (2021)), and Half-Space Trees
160 (Wetzig et al. (2019); Lyu et al. (2020)).

161 *1.4. Broader Impact*

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

173 The main contribution of the proposed solution to the developed body of
174 research is that it:

- 175 • Enriches interpretable anomaly detection with adaptive capabilities
- 176 • Isolates root cause of anomalies while considering interactions
- 177 • Uses self-learning approach on streamed IoT data
- 178 • Demonstrates interpretability by providing process limits for signals
- 179 • Utilizes existing IT infrastructure

180 1.5. Paper Organization

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

187 2. Preliminaries

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

197 2.1. Welford’s Online Algorithm

198 Welford introduced a numerically stable online algorithm for calculating
199 mean and variance in a single pass through data. Therefore, the algorithm
200 allows the processing of IoT device measurements without the need to store
201 their values Welford (1962).

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

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

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

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

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

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

208 as it is less prone to numerical instability due to catastrophic cancellation,
 209 significant loss of precision due to subtracting two nearly equal numbers.
 210 Finally, the corresponding unbiased variance is

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

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

213 *2.2. Inverting Welford's Algorithm*

214 Based on (2), it is clear that the influence of the latest sample over the
 215 running mean decreases as the population n grows. For this reason, regulat-
 216 ing the number of samples used for sample mean and variance computa-
 217 tion has crucial importance over adaptation. Given access to the instances used
 218 for computation and expiration period $t_e \in \mathbb{N}_0^{n-1}$, reverting the impact of
 219 x_{n-t_e} can be written as follows

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

220 where the reverted mean is given as

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

221 Finally, the unbiased variance follows the formula:

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

222 Notably, inversion allows the algorithm to keep constant rate of adapta-
 223 tion at cost of storing a bounded data buffer.

224 *2.3. Statistical Model of Multivariate System*

225 Multivariate normal distribution generalizes the multivariate systems to
 226 the model where the degree to which variables are related is represented by
 227 the covariance matrix. Gaussian normal distribution of variables is a reason-
 228 able assumption for process measurements, as it is a common distribution
 229 that arises from stable physical processes measured with noise (Mishra and
 230 Datta-Gupta (2018)). The general notation of multivariate normal distribu-
 231 tion is:

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

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

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

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

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

239 The cumulative distribution function (CDF) of a multivariate Gaussian
 240 distribution describes the probability that all components of the random
 241 vector \mathbf{X} take on a value less than or equal to a particular point q in space,
 242 and can be used to evaluate the likelihood of observing a particular set of
 243 measurements or data points. In other words, it gives the probability of
 244 observing a random vector that falls within a certain region of space. The
 245 standard notation of CDF is as follows:

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

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

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

249 Given the PDF, we can also determine the value of \mathbf{x} that corresponds to a
 250 given quantile q using a numerical method for inversion of CDF (ICDF) often
 251 denoted as percent point function (PPF) or $F(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})^{-1}$. An algorithm that
 252 calculates the value of the PPF is part of standard statistical software tools.

253 2.4. Conditional Probability Distribution

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

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

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

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

262 and

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

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

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

266 where $\mu_{a|\mathbf{b}}$ denotes the conditional mean and $\sigma_{a|\mathbf{b}}^2$ represents the conditional
 267 variance. These crucial parameters can be computed applying the
 268 Schur complement as follows:

$$\sigma_{a|\mathbf{b}}^2 = \sigma_{aa}^2 - \boldsymbol{\Sigma}_{ab}\boldsymbol{\Sigma}_{bb}^{-1}\boldsymbol{\Sigma}_{ba}, \quad (15)$$

269 for the conditional variance $\sigma_{a|\mathbf{b}}^2$, while the conditional mean, denoted as
 270 $\mu_{a|\mathbf{b}}$, is determined by:

$$\mu_{a|\mathbf{b}} = \mu_a + \boldsymbol{\Sigma}_{ab}\boldsymbol{\Sigma}_{bb}^{-1}(\mathbf{x}_b - \boldsymbol{\mu}_b). \quad (16)$$

271 The conditional variance $\sigma_{a|\mathbf{b}}^2$ essentially represents the Schur complement
 272 of $\boldsymbol{\Sigma}_{bb}$ within the overall covariance matrix $\boldsymbol{\Sigma}$.

273 2.5. Gaussian Anomaly Detection

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

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

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

$$q = P\{|x_a - \mu_{a|\mathbf{b}}| < 3\sigma_{a|\mathbf{b}}^2\} = 0.99735. \quad (17)$$

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

$$x_l = F((1 - P\{|x_a - \mu_{a|\mathbf{b}}| < 3\sigma_{a|\mathbf{b}}^2\}); \mu_{a|\mathbf{b}}, \sigma_{a|\mathbf{b}}^2)^{-1}, \quad (18)$$

291 for the lower limit, and

$$x_u = F((P\{|x_a - \mu_{a|\mathbf{b}}| < 3\sigma_{a|\mathbf{b}}^2\}); \mu_{a|\mathbf{b}}, \sigma_{a|\mathbf{b}}^2)^{-1}, \quad (19)$$

292 for the upper limit. These lower and upper limits together form vectors
 293 \mathbf{x}_l and \mathbf{x}_u , respectively, defining the region of normal system operation. This
 294 region is conceptualized as a hypercube in the feature space, with each di-
 295 mension bounded by the corresponding feature limits, as computed using
 296 Equations (18) and (19) for all $a = 1, \dots, k$; $\mathbf{b} = \{1, 2, \dots, k\}$ where $a \notin \mathbf{b}$.
 297 The approximation of a confidence ellipse as a hypercube can be employed
 298 to represent the region of normal system operation for individual variables
 299 of a multivariate system, rendering it as an aid for visual representation.

300 The predicted state of the system, denoted as y_i , and the normality of
 301 signals $\mathbf{y}_{s,i}$ at time i are determined based on the maximum distance of
 302 observations from the center of the probabilistic density. The center of the
 303 probabilistic density corresponds to the vector of conditional means $\mu_{a|\mathbf{b}}$ with
 304 respect to other features. The calculation of this distance involves the cumu-
 305 lative distribution function (CDF) of observations and conditional distribu-
 306 tions, as follows:

$$F(x_a; \mu_{a|\mathbf{b}}, \sigma_{a|\mathbf{b}}^2) : a = 1, \dots, k; \mathbf{b} = \{1, 2, \dots, k\} \text{ where } a \notin \mathbf{b}. \quad (20)$$

307 Subsequently, operation states of individual inputs are defined as follows:

$$\mathbf{y}_{s,i} = \begin{cases} 0 & \text{if } T \leq (20) \\ 1 & \text{if } T > (20), \end{cases} \quad (21)$$

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

310 For the overall abnormality of the system, any anomaly in signals $\mathbf{y}_{s,i}$ is
311 considered, resulting in:

$$y_i = \begin{cases} 1 & \text{if } 1 \in \mathbf{y}_{s,i} \\ 0 & \text{otherwise,} \end{cases} \quad (22)$$

312 defining the discrimination boundary between system operation where
313 $y_i = 0$ indicates normal system operation, and $y_i = 1$ indicates anomalous
314 operation.

315 2.6. Anomaly Definition

316 This subsection provides an overview of the definition of anomalies in
317 data analysis and their categorization, setting convention for this paper.

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

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

328 Nevertheless, this conventional method may not be suitable for scenarios
329 characterized by collective anomalies, where clusters of abnormal data points
330 coexist. A more pragmatic approach defines anomalies as deviations from
331 established "normal" patterns, resembling the principles of semi-supervised
332 learning. Change point detection, in a similar vein, can be regarded as a
333 relative approach that takes into account the varying dynamics of changes,
334 whether they occur gradually or abruptly (Iglesias Vázquez et al. (2023)).

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

341 Notwithstanding the differences in terminology, methods employed for the
342 identification of data points residing in low-probability regions, irrespective
343 of whether they are referred to as "anomaly detection," "outlier detection,"
344 or "novelty detection," share fundamental similarities (Iglesias Vázquez et al.
345 (2023)).

346 **3. Adaptive Anomaly Detection and Interpretation Framework**

347 In this section, we propose an adaptive and interpretable detection frame-
348 work (AID) for multivariate systems with streaming IoT devices. This ap-
349 proach models the system as a dynamic joint normal distribution, enabling
350 it to effectively adapt to pervasive nonstationary effects on processes. Our
351 method handles various factors, including change points, concept drift, and
352 seasonal effects. Our primary contribution lies in the fusion of an adaptable
353 self-supervised system with root cause identification capabilities. This combi-
354 nation empowers the online statistical model to diagnose anomalies through
355 two distinct mechanisms. Firstly, it employs conditional probability calcu-
356 lations to assess the system's operating conditions' normality. Secondly, it
357 identifies outliers within individual signal measurements and features based
358 on dynamic alert-triggering process limits. In the following sections, we de-
359 scribe our proposed methodology across three subsections. The initial sub-
360 section delves into the process of initializing the model's parameters. The
361 subsequent section describes online training and adaptation, while the final
362 subsection expounds upon the model's detection and diagnostic capabilities.
363 For a concise representation of the proposed method, Algorithm 1 is provided.

364 *3.1. Model Parameters Initialization*

365 The model initialization is governed by defining two tunable hyperparam-
366 eters of the model: the expiration period (t_e) and the threshold (T). The
367 expiration period determines the window size for time-rolling computations,
368 impacting the proportion of outliers within a given timeframe, and directly
369 influencing the relaxation (with a longer expiration period) or tightening
370 (with a shorter expiration period) of dynamic signal limits. Additionally, we
371 introduce a grace period, which defaults to $3/4t_e$, allowing for model calibra-
372 tion. During this grace period, system anomalies are not flagged to prevent
373 false positives and speed up self-supervised learning, introduced in Subsec-
374 tion 3.2. The length of the expiration period inversely correlates with the
375 model's ability to adapt to sudden changes. The adaptation and detection

376 of shifts in the data-generating process, such as changes in mean or variance,
 377 is managed through the adaptation period t_a . A longer t_a results in slower
 378 adaptation but potentially longer alerts, which can be valuable when collective
 379 anomalies are expected to occur. In most cases, $t_a = t_e$ offers optimal
 380 performance.

381 As a general rule of thumb, expiration period t_e should be determined
 382 based on the slowest observed dynamics within the multivariate system. The
 383 threshold T defaults to the three-sigma probability of q in (17). Adjusting
 384 this threshold can fine-tune the trade-off between precision and recall. A
 385 lower threshold boosts recall but may lower precision, while a higher thresh-
 386 old enhances precision at the cost of recall. The presence of one non-default
 387 easily interpretable hyperparameter facilitates adaptability to various sce-
 388 narios. We recommend starting with the default values of other parameters
 389 and making adjustments based on real-time model performance.

390 3.2. Online training

391 Training in AID follows an incremental learning approach, processing
 392 each new sample upon arrival. Incremental learning allows online parame-
 393 ter updates, albeit with a potential computational delay affecting response
 394 latency.

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

402 It is worth noting that adaptation relies on two self-supervised methods.
 403 Adaptation routine runs if the observation at time instance i is considered
 404 normal. Adaptation period t_a allows the model to update the distribution
 405 on outliers as well. Given the predicted system anomaly state from (22) as
 406 y_i over the window of past observations $\mathbf{y}_i = \{y_{i-t_a}, \dots, y_i\}$, the following test
 407 holds when adaptation is performed on outlier:

$$\frac{\sum_{y \in \mathbf{y}_i} y}{n(\mathbf{y}_i)} > 2 * (T - 0.5). \quad (23)$$

408 Here $n(\mathbf{y}_i)$ denotes the dimensionality of \mathbf{y}_i . The logic of the (23) follows the

409 probabilistic approach to anomalies that assumes a number of anomalies are
410 lower or equal to the conditional probability at both tails of the distribution

411 *3.3. Online prediction*

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

414 Firstly, we calculate the parameters of the conditional distribution con-
415 cerning the dynamic multivariate Gaussian distribution. These calculations
416 are performed for the process observation vector \mathbf{x}_i at time instance i . Spec-
417 ifically, we compute the conditional mean using (16) and the conditional vari-
418 ance using (15). These computations yield univariate conditional distribu-
419 tions for individual signals and features. These conditional distributions play
420 a crucial role in assessing the abnormality of signals and features concerning
421 other observed values. This assessment relies on the strength of relation-
422 ships defined by the covariance matrix of the dynamic multivariate Gaussian
423 distribution. Consequently, our approach inherently considers the interac-
424 tions between input signals and features. The determination of anomalous
425 behavior is governed by (21).

426 Any anomaly detected within one of the features triggers an alert at the
427 system level. The decision regarding the overall system's anomalous behavior
428 is guided by (22). Nevertheless, individual determinations of anomalies serve
429 as a diagnostic tool for isolating the root cause of anomalies.

430 To assist operators in their assessments, we establish a hypercube defined
431 by lower and upper threshold values, denoted as \mathbf{x}_l and \mathbf{x}_u , respectively.
432 These thresholds are derived from (18) and (19), incorporating updated
433 model parameters. Lower and upper thresholds play a pivotal role as dy-
434 namic process limits. They replace the conservative process limits provided
435 in sensor documentation, accounting for spatial factors, such as multipoint
436 measurements and temporal factors such as aging, and actual environmental
437 conditions that influence sensor operation.

438 Our framework anticipates unexpected novel behavior, including signal
439 loss. This anticipation involves calculating the cumulative distribution func-
440 tion (CDF) over the univariate normal distribution of sampling, focusing
441 on the differences between subsequent timestamps. We operate under the
442 assumption that, over the long term, the distribution of sampling times re-
443 mains stable. As a result, we employ a one-pass update mechanism utilizing
444 (2) and (4), for efficiency. To proactively detect subtle changes in sampling

⁴⁴⁵ patterns, self-supervised learning is employed, leveraging anomalies weighted
⁴⁴⁶ by the deviation from $(1 - F(x_i; \mu, \sigma^2))$ for training.

⁴⁴⁷ The system is vigilant in identifying change points. When the adaptation
⁴⁴⁸ test specified in (23) is satisfied, change points are flagged and isolated. This
⁴⁴⁹ initiation of change points triggers updates to the model, ensuring it adapts
⁴⁵⁰ to evolving data patterns, such as changes in operation state, effectively.

Algorithm 1 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: $i \leftarrow 1$; $n \leftarrow 1$; $T \leftarrow (17)$; $\boldsymbol{\mu} \leftarrow \mathbf{x}_0$; $\boldsymbol{\Sigma} \leftarrow \mathbf{1}_{k \times k}$; $\mu_t \leftarrow 0$; $\sigma_t^2 \leftarrow 1$;
- 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 (21);
 - 6: $y_i \leftarrow \text{PREDICT}(\mathbf{y}_{s,i})$ using (22);
 - 7: $\mathbf{x}_{l,i}, \mathbf{x}_{u,i} \leftarrow \text{GET}(T, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ using Algorithm ??;
 - 8: $y_{t,i} \leftarrow \text{PREDICT}(t_i - t_{i-1})$ using (21);
 - 9: $\mu_t, \sigma_t^2 \leftarrow \text{UPDATE}(t_i - t_{i-1}, \mu_t, \sigma_t^2)$ using (2), (4);
 - 10: **if** (22) = 0 **or** (23) **then**
 - 11: $\boldsymbol{\mu}, \boldsymbol{\Sigma} \leftarrow \text{UPDATE}(\mathbf{x}_i, \boldsymbol{\mu}, \boldsymbol{\Sigma}, n)$ using (2), (4);
 - 12: **if** (23) **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**
-

451 **4. Case Study**

452 This section provides a benchmark and two case studies that showcase
453 the effectiveness and applicability of our proposed approach. In the following
454 Subsections, we investigate the properties and performance of the approach
455 using streamed benchmark system data and signals from IoT devices in a mi-
456 crogrid system. The successful deployment demonstrates that this approach
457 is suitable for existing process automation infrastructure.

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

460 *4.1. Benchmark*

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

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

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

492 *4.2. Battery Energy Storage System (BESS)*

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

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

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

$$\begin{aligned}
 T_{\text{bat},i+1} = & T_{\text{bat},i} + T_s(q_{\text{fan}}V_{\text{b,max}}\rho c_p(T_{\text{out}} - T_{\text{bat},i}) + V_{\text{c,max}}q_{\text{circ,fan}}\rho c_p T_{\text{bat},i} \\
 & + q_{\text{circ,fan}}(P_{\text{cool}}q_{\text{cool}}P_{\text{heat}}q_{\text{heat}}) + c_{\text{scale}}Q_{\text{bat}} + q_{\text{inner fans}} \\
 & - (V_{\text{b,max}}q_{\text{fan}}V_{\text{c,max}}q_{\text{circ,fan}})\rho c_p T_{\text{bat},i})/(m_{\text{bat}}c_{\text{p,b}})
 \end{aligned} \tag{24}$$

509 When combined with an averaged measurement of battery cell temper-
510 ature, we could compute the difference between real and predicted temper-
511 ature. Such deviation can be useful in detecting unexpected patterns in
512 temperature. Nevertheless, it may be inaccurate as the physical model is
513 simplified and does not account for spatial aspects, like temperature gra-
514 dients as well as different dynamic effects of charging and discharging on
515 temperature. Therefore, the raw measured temperature is used as well. The
516 deviation between demanded power and delivered power was used to aid the
517 identification of the state, as the increased difference might be related to
518 other unexpected and novel patterns.

519 Fig. 1 depicts the operation of the BESS over March 2022. Multiple
520 events of anomalous behavior happened within this period, confirmed by the
521 operators, that are observable through a sudden or significant shift in mea-
522 surements in a given period. As the first step, the detection mechanism was
523 initialized, following the provided guidelines for parameter selection in Sub-
524 section 3.1. The expiration period was set to $t_e = 7$ days, due to the weekly
525 seasonality of human behavior impacting battery usage. The threshold was
526 kept at default value $T = 0.99735$. A grace period, during which the model
527 learns from both normal and anomalous data (though normal are expected,
528 yet not required here), is shortened to 2.5 days to observe the effect of BESS
529 calibration happening on 3rd day from deployment.

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

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

541 The system identified 6 deviations in sampling, denoted by the red bars
542 in Fig. 1. 4 anomalies with shorter duration represented packet loss. The
543 prolonged anomaly was notified during the transfer of the battery pack. The
544 longest dropout observed happened across 20th March up to 21st. Unexpect-
545 edly, the change point detection module triggered an alarm at the end of
546 the loss, resulting in adaptation and a sharp shift in drawn limits for Power

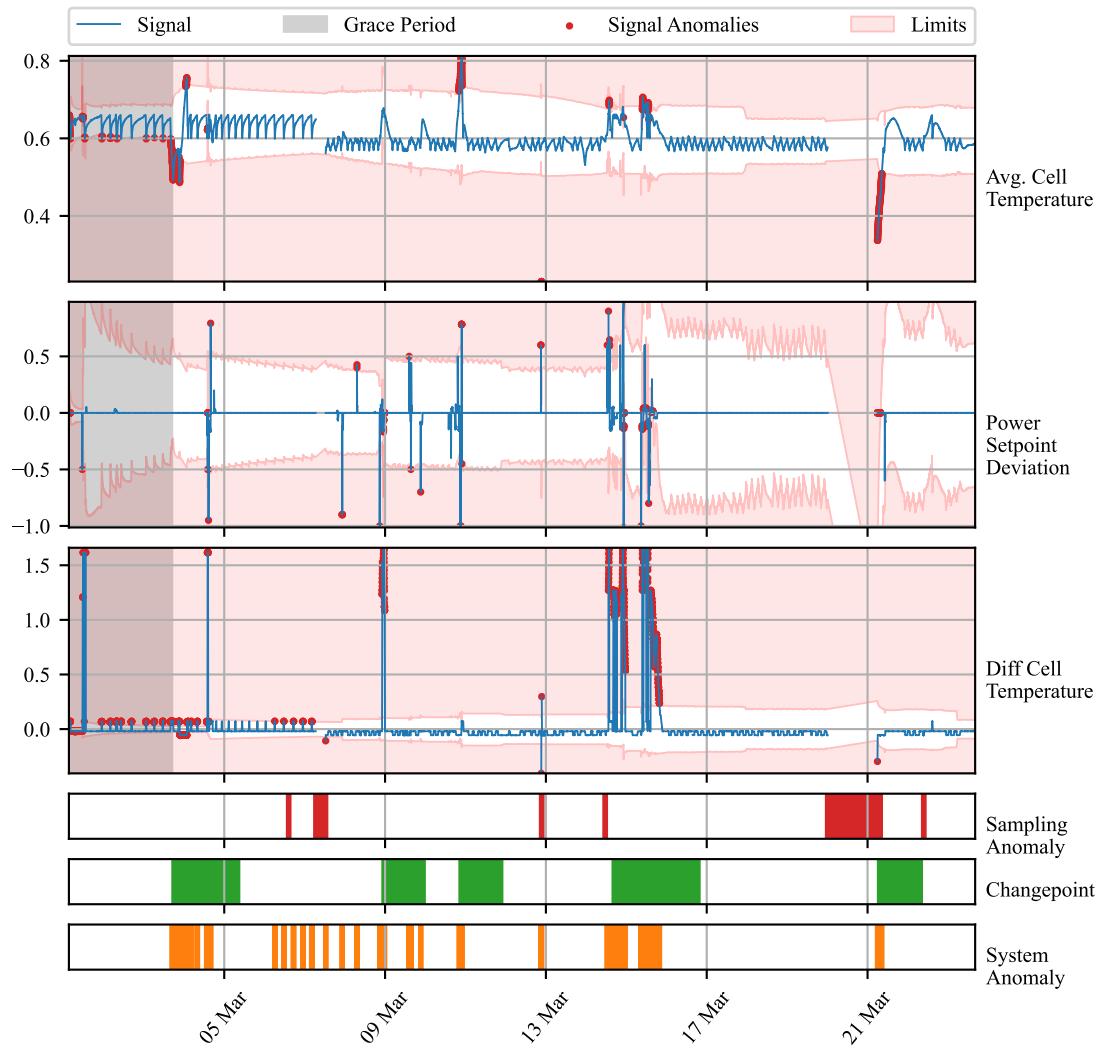


Figure 1: Time Series of BESS measurements (blue line) of process variables. The y-axis renders the values after the normalization of raw inputs. Root causes of anomalies are marked within specific signals as red dots. The light red area represents out-of-limits values for individual signals. Non-uniform 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.

547 Setpoint Deviation. Red dots represent anomalies at the signal level given
548 by equation (21). The dynamic signal limits are surpassed in one or multiple
549 signals during the system's anomalies. The root cause isolation allows the
550 pairing of anomalies with specific features. Conditional probability, against
551 which the anomalies are evaluated allows consideration of signal relationships
552 within individual limits.

553 *4.3. Kokam Battery Temperature Module*

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

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

567 In Figure 2 we observe 5 days of deviations between the observed tem-
568 perature measured by channels of module 9 and the average temperature
569 of all modules. After the grace period, we observe multiple system alarms
570 raised by various channels. Until the noon of 22nd August, they seem to be
571 spread out randomly between individual channels. During the late evening
572 of 22nd, anomalies were reported by both channels 4 and 5 for a prolonged
573 period, followed by an anomalous rise in temperature measured by channel
574 6 early in the morning on 23rd August. The fan fault was observed approx-
575 imately at 5 pm on 23rd August. Our anomaly detection system instantly
576 raised an alarm, notifying us of anomalous behavior reported by channels 1
577 - 3. The prolonged duration of the alarm triggered the changepoint alarm
578 approximately 2 hours later. This resulted in a slightly faster adaptation of
579 the system to the new operation under increased temperature. Surprisingly,
580 the temperature decreased during the next day, notifying us of the fan be-
581 ing in operation for a brief period, to fail again 30 minutes later after the
582 battery modules were cooled down to the previous setpoint. The anomaly
583 detection system was triggered once again, although adaptation loosened the

584 region of normal operation to allow itself to adapt. No significant anomalies
585 in sampling were observed during the period.

586 5. Conclusion

587 In this paper, we examine the capacity of adaptive conditional probability
588 distribution to model the normal operation of dynamic systems employing
589 streaming IoT devices and isolate the root cause. The dynamics of the sys-
590 tems are elaborated in the model using Welford’s online algorithm with the
591 capacity to update and revert sufficient parameters of underlying multivariate
592 Gaussian distribution in time making it possible to elaborate non-stationarity
593 in the process variables. Moreover, the self-supervision allows protection of
594 the distribution from the effect of outliers and increased speed of adaptation
595 in cases of changes in operation.

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

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

615 The ability to detect and identify anomalies in the system, isolate the
616 root cause of anomaly to specific signal or feature, and identify signal losses
617 is shown in two case studies on real data. Unlike many anomaly detec-
618 tion approaches, the proposed AID method does not require historical data
619 or ground truth information about anomalies, relieving general limitations.

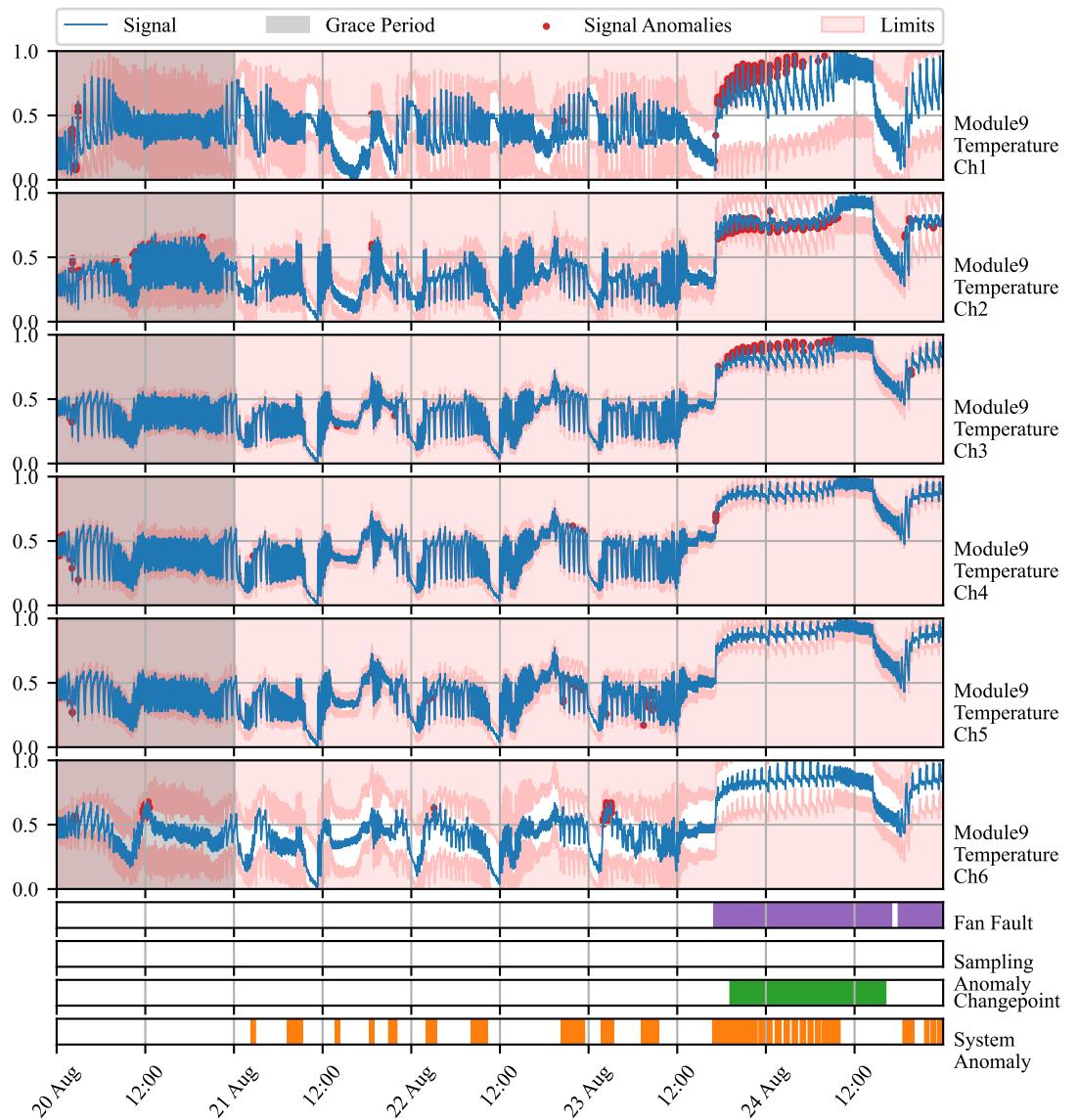


Figure 2: Time Series of battery module 9 measurements (blue line) of process variables. The y-axis renders the normalized deviations of temperature from average of all 12 modules. Signal anomalies are marked as red dots. The light red area represents out-of-limits values for individual signals. True fan faults are marked by purple bars. Green bars represent the times, at which changepoint was detected. All the signal anomalies are depicted as orange bars below the graph.

620 Moreover, it combines adaptability and interpretability, which is an area yet
621 to be explored.

622 The benchmark performed on industrial data showed the ability to pro-
623 vide comparable results to other self-learning adaptable anomaly detection
624 methods. This is an important property for our model which allows, in ad-
625 dition, the root cause isolation.

626 The first case study, performed on real operation data of Battery Energy
627 Storage Systems (BESS), demonstrated AID's effectiveness in capturing sys-
628 tem anomalies, providing less conservative signal limits, and leveraging a
629 physical model for temperature anomaly detection.

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

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

641 References

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

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

651 A. Kejariwal, Introducing practical and robust anomaly
652 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.
653
654

- 655 A. A. Cook, G. Mısırlı, Z. Fan, Anomaly detection for iot time-series data:
656 A survey, *IEEE Internet of Things Journal* 7 (2020) 6481–6494. doi:10.
657 1109/JIOT.2019.2958185.
- 658 K. Zhang, J. Chen, C.-G. Lee, S. He, An unsupervised spatiotemporal fusion network augmented with random mask and time-
659 relative information modulation for anomaly detection of machines
660 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>.
- 665 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>.
- 670 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>.
- 675 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>.
- 680 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>.
- 685 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/S0957417423018067>.

- 688 com/science/article/pii/S0957417423016639. doi:<https://doi.org/10.1016/j.eswa.2023.121161>.
- 690 M. Salehi, L. Rashidi, A survey on anomaly detection in evolving data:
691 [with application to forest fire risk prediction], SIGKDD Explor. Newsl.
692 20 (2018) 13–23. URL: <https://doi.org/10.1145/3229329.3229332>.
693 doi:10.1145/3229329.3229332.
- 694 N. Barbosa Roa, L. Travé-Massuyès, V. H. Grisales-Palacio, Dy-
695 clee: Dynamic clustering for tracking evolving environments, Pat-
696 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>.
- 699 A. G. Tartakovsky, A. S. Polunchenko, G. Sokolov, Efficient computer net-
700 work anomaly detection by changepoint detection methods, IEEE Journal
701 of Selected Topics in Signal Processing 7 (2013) 4–11. doi:10.1109/JSTSP.
702 2012.2233713.
- 703 H. Wu, J. He, M. Tömösközi, Z. Xiang, F. H. Fitzek, In-network processing
704 for low-latency industrial anomaly detection in softwarized networks, in:
705 2021 IEEE Global Communications Conference (GLOBECOM), 2021, pp.
706 01–07. doi:10.1109/GLOBECOM46510.2021.9685489.
- 707 H. S. Pannu, J. Liu, S. Fu, Aad: Adaptive anomaly detection system for cloud
708 computing infrastructures, in: 2012 IEEE 31st Symposium on Reliable
709 Distributed Systems, 2012, pp. 396–397. doi:10.1109/SRDS.2012.3.
- 710 S. Ahmad, A. Lavin, S. Purdy, Z. Agha, Unsupervised real-time anomaly
711 detection for streaming data, Neurocomputing 262 (2017) 134–
712 147. 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.
- 715 H. H. Bosman, G. Iacca, A. Tejada, H. J. Wörtche, A. Liotta, Ensembles
716 of incremental learners to detect anomalies in ad hoc sensor networks,
717 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
719 Sensing, Processing and Networking Technologies.

- 721 M. Carletti, C. Masiero, A. Beghi, G. A. Susto, Explainable machine learning
722 in industry 4.0: Evaluating feature importance in anomaly detection to
723 enable root cause analysis, in: 2019 IEEE International Conference on
724 Systems, Man and Cybernetics (SMC), 2019, pp. 21–26. doi:10.1109/SMC.
725 2019.8913901.
- 726 Q. P. Nguyen, K. W. Lim, D. M. Divakaran, K. H. Low, M. C. Chan, Gee: A
727 gradient-based explainable variational autoencoder for network anomaly
728 detection, in: 2019 IEEE Conference on Communications and Network
729 Security (CNS), 2019, pp. 91–99. doi:10.1109/CNS.2019.8802833.
- 730 K. Amarasinghe, K. Kenney, M. Manic, Toward explainable deep neural
731 network based anomaly detection, in: 2018 11th International Conference
732 on Human System Interaction (HSI), 2018, pp. 311–317. doi:10.1109/HSI.
733 2018.8430788.
- 734 X. Zhang, J. Shi, X. Huang, F. Xiao, M. Yang, J. Huang,
735 X. Yin, A. Sohail Usmani, G. Chen, Towards deep probabilistic
736 graph neural network for natural gas leak detection and localiza-
737 tion without labeled anomaly data, Expert Systems with Appli-
738 cations 231 (2023) 120542. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423010448>. doi:<https://doi.org/10.1016/j.eswa.2023.120542>.
- 741 W.-T. Yang, M. S. Reis, V. Borodin, M. Juge, A. Roussy, An interpretable
742 unsupervised bayesian network model for fault detection and diagno-
743 sis, Control Engineering Practice 127 (2022) 105304. URL: <https://www.sciencedirect.com/science/article/pii/S0967066122001502>.
744 doi:<https://doi.org/10.1016/j.conengprac.2022.105304>.
- 746 L. C. Brito, G. A. Susto, J. N. Brito, M. A. V. Duarte, Fault diagno-
747 sis using explainable ai: A transfer learning-based approach for rotating
748 machinery exploiting augmented synthetic data, Expert Systems with
749 Applications 232 (2023) 120860. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423013623>. doi:<https://doi.org/10.1016/j.eswa.2023.120860>.
- 752 Z. Wu, X. Yang, X. Wei, P. Yuan, Y. Zhang, J. Bai, A self-supervised
753 anomaly detection algorithm with interpretability, Expert Systems with
754 Applications 237 (2024) 121539. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423013623>.

- 755 com/science/article/pii/S0957417423020419. doi:<https://doi.org/10.1016/j.eswa.2023.121539>.
- 756
- 757 M. Wadinger, M. Kvasnica, Real-time outlier detection with dynamic process
758 limits, in: 2023 24th International Conference on Process Control (PC),
759 2023, pp. 138–143. doi:[10.1109/PC58330.2023.10217717](https://doi.org/10.1109/PC58330.2023.10217717).
- 760 K. Yamanishi, J.-i. Takeuchi, A unifying framework for detecting outliers and
761 change points from non-stationary time series data, in: Proceedings of the
762 Eighth ACM SIGKDD International Conference on Knowledge Discovery
763 and Data Mining, KDD '02, Association for Computing Machinery, New
764 York, NY, USA, 2002, pp. 676–681. URL: <https://doi.org/10.1145/775047.775148>. doi:[10.1145/775047.775148](https://doi.org/10.1145/775047.775148).
- 765
- 766 K. Yamanishi, J.-i. Takeuchi, G. Williams, P. Milne, On-line unsupervised outlier detection using finite mixtures with discounting learning algorithms, Data Mining and Knowledge Discovery 8 (2004) 275–
767 300. URL: <https://doi.org/10.1023/B:DAMI.0000023676.72185.7c>.
768 doi:[10.1023/B:DAMI.0000023676.72185.7c](https://doi.org/10.1023/B:DAMI.0000023676.72185.7c).
- 769
- 770
- 771 B. Steenwinckel, Adaptive anomaly detection and root cause analysis by fusing
772 semantics and machine learning, in: A. Gangemi, A. L. Gentile, A. G.
773 Nuzzolese, S. Rudolph, M. Maleshkova, H. Paulheim, J. Z. Pan, M. Alam
774 (Eds.), The Semantic Web: ESWC 2018 Satellite Events, Springer Interna-
775 tional Publishing, Cham, 2018, pp. 272–282.
- 776
- 777 B. Steenwinckel, D. De Paepe, S. Vanden Hautte, P. Heyvaert, M. Bentefrit,
778 P. Moens, A. Dimou, B. Van Den Bossche, F. De Turck, S. Van
779 Hoecke, F. Ongena, Flags: A methodology for adaptive anomaly
780 detection and root cause analysis on sensor data streams by fusing
781 expert knowledge with machine learning, Future Generation Com-
782 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>.
- 783
- 784 M. Amer, M. Goldstein, S. Abdennadher, Enhancing one-class support vec-
785 tor machines for unsupervised anomaly detection, in: Proceedings of the
786 ACM SIGKDD Workshop on Outlier Detection and Description, ODD '13,
787 Association for Computing Machinery, New York, NY, USA, 2013, pp.

- 788 8–15. URL: <https://doi.org/10.1145/2500853.2500857>. doi:10.1145/2500853.2500857.
- 790 B. Liu, Y. Xiao, P. S. Yu, L. Cao, Y. Zhang, Z. Hao, Uncertain one-class learning and concept summarization learning on uncertain data streams, *IEEE Transactions on Knowledge and Data Engineering* 26 (2014) 468–484. doi:10.1109/TKDE.2012.235.
- 794 B. Krawczyk, M. Woźniak, One-class classifiers with incremental learning and forgetting for data streams with concept drift, *Soft Computing* 19 (2015) 3387–3400. URL: <https://doi.org/10.1007/s00500-014-1492-5>. doi:10.1007/s00500-014-1492-5.
- 798 X. Miao, Y. Liu, H. Zhao, C. Li, Distributed online one-class support vector machine for anomaly detection over networks, *IEEE Transactions on Cybernetics* 49 (2019) 1475–1488. doi:10.1109/TCYB.2018.2804940.
- 801 Ö. Gözüaçık, F. Can, Concept learning using one-class classifiers for implicit drift detection in evolving data streams, *Artificial Intelligence Review* 54 (2021) 3725–3747. URL: <https://doi.org/10.1007/s10462-020-09939-x>. doi:10.1007/s10462-020-09939-x.
- 805 R. Wetzig, A. Gulenko, F. Schmidt, Unsupervised anomaly alerting for iot-gateway monitoring using adaptive thresholds and half-space trees, in: 2019 Sixth International Conference on Internet of Things: Systems, Management and Security (IOTSMS), 2019, pp. 161–168. doi:10.1109/IOTSMS48152.2019.8939201.
- 810 Y. Lyu, W. Li, Y. Wang, S. Sun, C. Wang, Rmhsforest: Relative mass and half-space tree based forest for anomaly detection, *Chinese Journal of Electronics* 29 (2020) 1093–1101. doi:<https://doi.org/10.1049/cje.2020.09.010>.
- 814 B. P. Welford, Note on a method for calculating corrected sums of squares and products, *Technometrics* 4 (1962) 419–420. doi:10.1080/00401706.1962.10490022.
- 817 S. Mishra, A. Datta-Gupta, Chapter 3 - distributions and models thereof, in: S. Mishra, A. Datta-Gupta (Eds.), *Applied Statistical Modeling and Data Analytics*, Elsevier, 2018, pp. 31–67. URL: <https://www.sciencedirect.com>.

- 820 com/science/article/pii/B9780128032794000031. doi:<https://doi.org/10.1016/B978-0-12-803279-4.00003-1>.
- 821
- 822 A. Genz, Numerical computation of multivariate normal probabilities, Journal
823 of Computational and Graphical Statistics 1 (2000). doi:10.1080/
824 10618600.1992.10477010.
- 825 F. Iglesias Vázquez, A. Hartl, T. Zseby, A. Zimek, Anomaly detection in
826 streaming data: A comparison and evaluation study, Expert Systems with
827 Applications 233 (2023) 120994. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423014963>. doi:<https://doi.org/10.1016/j.eswa.2023.120994>.
- 826
- 827 L. Ruff, J. R. Kauffmann, R. A. Vandermeulen, G. Montavon, W. Samek,
828 M. Kloft, T. G. Dietterich, K.-R. Müller, A unifying review of deep and
829 shallow anomaly detection, Proceedings of the IEEE 109 (2021) 756–795.
830 doi:10.1109/JPROC.2021.3052449.
- 831
- 832 I. D. Katser, V. O. Kozitsin, Skoltech anomaly benchmark (skab),
833 <https://www.kaggle.com/dsv/1693952>, 2020. doi:10.34740/KAGGLE/DSV/1693952.
- 834
- 835
- 836