electronics

MDPI

Article

An Interpretable Agent-Assisted Pipeline for Statistical Anomaly Detection in IoT Temperature Time Series

Luis Miguel Pires [1,2,3,*] and José Braga de Vasconcelos [3,4,*]

[1] Technologies and Engineering School (EET), Instituto Politécnico da Lusofonia (IPLuso), 1700-098 Lisbon, Portugal
[2] Department of Electronical Engineering, Telecommunications and Computers (DEETC), Instituto Superior de Engenharia de Lisboa (ISEL), 1959-007 Lisbon, Portugal
[3] School of Communication, Arts and Information (ECATI), Lusofona University, 1749-024 Lisbon, Portugal
[4] Faculty of Natural Sciences, Engineering, and Technology (FCNET), Lusofona University, 4000-098 Porto, Portugal
* Correspondence: luis.pires@ipluso.pt (L.M.P.); jose.vasconcelos@ulusofona.pt (J.B.d.V.)

Abstract

The research presents an interpretable framework which detects anomalies in IoT temperature time-series data with low complexity for use in edge environments that lack resources. The proposed solution uses three traditional statistical filters which include Hampel and Interquartile Range (IQR) and Z-Score to build an agent-assisted decision layer which selects the best method through a multi-criteria cost function. The framework runs tests on a structured synthetic dataset which contains seven different anomaly tests and on an actual IoT dataset which was gathered from eight separate sensor points. The researchers use standard anomaly detection metrics which include precision and recall and F1-score and false positive rate to conduct their complete evaluation. The proposed method is tested against two machine learning baseline methods which are Isolation Forest and One-Class Support Vector Machine (OC-SVM). The results show that the agent-assisted method achieves detection results which match industry standards while showing high interpretability and low processing needs. The framework demonstrates its ability to function in actual IoT environments through its use of authentic real-world data, and also basic statistical techniques together with an adjustable decision system create a strong and understandable method to detect anomalies in IoT sensing systems.

Keywords: Internet of Things (IoT); time-series anomaly detection; statistical anomaly detection; Hampel filter; interquartile range (IQR); Z-score; interpretable anomaly detection; sensor data monitoring

Academic Editor: Shuo Yu

Received: 12 March 2026
Revised: 17 April 2026
Accepted: 23 April 2026
Published: 27 April 2026

1. Introduction

Internet of Things (IoT) experiences rapid growth which results in extensive environmental sensor data collection through its various applications that include smart buildings, environmental monitoring, industrial supervision and Industry 4.0. The systems use temperature sensors because these sensors provide affordable and energy-efficient solutions that are simple to implement. Long-term operation of IoT monitoring systems encounter data anomalies which result from sensor malfunctions, communication issues, calibration errors and environmental factors. Monitoring systems experience severe statistical analysis disruptions because these anomalies cause data integrity issues which decrease system reliability for monitoring and decision-support functions.

Detection of anomalies serves as an essential element which supports the entire IoT data processing system. The deployment of anomaly detection systems needs to establish a detection system which can identify anomalies while safeguarding important signals and it should operate with minimal processing requirements which are appropriate for edge and embedded systems. The IoT monitoring field still finds classical statistical methods attractive despite the emergence of advanced machine learning methods because these methods offer high interpretability and strong performance and they require minimal computational power. Edge computing paradigms have emerged as a key solution to address latency and computational constraints in IoT systems [1–4].

Multiple real-world applications use IoT temperature sensing systems in cold-chain logistics which need to maintain specific temperature ranges such as for food and pharmaceutical safety, smart buildings which use temperature monitoring to improve energy efficiency and resident comfort, and industrial settings which need to detect equipment problems through temperature monitoring. Existing systems require dependable methods to discover irregularities which maintain their ability to operate under multiple environmental circumstances. The field of anomaly detection has developed new techniques which combine statistical methods with deep learning methods according to research studies which show the benefits and drawbacks of each approach [5–9]. Sensor-based systems rely on time-series anomaly detection which researchers consider an important research area [9] while deep learning techniques handle challenging tasks in complex settings according to results from their implementation [5,6]. The public continues to prefer traditional methods because they provide clear understanding and require less processing power according to research findings [8].

To solve these problems, the paper presents a framework which enables researchers to conduct comparative analyses while detecting anomalies and repairing signals in IoT temperature time-series data. The system integrates statistical filters with a decision-making component which determines the best solution based on various performance metrics.

The system requires less manual adjustment work because it maintains both signal quality and the ability to understand results. All components of this study including the dataset used in the research are accessible to the public which enables others to reproduce and reuse the study results [10,11]. The research establishes a unified testing system which allows for simultaneous assessment of multiple statistical detectors because previous studies only tested individual detectors. The research adopts three primary components which include a multi-scenario IoT temperature dataset that represents various anomaly patterns and a systematic assessment of different statistical filters. Rather than being an agent of reinforcement learning, it should be seen as an interpretable rule selection mechanism. The study used established statistical methods, yet its primary achievement resides in creating an evaluation system that enables transparent assessment of IoT temperature monitoring systems. The proposed architecture combines several statistical detectors into a flexible data processing system which employs a simple decision system that chooses the best detector based on the specific anomaly patterns detected in the signal. Research presents a complete synthetic dataset which simulates actual IoT anomaly detection scenarios, thus providing researchers with a tool to conduct controlled tests and verify their worry detection methods. The proposed framework offers essential requirements for IoT edge deployments through its three core features because it needs to show results in a way that users can understand and reproduce its work while maintaining low operating demands.

The detection of temperature time-series anomalies through IoT sensing systems remains difficult because temperature signals experience different types of disturbances and researchers have not developed effective methods that can adapt yet still provide clear explanations. Researchers prefer classical statistical filters because these filters offer easy

application and require minimal processing resources although their effectiveness depends on how the actual signals behave. In this context, authors developed an interpretable system which unites various statistical filtering techniques through a financial decision-making system to overcome existing constraints. The system determines which filter to use by assessing the condition of the signal through integrity metrics which allows the system to modify its operation while maintaining clear understanding of its functions. The main contributions of this work are: (i) a structured multi-scenario dataset for systematic evaluation, (ii) a comparative analysis of statistical filters under diverse conditions, and (iii) an agent-assisted decision mechanism for adaptive filter selection.

The remainder of this paper is organized as follows. Section 2 reviews the related work on statistical and machine learning approaches for time-series anomaly detection. The proposed anomaly detection framework together with its statistical filtering methods are presented in Section 3. The dataset generation process together with the experimental scenarios are explained in Section 4. Section 5 presents the results of the experiments together with the findings from the comparative evaluation. The paper ends in Section 6 which summarizes the content and provides recommendations for upcoming research.

2. State of the Art and Related Works

This section presents the current state of the art in Section 2.1, and in Section 2.2 are some related works about the topic of study.

2.1. State of the Art

Anomaly detection in IoT temperature time-series data is a fundamental preprocessing task in automated monitoring systems, where sensor readings directly influence control actions, alarm generation, and compliance verification. In long-term deployments, temperature sensors are frequently exposed to impulsive noise, gradual drift, sensor degradation, and non-Gaussian disturbances, which significantly limit the effectiveness of conventional threshold-based approaches.

Despite these challenges, statistical methods remain widely adopted in this context due to their low computational requirements, predictable behavior, and transparent decision logic. The development of robust statistics strengthened this position by promoting estimators that explicitly limit sensitivity to extreme values and heavy-tailed distributions [12]. As a result, median- and quantile-based measures are particularly suitable for IoT environments, where even a small number of outliers can severely distort mean- and variance-based estimators.

Among statistical filtering techniques, Hampel filter is well known for its robustness and suitability for real-time applications [13]. IQR-based detection relies on non-parametric bounds derived from exploratory data analysis principles, offering distribution-agnostic behavior that is especially useful when the underlying signal statistics are unknown or time-varying [14]. Z-Score-based detection remains a common baseline due to its simplicity and ease of implementation; however, its performance is strongly dependent on variance stability and near-Gaussian assumptions, which are often violated in real IoT scenarios [15].

Beyond algorithmic considerations, there is a growing demand for trustworthy and traceable data pipelines. Sustainability-oriented monitoring, as promoted by the United Nations (UN) Sustainable Development Goals (SDGs), relies heavily on accurate and trustworthy sensor data to support resilient infrastructures and responsible resource usage [16]. In parallel, emerging regulatory initiatives such as the European Union (EU) Digital Product Passport (DPP) require auditable and verifiable data streams to ensure lifecycle transparency and regulatory compliance across value chains [17,18]. In such contexts,

anomaly detection mechanisms based on explicit decision rules are clearly preferable to opaque or black-box solutions.

The authors established interpretability through their definition which enables users to comprehend and elucidate how decisions are made during the process of detecting anomalies and choosing filtering methods. The framework uses transparent statistical metrics together with a deterministic cost function which includes components that have physical significance to create its assessment framework while black-box machine learning models maintain hidden operational methods. Researchers assessed interpretability through three evaluation criteria which included transparency and reproducibility and direct connection between input signal characteristics and chosen filtering methods.

2.2. Related Works

The methodological foundations of the statistical filters considered in this study are supported by several well-established works. Hampel filter [19] was introduced by Pearson et al. as an efficient and robust outlier detection algorithm suitable for real-time signal processing [13]. Classical statistical approaches remain relevant due to their interpretability and low computational cost [19–22], while recent methods rely on deep learning and unsupervised techniques [5–9]. More recently, Roos-Hoefgeest Toribio et al. proposed computational optimizations that significantly reduce the execution time of the Hampel filter, further strengthening its applicability to embedded and IoT systems [23].

Z-Score-based anomaly detection has been applied in various time-series sensor contexts. Yaro et al. demonstrated its use in indoor localization by incorporating a robust scale estimator to reduce sensitivity to extreme values [15]. Nevertheless, variance-based approaches remain vulnerable to non-stationarity and variance distortion under heavy-tailed noise, limiting their robustness in long-term deployments.

Quantile-based approaches originate from Tukey's [20] exploratory data analysis framework, which laid the foundation for distribution-agnostic outlier detection methods [15]. These techniques are particularly effective when the underlying data distribution is unknown or clearly non-Gaussian, a common situation in IoT temperature monitoring.

From an engineering and evaluation perspective, classical references emphasize the importance of comparative assessment and multi-criteria analysis when selecting data preprocessing techniques, rather than relying on a single performance indicator [24]. However, the existing literature rarely provides openly available datasets that simultaneously include multiple anomaly regimes, reconstructed signals, and integrity-oriented performance metrics within a single reproducible framework.

The field of time-series anomaly detection has advanced from traditional statistical methods to machine learning techniques which enable the detection of intricate patterns that fixed statistical methods cannot recognize. Unsupervised algorithms such as Isolation Forest (IF) [25], One-Class Support Vector Machines (OC-SVM) [26], and local outlier factors have been widely adopted for anomaly detection in high-dimensional and streaming data environments. The approaches become useful in situations when there is a lack of anomaly labels because this scenario occurs frequently in actual IoT systems which experience rare abnormal events and possess limited ground truth information. Recent developments in edge intelligence further enable the deployment of lightweight and adaptive anomaly detection mechanisms closer to the data source [2].

Deep learning methods [27] now exist to model both the temporal dependencies and contextual relationships present in time-series signals. Research includes Long Short-Term Memory (LSTM)-based autoencoders, one-class neural architecture, and recently developed diffusion-based models which function as tools for anomaly detection. The complex datasets enable these methods to achieve high detection accuracy, but the methods

need more training data and better computing power while becoming harder to understand than simpler statistical detection methods. The system needs these features, especially for monitoring environments that lack resources and for systems which demand quick processing and understandable results.

Beyond classical statistical techniques, anomaly detection in time-series data has also been extensively studied using machine learning and deep learning approaches. Examples include IF, OC-SVM, and local outlier factors for unsupervised anomaly detection. More recent approaches employ deep neural architecture such as LSTM autoencoders and diffusion-based models to capture complex temporal dependencies. Benchmarking frameworks such as Time-Series Benchmarking (TSB)–Univariate Anomaly Detection (UAD) [28] provide standardized datasets and evaluation protocols for comparing anomaly detection algorithms across diverse scenarios. The methods produce high accuracy for detection but need extensive training data and high processing power which restricts their use in IoT monitoring systems that have limited available resources.

Benchmarking initiatives help researchers identify which aspects of different anomaly detection methods function effectively and which aspects exhibit limitations. The univariate time-series anomaly detection methods which are evaluated through large-scale testing prove that no single detection method can achieve superior results over all other methods in every possible anomaly detection scenario. This research shows that signal monitoring systems should use adaptive methods which select detection techniques based on the monitored signal's statistical properties. Temperature monitoring systems in IoT environments need to consider these factors because their signals can maintain stable behavior while experiencing sudden changes, slow changes, complete segment corruption, or any combination of these patterns.

The literature provides multiple anomaly detection methods which include both traditional statistical methods and machine learning and deep learning techniques. The main features of different methods are presented in Table 1 which shows their ability to explain results, their processing demands, and their effectiveness in monitoring Internet of Things environments.

Table 1. Comparison of representative anomaly detection approaches for IoT time-series monitoring.

Method	Category	Interpretability	Computational Cost	Typical IoT Suitability
IF	Machine Learning	Low	Medium	Moderate
OC-SVM	Machine Learning	Low	High	Limited
LSTM Autoencoder	Deep Learning	Very Low	Very High	Limited
TSB-UAD Benchmark Methods	Mixed	Variable	Variable	Evaluation Framework
Proposed Framework	Statistical and Decision Agent	High	Low	High

This study addresses this gap by releasing a structured dataset, performing a comparative evaluation of three statistical filters, and introducing an interpretable agent-assisted mechanism for adaptive filter selection. Table 2 summarizes the related work and positions of the present contribution.

Table 2. Summary of related works and positioning of this study.

Ref.	Focus	Method	Key Limitation	Contribution
[13]	Real-time outlier detection	Hampel filter	Single-method focus	Comparative multi-filter evaluation
[20]	Computational efficiency	Accelerated Hampel	No multi-scenario analysis	Integrated into full pipeline
[15]	Time-series outliers	Z-Score and Sn	Sensitive to non-stationarity	Baseline under diverse scenarios
[14]	Exploratory analysis	IQR bounds	Not time-series specific	Sliding-window IoT application
[12]	Robust statistics	Theoretical foundation	No IoT focus	Practical dataset implementation
[27]	Deep anomaly detection	LSTM/neural models	Higher computational complexity	Captures complex temporal patterns
[28]	Benchmark evaluation	Multiple anomaly detection algorithms	Not specific to IoT temperature monitoring	Provides standardized evaluation datasets
This work	IoT temperature	Hampel, IQR, Z-Score and agent selection	No real temperature sensors data	Adaptive, interpretable selection

The recent surveys conducted in [29,30] demonstrate that research on anomaly detection for IoT networks and sensor networks has evolved into a significant research domain which uses statistical techniques, machine learning methods, and AI-based systems to solve problems. The research demonstrates that anomaly detection methods need evaluation through three specific criteria which include their accuracy performance, their suitability for different application domains, their computational demands, and their implementation limitations.

The methods together establish three different statistical approaches which include robust statistical methods, parametric statistical methods, and unsupervised machine learning (ML) techniques to create an equal foundation for conducting comparative assessments.

3. Data Generation and Processing Methods

In this section, the methodology that was employed to produce, handle, and analyze the IoT temperature dataset is detailed. The goal is to give an open and replicable account of the entire data pipeline which comprises signal generation, anomaly modeling, statistical filtering, and metric computation.

3.1. Overall Data Processing Pipeline

Figure 1 summarizes the complete processing workflow adopted in this study. The pipeline follows a sequential and modular structure comprising data loading, anomaly detection, signal reconstruction, metric computation, and visualization. All statistical filters are applied under identical conditions to ensure fair comparison across scenarios. This design ensures traceability from raw data to reconstructed signals and reported metrics, supporting reproducible experimental evaluation. When the input file is ready, it is the temperature time-series data that are loaded into a structured data frame comprising the timestamp and temperature values. From here, the signal is processed in a series of three statistical anomaly detection methods, namely the Hampel filter, the IQR filter, and the Z-

Score filter, applied sequentially. Every single method outputs a repair version of the signal along with the anomaly flag vector that corresponds to it and thus enables comparison of all filters to be done consistently.

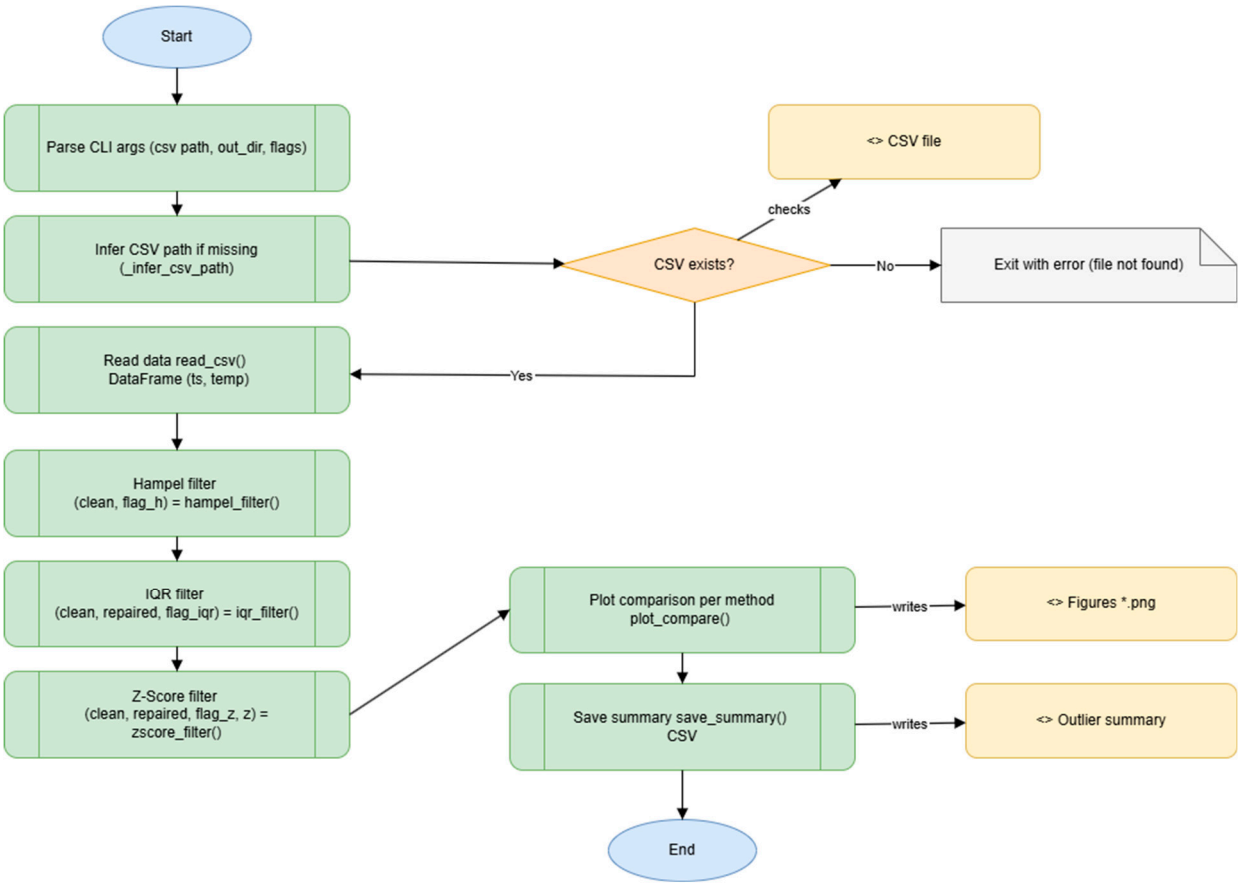

Figure 1. Main pipeline of Unified Modeling Language (UML) activity diagram.

All statistical filters operate in parallel, each receiving the same original time-series input signal. No sequential dependency exists between filtering nodes, ensuring a fair and consistent comparison across methods.

3.2. Statistical Filtering Models

Three statistical filters are implemented to detect and repair anomalies in the temperature time series. Each filter operates on a sliding window and applies a different statistical assumption regarding central tendency and dispersion.

To ensure a fair comparison, all filters employ consistent anomaly repair strategies. Detected anomalies are replaced using local statistical estimators: Hampel filter replaces anomalies with the window median, while IQR- and Z-Score-based detections are followed by median-based local interpolation. This standardization ensures that performance differences arise from detection capabilities rather than reconstruction inconsistencies. These methods are grounded in robust statistics and exploratory data analysis principles [19–22], ensuring reliable performance under noisy conditions.

Hampel filter [13,14] is a resilient median-based outlier detection method. For a given window $W = \{x_{i-k}, \ldots, x_i, \ldots, x_{i+k}\}$, the central tendency is estimated using the median:

$$m_i = \text{median}(W) \quad (1)$$

The dispersion is estimated using the Median Absolute Deviation (MAD):

$$\mathrm{MAD}_i = \mathrm{median}(|\, x_j - m_i \,|), x_j \in W \tag{2}$$

An observation x_i is classified as an outlier if

$$|\, x_i - m_i \,| > \alpha \cdot \mathrm{MAD}_i \tag{3}$$

where α is a tunable threshold parameter. When an outlier is detected, the value is replaced by the median m_i, resulting in a repaired signal that is less sensitive to extreme deviations while preserving the local trend. Figure 2 presents the Unified Modeling Language (UML) activity diagram of the Hampel filtering module, highlighting its median-based decision logic, robust MAD estimation, and deterministic outlier replacement strategy.

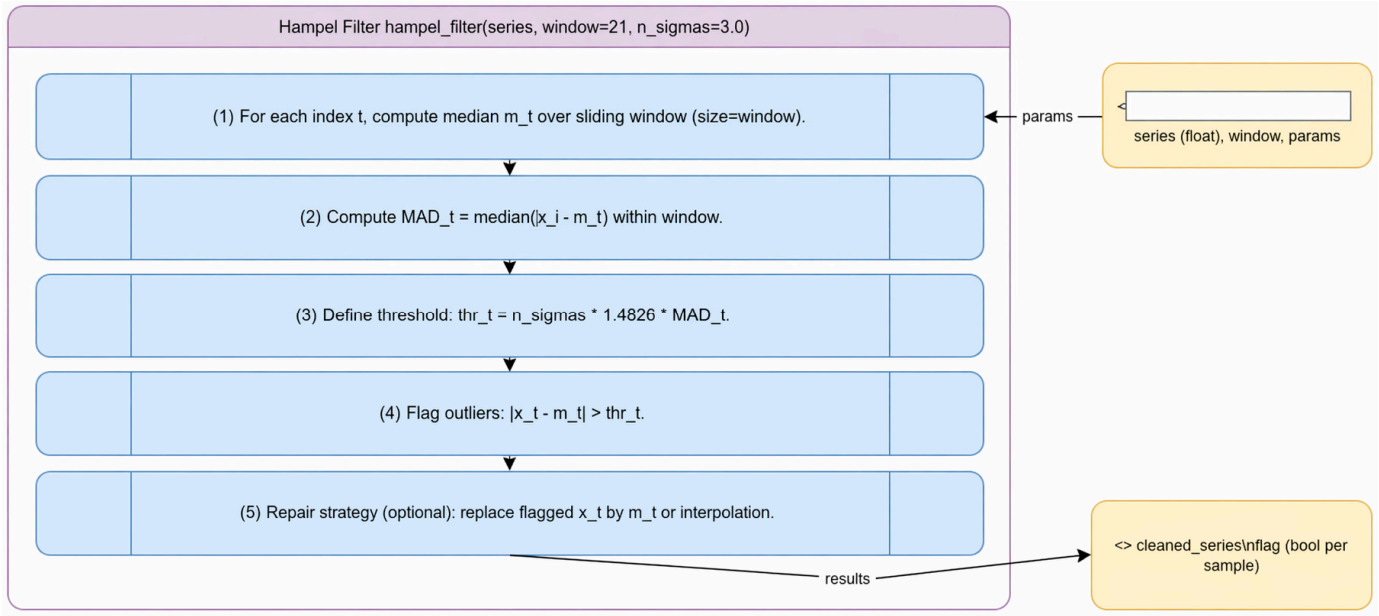

Figure 2. Hampel filter UML diagram.

IQR filter [12,14,15] is a non-parametric method based on quartile statistics. For each sliding window, the first and third quartiles, Q_1 and Q_3, are computed:

$$\mathrm{IQR} = Q_3 - Q_1 \tag{4}$$

Outliers are identified using Tukey-style bounds [6]:

$$x_i < Q_1 - \beta \cdot \mathrm{IQR or}\ x_i > Q_3 + \beta \cdot \mathrm{IQR} \tag{5}$$

where β controls the aggressiveness of the filter. Detected outliers are replaced by a robust central estimate (typically the window median), ensuring continuity of the reconstructed signal.

This approach is particularly effective in the presence of heavy-tailed noise and dense impulsive anomalies, where variance-based methods tend to fail. Figure 3 presents the UML activity diagram of the IQR filtering module, illustrating the computation of quartile-based bounds and the detection of anomalies using non-parametric statistics.

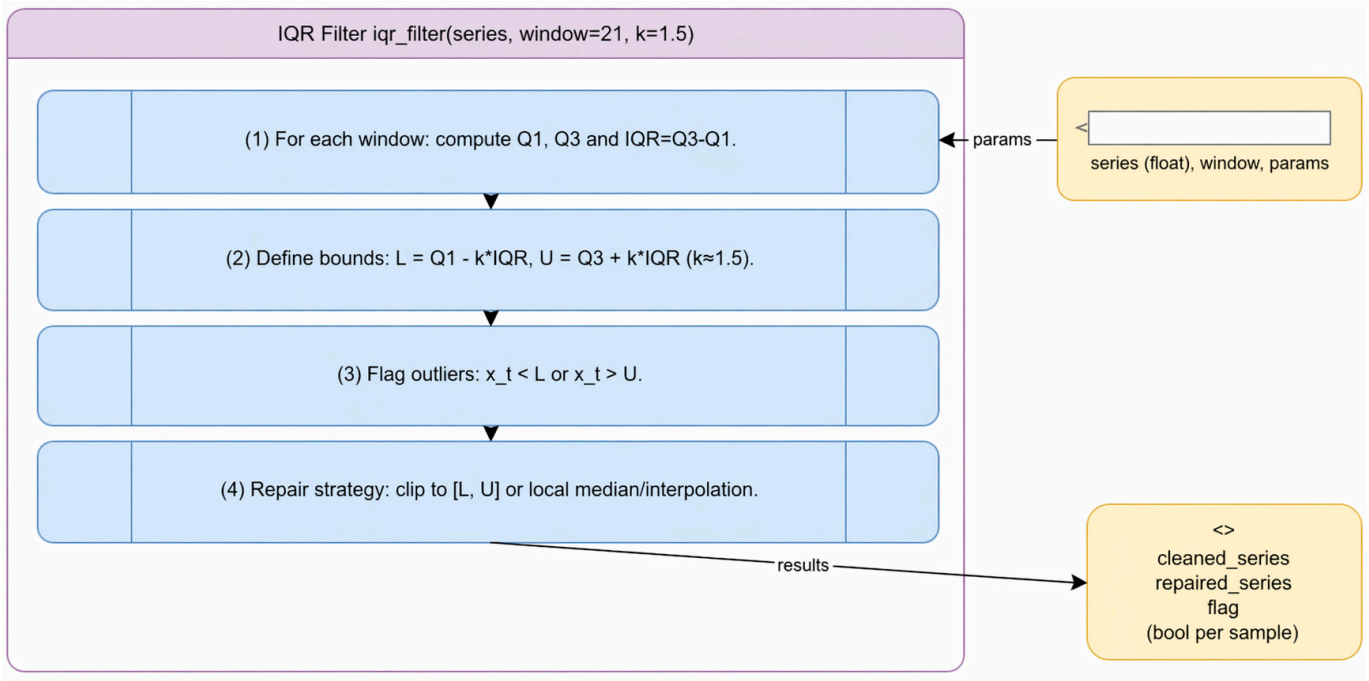

Figure 3. IQR filter UML diagram.

Z-Score filter [12,14,15] assumes approximate normality within each window. The mean μ_i and standard deviation σ_i are computed as

$$\mu_i = \frac{1}{N}\sum_{j=1}^{N} x_j, \sigma_i = \sqrt{\frac{1}{N}\sum_{j=1}^{N}(x_j - \mu_i)^2} \quad (6)$$

The standardized score is then defined as

$$z_i = \frac{x_i - \mu_i}{\sigma_i} \quad (7)$$

An observation is flagged as anomalous if

$$| z_i | > \gamma \quad (8)$$

where γ is a predefined threshold. Anomalous values are replaced by the local mean or by an interpolated value, depending on the reconstruction strategy. While computationally efficient, this method is sensitive to variance inflation and non-Gaussian behavior, making it less robust under heavy-tailed or highly non-stationary conditions. Figure 4 presents the UML activity diagram of the Z-Score filtering module, emphasizing its reliance on local mean and variance estimation and its sensitivity to non-Gaussian behavior.

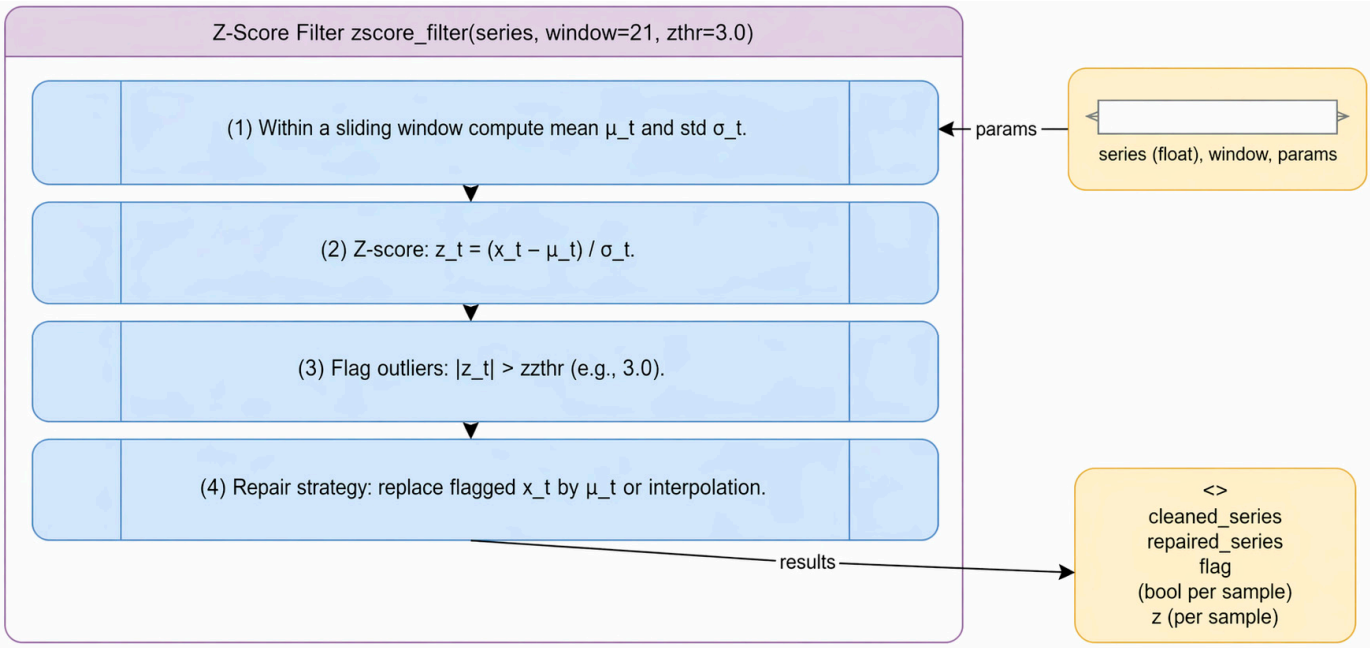

Figure 4. Z-Score filter UML diagram.

3.3. Machine Learning-Based Baseline Methods

The study used two traditional statistical filtering methods together with two lightweight machine learning-based anomaly detection methods (IF and OC-SVM) which function as baseline comparison points. The methods were chosen because they are commonly used to detect anomalies and they can work without needing any labeled training data. Models should be included because they will help assess performance when testing the proposed statistical framework against more flexible data-driven testing conditions. The ensemble-based anomaly detection method IF uses recursive data space partitioning to identify abnormal observations. The main concept establishes that an anomaly needs fewer tree structure splits because it becomes easier to isolate from regular data. The anomaly score for a data point x is defined as [25]

$$s(x,n) = 2^{-\frac{E(h(x))}{c(n)}} \tag{9}$$

where

- $s(x,n)$ is the anomaly score of sample x given a dataset of size n;
- $E(h(x))$ is the expected path length of x across the ensemble of isolation trees;
- $c(n)$ is a normalization factor corresponding to the average path length of unsuccessful searches in a binary tree, defined as [25]

$$c(n) = 2H(n-1) - \frac{2(n-1)}{n} \tag{10}$$

with

- $H(n)$ being the harmonic number, approximated by $\ln(n) + \gamma$;
- $\gamma \approx 0.5772$ (Euler–Mascheroni constant).

Shorter path lengths indicate that a point is easier to isolate and therefore more likely to be anomalous. IF is particularly suitable for high-dimensional or complex data distributions, although its computational cost is higher than that of simple statistical filters.

OC-SVM is a boundary-based method that learns a decision function describing the region in feature space where normal data resides. It aims to separate the data from the

origin with maximum margin in a transformed feature space. The optimization problem is defined as [26]

$$\min_{w,\rho,\xi}\left(\frac{1}{2}\parallel w\parallel^2+\frac{1}{\nu n}\sum_{i=1}^{n}\xi_i-\rho\right) \tag{11}$$

subject to

$$w\cdot\phi(x_i)\geq\rho-\xi_i,\xi_i\geq 0 \tag{12}$$

where

- x_i represents the input samples;
- $\phi(x_i)$ is a mapping to a higher-dimensional feature space;
- w is the normal vector defining the decision boundary;
- ρ is the offset (threshold) of the decision function;
- ξ_i are slack variables allowing soft margin violations;
- $\nu\in[0,1]$ is a hyperparameter controlling the trade-off between the fraction of outliers and the model complexity;
- n is the number of training samples.

A sample is considered anomalous if it lies outside the learned boundary:

$$f(x)=\text{sign}(w\cdot\phi(x)-\rho) \tag{13}$$

OC-SVM is effective for capturing non-linear structures when combined with kernel functions, such as the Radial Basis Function (RBF). However, it introduces additional computational overhead and reduced interpretability compared to statistical methods.

Although both methods provide flexible modeling capabilities, they present limitations in the context of resource-constrained IoT systems:

- Higher computational cost (especially IF);
- Increased memory requirements;
- Reduced interpretability compared to statistical filters.

Therefore, these models are used strictly as comparative baselines, allowing a balanced evaluation between classical statistical approaches and more flexible machine learning techniques.

Although machine learning-based methods such as IF and OC-SVM introduce greater flexibility in handling complex data patterns, their performance is not consistent across all scenarios. In practice, no single method proved to be universally optimal, with each approach showing strengths under specific conditions. This observation highlights the need for a more adaptive strategy, capable of selecting the most appropriate method depending on the characteristics of the signal. To address this, the following section introduces a cost-based decision framework, where an agent evaluates multiple criteria and selects the filtering approach that best balances detection performance and signal reconstruction quality.

3.4. Agent Cost Function and Filter Evaluation Strategy

To support the automatic selection of the most appropriate statistical filter for each dataset scenario, an agent-based evaluation mechanism was introduced. Rather than performing full reinforcement learning, the proposed agent operates as a deterministic decision-support mechanism that evaluates candidate filters using a multi-criteria statistical cost function. The objective of the agent is to balance three key properties of the reconstructed time series:

- Effective suppression of anomalous measurements;
- Preservation of the underlying signal dynamics;

- And adequate smoothing of noise without distorting the signal structure.

At a conceptual level, this objective can be expressed using the following abstract cost formulation:

$$Cost = w_1 \cdot A + w_2 \cdot S + w_3 \cdot D \tag{14}$$

where

- A represents anomaly suppression effectiveness;
- S represents signal smoothness;
- D represents deviation from the original signal;
- w_1, w_2, w_3 are weighting coefficients controlling the relative importance of each component.

This formulation provides an intuitive representation of the optimization objective guiding the agent. For implementation purposes, these conceptual components are expanded into measurable statistical metrics derived from the filtered signal.

The operational cost function used by the agent is defined as

$$C(f_i) = \omega_1 RMSE_n + \omega_2 f_{out} + \omega_3 Var_n + \omega_4 \Delta m + \omega_5 r_{spikes} \tag{15}$$

where

- $RMSE_n$ is the normalized reconstruction error;
- f_{out} is the fraction of detected outliers;
- Var_n is the normalized variance of the reconstructed signal;
- Δm represents the difference between the slopes of the original and filtered signals;
- r_{spikes} denotes the residual spike rate;
- ω_i are weighing coefficients controlling the contribution of each metric.

The selection of weights for the cost function was based on empirical methods to establish balance between multiple competing goals which included reconstruction accuracy and anomaly detection sensitivity and signal stability. The authors assigned more importance to normalized reconstruction error and outlier detection consistency while they used extra terms to measure variance preservation and trend deviation. The framework uses fixed weights because this setup maintains both interpretability and deterministic behavior while eliminating the requirement for hyperparameter tuning. The design decision complies with low-power IoT systems which demand both easy computation and exact result replication.

The weighting coefficients satisfy

$$0 \leq \omega_i \leq 1 \tag{16}$$

and

$$\sum_{i=1}^{5} \omega_i = 1 \tag{17}$$

ensuring a normalized and interpretable contribution of each component to the overall cost. This normalized weighting scheme allows the cost function to be interpreted as a convex combination of complementary signal quality metrics.

To ensure that the different metrics contribute comparably to the overall cost function, several components are normalized with respect to the statistical properties of the original signal.

Normalized reconstruction error is defined as

$$RMSE_n = \frac{RMSE}{\sigma_{orig}} \tag{18}$$

where $RMSE$ represents the root mean square error between the original and filtered signals and σ_{orig} denotes the standard deviation of the original time series.

Normalized variance is computed as

$$Var_n = \frac{Var_{clean}}{Var_{orig}} \tag{19}$$

where Var_{clean} and Var_{orig} correspond to the variances of the filtered and original signals.

Trend preservation is defined as

$$\Delta m = \mid m_{orig} - m_{clean} \mid \tag{20}$$

where m_{orig} and m_{clean} are the slopes obtained through linear regression of the original and filtered signals.

The agent also evaluates the presence of abrupt residual variations remaining after filtering. The residual spike rate is defined as

$$r_{spikes} = \frac{N_{spikes}}{N_{samples}} \tag{21}$$

where

- N_{spikes} is the number of detected spikes after filtering;
- $N_{samples}$ is the total number of samples in the time series.

A spike is defined as a sudden variation between consecutive samples exceeding a threshold derived from signal variability:

$$\mid x_t - x_{t-1} \mid > \theta \tag{22}$$

where θ represents a threshold determined from the statistical variability of the signal.

To improve methodological clarity, Table 3 summarizes the statistical metrics used by the agent to evaluate candidate filters. These metrics jointly capture anomaly suppression effectiveness, reconstruction fidelity, signal smoothness, and preservation of the original temporal dynamics. To enable a multi-criteria evaluation of filtering performance, five statistical metrics were defined to quantify reconstruction accuracy, anomaly detection behavior, signal smoothness, trend preservation, and residual spike activity.

Table 3. Statistical metrics used in the agent cost function.

Metric	Symbol	Description	Definition	Range
Normalized reconstruction error	$RMSE_n$	Deviation between original and filtered signals	$RMSE_n = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(x_i^{orig} - x_i^{clean}\right)^2}}{\sigma_{orig}}$	≥0
Fraction of detected outliers	f_{out}	Proportion of samples identified as anomalies	$f_{out} = \frac{N_{outliers}}{N_{samples}}$	[0, 1]
Normalized variance	Var_n	Relative smoothness of filtered signal	$Var_n = \frac{Var\ (x_{claen})}{Var\ (x_{orig})}$	≥0
Trend preservation	Δm	Difference between slopes of original and filtered signals	$\Delta m = \lvert m_{orig} - m_{clean} \rvert$	≥0
Residual spike rate	r_{spikes}	Frequency of abrupt residual variations	$r_{spijes} = \frac{N_{spikes}}{N_{samples}}$	[0, 1]

The slope values m_{orig} and m_{clean} are obtained from a first-order linear regression fitted to the original and filtered time series, respectively.

For each candidate statistical filter, the agent computes the corresponding cost value $C(f_i)$. The filter producing the minimum cost is selected as the optimal preprocessing method.

$$f^* = \underset{f_i \in F}{\operatorname{argmin}} C(f_i) \tag{23}$$

where f denotes the set of candidate statistical filters.

The complete processing workflow of the proposed framework is illustrated in Figure 5, which presents the interaction between the IoT temperature time-series input, the statistical filtering modules, and the agent-based evaluation mechanism. Within this architecture, the agent evaluates the filtered signals through the proposed statistical cost function and selects the most suitable filter for each dataset scenario.

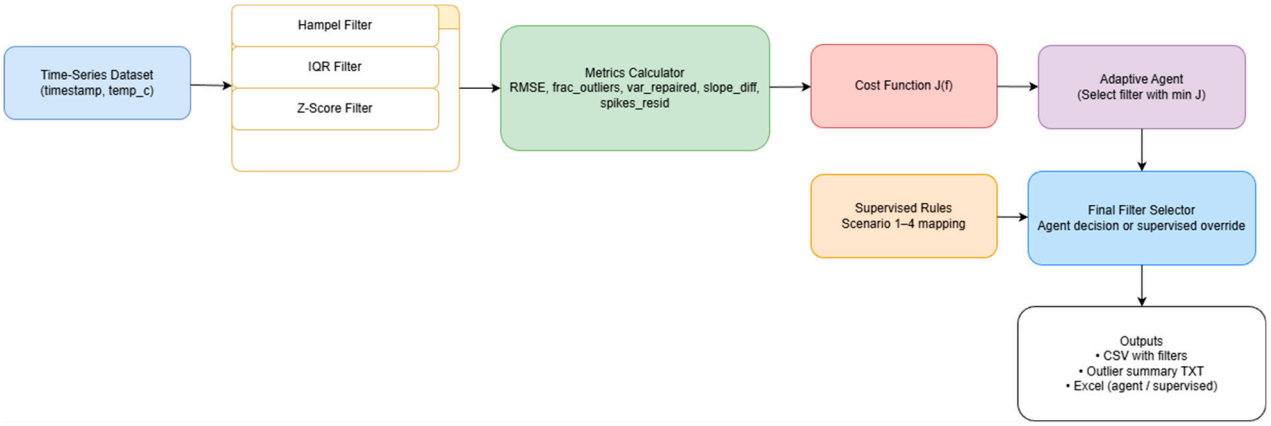

Figure 5. Agent overview UML diagram.

The agent-assisted filter selection mechanism is illustrated in a high-level UML fashion. The diagram depicts the parallel processing of the same time-series dataset through the Hampel, IQR, and Z-Score filters, which is then followed by the calculation of several performance metrics such as reconstruction error, anomaly rate, variance preservation, slope deviation, and residual spike rate. Evaluated metrics are incorporated into a composite cost function, which, in turn, is assessed by the adaptive agent who selects the filter with the minimum overall cost. Moreover, the diagram communicates the combination of supervised rules for uncomplicated scenarios and the eventual output artifacts, thereby illustrating that the selection process is driven by metrics, transparent and fully traceable.

The proposed agent should not be understood as a reinforcement learning or ML agent according to architectural standards. The system functions as a lightweight deterministic decision-making system which operates within resource-limited IoT applications.

Decision processes use a rule-based system which evaluates candidate filters by applying specific cost weights that have been established beforehand. The chosen approach has three specific goals which it needs to achieve because of its implementation requirements. The first goal requires the system to maintain basic computational processes which can function on embedded devices and edge nodes. The second goal needs system performance to show clear results which can be measured through actual signal characteristics. The system needs to exhibit deterministic behavior which enables both system testing and validation procedures to work. The decision-making system executes its functions through an evaluation and selection process which starts when statistical filters assess input signals. The system measures performance through multiple metrics which it uses to calculate a composite cost. The filter with the minimum cost is selected as the optimal reconstruc-

tion method. The research implementation uses a method which applies deterministic evaluation to Python programming language. Cost function combines normalized metrics which include root mean square error (RMSE) outlier fraction variance slope difference and residual spike rate. The filter which has the lowest cost value becomes the best filter choice for the specific situation being analyzed.

3.5. Scenario-Based Data Generation and Reproducibility

The released data consists of seven scenarios (Tables 4 and 5) designed to reproduce heterogeneous anomaly patterns commonly observed in long-term IoT temperature monitoring. The scenarios represent diverse signal degradation mechanisms, including isolated spikes and dense impulsive noise as well as gradual drifts, corrupted flat segments, and complex non-stationary combinations of all of them.

Table 4. Scenario-specific anomaly injection rules and parameter ranges used in the synthetic benchmark.

Parameter	Description	Typical Value
T_0	Initial baseline temperature	20–22 °C
σ	Sensor noise standard deviation	0.05–0.2 °C
A_s	Spike magnitude	2–5 °C
p_{spike}	Spike occurrence probability	0.5–2%
L_{imp}	Impulsive noise duration	2–5 samples
β	Drift coefficient	0.001–0.01 °C/sample
L_{flat}	Length of flat corrupted segment	20–50 samples

Table 5. Overview of benchmark scenarios and characteristic anomaly patterns.

Scenario	Signal Characteristics	Anomaly Type	Typical Magnitude	Duration	Injection Frequency
Scenario 1 (S1): Stable Spikes	Stable baseline temperature with small natural noise	Isolated spikes	±3–6 °C deviation	1–2 samples	Rare (≈1–2% of samples)
Scenario 2 (S2): Impulsive Noise	Stable signal with frequent abrupt disturbances	Dense impulsive noise	±2–5 °C deviation	1 sample	Frequent (≈5–10%)
Scenario 3 (S3): Gradual Drift	Slowly increasing baseline	Drift anomaly	Gradual ±3–8 °C shift	Long segments (50–150 samples)	Continuous
Scenario 4 (S4): Flat Corrupted Segments	Artificially constant temperature	Sensor freeze/flat signal	Constant value	40–120 samples	Occasional
Scenario 5 (S5): Mixed Anomalies	Non-stationary signal	Spikes, drift and flat segments	±3–8 °C	Variable	Mixed
Scenario 6 (S6): Cyclic Variations	Periodic environmental pattern	Periodic anomalies and spikes	±2–6 °C	Short bursts	Irregular
Scenario 7 (S7): Heavy-Tailed Noise	High variance signal	Extreme events	±5–10 °C	1–3 samples	Rare extreme

The first four scenarios were designed so that each of the degradation mechanisms can individually be addressed. This allows filter behavior to be evaluated under simplified yet

realistic conditions, revealing the intrinsic response of each statistical method without the disturbance of multiple anomaly sources. On the other hand, scenarios 5, 6, and 7 represent real operational environments where signal conditions change over time, as they progressively combine and vary the types and levels of disturbances applied.

Detailed presentations and visual descriptions of the seven scenarios are given in Section 4, where the scenarios are looked at both structurally and functionally. The distinction made between scenario generation and dataset characterization can be regarded as a source of conceptual clarity while at the same time allowing a smooth flow of the narrative between the methodological and experimental parts of the manuscript.

The data generation and processing steps are implemented in Python 3.11 version [11] using a modular software architecture that separates filtering logic, evaluation metrics, and visualization components. This design ensures uniform processing across all scenarios. The complete pipeline, datasets, figures, and summary artifacts are publicly released through an open access repository [10], enabling full reproducibility, independent verification, and reuse. This study focuses on statistical anomaly detection methods due to their interpretability and low computational cost.

4. Dataset Characterization and Experimental Design

The dataset is thoroughly characterized in this part according to the scenario-based organization which was introduced at the stage of data generation. Each scenario is depicted by the corresponding dataset figures, with particular attention being paid to the signal characteristics and anomaly patterns. UML-based pipeline documentation is provided as supporting material to ensure the transparency and reproducibility of the experimental design [10]. Figure 6 shows the algorithmic flow of the proposed agent-assisted anomaly detection pipeline.

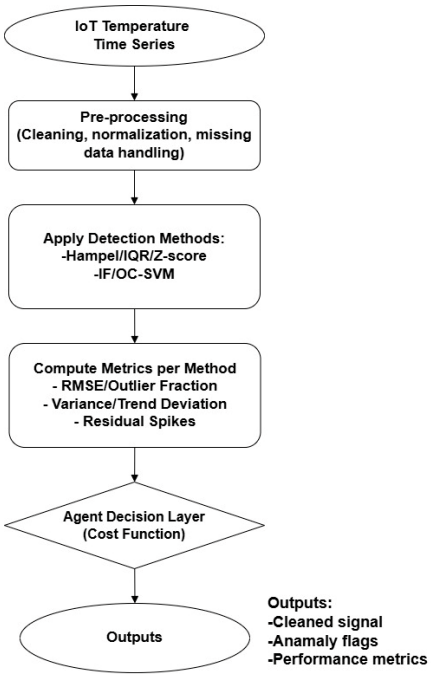

Figure 6. Algorithmic flow of the proposed agent-assisted anomaly detection pipeline, illustrating the sequential processing from raw IoT temperature data to final anomaly detection and signal reconstruction through multi-method evaluation and cost-based decision selection.

4.1. Dataset Overview and Scenario Design

Each anomaly scenario was designed to reflect real-world IoT conditions: impulsive noise corresponds to sensor interference or transmission errors; drift represents calibration degradation over time; flat corrupted segments model sensor freezing or communication loss; and multi-node correlated anomalies reflect distributed failures in networked sensing systems. This mapping enhances the practical applicability of the proposed dataset.

The observed temperature signal $x(t)$ is generated as the sum of three components:

$$x(t) = s(t) + n(t) + a(t) \tag{24}$$

where

- $s(t)$ represents the baseline environmental temperature signal;
- $n(t)$ represents measurement noise produced by the sensor;
- $a(t)$ represents injected anomalies simulating sensor faults or disturbances.

The baseline temperature signal $s(t)$ is modeled as a slowly varying process:

$$s(t) = T_0 + \alpha t + \epsilon(t) \tag{25}$$

where

- T_0 is the initial temperature level;
- α represents a small drift coefficient;
- $\epsilon(t)$ is low-amplitude Gaussian noise representing natural environmental fluctuations.

The measurement noise component $n(t)$ is modeled as

$$n(t) \sim \mathcal{N}\left(0, \sigma^2\right) \tag{26}$$

where σ controls the magnitude of sensor noise.

In addition to the synthetic benchmark, a real-world IoT environmental sensing dataset was incorporated to assess the practical applicability of the proposed framework. The real dataset contains temperature measurements collected from multiple sensor nodes under realistic operating conditions, including natural environmental variability, sensor noise, and irregular local disturbances. Since manually verified anomaly labels were not available, a weak labeling strategy was adopted using robust deviations in temperature and temporal variation, complemented by contextual environmental variables when available. This real-data extension does not replace the controlled synthetic benchmark but provides an additional layer of validation under more realistic deployment conditions.

Anomaly components $a(t)$ are injected according to predefined scenario-specific rules that control anomaly magnitude, duration, and occurrence frequency.

The anomaly injection mechanism uses a controlled stochastic process to achieve identical experimental results in all its test runs. The system defines each anomaly type through parameters that determine its intensity and length of time and its likelihood of happening. For spike anomalies, disturbances are generated as

$$a_{spike}(t) = \begin{cases} A_s & \text{if } t \in \mathcal{T}_{spike} \\ 0 & \text{otherwise} \end{cases} \tag{27}$$

where A_s represents the spike magnitude and $\mathcal{T}_{spike}$ denotes randomly selected timestamps.

Impulsive noise is modeled as short bursts of high-amplitude deviations applied over small time windows. Gradual drift anomalies are generated as

$$a_{drift}(t) = \beta t \tag{28}$$

where β represents a small drift rate simulating sensor calibration degradation.

Flat corrupted segments are generated by forcing the signal to remain constant during predefined time intervals:

$$x(t) = c \tag{29}$$

where c is a constant value representing a frozen sensor output.

Finally, mixed anomalies combine multiple disturbance mechanisms simultaneously to emulate realistic long-term IoT deployments. Table 4 summarizes the parameters used for synthetic anomaly injections in the dataset generated.

Baseline temperature signal is modeled as a slowly varying stochastic process with additive Gaussian noise, while anomalies are injected using parameterized perturbations controlling amplitude, duration, and frequency. The parameter values used in the experiments were selected according to common practices reported in the robust statistics literature and validated empirically through preliminary tests.

To ensure experimental reproducibility and transparency, the main characteristics of the generated dataset and the anomaly injection parameters used in each scenario are summarized in Table 5.

4.2. Scenario 1 (S1): Stable Signal with Sparse Spikes

In S1 (Figure 7), an ideal operating condition is depicted wherein the temperature signal is mostly constant except for a few single short spikes. These irregularities mimic rare sensor malfunctions or minor disruptions that are typically noticed in the case of long-term IoT deployments.

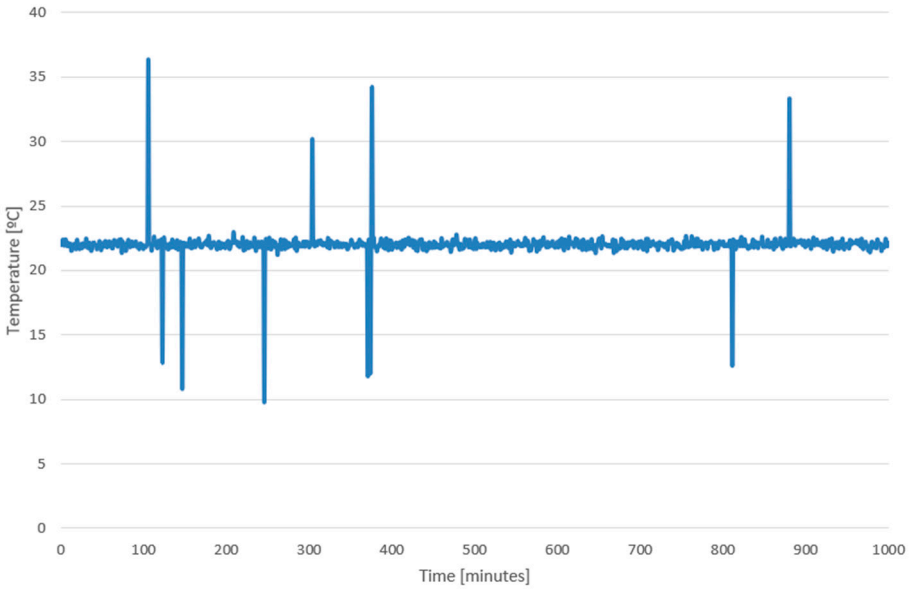

Figure 7. S1: Stable spikes.

Under these conditions, the outliers are detected solely based on their high volatility characteristic, as exhibited in Figure 7. This scenario serves as a baseline case for evaluating statistical filters under minimal interaction between anomalies and long-term signal behavior.

4.3. Scenario 2 (S2): Dense Impulsive Noise

S2 brings in a much higher density of impulsive anomalies all over the time series. In this instance, outliers come up a lot and could even form groups over time, which would

make it harder to tell the difference between the samples with anomalies and the ones with legitimate variations.

Figure 8 illustrates that the signal is mostly composed of several impulsive deviations that happen repeatedly, which makes it more likely that there will be over-correction and excessive smoothing. This scenario assesses how filtering techniques handle dense impulsive disturbances without distorting the underlying signal structure.

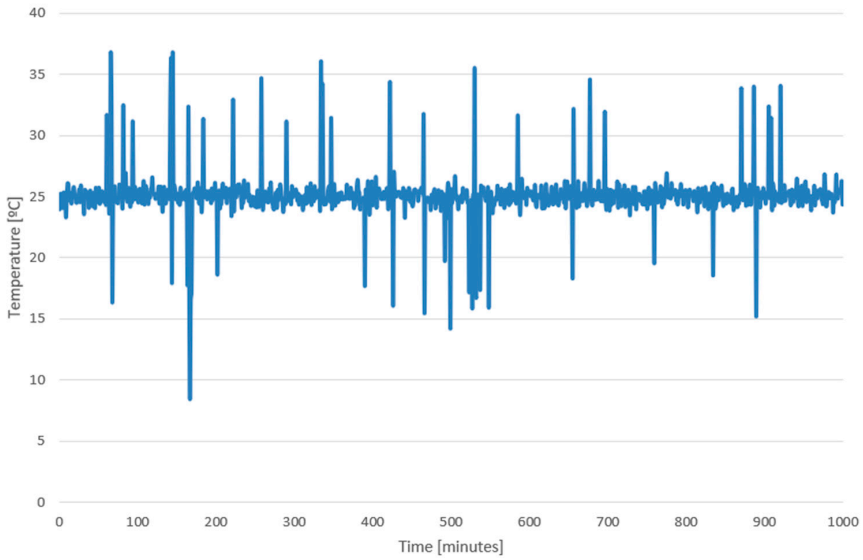

Figure 8. S2: Impulsive noise.

4.4. Scenario 3 (S3): Gradual Drift

S3 portrays a slow and steady increase in temperature over time, an illustration of sensor aging, calibration shifts, or gradual changes in the environment. Point-wise criteria-based detection is inapplicable for this scenario because it is very hard to detect the daily accumulation of deviations, which are nonetheless treated as anomalies.

In the illustration, the signal's baseline has been deviating slowly but steadily, as shown in Figure 9. This scenario evaluates the ability of anomaly detection methods to preserve long-term trends while suppressing localized deviations.

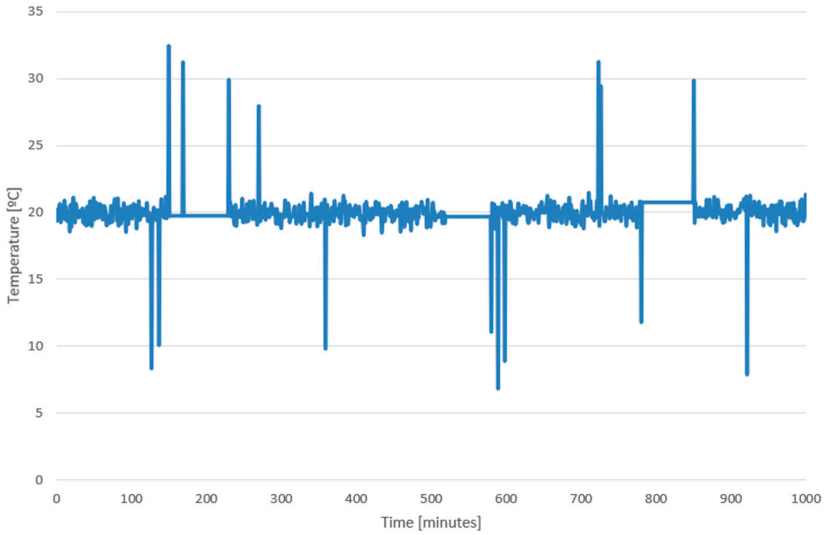

Figure 9. S3: Periodic drift.

4.5. Scenario 4 (S4): Corrupted Flat Segments

S4 portrays the occurrence of corrupted flat segments where the temperature signal remains artificially constant for a long time. Such cases can be caused by sensor freezing, communication breakdowns, or data acquisition errors.

In Figure 10, the prolonged plateaus are clearly visible, and they do not conform to the realistic temperature dynamics. The present scenario examines the capability of statistical filters to recognize structural anomalies which are not amenable to simple threshold-based methods.

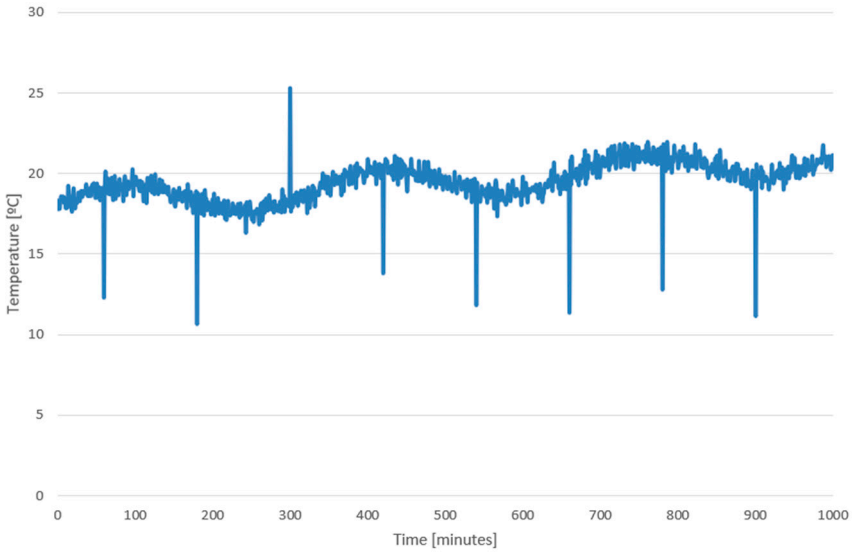

Figure 10. S4: Flat corrupted.

4.6. Scenario 5 (S5): Mixed Anomalies with Non-Stationary Behavior

S5 unites several anomaly types in the setting of a non-stationary signal. Impulsive noise, gradual drift, and intermittent flat segments together characterize a very realistic long-term deployment where the signal's statistical properties change continuously over time.

The overlap of the different disturbance mechanisms is shown in Figure 11, which makes it quite difficult to interpret the data in isolation. This scenario examines filter performance under non-stationary conditions with overlapping anomaly mechanisms.

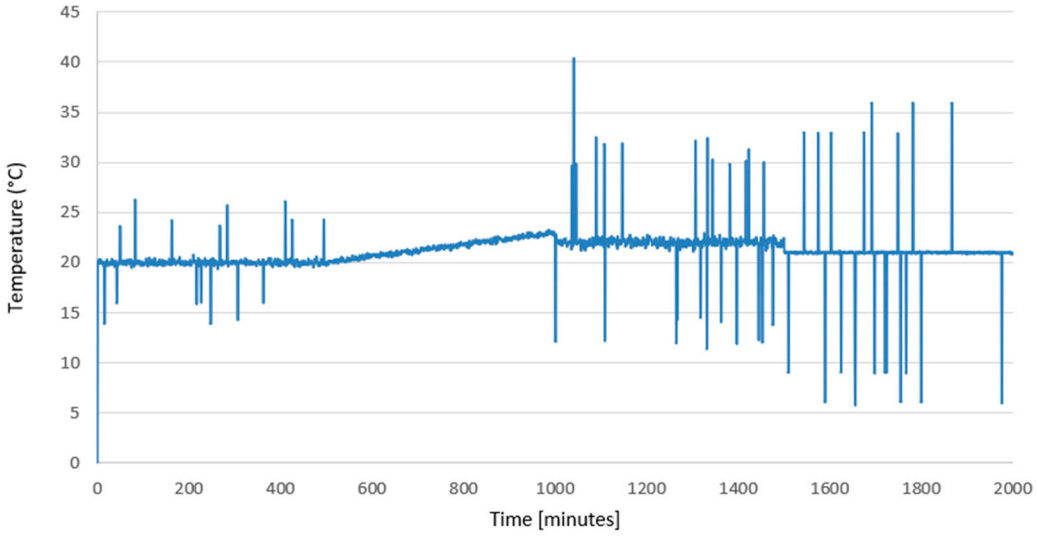

Figure 11. S5: Complex mixed noise.

4.7. Scenario 6 (S6): Cyclic Variations with Embedded Anomalies

Cyclic temperature variations, like diurnal or periodic environmental effects, along with superimposed anomalies are presented in S6. The main difficulty presented in this scenario is to recognize the true periodic behavior from the anomalous deviations.

As depicted in Figure 12, the basic cyclic pattern needs to be kept during the removal of irregular disturbances. This scenario focuses on preserving cyclic patterns while mitigating superimposed anomalous deviations.

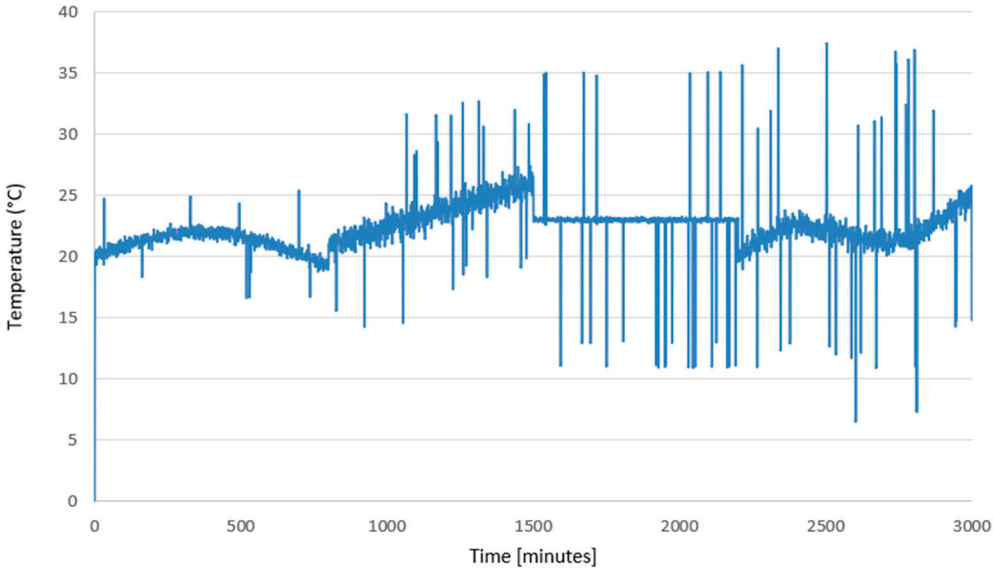

Figure 12. S6: Complex plus.

4.8. Scenario 7 (S7): Heavy-Tailed Noise and Extreme Events

S7 is the hardest condition to operate under as it has heavy-tailed noise distributions and extreme temperature excursions. It is the same as having very harsh environments or very bad sensors producing rare but extreme deviations.

Significant variance inflation and extreme outliers severely affecting mean and variance estimators are illustrated in Figure 13. This scenario is a trial for the statistical filters' robustness and the agent-assisted selection mechanism's effectiveness.

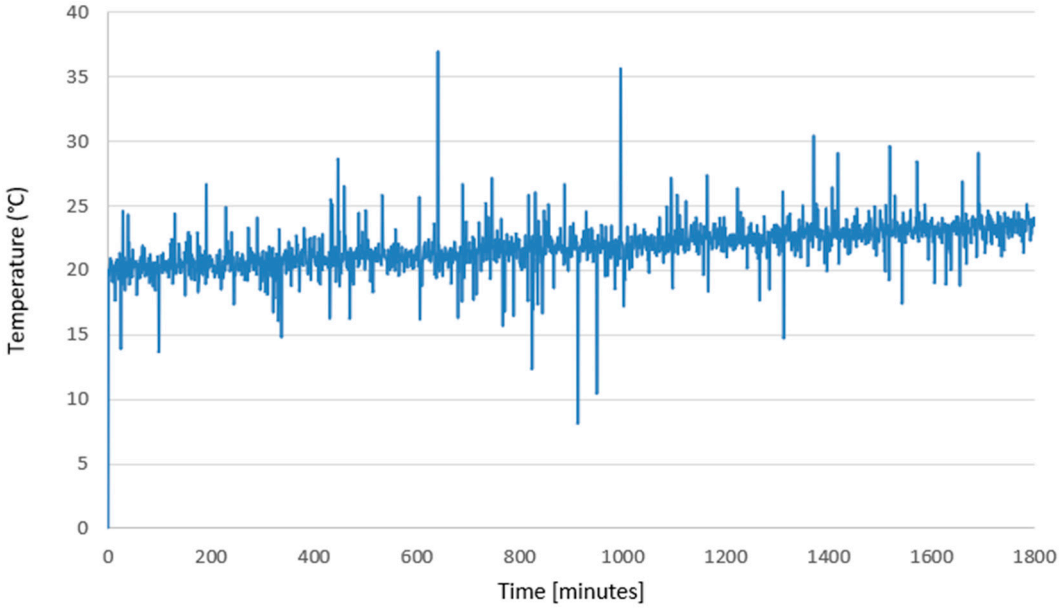

Figure 13. S7: Heavy-tail drift.

The complete data generation and evaluation pipeline was implemented in Python [11] and is publicly available in the accompanying source code repository [10]. The filter evaluation scripts together with the agent-assisted selection scripts execute the dataset processing pipeline and filter selection process which uses cost-based methods.

4.9. Field-like Scenario (SF)

The synthetic benchmark scenarios (S1–S7) received additional support from the field-like scenario (SF) which used actual IoT temperature data as its foundation. The test uses multiple realistic conditions which include environmental disturbances and data loss and partially identified unusual events to assess how well the proposed pipeline works in actual situations. The dataset was obtained from the public repository [31].

The dataset consists of environmental sensing data which scientists collected at multiple locations throughout extended time periods with measurements taken every five minutes.

The study analyzed eight different sensor locations which each represented a unique environmental situation. The SF scenario contains unidentified abnormalities which researchers track through statistical analysis and predefined thresholds based on domain knowledge. The SF scenario introduces the following challenges:

- Non-stationary temperature behavior;
- Real measurement noise;
- Irregular anomaly patterns;
- Sensor drift and environmental variability.

The scenario provides the conditions required to assess the following:

- Detection consistency across methods;
- Reconstruction stability;
- Generalization capability of the agent-based selection.

Real IoT temperature data shows its field-like scenario through Figure 14. The upper plot presents the complete time series, where it is possible to observe long-term variability, seasonal-like behavior, and the presence of measurement noise typical of real sensing environments. The signal from the synthetic scenarios shows non-stationary behavior because its signal pattern contains multiple unpredictable elements. The lower plot displays a representative temporal segment through which viewers can study local dynamics in closer detail. The interval shows three types of temperature changes which include short-term variations and sudden temperature shifts and single temperature spikes that probably result from temporary anomalies and environmental interruptions and sensor problems such as drift or noise bursts. Figure 14 demonstrates how real-world data becomes more complex because its anomalies lack clear definitions and labels which makes adaptive detection systems essential for handling weakly supervised data with different signal types.

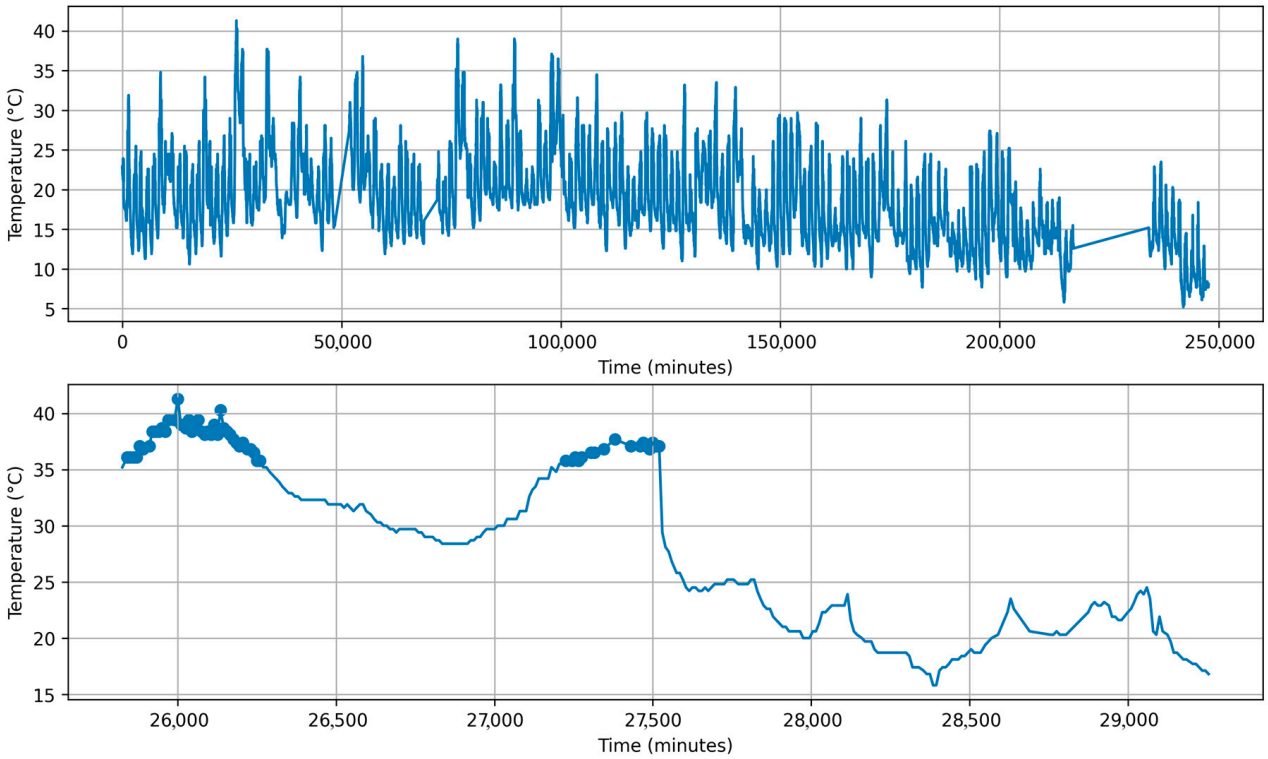

Figure 14. Field-like scenario (SF): Real IoT temperature time series (**top**) and zoomed segment highlighting local variability and anomaly behavior (**bottom**).

5. Results

Experimental tests of the proposed anomaly detection system validate its three essential testing areas. The first dimension of testing the measures detection performance through three metrics includes precision, recall, and F1-score. The second dimension of testing the measures system performance through two metrics includes execution time and memory usage. The third dimension of testing the evaluated system performance through multiple scenarios includes testing with actual data from the field. The evaluation combines the following:

- A controlled synthetic dataset with seven anomaly scenarios;
- A real-world IoT dataset with weak or missing labels.

The structure of the results ensures their direct application to support the discussion and conclusions that Section 6 presents.

The main configuration parameters used in the anomaly detection pipeline are summarized in Table 6. According to these values the statistical filters will operate, and the decision agent will use cost function composite weights to make decisions. The same parameter settings were applied across all scenarios to ensure consistent and reproducible experimental evaluation.

The SF dataset represents a real-world IoT deployment consisting of eight sensor locations, with temperature measurements acquired every 5 min. Table 7 presents a summary of the dataset which includes both sample counts and the characteristics of anomalies, while Section 4 contains detailed statistics for each location.

Table 6. Parameter settings are used for statistical filters and cost function agents.

Parameter	Value	Justification	Reference
Hampel window size	21	Provides a balance between local sensitivity and noise robustness in slowly varying temperature signals	[13,19]
Hampel threshold	3	Standard choice for outlier detection using MAD, equivalent to ~3σ rule	[13,14]
IQR multiplier	1.5	Tukey's rule for detecting moderate outliers in non-parametric distributions	[6]
Z-Score threshold	3	Common threshold for Gaussian-based anomaly detection	[12]
w_1	0.4	Normalized reconstruction error	This work
w_2	0.3	Outlier fraction	This work
w_3	0.15	Signal variance preservation	This work
w_4	0.1	Trend deviation	This work
w_5	0.05	Residual spike penalty	This work

Table 7. Summary of synthetic and real-world datasets used in the evaluation.

Scenario	Type	Samples	Anomaly Rate (%)	Description
S1	Synthetic	10,000	1.0	Stable signal with sparse spikes
S2	Synthetic	10,000	5.0	Dense impulsive noise
S3	Synthetic	10,000	3.0	Drift with embedded anomalies
S4	Synthetic	10,000	4.0	Flat segments with spikes
S5	Synthetic	10,000	6.0	Mixed anomalies and regime switching
S6	Synthetic	10,000	3.5	Cyclic signal with anomalies
S7	Synthetic	10,000	7.0	Heavy-tailed noise and extreme events
SF	Real (8 sensors)	~200,000+	~0.8–1.0	Real-world IoT dataset (8 locations)

5.1. Aggregated Detection Performance

Table 8 provides the primary quantitative reference which establishes method comparison results for all datasets. The results show the following:

- IF achieves the highest overall F1-score (0.304), which demonstrates its ability to handle various types of abnormal behavior.
- Hampel filtering delivers competitive results through its F1-score of 0.303 while using less computational power, which makes it an efficient lightweight solution.
- The IQR method and OC-SVM demonstrate moderate success because they achieve only slight improvements beyond what statistical methods accomplish.
- Z-Score achieves the highest precision (0.734) and lowest false positive rate, but at the expense of recall, resulting in a lower F1-score.

Table 8. Mean quantitative performance across all datasets.

Method	Precision	Recall	F1-Score	FPR	Accuracy	RMSE	MAE	Runtime (ms)	Peak Memory (kB)
Hampel	0.413	0.362	0.303	0.020	0.903	0.424	0.218	3.481	92.260
IQR	0.390	0.277	0.249	0.009	0.906	0.446	0.225	3.110	95.410
IF	0.482	0.317	0.304	0.017	0.905	0.451	0.220	1007.105	821.393
OC-SVM	0.431	0.288	0.265	0.019	0.902	0.464	0.222	5.167	78.449
Z-Score	0.734	0.207	0.230	0.000	0.911	0.473	0.229	3.201	83.268

From a computational perspective,

- Edge IoT deployment requires statistical methods which operate at milliseconds speed according to the test results;
- IF requires roughly 1000 milliseconds to run while it consumes more memory than other methods, which creates a considerable performance versus operational effectiveness problem.

Corresponding trends are illustrated in Figures 15–17, which provide a visual comparison of F1-score, precision, and recall. However, the interpretation is primarily supported by Table 7, avoiding redundancy.

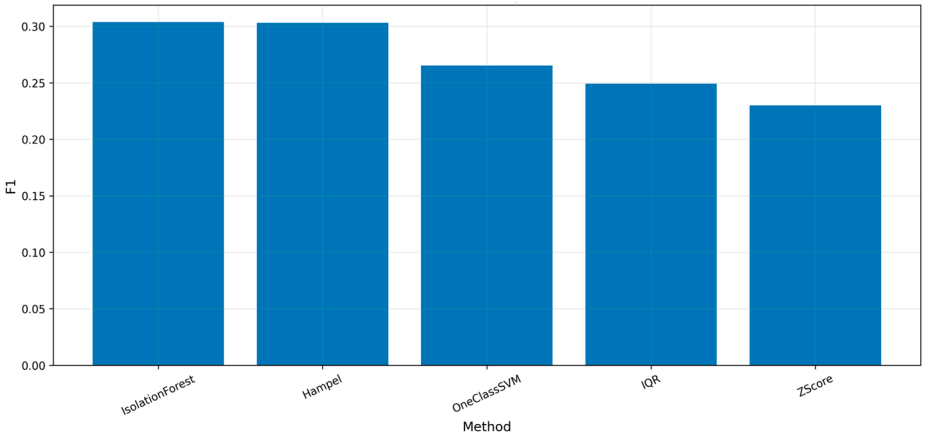

Figure 15. Mean F1-score by method.

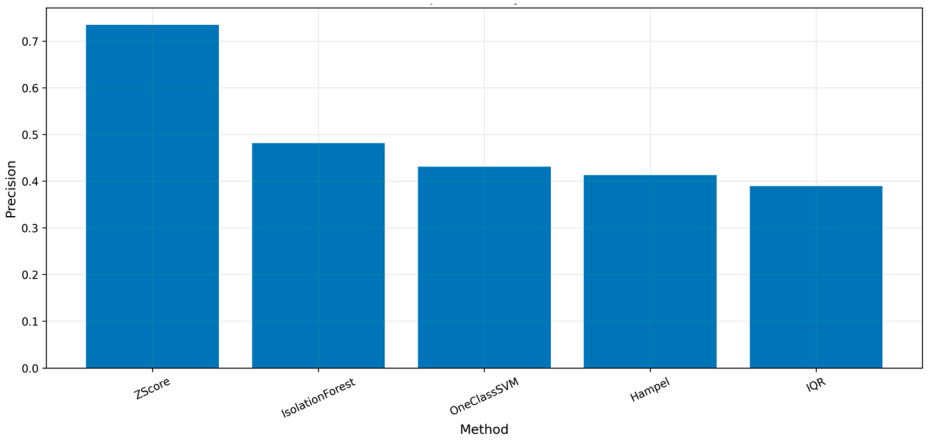

Figure 16. Mean precision by method.

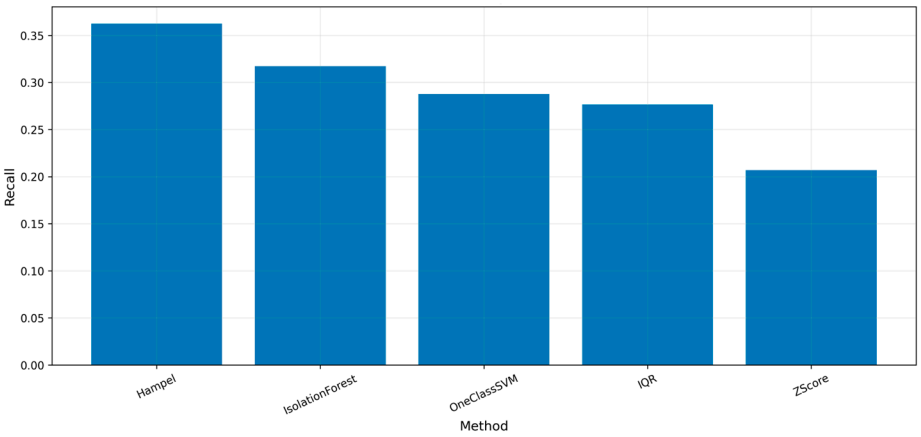

Figure 17. Mean recall by method.

5.2. Scenario-Based Detection Analysis

A scenario-wise comparison is presented in Table 9, which provides a compact and comprehensive view of performance variability across anomaly types.

Table 9. Scenario-wise comparative summary.

Scenario	Hampel	IQR	Z-Score	IF	OC-SVM
S1	0.326	0.609	0.933	0.378	0.368
S2	0.551	0.254	0.000	0.651	0.651
S3	0.062	0.024	0.018	0.106	0.106
S4	0.034	0.037	0.040	0.034	0.034
S5	0.058	0.036	0.031	0.108	0.120
S6	0.236	0.024	0.000	0.362	0.210
S7	0.867	0.909	0.712	0.500	0.353
SF	0.230	0.038	0.040	0.142	0.142

The results demonstrate that detection performance is strongly dependent on scenario characteristics:

- The Z-Score method achieves its best results in S1 (stable spikes) because it can effectively identify outlier data points.
- Statistical methods do not match the performance of IF and OC-SVM because these two methods can effectively analyze non-linear data distributions.
- Methods show worse results in S3 (periodic drift) because they need to handle the challenge of detecting slow-moving anomalies.
- Methods in S4 (flat corrupted) reach their lowest F1-score results because they can only detect a small number of anomalies.
- The mixed complex conditions in S5 to S6 demonstrate that machine learning techniques enhance system endurance.
- IQR method delivers optimal results in S7 (multiple locations/structured variability) with an F1-score of 0.909.
- Hampel method shows optimal stability performance in SF (field-like scenario).

Table 8 presents the best method for each scenario based on their F1-score and RMSE performance according to the formalized observations. The results confirm the following:

- No method consistently dominates across all scenarios;
- Optimal performance is inherently scenario-dependent;
- Static filter selection is suboptimal in heterogeneous environments.

5.3. Reconstruction Quality Evaluation

Quality assessment uses both quantitative metrics that include the RMSE and MAE measurements in Table 8 and qualitative assessment methods. Table 10 presents a systematic evaluation that shows all strengths and weaknesses of the comparison. Hampel provides strong performance through its ability to handle sparse data and mixed anomaly situations. IQR effectively handles impulsive noise but becomes unstable when there are drift conditions. Z-Score focuses on accuracy which leads to it missing distributed anomalies. The strong detection abilities of IF require substantial computational resources to operate. OC-SVM provides effective performance, but its improved interpretability remains restricted.

Table 10. Comparative qualitative summary of strengths and limitations.

Method	Main Strength	Main Limitation	Best-Suited Scenarios
Hampel	Strong balance between recall, robustness, and computational efficiency while remaining fully interpretable	Slightly lower precision compared to Z-Score in low-noise conditions	Sparse spikes, mixed anomaly scenarios, and field-like IoT signals
IQR	Simple and fast method with good performance for well-defined impulsive outliers	Less stable under non-stationary signals and gradual drift patterns	Impulsive noise and moderate anomaly conditions
Z-Score	Achieves the highest precision with very low false positive rate	Low recall due to conservative behavior, missing subtle or distributed anomalies	Scenarios where minimizing false alarms is critical
IF	Strong overall F1-score and good performance in complex or non-linear data distributions	High computational cost and memory usage compared to statistical methods	Complex scenarios with overlapping anomalies or dense noise
OC-SVM	Lightweight machine learning baseline with balanced performance	Limited interpretability and marginal performance gains compared to statistical approaches	Intermediate anomaly scenarios and benchmarking purposes

The reconstructed results show that through Figures 18–20, different statistical approaches maintain the signal structure which shows that different levels of sensitivity to ML approaches create smooth corrections which reduce the ability to understand their impact.

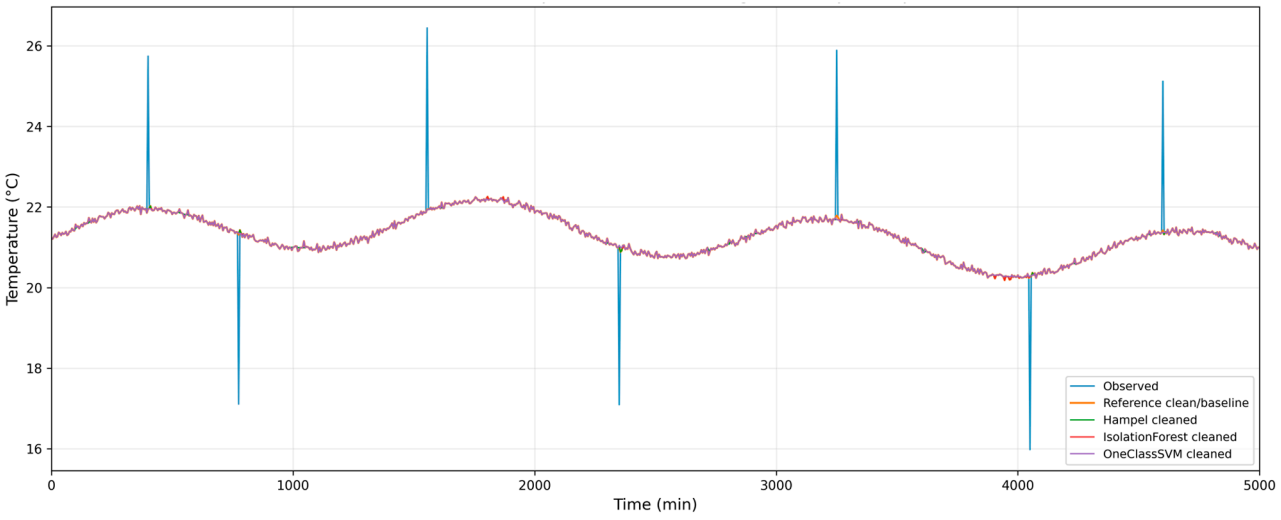

Figure 18. Reconstruction comparison for S1.

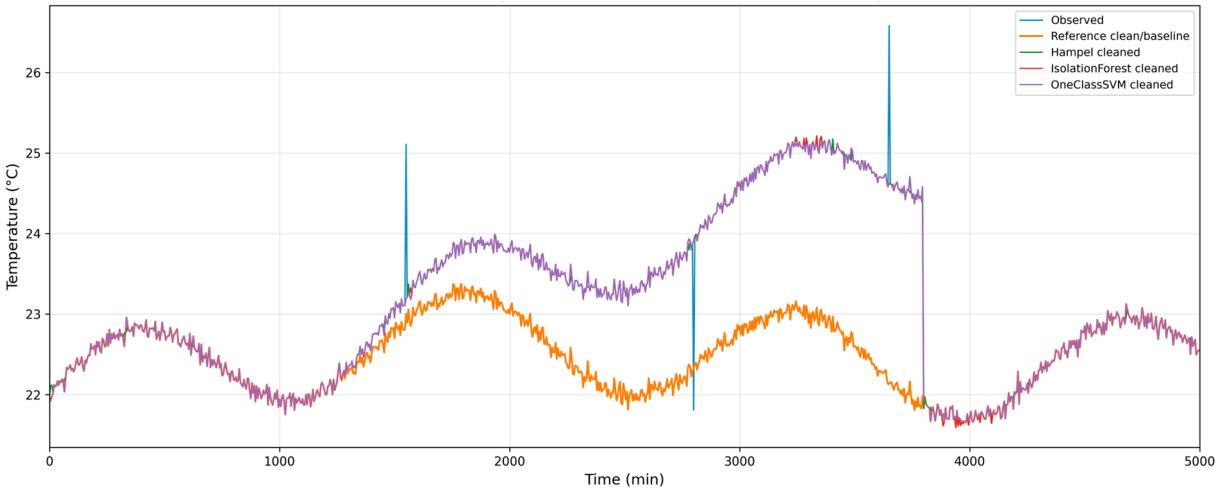

Figure 19. Reconstruction comparison for S3.

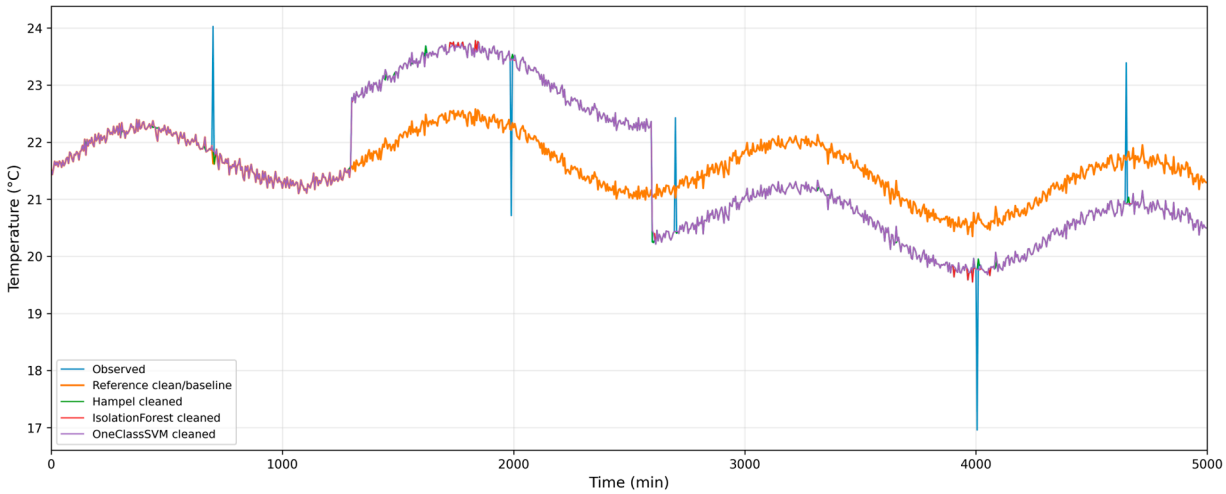

Figure 20. Reconstruction comparison for S5.

5.4. Computational Performance

The primary source of computational efficiency emerges from Table 7, which is associated with the additional evidence found in Figures 21 and 22. The results demonstrate that statistical methods which include Hampel IQR and Z-Score maintain a consistent low resource requirement. The agent-assisted mechanism introduces minimal extra costs for its operation. The IF method requires much more processing power and system memory compared to other methods. The system demonstrates a fundamental trade-off between its ability to detect threats and the system resources it requires for operation, which becomes essential in IoT edge deployment situations.

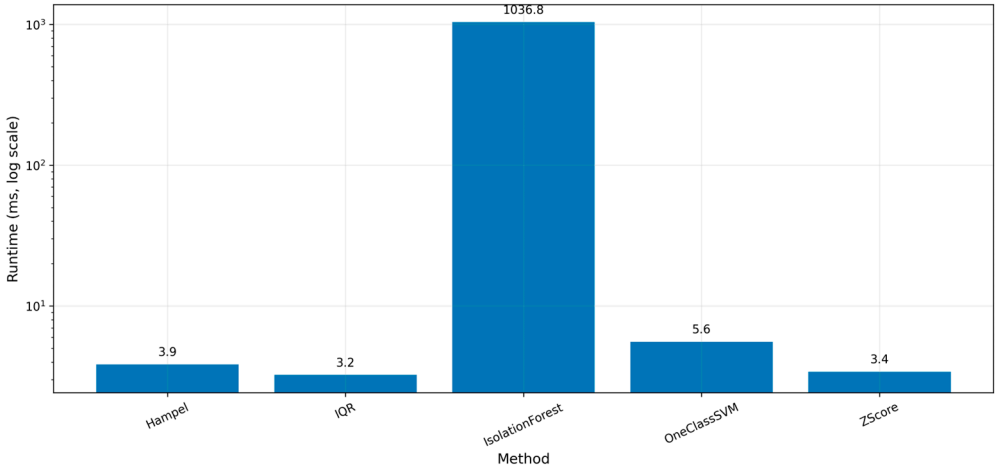

Figure 21. Mean execution time by method.

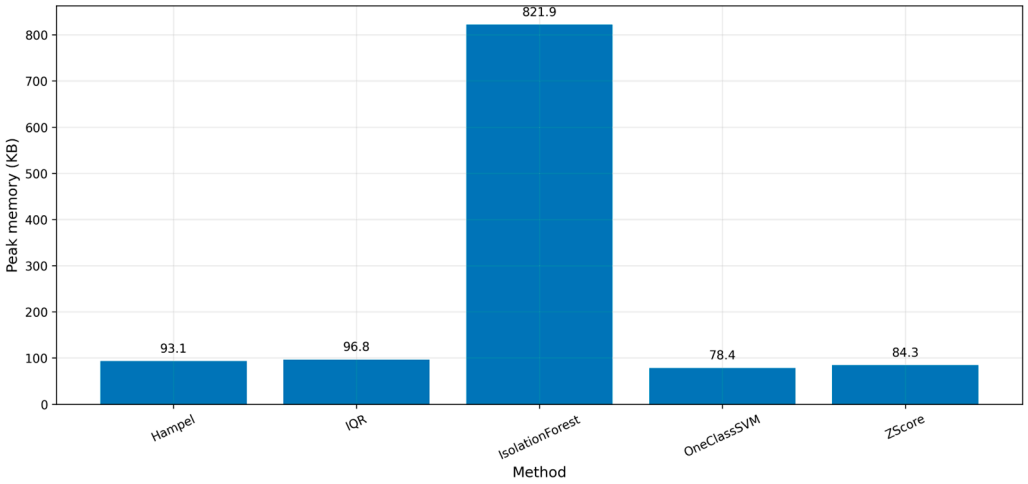

Figure 22. Mean peak memory by method.

5.5. Discussion and Practical Insights

The experimental results demonstrate that no single anomaly detection method consistently outperforms the others across all evaluated scenarios. As observed in Tables 7 and 8, each method exhibits strengths and limitations depending on the underlying anomaly characteristics, signal dynamics, and noise conditions. This reinforces the fundamental challenge in IoT anomaly detection: the variability of real-world data prevents the existence of a universally optimal static method.

To provide a clearer and more actionable interpretation of these results, Table 10 summarizes the best-performing method for each scenario, considering F1-score as the primary metric and RMSE as a secondary indicator of reconstruction quality. The results highlight a strong dependency between anomaly type and method suitability. For instance, Z-Score achieves optimal performance in scenarios with well-defined statistical deviations (S1 and S4), while IF performs better in more complex and less structured anomaly patterns (S2, S3, and S6). In contrast, IQR demonstrates superior performance in highly correlated anomaly conditions (S7), and Hampel remains a robust baseline for field-like scenarios (SF), where noise and variability are less predictable.

This variability confirms that method selection cannot rely solely on global performance averages. Instead, it must consider scenario-specific characteristics, which justifies the introduction of adaptive or agent-assisted decision mechanisms. The results presented in Table 11 therefore provide strong empirical support for the proposed framework, where

method selection is dynamically guided by multiple performance indicators rather than predefined assumptions.

Table 11. Scenario-wise selection of the best-performing method based on F1-score (primary metric) and RMSE (secondary metric).

Scenario	Best Method	Precision	Recall	F1-Score	RMSE
S1	Z-Score	0.875	1.000	0.933	0.007
S2	IF	0.900	0.509	0.651	0.466
S3	IF	0.633	0.058	0.106	1.073
S4	Z-Score	1.000	0.020	0.040	0.064
S5	OC-SVM	0.724	0.065	0.120	0.824
S6	IF	0.633	0.253	0.362	0.342
S7	IQR	0.897	0.921	0.909	0.017
SF	Hampel	0.596	0.143	0.230	0.572

Transition from synthetic to real-world data further reinforces these observations. The real-world dataset [31], illustrated in Figure 23 and further analyzed in Figure 24, consists of environmental sensor measurements collected at multiple urban locations under practical operating conditions. Figures 23 and 24 show the actual sensor data from the real-world monitoring system installed at board 509 of the Docklands Library. The dataset includes eight separate monitoring sites which are listed in Table 11 but only one site is shown because the demonstration needs to display regular operating patterns instead of showing all monitoring data. As summarized in Table 12, the dataset includes eight independent sensor nodes, each with distinct statistical properties and low anomaly rates (typically below 1%). This reflects realistic IoT deployments, where anomalies are rare and often subtle, making detection inherently challenging.

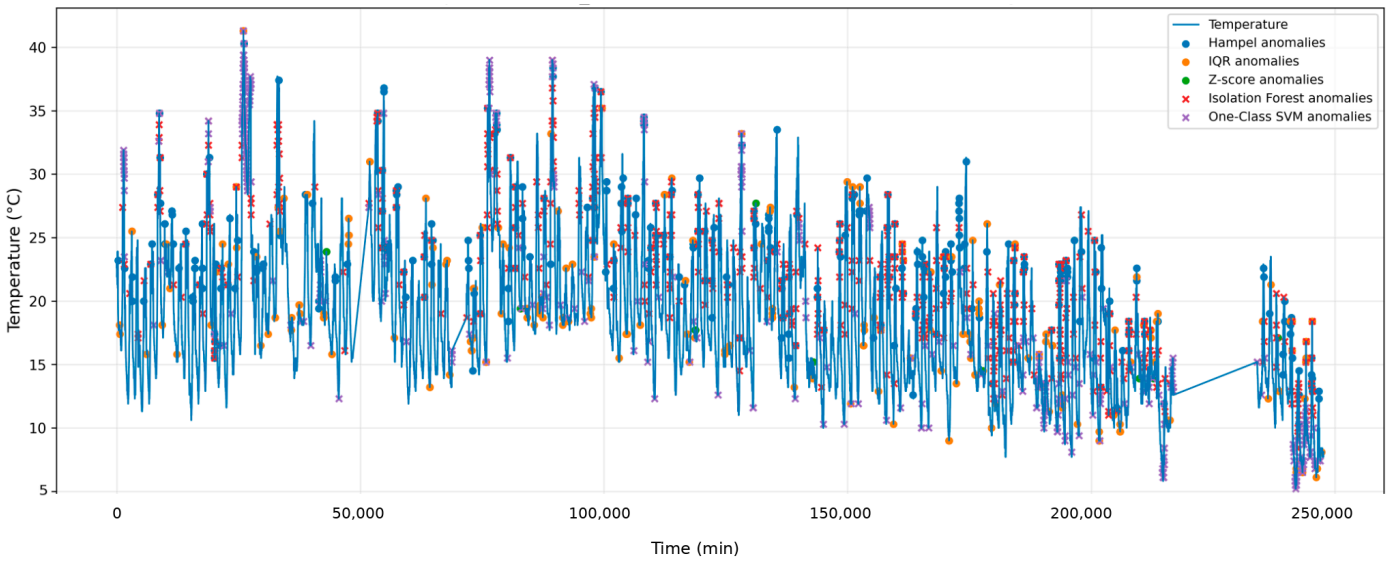

Figure 23. Detection comparison on a representative real-world IoT temperature series (board 509, Docklands Library [31]). Anomalies identified by statistical and machine learning methods are overlaid on the observed signal.

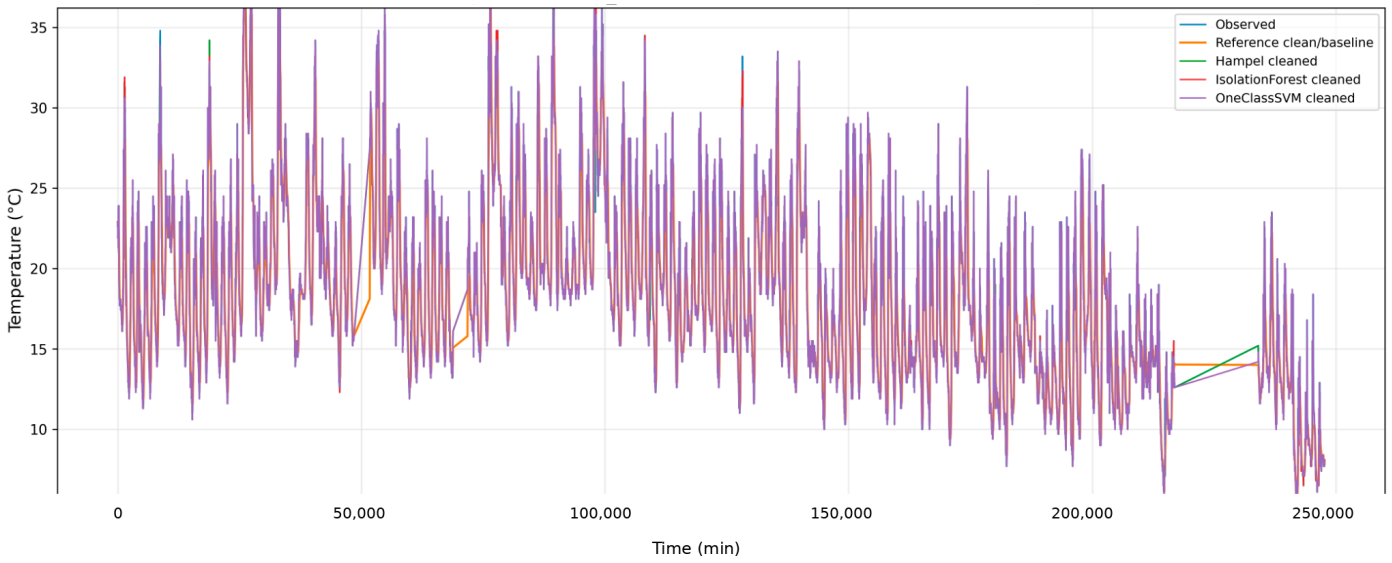

Figure 24. Reconstruction comparison for the real-world dataset (board 509). The cleaned signals obtained by different methods are compared with the observed time series.

Table 12. Real-world dataset summary and weak-label characteristics [31].

Location	Samples	Duration (Days)	Weak Anomalies	Weak Anomaly Rate (%)	Mean Temp (°C)	Std Temp (°C)
509	19,119	66.39	178	0.93	18.18	5.52
510	12,038	41.79	121	1.00	18.09	6.06
506	6626	23.01	64	0.97	16.88	5.38
511	4598	15.96	38	0.83	19.51	5.22
507	2918	10.13	25	0.86	19.73	4.84
505	2915	10.12	27	0.93	20.42	4.37
501	2903	10.07	26	0.90	19.86	5.26
508	2728	9.47	22	0.81	19.91	4.43
509	19,119	66.39	178	0.93	18.18	5.52

Since ground-truth annotations are not available, weak anomaly labels were generated using a statistical baseline approach based on Z-Score thresholding ($|z| > 3$). Although approximate, this labeling strategy provides a consistent reference for comparative evaluation. The results indicate that methods with high precision, such as Z-Score, tend to minimize false positives but may fail to detect subtle anomalies, whereas more flexible methods such as Isolation Forest or Hampel can better capture irregular patterns at the cost of increased variability.

Importantly, the behavior observed in the real-world dataset aligns with the trends identified in synthetic scenarios. No method dominates across all locations, and performance remains dependent on local signal characteristics, including variance, noise level, and temporal dynamics. This consistency between controlled and field-like conditions strengthens the validity of the experimental framework and supports its applicability to real IoT deployments.

Overall, these findings highlight the limitations of static anomaly detection approaches and emphasize the need for adaptive strategies. The proposed agent-assisted framework addresses this challenge by enabling context-aware method selection based on multiple criteria, including detection accuracy, reconstruction quality, and computational cost. This approach provides a practical balance between interpretability, performance, and efficiency, making it particularly suitable for resource-constrained IoT systems.

5.6. Synthesis of Results

Detection performance exhibits strong dependency on specific scenarios according to Table 8 which shows that different anomaly patterns need different methods for their detection. Table 10 demonstrates that no detection method achieves matching performance results across all tested scenarios.

Classical statistical methods maintain their competitive status because they offer researchers three essential advantages which include low computational requirements and stable performance and easy-to-understand results. ML techniques, IF and OC-SVM achieve better results when dealing with advanced situations which involve non-linear patterns because these methods require higher processing power and memory capacity. The framework achieves its objectives of detecting performance and reconstruction quality and computational efficiency by using its multi-criteria evaluation process together with its agent-assisted selection method. The system uses its adaptive feature to choose between different methods based on the input signal characteristics.

The real-world data validation process uses multiple sensor locations to verify Table 11 which proves that the proposed method operates successfully under real-world conditions. The evaluation process uses weak labeling as a baseline which demonstrates consistent results that match synthetic scenarios despite the absence of ground-truth labels.

The outcome demonstrates that the proposed framework maintains both stability and adaptability, which serves as a basis to support the discussion in Section 6.

6. Dataset Insights, Limitations, and Potential Extensions

This section presents main points which include the main findings from the dataset and the identification of its limitations and the description of future research possibilities which the proposed framework will enable.

6.1. Insights from Comparative Filter Behavior

Filter performance in various scenarios shows that its results depend on the specific anomaly structure which exists in the tested environment. Median and quantile filters show greater resistance against disruptive and extreme tail disturbances while the variance-based techniques show higher susceptibility to both non-stationary conditions and increasing variance patterns. The study found that different evaluation metrics lead to different performance rankings of the assessed items. Filters which aim to minimize reconstruction error create distorted trend patterns and periodic movement which the researchers observed in S3 and S6. The study proves that one metric evaluation fails to meet the requirements of IoT anomaly detection and it shows the need for multiple criteria decision-making systems.

6.2. Interpretability and Metric-Driven Decision Making

The proposed agent does not use concealed learning techniques which enable it to choose filters through direct assessment of signal quality. The cost function combines reconstruction error, anomaly rate, variance preservation, slope deviation, and residual spikes. The current system provides a decision-making framework that supports both transparent and traceable processes which prove essential for IoT systems that require long-term operational transparency and ability to conduct audits.

6.3. Limitations

The dataset contains synthetic elements which simulate realistic movements according to its intended purpose. Real-world implementation of the system will face obstacles because it will deal with incomplete information and nonstandard data collection methods and transmission errors. The study investigates temperature signals which exist in one

dimension while it lacks the ability to detect temperature signals from multiple sensors. The analysis of dynamic signals needs different window sizes because fixed window sizes do not provide suitable results. The framework creates two different obstacles which include its cost function design that requires fixed weight use and its statistical filters which face difficulties when handling complex non-linear patterns. These aspects define the scope of applicability.

6.4. Extensions

The dataset provides researchers with resources to conduct research about adaptive filtering methods and the process of parameter tuning and the development of hybrid statistical and learning-based approaches [32]. The system enables scientists to test different reinforcement learning techniques while using lightweight models which allow deployment in edge environments that require limited computational power. The proposed framework exists as the main foundation for its development through interpretable statistical filters, but the experimental results show that machine learning methods including IF and OC-SVM deliver comparable results with their advanced anomaly detection capabilities in S2, S3, S5, and S6. The results indicate that these methods can function as additional elements in the decision-making process which creates a hybrid system that combines understandable outputs with efficient processing and effective detection abilities. The integration of these systems creates a valuable research path for upcoming studies which focus on detecting non-stationary or highly irregular anomalies.

6.5. Evaluation of Field-like Scenario (SF)

The field-like scenario evaluates the pipeline performance during its operation in actual IoT environments. The assessment uses three evaluation methods because complete label data is not available for testing purposes. The results demonstrate that statistical techniques maintain their robustness and stability. Learning-based methods, IF and OC-SVM produce increased false positive rates when they operate in environments with excessive noise. The results show that lightweight interpretable methods work effectively in practical IoT deployments.

7. Conclusions and Future Work

The authors developed a framework which allows users to detect anomalies and restore signals from IoT temperature time-series data using an easily understandable and repeatable method. The system uses three traditional statistical filters, which are Hampel, IQR and Z-Score, to create a decision-making system that determines the best filtering method based on the signal attributes. The system enables adaptive functioning because it avoids utilizing models which create hidden processes and require high computational resources.

Seven synthetic scenarios, S1 to S7, produced results which demonstrate that filter performance depends on the different ways for creating anomalies. Median- and quantile-based methods show greater resistance against impulsive disturbances while variance-based methods experience performance loss when faced with non-stationary conditions. The decision mechanism uses multiple signal integrity criteria to choose between different methods because this approach produces better results across various situations.

The dataset introduced in this work also plays an important role, because it contains reconstructed signals and metric-based evaluations which enable systematic comparison of different approaches because it includes anomaly labels. The study creates reproducibility through its research methods which establish a standard for future IoT data quality research.

The authors used a field-like scenario as their experimental setup to connect laboratory experiments with actual environmental conditions. The results demonstrate that the proposed pipeline maintains its operational stability and system transparency when processed through actual sensor data. A combination of lightweight statistical methods with a basic decision system provides an effective solution which achieves performance results while maintaining system transparency and keeping operational expenses low. The research framework develops transparent statistical filters which scientists can interpret. The testing results show that the machine learning techniques Isolation Forest and One-Class SVM can function as additional parts of the system. The complex anomaly scenarios of S2, S3, S5, and S6 demonstrate this capability.

The authors plan to expand their validation process by testing larger datasets from actual field conditions and by studying deployment challenges which include sensor drift, calibration problems and communication failures. Also, the authors will investigate hybrid methods which combine statistical techniques with learning-based systems to enhance adaptability in complex environments.

Author Contributions: Conceptualization, L.M.P.; methodology, L.M.P.; software, L.M.P.; validation, L.M.P. and J.B.d.V.; formal analysis, L.M.P. and J.B.d.V.; investigation, L.M.P.; resources, L.M.P.; data curation, L.M.P. and J.B.d.V.; writing—original draft preparation, L.M.P.; writing—review and editing, L.M.P. and J.B.d.V.; visualization, L.M.P. and J.B.d.V.; supervision, J.B.d.V. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding authors.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

IF	Isolation Forest
IoT	Internet of Things
IQR	Interquartile Range
LSTM	Long Short-Term Memory
MAD	Median Absolute Deviation
OC-SVM	One Class SVM
RMSE	Root Mean Square Error
SVM	Support Vector Machines
TSB-UAD	Time-Series Benchmark for Univariate Anomaly Detection
UML	Unified Modeling Language

References

1. Shi, W.; Cao, J.; Zhang, Q.; Li, Y.; Xu, L. Edge computing: Vision and challenges. *IEEE Internet Things J.* **2016**, *3*, 637–646. [CrossRef]
2. Zhou, Z.; Chen, X.; Li, E.; Zeng, L.; Luo, K.; Zhang, J. Edge intelligence: Paving the last mile of artificial intelligence with edge computing. *Proc. IEEE* **2019**, *107*, 1738–1762. [CrossRef]
3. Premsankar, G.; Di Francesco, M.; Taleb, T. Edge computing for the Internet of Things: A case study. *IEEE Internet Things J.* **2018**, *5*, 1275–1284. [CrossRef]
4. Yousefpour, A.; Ishigaki, G.; Jue, J.P. Fog computing: Towards minimizing delay in the Internet of Things. *J. Syst. Archit.* **2019**, *91*, 1–16.
5. Zhang, Y.; Chen, X.; Jin, L.; Wang, X.; Guo, X. Network anomaly detection based on deep learning: A survey. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 2019–2021. [CrossRef]
6. Chalapathy, R.; Chawla, S. Deep learning for anomaly detection: A survey. *ACM Comput. Surv.* **2019**, *51*, 1–36.

7. Hundman, K.; Constantinou, V.; Laporte, C.; Colwell, I.; Soderstrom, T. Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, London, UK, 19–23 August 2018*; ACM: New York, NY, USA, 2018.
8. Ahmed, M.; Mahmood, A.N.; Hu, J. A survey of network anomaly detection techniques. *J. Netw. Comput. Appl.* **2016**, *60*, 19–31. [CrossRef]
9. Blázquez-García, A.; Conde, A.; Mori, U.; Lozano, J.A. A review on outlier/anomaly detection in time series data. *ACM Comput. Surv.* **2021**, *54*, 1–33. [CrossRef]
10. Pires, L.M. IoT Anomaly Detection Repository (Hampel, IQR, Z-Score). GitHub. 2025. Available online: https://github.com/prof-luispires/iot-anomaly-detection.git (accessed on 8 April 2026).
11. *Python*, version 3.11; Python Software Foundation: Beaverton, OR, USA, 2022. Available online: https://www.python.org (accessed on 8 April 2026).
12. Maronna, R.A.; Martin, R.D.; Yohai, V.J.; Salibián-Barrera, M. *Robust Statistics: Theory and Methods*, 2nd ed.; Wiley: Hoboken, NJ, USA, 2019.
13. Pearson, R.K.; Neuvo, Y.; Astola, J.; Gabbouj, M. The Hampel filter: An efficient robust outlier detection algorithm for real-time applications. *Digit. Signal Process.* **2016**, *59*, 1–18. [CrossRef]
14. Castro, L.N. *Exploratory Data Analysis: Descriptive Analysis, Visualization, and Dashboard Design*; CRC Press: Boca Raton, FL, USA, 2025.
15. Yaro, A.S.; Maly, F.; Prazak, P. Outlier detection in time-series RSS using Z-score with Sn estimator. *Appl. Sci.* **2023**, *13*, 3900. [CrossRef]
16. United Nations. *Transforming Our World: The 2030 Agenda for Sustainable Development*; United Nations: New York, NY, USA, 2015. Available online: https://sdgs.un.org/2030agenda (accessed on 8 April 2026).
17. European Parliament and Council. *Regulation (EU) 2024/1781 on Ecodesign Requirements for Sustainable Products*; European Parliament and Council: Washington, DC, USA, 2024. Available online: https://eur-lex.europa.eu (accessed on 8 April 2026).
18. European Commission. *Digital Product Passport: Transparency and Sustainability*; European Commission: Brussels, Belgium, 2024. Available online: https://data.europa.eu (accessed on 8 April 2026).
19. Hampel, F.R. The influence curve and its role in robust estimation. *J. Am. Stat. Assoc.* **1974**, *69*, 383–393. [CrossRef]
20. Tukey, J.W. *Exploratory Data Analysis*; Addison-Wesley: Reading, MA, USA, 1977.
21. Iglewicz, B.; Hoaglin, D.C. *How to Detect and Handle Outliers*; ASQC: Milwaukee, WI, USA, 1993.
22. Rousseeuw, P.J.; Hubert, M. Robust statistics for outlier detection. *WIREs Data Min. Knowl. Discov.* **2011**, *1*, 73–79. [CrossRef]
23. Roos-Hoefgeest Toribio, M.; Garnung Menéndez, A.; Roos-Hoefgeest Toribio, S.; Álvarez García, I. Speed-up Hampel filter for outlier detection. *Sensors* **2025**, *25*, 3319. [CrossRef] [PubMed]
24. Montgomery, D.C.; Runger, G.C. *Applied Statistics and Probability for Engineers*, 7th ed.; Wiley: Hoboken, NJ, USA, 2020.
25. Liu, F.T.; Ting, K.M.; Zhou, Z.-H. Isolation forest. In *Proceedings of the IEEE International Conference on Data Mining (ICDM), Pisa, Italy, 15–19 December 2008*; IEEE: New York, NY, USA, 2008; pp. 413–422.
26. Schölkopf, B.; Platt, J.C.; Shawe-Taylor, J.C.; Smola, A.J.; Williamson, R.C. Estimating the support of a high-dimensional distribution. *Neural Comput.* **2001**, *13*, 1443–1471. [CrossRef] [PubMed]
27. Shen, L.; Li, Z.; Kwok, J.T. Time-series anomaly detection using temporal hierarchical one-class network. In *Proceedings of the Neural Information Processing Systems (NeurIPS), Vancouver, BC, Canada, 6–12 December 2020*; NeurIPS: San Diego, CA, USA, 2020.
28. Paparrizos, J.; Kang, Y.; Boniol, P.; Tsay, R.S.; Palpanas, T.; Franklin, M.J. TSB-UAD: Benchmark suite for time-series anomaly detection. *Proc. VLDB Endow.* **2022**, *15*, 1697–1711. [CrossRef]
29. de Medeiros, K.; da Costa, K.A.; Papa, J.P.; Lisboa, C.O.; Munoz, R. A survey on anomaly detection in Internet of Things data using machine learning. *Sensors* **2023**, *23*, 3245. [CrossRef]
30. Chatterjee, A.; Ahmed, B.S. IoT anomaly detection methods and applications: A survey. *Internet Things* **2022**, *19*, 100568. [CrossRef]
31. DataVic. Sensor Readings with Temperature, Light, Humidity Every 5 Minutes at 8 Locations (2014–2015). Available online: https://discover.data.vic.gov.au/dataset/sensor-readings-with-temperature-light-humidity-every-5-minutes-at-8-locations-trial-2014-2015 (accessed on 8 April 2026).
32. Susto, G.A.; Schirru, A.; Pampuri, S.; McLoone, S.; Beghi, A. Machine learning for predictive maintenance: A multiple classifier approach. *IEEE Trans. Ind. Inform.* **2015**, *11*, 812–820. [CrossRef]