**Hochschule Konstanz**
Department of Computer Science

**Submitted by**
Samuel Tim
Student Number 307636

samuel.tim200yahoo.de

# Bachelor Thesis

## Energy Anomaly Detection with Machine Learning

**Konstanz, 31st December 2025**

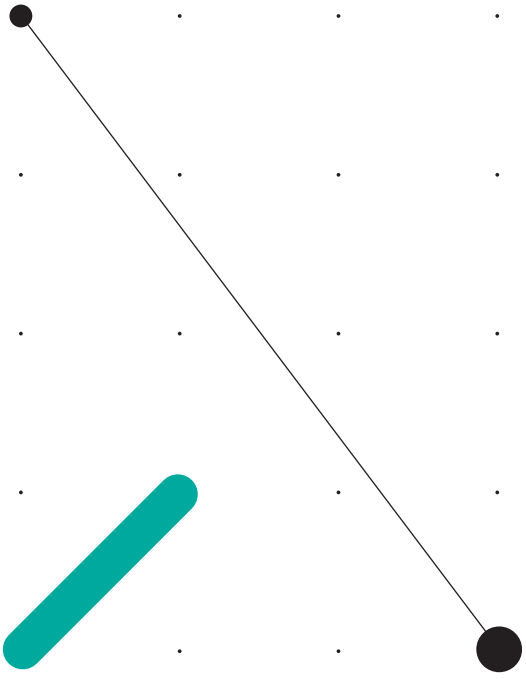
**Hochschule Konstanz**
Department of Computer Science

Bachelor Thesis

# Energy Anomaly Detection with Machine Learning

by

**Samuel Tim**



in Partial Fulfillment of the Requirements for the Degree of

**Bachelor of Science**
in Applied Computer Science

at the Hochschule Konstanz University of Applied Sciences,

| | |
|---|---|
| Student Number: | 307636 |
| Date of Submission: | 31st December 2025 |
| Supervisor: | **Prof. Dr. Marko Boger** |
| Second Examiner: | **Dipl.-Inf. Björn Erb** |

An electronic version of this thesis is available at http://repository.tudelft.nl/.

# Abstract

Buildings account for approximately 30% of global final energy consumption, while empirical studies estimate that between 4% and 18% of building energy use is attributable to anomalies such as technical faults, control and scheduling errors, and behaviour-induced energy misuse, which result in avoidable operational inefficiencies.

This thesis presents an integrated methodology for contextual anomaly detection in multivariate, non-stationary building-energy time series, enabling financial-impact estimation and automated root-cause attribution. The approach is fully implemented within an existing IoT building-management platform and includes a production-ready frontend for anomaly visualization and operational analysis.

To address structural limitations of existing anomaly-detection benchmarks for building-energy data, a dedicated evaluation dataset was constructed using the BOPTEST simulation environment, comprising a clean baseline and systematically injected multivariate, context-dependent anomaly scenarios. Multiple detection methods, including statistical, deep-learning, and foundation-model-based approaches, were evaluated on this benchmark.

The results indicate that stochastic prediction models with probabilistic output distributions are more suitable than deterministic point predictors for modelling multimodal building-energy behaviour and identifying contextual anomalies. Chronos-2 enables the practical application of time-series foundation models to multivariate energy telemetry without per-asset training, while mixture-density modelling was identified as a promising architectural direction for future research. The findings establish a methodological basis for a universal energy foundation model supporting zero-shot anomaly detection and standardized baseline comparison in accordance with IPMVP.

# Contents

# Glossary

**anomaly** observation or pattern that deviates significantly from a defined notion of normality. viii

**benchmark** standardized dataset and evaluation protocol used to compare the performance of different anomaly detection methods. viii

**confusion matrix** tabular summary of prediction results that counts true positives, true negatives, false positives, and false negatives. viii

**ground truth** reference labels that indicate for each observation whether it is considered normal or anomalous, used as a standard when evaluating detection performance. viii

**mislabeling** inconsistent assignment of anomaly labels to similar or identical patterns, which distorts evaluation by inflating false-negative rates. viii, 20

**precision** for anomaly detection, the proportion of predicted anomalous points or segments that are actually anomalous (true positives divided by all positive predictions). viii

**recall** for anomaly detection, the proportion of truly anomalous points or segments that are correctly detected (true positives divided by all actual anomalies). viii

**run-to-failure bias** systematic placement of anomalies at the end of a time series, which favors models that exploit positional cues rather than genuine signal deviations. viii, 20

**unrealistic anomaly ratio** an artificially high proportion of anomalous observations in a dataset compared to real-world systems, which can lead to over-optimistic performance estimates. viii, 20

# Acronyms

**AMI** Advanced Metering Infrastructure. 1

**BAS** Building Automation Systems. 1

**BOPTEST** Building Optimization Performance Test Framework. 5

**CNN** convolutional neural network. 20, 21

**IPMVP** International Performance Measurement and Verification Protocol. 5

**MCPA** Multivariate Context Point Anomaly. 4

**ML** machine learning. 21

**OmniAnomaly** stochastic recurrent neural network model OmniAnomaly. 20

**PA-F1** Point-Adjustment F1 score. 20

**Sub-PCA** subspace principal component analysis. 20

**TSAD** time series anomaly detection. 20

**TSB-AD** Time Series Benchmark for Anomaly Detection. 20, 21

**TSFM** time-series foundation model. 3

**VUS-PR** Volume Under the Surface–Precision Recall. 20, 21

# 1

# Introduction

## 1.1. Motivation and Economic Context

Buildings account for approximately 30% of global final energy consumption and more than 50% of global electricity consumption [Alš24]. Empirical studies indicate that avoidable operational anomalies—encompassing technical faults, suboptimal control strategies, and persistent behavioural misuse—account for between 4% and 18% of building energy use [Rot+04]. These inefficiencies frequently remain undetected because conventional threshold-based monitoring systems are not triggered.

Non-technical losses represent a quantifiable economic burden. Electricity theft results in annual losses exceeding 6 billion USD in the United States alone [MM09]. Furthermore, reports from the World Bank indicate that in some developing countries up to 50% of distributed electricity is lost due to theft [Ant09]. Such patterns of energy misuse constitute an economically relevant class of anomalies. While modern Building Automation Systems (BAS) are capable of detecting deviations from nominal operation, they typically neither quantify the associated financial impact nor provide systematic root-cause attribution, thereby limiting their operational and economic usefulness.

## 1.2. Digitalization of Buildings and Data Explosion

The implementation of Advanced Metering Infrastructure (AMI), which combines smart meters with communication networks, is expanding globally. In the United States, smart

meters had been deployed for approximately 77% of households and businesses by 2022, with the installed base projected to grow to about 134 million devices in 2024 and 142 million in 2026 [Edi24]. The increasing integration of digital infrastructure and sub-metering in modern building environments generates vast repositories of high-frequency telemetry. This abundance of data provides a unique opportunity for the application of advanced artificial-intelligence techniques that thrive on large-scale, high-resolution multivariate data to identify previously undetectable deviations.

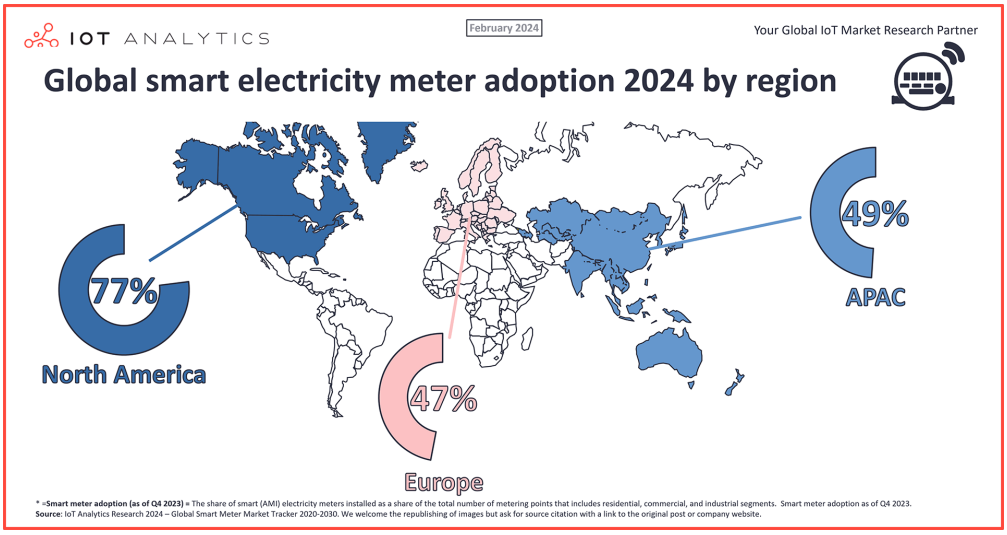



Figure 1.1: Global smart electricity meter adoption by region in 2024, illustrating the varying levels of AMI penetration across markets [IoT24].

## 1.3. Why Current Building Automation Systems Fail

Most deployed Building Automation Systems (BAS) rely on static rule-based logic and univariate statistical thresholds applied to individual sensor streams. Such approaches are structurally incapable of capturing the multivariate, context-dependent nature of building-energy behaviour and are therefore unable to distinguish between legitimate operational regime changes and true anomalous states.

More recent data-driven and machine-learning-based detection methods exhibit fundamental limitations. Many approaches operate on deterministic point predictions that fail to represent the stochastic, multimodal, and non-stationary characteristics of building-energy time series. As a result, these models impose context-agnostic deviation boundaries that treat identical absolute residuals as equally anomalous across fundamentally different operational regimes, leading to structurally incorrect anomaly semantics.

Sequential forecasting-based detectors further suffer from two critical failure modes when applied to sustained anomalies: (i) error propagation, where anomalous values corrupt the sliding input window and destabilize future predictions, and (ii) rapid baseline adaptation, where models quickly absorb anomalous states as normal operation, causing long-duration anomalies to disappear from the anomaly score signal :contentReference[oaicite:1]index=1. These effects undermine both detection reliability and financial loss quantification.

Furthermore, the majority of published anomaly-detection benchmarks rely on inadequately annotated datasets, implicit anomaly assumptions, and global point-anomaly definitions that can be captured by trivial threshold rules but fail to represent contextual and multivariate operational anomalies. These limitations significantly reduce the transferability of reported performance to real-world building operation.

Finally, existing systems rarely provide automated root-cause attribution or translate detected deviations into quantifiable financial impact, thereby limiting their practical value for operational decision-making and maintenance prioritization.

## 1.4. Emergence of Foundation Models for Time Series

The emergence of time-series foundation model (TSFM), such as Chronos-2 [Ans+25], marks a new paradigm in building-energy analytics. These models are designed to process multivariate, non-stationary, and stochastic data and provide probabilistic output distributions instead of single-value predictions. This enables the construction of normative operational bands that capture multimodal building behaviour and allow contextual deviations to be distinguished from normal variability.

A key advantage of foundation models is their zero-shot generalization capability. In contrast to asset-specific forecasting models, TSFMs do not require per-meter training or frequent retraining. Multivariate building telemetry can be provided directly as contextual input, while optional fine-tuning can be performed jointly across entire building portfolios. This makes foundation models particularly well suited to the inherently non-stationary nature of building-energy data and enables scalable deployment across large building estates.

## 1.5. System Context and Industrial Relevance

This research is conducted in the context of the Eliona IoT Building Management Platform (see Section 4.1), a production-grade multi-tenant system deployed in commercial and industrial building portfolios worldwide. Eliona integrates heterogeneous building automation systems, smart meters, and environmental sensors into a unified telemetry and analytics layer.

The anomaly-detection framework developed in this thesis is not a laboratory prototype, but a fully integrated subsystem within Eliona's operational architecture. It processes live building telemetry, performs stochastic anomaly detection, quantifies financial impact, localizes probable root causes, and exposes actionable insights through a production-ready frontend used by facility managers and energy operators.

This real-world deployment context defines both the functional requirements and the architectural constraints of the proposed methodology, including scalability, robustness to missing data, non-stationary baselines, explainability, and economic interpretability.

## 1.6. Problem Scope and Research Contributions

This thesis addresses the problem of detecting, economically quantifying, and diagnostically localizing contextual anomalies in multivariate building-energy time series within large-scale, non-stationary operational environments.

In contrast to traditional threshold-based and deterministic forecasting approaches, this work formulates anomaly detection as a stochastic, multivariate, context-dependent modeling problem. The objective is the design and implementation of an integrated, production-ready anomaly intelligence system that not only detects deviations, but also explains their technical origin, quantifies their economic impact, and derives operationally meaningful recommendations.

The proposed framework models expected building-energy behaviour as a multivariate, multimodal mixture distribution, allowing deviations to be evaluated probabilistically rather than against static thresholds. This formulation explicitly accounts for non-stationary baselines caused by seasonal shifts, occupancy changes, and long-term system drift, preventing legitimate regime transitions from being misclassified as anomalies. Methodologically, the work focuses on detecting Multivariate Context Point Anomaly (MCPA) and translating them into financially interpretable metrics.

The primary contributions of this thesis are:

- A formalization of building-energy anomaly detection as multivariate contextual point anomaly detection under multimodality and non-stationarity.
- An empirical and theoretical critique of deterministic forecasting-based anomaly detectors and their structural failure modes.
- A stochastic detection framework based on probabilistic normative bands derived from foundation models.
- A financially interpretable quantification layer that transforms deviations into monetary loss estimates.
- A hierarchical root-cause attribution pipeline grounded in building ontologies and causal dependencies.
- A domain-specific multivariate benchmark dataset generated via Building Optimization Performance Test Framework (BOPTEST) with labeled contextual fault scenarios (see Section 4.7).
- A fully integrated, scalable, multi-tenant implementation deployed within a production IoT building platform.
- A methodological foundation for the development of a universal energy foundation model supporting zero-shot anomaly detection and standardized baseline comparison in accordance with the International Performance Measurement and Verification Protocol (IPMVP).

This work assumes that the historical baseline used for model context represents nominal building operation. The framework is therefore designed to detect deviations emerging after baseline establishment and does not aim to retroactively identify faults that were already persistently present in historical reference data. Furthermore, the scope is limited to aggregated building-energy telemetry and does not target high-frequency electrical fault detection, equipment-level vibration analysis, or cybersecurity intrusion detection.

# 2

# Foundations

This chapter establishes the formal foundations required for contextual anomaly detection in building-energy telemetry. It characterizes the structural, statistical, and causal properties of building-energy data, defines the relevant anomaly taxonomies, and introduces the methodological and benchmarking concepts used throughout this thesis.

Based on these foundations, the chapter derives formal modeling requirements that constrain the design of detection, quantification, and attribution methodologies developed in the subsequent chapters.

## 2.1. Characteristics of Building Energy Data

Building-energy telemetry constitutes a multivariate, multimodal, and non-stationary stochastic process governed by physical, behavioural, and technical drivers. Effective anomaly detection therefore requires formal consideration of the structural and statistical properties of these data.

### 2.1.1. Multivariate Structure

Building energy data is inherently multivariate and interdependent. In addition to aggregate meter readings, relevant variables include environmental conditions, occupancy, and subsystem states. Cross-variable dependencies are fundamental: changes in environmental drivers induce correlated changes in technical system loads. Consequently,

anomaly detection must operate on multivariate joint behaviour rather than on isolated univariate series.

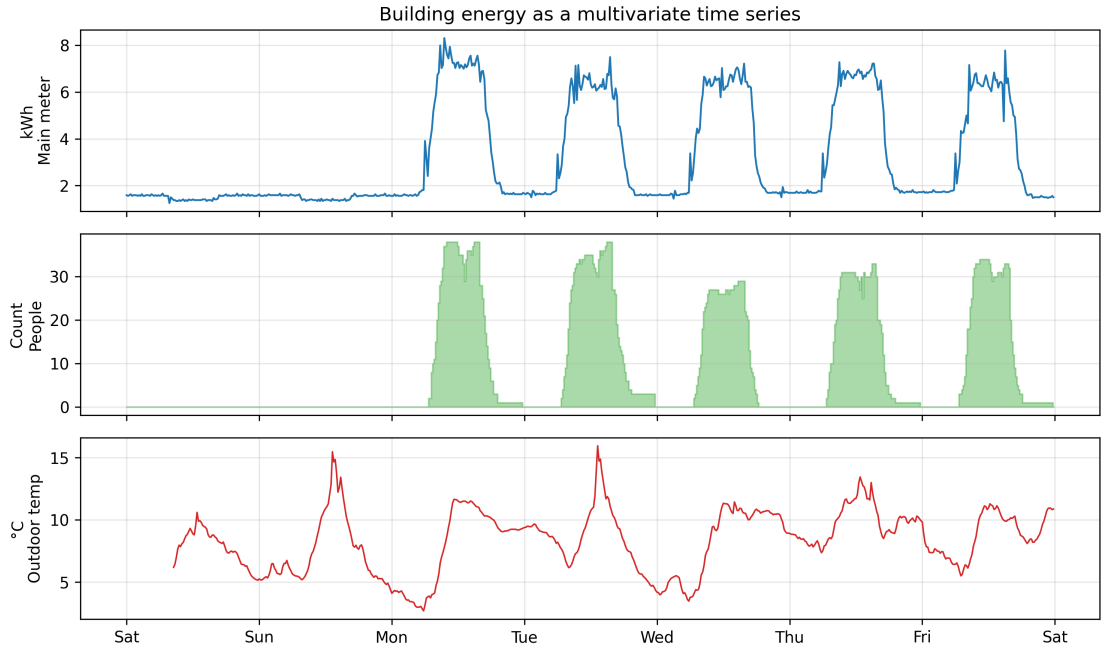



Figure 2.1: Representative multivariate time series showing the main meter load together with occupancy (people count) and outdoor temperature. The plot illustrates how multiple interdependent variables evolve jointly over time.

### 2.1.2. Causal Chain of Energy Consumption

Energy consumption emerges from a causal chain spanning demand generation, control logic, and mechanical execution. Environmental and occupancy conditions generate service demand; controllers translate demand into actuation commands; mechanical subsystems execute these commands, producing measurable energy use. Deviations observed at aggregate meters therefore frequently originate from faults located in upstream sensing or control layers.

Understanding the causal chain, as illustrated in Figure 2.2, is a prerequisite for localizing anomalous behavior within building systems. A deviation observed in the building's main meter often originates from a fault located in a preceding stage of the technical hierarchy, such as a sensor error or a logic failure in the control layer.

For instance, a malfunctioning temperature sensor reporting an erroneous heat spike triggers a cascade of responses. The control layer interprets this false data as a thermal requirement and initiates a cooling command to counteract the perceived heat. This signal causes the supply layer to activate mechanical components, such as

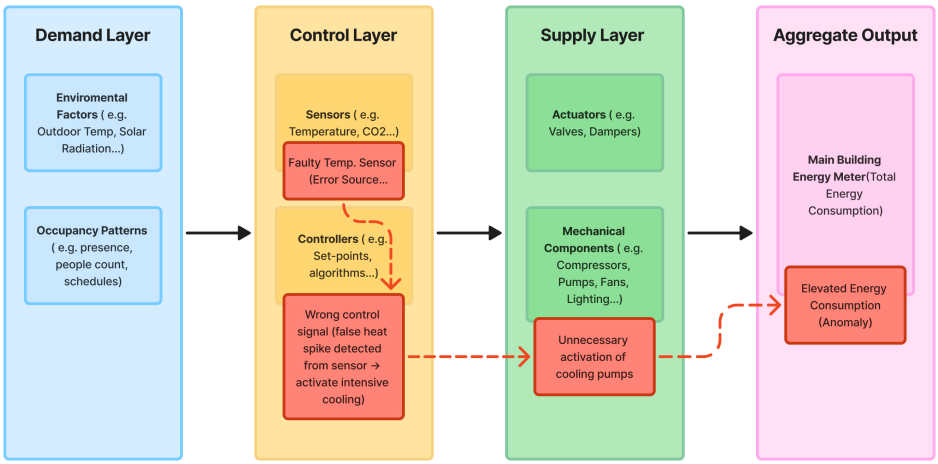



Figure 2.2: Causal chain of building energy consumption from demand over control to supply layer.

cooling pumps and compressors. These devices consume electrical energy to satisfy the requested cooling load. Consequently, the aggregate output layer, represented by the building's main energy meter, records a significant increase in consumption. In this scenario, the measured energy spike is not a result of an actual physical need but acts as a symptom of a failure located deeper in the technical hierarchy.

### HVAC and Environmental Drivers

HVAC systems dominate building energy demand. Thermal gradients, solar radiation, humidity, and scheduling logic determine cooling and heating loads. Suboptimal control strategies, scheduling conflicts, and mechanical degradation induce baseline drift and excessive consumption, generating anomalies that are often operationally normal yet energetically inefficient.

### Occupancy and Internal Loads

Human activity introduces stochastic variability through lighting, appliance use, and thermal gains. Behavioural interventions can decouple consumption from environmental drivers, while IT infrastructure introduces discrete operational regimes. These effects contribute to multimodality and regime-dependent energy patterns.

### Structural Moderators and Data Integrity

Building envelope characteristics and thermal inertia modulate system response dynamics. Interdependencies between subsystems propagate anomalies across services. Digital measurement infrastructure introduces non-physical artifacts, including

missing values and aggregation spikes, which must be distinguished from physical faults during preprocessing.

### 2.1.3. Temporal Dependence and Persistence

Building-energy telemetry exhibits strong temporal autocorrelation caused by thermal inertia, operational ramp-up dynamics, and persistent high-load device states. Consequently, short-term system behaviour is highly predictable under nominal operation, while slow-developing faults and sustained inefficiencies may remain concealed within otherwise smooth trajectories.

This persistence simultaneously stabilizes short-term forecasting and undermines detection of long-duration anomalies, particularly when sequential models rapidly absorb anomalous regimes into their predictive baseline.

### 2.1.4. Seasonality and Periodicity

Building-energy consumption follows pronounced daily, weekly, and seasonal periodicities driven by occupancy cycles, control schedules, and climatic seasons. These regime-dependent patterns form a repetitive operational fingerprint.

Anomaly detection must therefore distinguish contextual violations of expected periodic regimes (e.g., weekday-level consumption during weekends) from absolute deviations.

### 2.1.5. Statistical Distribution and Non-Stationarity

Empirical building-energy distributions deviate substantially from unimodal Gaussian assumptions and exhibit multimodal mixture structures with heavy tails due to discrete operational regimes and heterogeneous subsystem interactions. Deterministic point estimates are therefore insufficient to represent normative behaviour.

Sparse coverage of extreme weather and rare operational states introduces causal ambiguity and increases false anomaly rates for previously unobserved but physically valid conditions. Furthermore, building-energy telemetry is non-stationary; long-term baseline drift caused by seasonal transitions, equipment degradation, and persistent occupancy changes continuously shifts normative distributions, necessitating probabilistic, context-aware modeling.

Figure 2.3 illustrates this effect on real data: the measured main-meter consumption concentrates in several dense regions at lower loads and exhibits a pronounced right

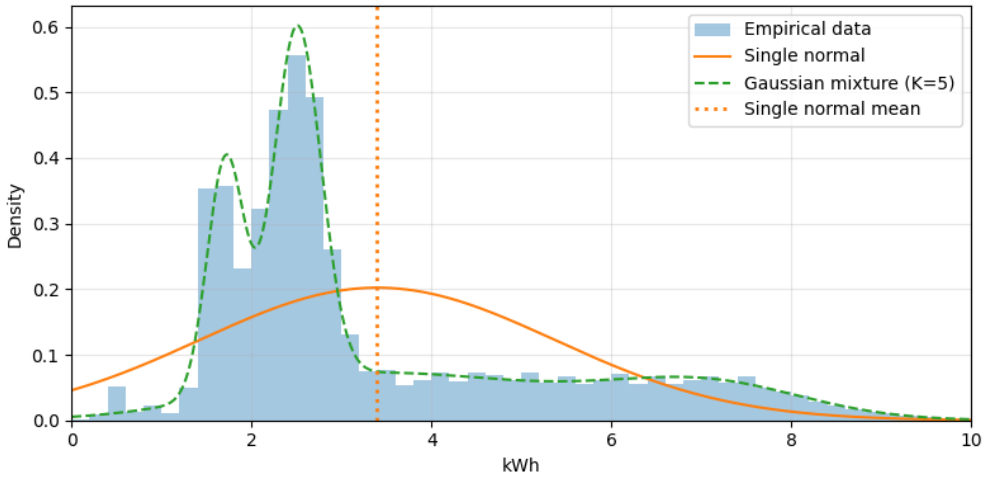



Figure 2.3: Empirical distribution of the building's main meter (15-minute kWh values, histogram) with an overlaid single normal distribution and a Gaussian mixture model with five components, illustrating the mismatch between a unimodal Gaussian model and the multimodal, heavy-tailed structure of real building energy data.

tail. A single normal distribution smooths over these structures and underestimates tail probabilities, whereas the fitted Gaussian mixture adapts to the multiple modes and better traces the empirical density.

### 2.1.6. Data Acquisition and Semantic Structure

The transformation of physical energy consumption into digital telemetry follows a multi-stage acquisition pipeline that converts electrical quantities into structured multivariate time series suitable for algorithmic analysis.

**Ontological Modeling:** To ensure interpretability and causal localization, the telemetry is mapped to a semantic ontology that encodes the physical and logical relationships between meters, subsystems, and devices [Eli25c]. This ontological layer enables detected anomalies to be localized within the technical hierarchy rather than remaining aggregated deviations at the main meter level.

**Standardized Units:** Raw meter readings are converted into standardized physical units (e.g., kWh) to ensure consistency across heterogeneous hardware and communication interfaces.

### 2.1.7. Data Continuity and Transmission Artifacts

The integrity of telemetry streams depends on the stability of the communication infrastructure. Network-level distortions introduce non-physical artifacts that must be distinguished from actual building faults.

**Transmission Gaps:** Communication failures produce missing values that interrupt temporal continuity and require correction during preprocessing.

**Aggregation Spikes:** Buffered data retransmission following outages may produce virtual load spikes, reflecting delayed reporting rather than physical surges in energy demand.

## 2.2. Foundations of Anomaly Detection

Anomaly detection aims to identify observations or patterns that deviate from an implicit notion of normality. In time-series anomaly detection (TSAD), deviations are defined relative to temporal structure, persistence, and regime-dependent behaviour rather than isolated numerical values. The taxonomy adopted in this work follows the benchmark framework proposed by Paparrizos et al. [Pap+22].

### 2.2.1. Dimensionality and Normality Regimes

The complexity of anomaly detection is governed by the dimensionality of the time series and the number of normative operational regimes.

**Dimensionality:** Univariate time series describe a single system variable, whereas multivariate time series jointly model multiple interdependent variables. The dataset analyzed in this work is multivariate, combining aggregate energy consumption with environmental and occupancy drivers to resolve causal ambiguity.

**Normality Regimes:** Building-energy telemetry operates under multiple normative regimes driven by seasonal, operational, and occupancy-dependent contexts. Consequently, “normal” behaviour is regime-specific rather than globally invariant.

The analyzed data is therefore classified as multivariate with multi-mode normality, necessitating detection models that adapt to shifting baselines and cross-variable dependencies.

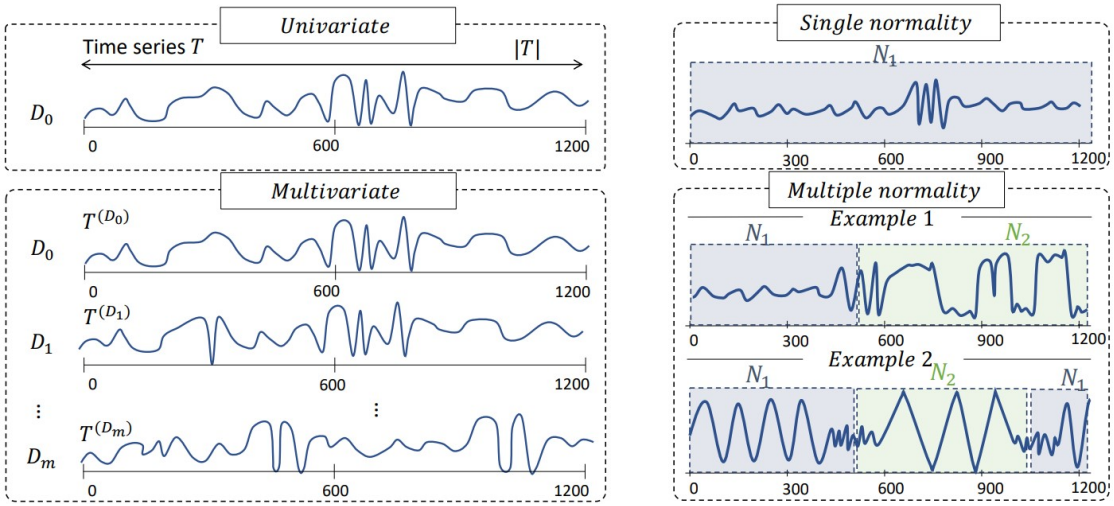



Figure 2.4: Schematic illustration of time series types along two axes—dimensionality (univariate vs. multivariate) and normality regimes (single-mode vs. multi-mode). Adapted from Boniol et al.'s tutorial on new trends in time series anomaly detection [**Boniol2023NewTrends**].

### 2.2.2. Terminology: Multivariate and Multi-Target Time Series

Throughout this thesis, the term *multivariate* denotes covariate-conditioned time-series modelling. That is, anomaly detection is performed on a single primary energy meter while explicitly conditioning on multiple exogenous driver variables such as weather, occupancy and calendar information. This formulation reflects the standard analytical view in building-energy modelling, where contextual variables are required to resolve causal ambiguity and to distinguish contextual anomalies from physically normal load variations.

In contrast, some anomaly-detection literature uses the term multivariate to describe joint modelling of multiple sensor channels as simultaneous prediction targets. In order to avoid ambiguity, this thesis refers to such settings as *multi-target* (or multi-sensor) time-series modelling.

Accordingly, all experimental investigations in this work address single-target, covariate-conditioned contextual anomaly detection rather than joint multi-target anomaly detection.

### 2.2.3. Structural Classes of Anomalies

**Point Anomalies:** Individual observations that deviate from expected behaviour.

**Global Point Anomalies:** Deviations outside the global historical range.

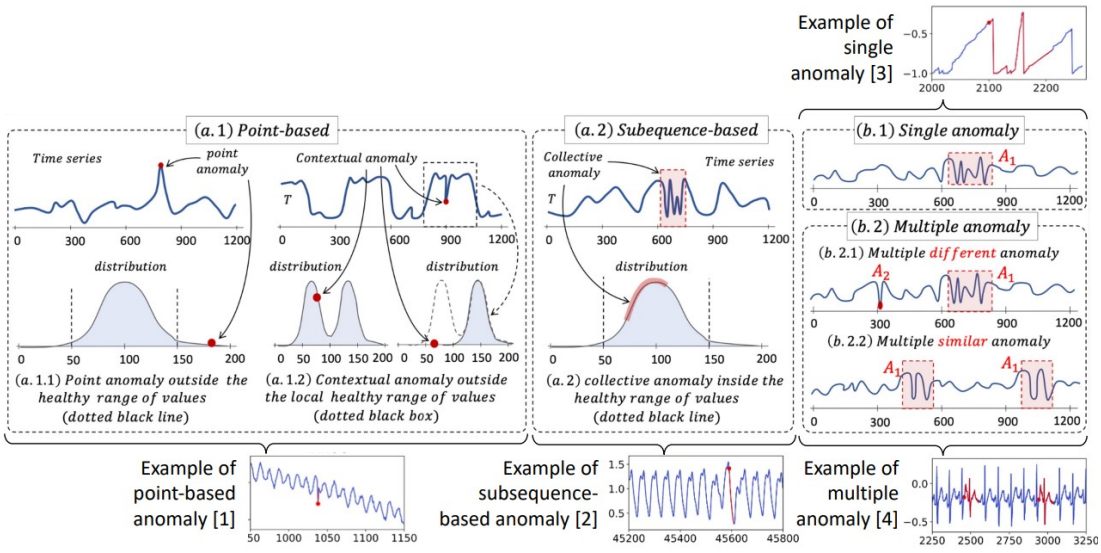



Figure 2.5: Taxonomy of time series anomalies along structural and multiplicity dimensions, distinguishing global and contextual point anomalies, subsequence-based anomalies, and their occurrence as single, multiple different, or multiple similar events. Adapted from Boniol et al.'s tutorial on new trends in time series anomaly detection [**Boniol2023NewTrends**].

**Contextual Point Anomalies:** Deviations from regime-dependent normative behaviour defined by temporal or exogenous context.

**Subsequence Anomalies:** Deviations manifested through abnormal temporal patterns.

### 2.2.4. Multiplicity of Occurrence

**Single Anomalies:** Isolated anomalous events.

**Multiple Similar Anomalies:** Recurrent manifestations of the same anomaly pattern.

**Multiple Different Anomalies:** Co-occurring anomalies of heterogeneous types.

## 2.3. Methodological Approaches to Anomaly Detection

Anomaly detection transforms raw time-series telemetry into actionable information by assigning each observation a degree of abnormality and mapping it to a binary decision boundary.

### 2.3.1. Anomaly Scores

Most detection algorithms output an anomaly score $s_i$ per timestamp, which quantifies deviation from learned normative behaviour. Binary alerts are obtained by thresholding this score, yielding a time series of nominal and anomalous states.

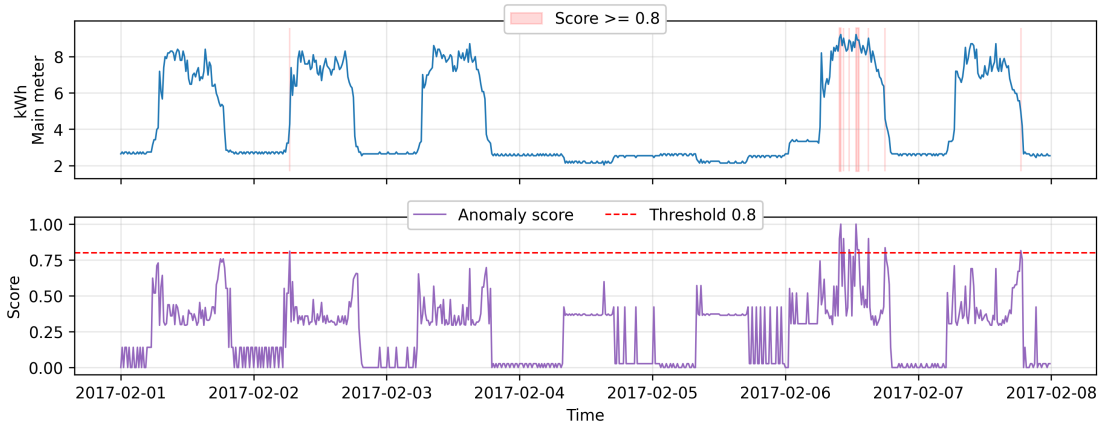



Figure 2.6: Example of an anomaly score $s_i \in [0, 1]$ aligned with the underlying time series. A threshold of 0.8 separates normal points from those flagged as anomalous.

### 2.3.2. Learning Paradigms

The applicability of detection methods is governed by the availability of labelled data:

**Supervised:** Requires explicit labels for both normal and anomalous states; rarely feasible in building operations.

**Semi-Supervised:** Learns normative behaviour from assumed healthy historical data; commonly used in building-energy monitoring.

**Unsupervised:** Operates without labelled baselines; typically applied during system commissioning or cold-start phases.

### 2.3.3. Families of Detection Methods

Anomaly detection approaches are grouped into three methodological families:

**Distance-Based:** Identify anomalous subsequences by pattern dissimilarity.

**Density-Based:** Detect low-probability observations in learned feature distributions.

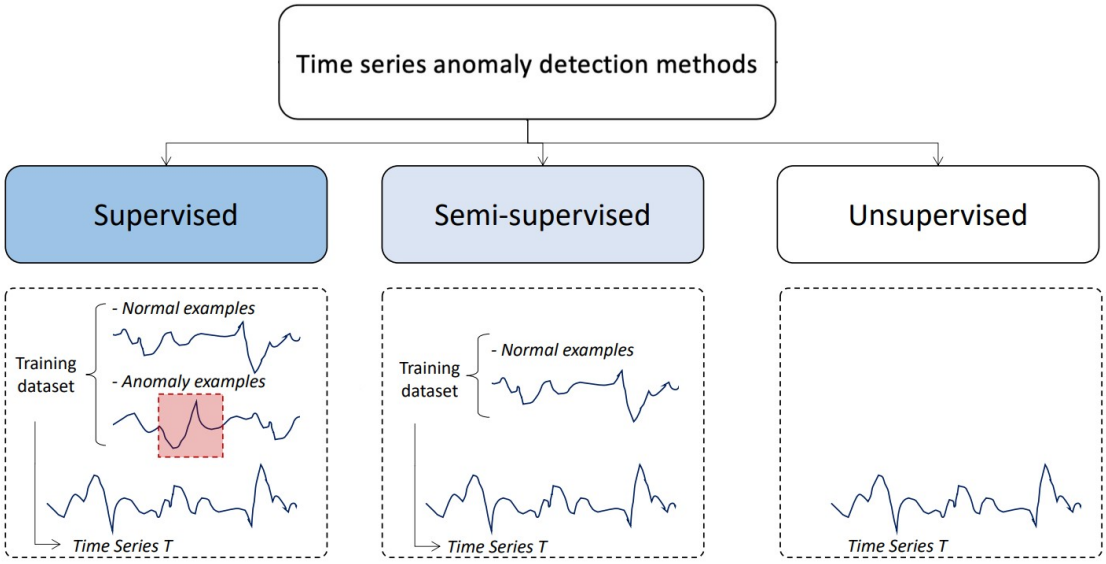



Figure 2.7: Schematic overview of supervised, semi-supervised, and unsupervised learning paradigms for anomaly detection, adapted from Boniol et al.'s tutorial on new trends in time series anomaly detection [**Boniol2023NewTrends**].

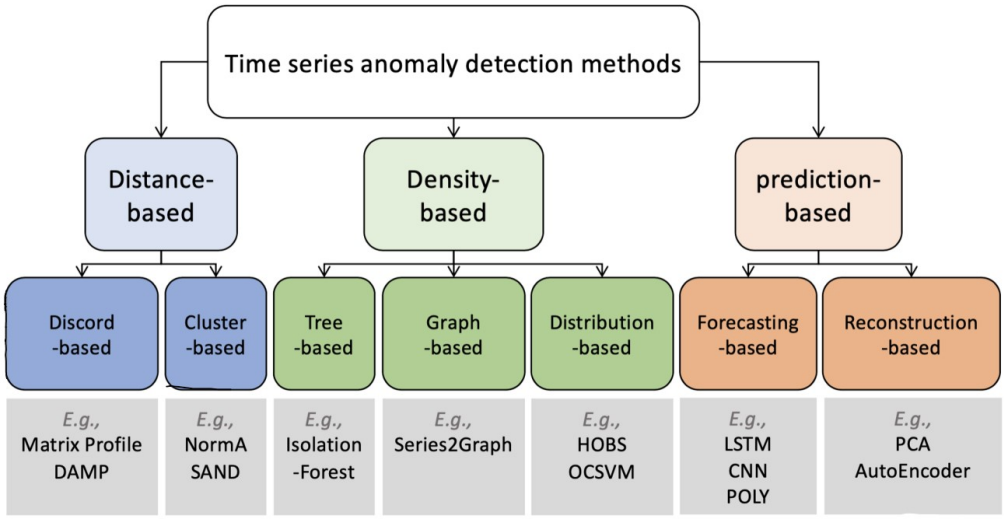



Figure 2.8: Hierarchical taxonomy of anomaly detection methods grouped into distance-based, density-based, and prediction-based families, adapted from Boniol et al.'s tutorial on new trends in time series anomaly detection [**Boniol2023NewTrends**].

**Prediction-Based:** Identify deviations via residuals between predicted and observed values, including forecasting- and reconstruction-based variants.

This thesis focuses on prediction-based approaches, as they are the only methodological family that provides an explicit expected-value baseline, enabling deviations to be quantified in physical units and directly translated into financial impact. This property is essential for contextual building-energy anomaly detection and economic loss estimation.

## 2.4. Benchmarking Foundations

Benchmarking evaluates anomaly detection performance against labelled reference datasets by comparing model decisions to ground-truth annotations.

### 2.4.1. Binary Labels and Confusion Matrix

Each observation is assigned a binary label (normal vs. anomalous). Model predictions are evaluated using the confusion matrix, yielding counts of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).

### 2.4.2. Evaluation Metrics

Performance is summarized using precision, recall, and the F1 score:

$$\text{Precision} = \frac{TP}{TP + FP}, \quad \text{Recall} = \frac{TP}{TP + FN}, \quad F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}.$$

These metrics quantify alarm reliability, detection sensitivity, and their balanced trade-off, respectively.

## 2.5. Synthesis of Foundations

Building-energy telemetry constitutes a multivariate, multimodal, and non-stationary stochastic process governed by a causal chain spanning demand, control, and me-

chanical execution. Deviations observed at aggregate meters therefore represent manifestations of upstream technical or behavioural faults rather than isolated numerical outliers.

From the structural and statistical properties of these data, the following formal requirements for anomaly detection follow:

1. **Probabilistic multivariate modeling**
   Expected behaviour must be represented as a multivariate mixture distribution rather than a deterministic point estimate, in order to capture multimodal operating regimes and cross-variable dependencies.

2. **Robustness to temporal dependence and persistence**
   Detection must remain stable under strong autocorrelation, seasonal regime shifts, and long-duration persistence, such that sustained deviations are not absorbed into the learned baseline.

3. **Separation of physical faults and digital artifacts**
   Transmission gaps, aggregation spikes, and other digital artifacts must be distinguished from physical anomalies through explicit preprocessing and data-quality handling.

Furthermore, building-energy telemetry is formally classified as a multivariate, multimode time series dominated by contextual point anomalies. Subsequence faults therefore manifest as contextual point deviations at the aggregation horizons relevant for energy monitoring. Persistent deviations present in historical baselines may be absorbed into learned normality (normality drift), representing a fundamental constraint for semi-supervised and unsupervised detection paradigms.

# 3

# Related Work

## 3.1. Classical Energy Baseline and Rule-Based Detection

Early work on energy anomaly detection in buildings is dominated by deterministic baselines and expert-driven rule systems that encode normative consumption behaviour explicitly. These approaches originate from energy engineering practice and are widely deployed in building management systems due to their transparency and low computational complexity.

Peña et al. [Peñ+16] present a representative rule-based framework for smart buildings in which energy efficiency indicators are derived from HVAC operation and expert knowledge is formalized into a set of anomaly detection rules using data mining techniques. Their system detects predefined inefficiency patterns based on threshold violations and logical conditions applied to multiple sensor streams. While such approaches provide interpretable diagnostics and are well suited for known fault patterns, they rely on static expert rules and lack adaptability to evolving building behaviour, seasonal regime changes, and unseen anomaly types.

Regression-based baselining methods constitute another classical detection paradigm. Liu and Nielsen [LN16] propose an online regression framework for smart-meter anomaly detection in which expected consumption is estimated via supervised learning models and anomalies are detected as residual deviations. These methods enable scalable real-time detection and support streaming deployment; however, they assume relatively stationary consumption baselines and primarily operate on deterministic point forecasts, limiting their robustness under multimodal and non-Gaussian energy distributions.

Overall, classical rule-based and regression-based approaches establish important foundations for energy anomaly detection, but their deterministic formulation and reliance on static baselines restrict their ability to resolve contextual, regime-dependent, and stochastic deviations that characterize modern building-energy telemetry.

## 3.2. Reliability and Benchmarking: The TSB-AD Framework

The selection of an appropriate detection methodology is constrained by systemic issues within the existing research landscape. Liu and Paparrizos [LP24] identify these issues as the "elephant in the room," demonstrating that apparent progress in time series anomaly detection (TSAD) is often an artifact of flawed evaluation practices rather than algorithmic superiority.

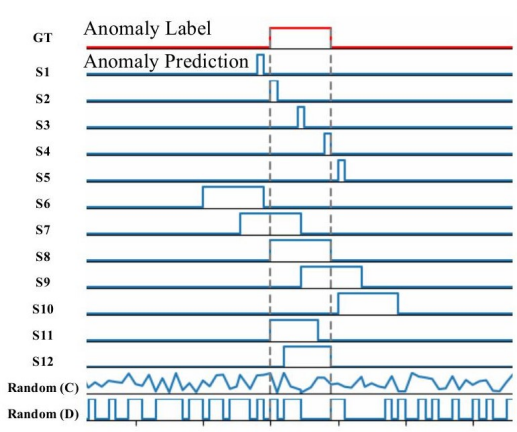
### 3.2.1. Systemic Flaws and Metric Reliability

Historical results are often compromised by three documented data-level flaws. First, mislabeling leads to artificially high false-negative rates. Second, a prevalent run-to-failure bias rewards models that simply prioritize temporal position. Finally, unrealistic anomaly ratios fail to reflect the rarity of faults in physical systems.

The "illusion of progress" is further attributed to point-wise metrics like Point-Adjustment F1 score (PA-F1), which facilitates a significant overestimation of model performance by rewarding a detection if even a single point within an anomalous segment is identified. To ensure accuracy, this research adopts Volume Under the Surface–Precision Recall (VUS-PR), established by Liu and Paparrizos [LP24] as the robust standard for providing threshold-independent evaluation resistant to temporal lags and noisy scoring.

### 3.2.2. Benchmark Evaluation and Model Hierarchy

Evaluation across 1 070 curated time series reveals that statistical methods like subspace principal component analysis (Sub-PCA) [LGW04] dominate univariate settings, whereas deep learning architectures demonstrate superior modeling capacity in multivariate scenarios (Time Series Benchmark for Anomaly Detection (TSB-AD)-M). As shown in Figure 3.2, convolutional neural networks (convolutional neural network (CNN)) [Wu17] and generative models like stochastic recurrent neural network model OmniAnomaly (OmniAnomaly) [Su+19] consistently outperform statistical baselines in capturing non-




Bias | Indiscrimination | Lack of adaptability

| Scenario | Threshold-independent | | | | | Threshold-dependent | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | AUC-PR | AUC-ROC | VUS-PR | VUS-ROC | PATE | Standard-F1 | PA-F1 | Event-based-F1 | R-based-F1 | Affiliation-F |
| S1 | 0.04 | 0.50 | 0.13 | 0.54 | 0.03 | 0.00 | 0.00 | 0.00 | 0.00 | 0.95 |
| S2 | 0.14 | 0.55 | 0.17 | 0.56 | 0.58 | 0.18 | 1.00 | 1.00 | 0.44 | 0.98 |
| S3 | 0.14 | 0.55 | 0.17 | 0.56 | 0.58 | 0.18 | 1.00 | 1.00 | 0.44 | 0.99 |
| S4 | 0.14 | 0.55 | 0.17 | 0.56 | 0.58 | 0.18 | 1.00 | 1.00 | 0.44 | 0.98 |
| S5 | 0.04 | 0.50 | 0.13 | 0.54 | 0.18 | 0.00 | 0.00 | 0.00 | 0.00 | 0.95 |
| S6 | 0.04 | 0.48 | 0.18 | 0.62 | 0.03 | 0.00 | 0.00 | 0.00 | 0.00 | 0.93 |
| S7 | 0.27 | 0.74 | 0.61 | 0.85 | 0.68 | 0.50 | 0.80 | 0.67 | 0.55 | 0.98 |
| S8 | **1.00** | **1.00** | **1.00** | **1.00** | **1.00** | **1.00** | **1.00** | **1.00** | **1.00** | **1.00** |
| S9 | 0.27 | 0.74 | 0.61 | 0.85 | 0.60 | 0.50 | 0.80 | 0.67 | 0.55 | 0.98 |
| S10 | 0.04 | 0.48 | 0.18 | 0.62 | 0.14 | 0.00 | 0.00 | 0.00 | 0.00 | 0.93 |
| S11 | 0.81 | 0.90 | 0.81 | 0.90 | 0.96 | 0.89 | 1.00 | 1.00 | 0.91 | 1.00 |
| S12 | 0.81 | 0.90 | 0.81 | 0.90 | 0.91 | 0.89 | 1.00 | 1.00 | 0.91 | 1.00 |
| Random (C) | 0.06 | 0.56 | 0.09 | 0.72 | 0.06 | 0.12 | 0.73 | 0.21 | 0.16 | 0.70 |
| Random (D) | 0.04 | 0.51 | 0.08 | 0.66 | 0.34 | 0.08 | 0.15 | 0.08 | 0.09 | 0.68 |

Figure 3.1: Reliability analysis of evaluation measures across different anomaly prediction scenarios. The red segment at the top represents the ground truth anomaly label, followed by various prediction signals (S1–S12 and random). The adjacent table indicates the resulting scores for threshold-independent and threshold-dependent metrics. Adapted from Liu and Paparrizos [LP24].

linear dependencies across multiple sensor channels [Su+19].

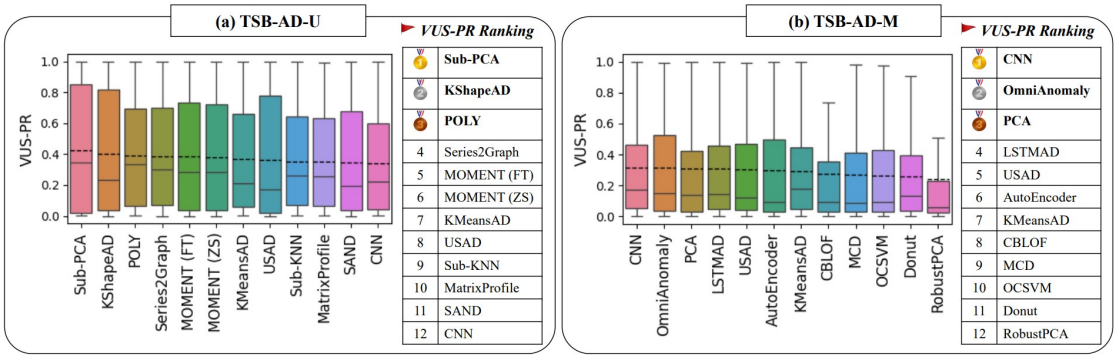



Figure 3.2: Accuracy evaluation of the top 12 methods on (a) univariate (TSB-AD-U) and (b) multivariate (TSB-AD-M) datasets based on the VUS-PR metric. Adapted from Liu and Paparrizos [LP24].

### 3.2.3. Implications for Multivariate Context Point Anomalies

While the TSB-AD benchmark provides a critical foundation for metric selection, its direct application to building energy telemetry is limited by several domain-specific gaps. The benchmark established that machine learning (ML) architectures like CNN excel in multivariate dependency modeling, while foundation models demonstrate superior efficacy in point anomaly identification. However, the TSB-AD-M partition contains a limited representation of multivariate point anomalies; the majority of its instances consist of sequence-based deviations or global outliers rather than contextual point anomalies.

Furthermore, Liu and Paparrizos [LP24] primarily evaluated foundation models in

univariate contexts, leaving their performance in multivariate environments unexplored. It is important to note that the term *multivariate* in Liu and Paparrizos [LP24] refers to joint multi-sensor anomaly detection, whereas in this thesis it denotes covariate-conditioned detection on a single primary meter. Consequently, the benchmark does not cover multivariate contextual anomaly detection in the sense addressed in this work.

For building energy systems, the benchmark lacks specific energy-sector data and does not account for the longitudinal nature of building operations. In real-world scenarios, researchers often have access to multiple years of historical data, which allows for the establishment of robust baselines. Unlike the static snapshots in many benchmarks, building data is subject to slow behavioral drifts (e.g., equipment aging). This necessitates a benchmark setup where models can continuously learn from historical patterns before being evaluated on anomalies. Consequently, while this thesis adopts the robust evaluation principles and metric recommendations of Liu and Paparrizos [LP24], the experimental design is explicitly adapted to the requirements of multivariate contextual anomaly detection in longitudinal building-energy telemetry, enabling continuous baseline learning under non-stationarity.

### 3.2.4. Large-Scale Supervised Energy Benchmarks: LEAD 1.0

A prominent large-scale benchmark for energy anomaly detection is LEAD 1.0 [GA22], which provides manually annotated hourly electricity consumption data from 1 413 commercial buildings. Anomalies are labeled based on visually observable deviations from daily and weekly load patterns and include global point anomalies and collective anomalies.

The availability of explicit anomaly labels has enabled supervised classification approaches to achieve extremely high reported detection scores. Recent competition results demonstrate that gradient-boosted tree ensembles combined with extensive change-of-value feature engineering can achieve ROC-AUC values above 0.98 by directly learning the human labeling patterns [Fu22].

However, LEAD 1.0 primarily captures globally visible pattern breaks and does not encode contextual inefficiencies, multivariate causal dependencies, long-term baseline drift, or economic impact semantics. Consequently, supervised models trained on LEAD effectively learn to imitate human visual judgments rather than to detect physically or economically relevant inefficiencies. These properties limit the transferability of LEAD-based detection results to real-world building energy management.

## 3.3. Comparative Analysis of Deep Learning and Foundation Models in Energy Systems

The landscape of time-series anomaly detection in energy systems has evolved from classical statistical heuristics toward complex deep learning architectures and, more recently, time-series foundation models. While Morshedi and Matinkhah [MM25] provide a comprehensive survey of convolutional, recurrent, and adversarial neural architectures in IoT anomaly detection, the specific structural properties of building-energy telemetry impose substantially different modeling requirements.

### 3.3.1. Deep Generative Models and the Advantage of Reconstruction

A central methodological distinction in building-energy research lies between deterministic forecasting models and probabilistic generative models. Azzalini et al. [Azz+25] demonstrate that recurrent autoencoder architectures consistently outperform convolutional variants due to their ability to capture long-range temporal dependencies in sequential meter data.

Within variational autoencoder frameworks, reconstruction probability (RP) has been shown to outperform simple reconstruction error (RE) by explicitly accounting for reconstruction variance, thereby increasing robustness against stochastic fluctuations inherent to building operations. This modeling principle underlies generative architectures such as OmniAnomaly, which employ stochastic recurrent neural networks to learn latent representations of multivariate building telemetry and to characterize normal operational behavior probabilistically [Su+19].

### 3.3.2. Time-Series Foundation Models in the Energy Domain

Time-series foundation models introduce a paradigm shift by enabling zero-shot and few-shot generalization across heterogeneous datasets. Hela, Handigol, and Arjunan [HHA25] evaluate foundation models such as TimeGPT [GCM23] and MOMENT [Gos+24] on the LEAD 1.0 benchmark, showing that these models exhibit strong zero-shot capabilities for detecting globally visible point anomalies in building-energy time series.

However, benchmark results further indicate that compact generative architectures may outperform foundation models in forecasting-residual-based anomaly detection tasks. Specifically, variational autoencoders trained from scratch surpass large foundation models such as MOMENT [Gos+24] on LEAD 1.0, while Liu and Paparrizos

[LP24] similarly report dominance of lightweight statistical and neural architectures in benchmark-driven TSAD competitions.

Crucially, these findings are confined to deterministic forecasting-residual paradigms and snapshot-based benchmark formulations. Existing benchmarks primarily encode globally visible or subsequence-based anomalies and evaluate models under unimodal Gaussian residual assumptions. They do not represent multivariate contextual inefficiencies, regime-dependent multimodality, long-term baseline drift, or probabilistic deviation semantics that characterize real-world building energy telemetry.

This work therefore departs from the conventional residual-based anomaly detection formulation by treating building-energy anomaly detection as probabilistic deviation from a contextual multivariate baseline. In this regime, foundation models capable of native distributional forecasting—such as Chronos-2—provide architectural capabilities that enable explicit modeling of regime-dependent mixture densities, which are structurally required to represent the multimodal operational states of buildings.

### 3.3.3. Synthesis of Related Work

The review of established methodologies demonstrates a technical transition from deterministic expert systems toward probabilistic deep learning architectures. Classical rule-based frameworks and regression-based baselining provide high interpretability and low computational complexity but exhibit restricted adaptability to the non-stationary and multimodal characteristics of building telemetry. The assessment of current methodologies is frequently compromised by systemic flaws in existing benchmarks, including mislabeling and biased evaluation metrics such as PA-F1. To resolve these deficiencies, recent research emphasizes robust evaluation standards like VUS-PR and the utilization of deep generative models such as OmniAnomaly. While large-scale supervised datasets like LEAD 1.0 enable high detection scores, they primarily reflect human visual judgments rather than contextual inefficiencies or multivariate causal dependencies. Foundation models such as Chronos-2 offer zero-shot generalization and the capacity for distributional forecasting, yet their application to multivariate contextual detection remains a significant research gap. Consequently, this work departs from conventional deterministic residuals in favor of a probabilistic formulation that explicitly models regime-dependent mixture densities to bridge the identified gap between technical detection and operational remediation.

# 4

# Methodology

## 4.1. System Context: The Eliona IoT Platform

The anomaly detection system is integrated into the Eliona IoT Platform, which serves as the operational environment for data ingestion, storage, and visualization[Eli25b; Eli25f]. The platform is designed to be deployment-agnostic, operating primarily as a high-scale, Azure-based Cloud environment while preserving on-premise capability for local installations. This flexibility allows the same anomaly detection logic to be applied consistently across multiple tenants and deployment models.

### 4.1.1. Modular System Architecture

The platform is organized into three functional layers to ensure data isolation, scalability, and high throughput. Computational logic and specialized microservices reside within the backend, while the frontend provides a comprehensive interface for visualization and user interaction.

**Device Layer** This layer connects physical assets via standard protocols such as MQTT, HTTP, BACnet, and Modbus. Each device is uniquely authenticated using credentials or tokens to ensure secure and traceable data ingestion.

**Server Layer (Backend)** This layer acts as the centralized processing hub. It manages asset registration, hosts the Rule Engine for automated data processing, and coordinates specialized microservices for distinct use cases[Eli25d; Eli25e]. Time-series data is stored in a single PostgreSQL instance extended with TimescaleDB,

using conventional relational tables for metadata and Hypertables for high-frequency telemetry.

**Application and Frontend Layer** This layer serves as the primary interface for end-users. It provides real-time dashboards, maps, reports, and analytics for monitoring energy health and interacting with the results produced by backend calculations.

### 4.1.2. Asset Modeling and Hierarchical Ontology

A central feature of the platform is its asset model and ontology, which provide a structured representation of entities and their relationships in building data [Eli25c; Eli25a]. Assets are created from reusable templates and organized into multiple hierarchies to reflect both physical layout and functional dependencies.

**Assets and Templates** Assets represent any entity in the system, including sensors, rooms, equipment, or entire buildings. Each asset is instantiated from an Asset Template that predefines attributes such as temperature, occupancy, or power demand, enabling consistent metadata across sites and tenants[Eli25b; Eli25a].

**Dual Hierarchies** Assets are structured into two complementary tree structures. The Local Tree captures physical location (e.g., Site $\rightarrow$ Building $\rightarrow$ Floor), while the Functional Tree represents technical relationships (e.g., Heating System $\rightarrow$ Pump $\rightarrow$ Flow Sensor)[Eli25f]. This dual representation enables both spatial and functional queries over the same telemetry.

**Tagging** Metadata tags are assigned to assets to group and query telemetry points across different buildings and tenants. Tags provide an additional semantic layer on top of the hierarchies, which is utilized by the anomaly detection system to retrieve relevant multivariate signals for model training and scoring.

## 4.2. Financial Impact Quantification

Beyond detection, practical energy anomaly management requires a reliable estimate of the associated financial impact. A common naïve approach computes the deviation between the observed consumption $x_t$ and a single expected value (typically the mean prediction $\mu_t$) and directly converts this difference into excess energy cost. However, as illustrated in Figure 2.3, this approach is fundamentally flawed under multimodal

operating regimes: the mean may lie in a low-density region that is never physically realized, leading to systematically inflated or misleading loss estimates.

### 4.2.1. Distribution-Aware Baseline Selection

When a full predictive distribution is available—specifically a mixture density representation—the expected baseline for financial quantification should be conditioned on the most plausible operating mode given the observation. Let the predictive distribution at time $t$ be given by a mixture model

$$p_t(x) = \sum_{k=1}^{K} \pi_{t,k} \mathcal{N}(x \mid \mu_{t,k}, \sigma_{t,k}^2),$$

where $\pi_{t,k}$, $\mu_{t,k}$, and $\sigma_{t,k}$ denote the mixture weights, means, and variances, respectively.

Instead of using the global mean $\mu_t = \sum_k \pi_{t,k}\mu_{t,k}$, the reference baseline $\tilde{\mu}_t$ is defined as the mean of the mixture component that maximizes the posterior responsibility for the observed value:

$$k^* = \arg\max_k \ \pi_{t,k} \mathcal{N}(x_t \mid \mu_{t,k}, \sigma_{t,k}^2), \qquad \tilde{\mu}_t = \mu_{t,k^*}.$$

This formulation assumes that, even in the presence of an anomaly, the observed value originates from a specific operational regime rather than from an unphysical average across regimes. The instantaneous excess consumption is then conservatively estimated as

$$\Delta x_t = \max(0,\ x_t - \tilde{\mu}_t).$$

### 4.2.2. Fallback Strategy Without Mixture Information

In scenarios where mixture components are not accessible and only unimodal uncertainty estimates are available, a conservative fallback strategy is employed. Instead of the mean prediction, an upper-confidence reference is used:

$$\tilde{\mu}_t = \mu_t + \sigma_t,$$

where $\sigma_t$ denotes the predictive standard deviation. This choice ensures that only deviations exceeding expected stochastic variability contribute to the estimated loss, thereby preventing systematic overestimation.

### 4.2.3. Design Rationale

Both strategies intentionally bias the financial impact estimate toward a lower bound. This is a deliberate design choice: in operational energy management, false inflation of financial losses is more detrimental than moderate underestimation, as it erodes trust in automated analytics and leads to suboptimal prioritization. By conditioning the baseline on the most plausible operating regime—or, alternatively, on a high-confidence envelope—the proposed approach ensures that reported financial impact reflects physically meaningful and economically defensible excess consumption.

## 4.3. Hierarchical Root Cause Analysis and Action Synthesis

Detecting an anomaly at an aggregate meter provides limited operational value unless its origin can be localized within the building system. To enable diagnostic interpretability, the proposed framework performs hierarchical root cause analysis (RCA) by leveraging the asset ontology and geographical hierarchy of the Eliona platform.

### 4.3.1. Ontology-Guided Hierarchical Attribution

Eliona models building assets in a hierarchical tree structure that reflects geographical containment (e.g., Site $\rightarrow$ Building $\rightarrow$ Floor $\rightarrow$ Room) as well as functional decomposition (e.g., Main Meter $\rightarrow$ Sub-meter $\rightarrow$ Device). When an anomaly is detected at an aggregate level, such as a building main meter, all descendant assets in the hierarchy are queried for their corresponding anomaly scores and financial impact estimates.

For each asset $a_i$ in the subtree, an impact measure $\Delta C_t(a_i)$ is computed based on the methodology described in Section 4.2. Root cause attribution is then performed by ranking assets according to their cumulative impact over the anomaly window:

$$\Delta C(a_i) = \sum_{t \in \mathcal{T}} \Delta C_t(a_i),$$

where $\mathcal{T}$ denotes the set of timestamps associated with the detected anomaly. Assets with the highest contribution are identified as the most probable sources of the aggregate deviation.

This hierarchical decomposition allows anomalies to be localized not only to specific meters, but also to concrete physical contexts such as floors, rooms, or functional subsystems.

### 4.3.2. Aggregation by Asset Type

In addition to per-asset attribution, impacts are aggregated by asset type using the semantic labels defined in the ontology (e.g., lighting, HVAC, smart plugs). While individual devices may exhibit only minor deviations, their combined impact can be substantial. Formally, the aggregated impact for an asset type $\tau$ is defined as:

$$\Delta C(\tau) = \sum_{a_i \in \tau} \Delta C(a_i).$$

This aggregation enables the identification of systematic inefficiencies caused by device groups rather than isolated components, such as multiple plug loads in a single room or lighting circuits spanning several zones.

### 4.3.3. Contextual Synthesis and Recommendation Generation

The outputs of the hierarchical and type-based RCA—localized assets, aggregated impacts, temporal patterns, and associated environmental context (e.g., time of day, weekday/weekend, weather conditions)—are consolidated into a structured diagnostic representation. This representation is subsequently provided to a large language model (LLM) configured with domain-specific expert knowledge.

Rather than performing detection or attribution, the LLM operates exclusively at the interpretation layer. Given the structured evidence, it generates human-readable explanations and actionable recommendations. For example, a sustained nighttime anomaly attributed to lighting meters in unoccupied rooms may be interpreted as lights left on after operating hours, accompanied by suggested mitigation actions such as automated shutdown schedules or occupancy-based control. Potential savings are quantified by extrapolating the estimated financial impact per occurrence.

### 4.3.4. Design Rationale

This two-stage approach deliberately separates statistical inference from semantic reasoning. Root cause localization and impact quantification are derived deterministically from measured data and the asset ontology, ensuring traceability and reproducibility. The LLM is used solely to translate these results into operational insights and recommendations, improving usability without compromising analytical rigor or introducing opaque decision-making into the detection pipeline.

## 4.4. Critique of Sequential Forecasting for Anomaly Detection

A dominant paradigm in time-series anomaly detection is the use of sequential forecasting models. In this approach, a model (e.g., RNN, LSTM, or Transformer) is trained to predict the next value $x_t$ based on a sliding window of historical values $(x_{t-w}, \ldots, x_{t-1})$ and potentially exogenous features. An anomaly is flagged if the deviation (residual) between the predicted value $\hat{x}_t$ and the actual value $x_t$ exceeds a threshold. While intuitively appealing, this autoregressive approach suffers from fundamental limitations when applied to sustained anomalies in industrial settings, particularly regarding error propagation and signal adaptation. To demonstrate these failure modes, a controlled synthetic experiment was conducted.

### 4.4.1. Synthetic Experimental Setup

A synthetic dataset was generated to simulate a predictable building energy profile: consumption is set to 10 units between 08:00 and 18:00 on weekdays, and 0 units otherwise. To evaluate detection capabilities, two distinct, sustained anomalies were injected:

1. A “night-shift” anomaly with sustained consumption of 10 units during nighttime hours.
2. A “weekend-work” anomaly with sustained consumption of 5 units over a weekend.

Three distinct forecasting models, plus an additional inference-time variant of the 24-hour model, were tested against this data to highlight different behavioral modes. The anomaly score is calculated as the absolute difference between actual and predicted values.

### 4.4.2. Failure Mode 1: Error Propagation and Instability

The first fundamental issue arises when a sequential model encounters substantial, previously unseen anomalous data. Because the model relies on past observations to generate future predictions, once an anomaly occurs, it enters the model’s input window for the subsequent $w$ steps.

Figure 4.1 illustrates this phenomenon using a model trained with a 24-hour historical window plus time-based features (time of day, `is_weekend`).

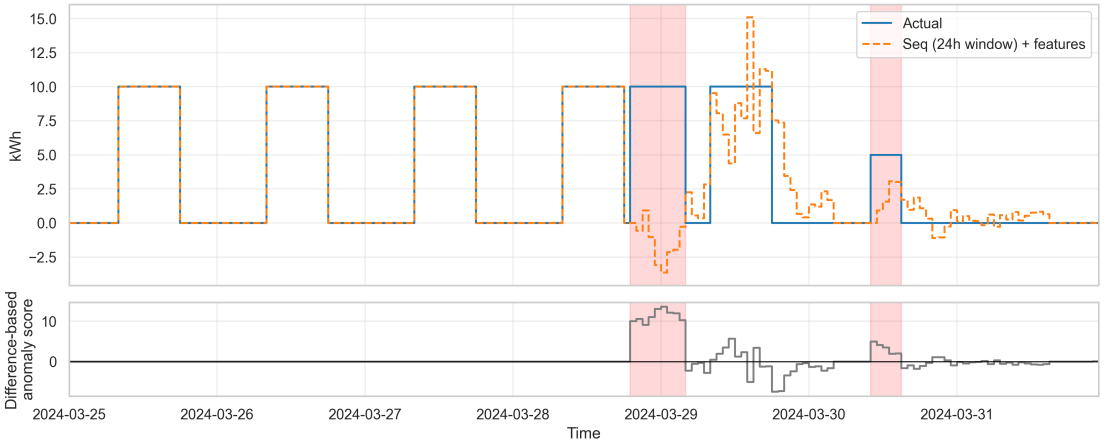



Figure 4.1: Prediction behavior of a model using a 24 h historical window plus time features. The top panel shows actual vs. predicted values; the bottom panel shows the difference-based anomaly score. Note the erratic predictions even after the anomaly ends as the unseen data propagates through the sliding window.

When the sustained nighttime anomaly hits, it represents data completely outside the model's training distribution. The model fails to predict the onset (generating a high anomaly score initially). However, as these anomalous 10-unit values fill the 24-hour input window, the model's internal state becomes corrupted. It begins making erratic predictions, sometimes overestimating, sometimes underestimating, resulting in a noisy anomaly score signal. Crucially, this instability persists even after the actual anomaly has finished, as the "poisoned" window takes 24 hours to clear.

### 4.4.3. Failure Mode 2: Rapid Adaptation and the PA-F1 Illusion

The second failure mode is conversely related to models relying heavily on short-term autocorrelation. In many time series, the best predictor of $x_t$ is simply $x_{t-1}$. If a model learns this dependency strongly, it will rapidly "adapt" to a sustained anomaly.

Figure 4.2 shows a model trained only on the past five historical values, without contextual features.

The model successfully flags the onset of both anomalies due to the sudden jump. However, within five time steps, the input window is filled with the anomalous values. The model quickly learns the new "normal" (e.g., that consumption is currently 10 at night) and predicts accordingly. The residual drops to near zero, and the anomaly is effectively missed for the majority of its duration.

#### Implications for Evaluation Metrics and Financial Impact

This behavior explains the heavy reliance in academic literature on Point Adjustment F1 (PA-F1) scores. In PA-F1, if a model detects a single point within a contiguous

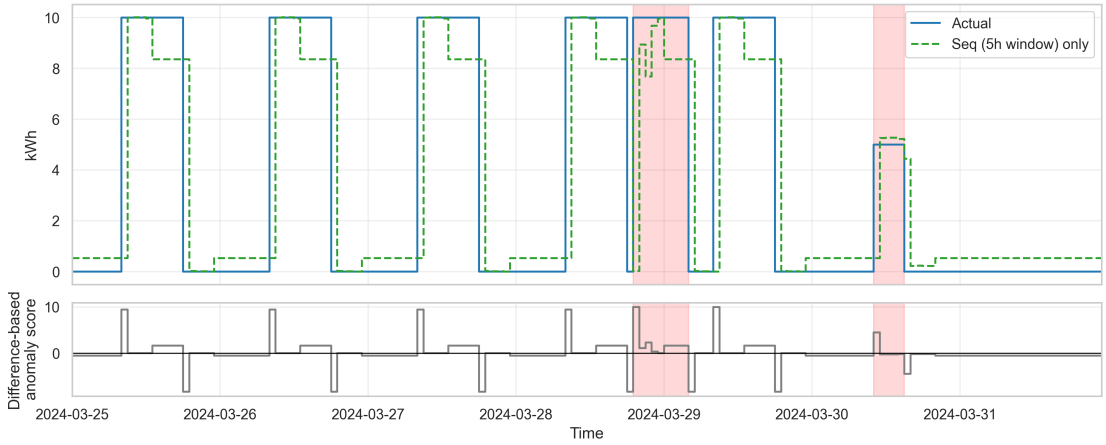



Figure 4.2: Prediction behavior of a model using only a short (5-step) historical window. The model correctly identifies the onset of anomalies but rapidly adapts to the new level, causing the anomaly score to drop back to near zero while the anomaly is still ongoing.

anomaly segment, the entire segment is counted as correctly detected. While this inflates benchmark scores, it masks the model's inability to track sustained deviations.

For industrial applications requiring financial impact quantification, this failure mode is catastrophic. Calculating financial loss requires integrating the deviation over the entire duration of the event. A model that only flags the first 15 minutes of a 4-hour energy spike is useless for quantifying the total wasted energy.

### 4.4.4. Mitigation Strategies

There are two primary architectural strategies to resolve these sequential dependence issues.

#### Strategy A: Contextual Feature-Only Modeling

The most direct solution is to remove the autoregressive dependency entirely. By training a model to predict consumption based solely on contextual features (time, weather, occupancy) and ignoring past consumption values, error propagation is impossible.

Figure 4.3 demonstrates this approach. The prediction remains stable regardless of the actual input, providing a clean, continuous anomaly score throughout the duration of both events. While highly effective for context anomalies, this approach sacrifices the ability to model complex temporal dynamics and cannot leverage powerful sequential foundation models.

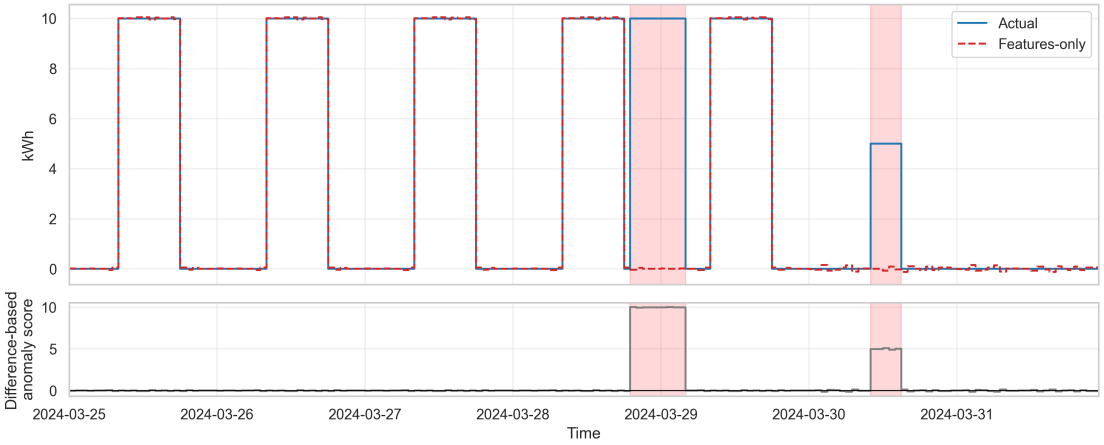



Figure 4.3: Behavior of a features-only model (no historical consumption input). The prediction relies solely on context (time/weekend), resulting in a stable baseline and accurate detection of sustained anomalies without adaptation.

### Strategy B: Inference-Time Input Imputation

To retain the benefits of sequential modeling while mitigating error propagation, an inference-time correction mechanism can be introduced. If the anomaly score at step $t$ exceeds a defined threshold, the actual value $x_t$ is considered contaminated. Instead of feeding $x_t$ into the sliding window for step $t+1$, the model's own prediction $\hat{x}_t$ is imputed as a "corrected" value. In an online or periodically retrained setting, this also prevents the model from adapting its baseline to these anomalous segments, so similar future events are not reinterpreted as normal behaviour despite the non-stationarity of the raw building signal.

Figure 4.4 applies this logic to the unstable 24-hour window model from Figure 4.1. By replacing anomalous inputs with predictions, the sliding window remains clean, preventing the model from adapting to the anomaly or becoming unstable. This allows for accurate tracking of sustained anomalies while still using sequential architectures.

## 4.5. Statistical Limitations of Point and Gaussian Predictions

To isolate the effect of distributional assumptions on anomaly detection, a synthetic "Variable Shift" dataset was created. Each hourly sample toggles between a low-power regime ($0$–$1$ kWh) and a high-power regime ($9$–$11$ kWh) with a stochastic morning/evening schedule (approximately 60/40 split). A stuck-at fault of $5$ kWh was injected during a regular weekday to emulate a latent control failure. Figure 4.5 shows that all three model families—deterministic dense regression, single-Gaussian prediction, and Mixture Density Networks (MDN)—deliver visually similar means, yet their anomaly

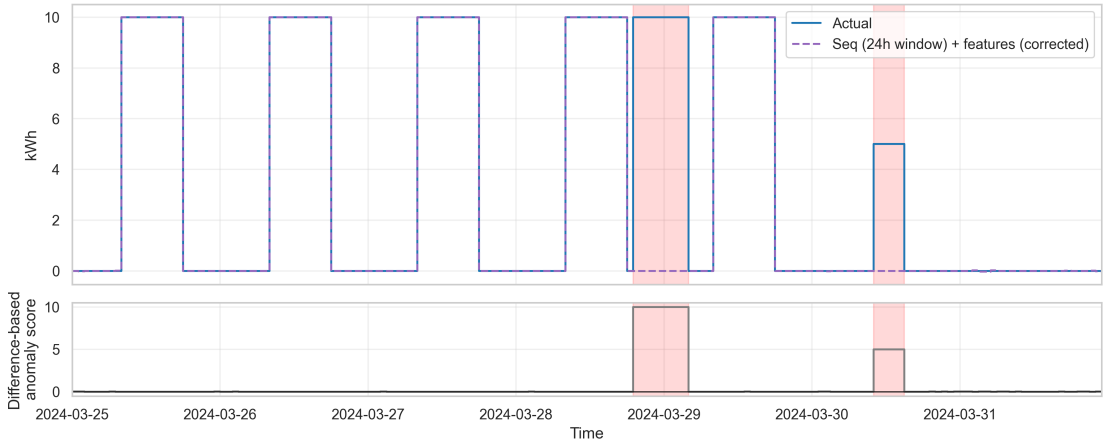



Figure 4.4: The same 24-hour window model from Figure 4.1, but applied with inference-time imputation. When an anomaly is detected, the predicted value replaces the actual value in the sliding window for future steps. This prevents error propagation and maintains a high anomaly score throughout the event.

scoring behavior diverges drastically.

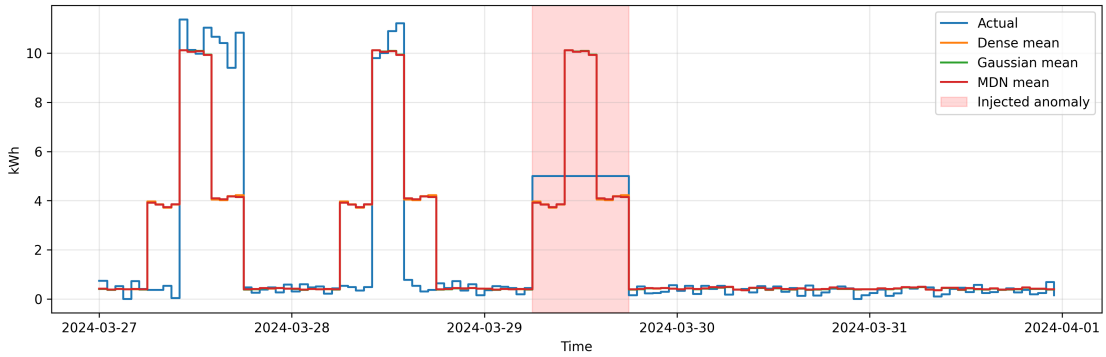



Figure 4.5: Predicted means over the Variable Shift horizon. Dense, Gaussian, and MDN models track the two regimes, masking the scoring deficiencies discussed in Sections 4.5.1–4.5.3.

### 4.5.1. The Failure of Mean Squared Error Minimization

Dense regressors trained with Mean Squared Error (MSE) converge toward the global average of both regimes. In bimodal settings this leads to systematic bias: the model predicts approximately $5\,\text{kWh}$ regardless of whether the system is in its “Off” (low) or “On” (high) state. Consequently, perfectly normal behavior is scored as highly anomalous, whereas the injected stuck-at-$5$ event appears deceptively healthy because it matches the biased mean. The residual trace in Figure 4.6 exposes this contradiction: the absolute error balloons whenever the device operates normally, yet it contracts when the genuine anomaly occurs.

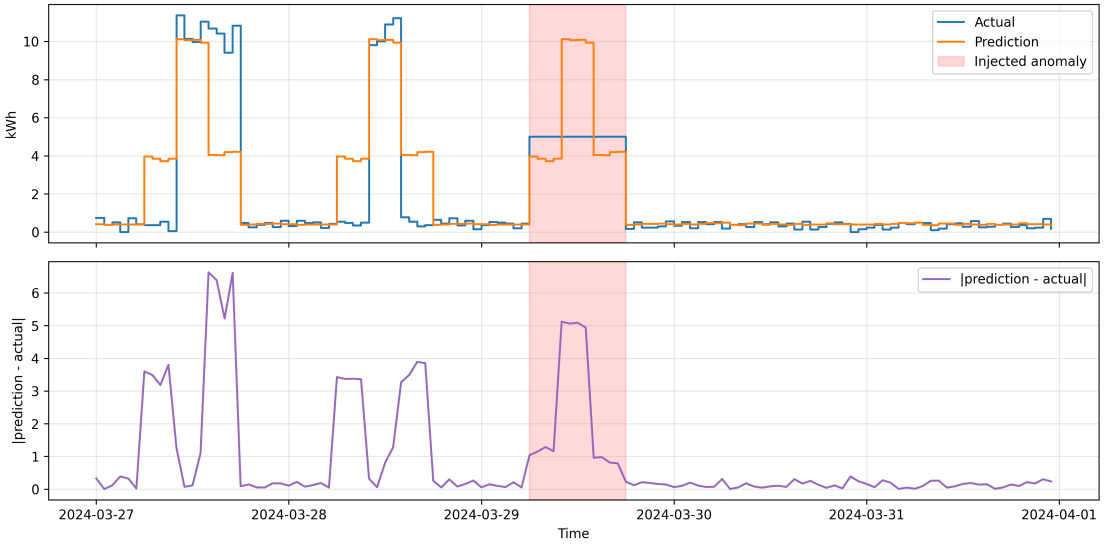



Figure 4.6: Dense regressor residuals over the Variable Shift dataset. The mid-range prediction inflates anomaly scores for legitimate operating states, while the stuck-at-$5$ fault yields a small residual.

### 4.5.2. The Gaussian Distribution Paradox

A single-component Gaussian attempts to reconcile bimodality by inflating its variance. The resulting Probability Density Function (PDF) concentrates probability mass near the center—a region never visited by real data. The normalized log-likelihood trace (Figure 4.7) confirms that the stuck-at-$5$ anomaly sits inside the "most likely" area of the Gaussian, generating a low penalty. Meanwhile, legitimate regime values land in lower-density shoulders and spuriously raise the score. The heatmap in Figure 4.8 makes the distortion visible: the green, high-probability band spans the median instead of the true modes.

### 4.5.3. Solution: Mixture Density Networks

Mixture Density Networks address both issues by learning multiple kernels simultaneously. Each component can specialize in a particular operating mode, while the regions between components retain near-zero probability. Figure 4.10 shows how the MDN assigns green (high probability) bands only where data is observed, keeping the mid-range red. When log-likelihood is used as the anomaly score, the stuck-at-$5$ fault immediately falls into the valley between components, producing a sharp increase in $|\log p(x)|$ (Figure 4.9). This probabilistic separation allows the MDN to quantify financial impact reliably: integrating the residual energy over time now reflects the true magnitude of the fault rather than artifacts of model bias.

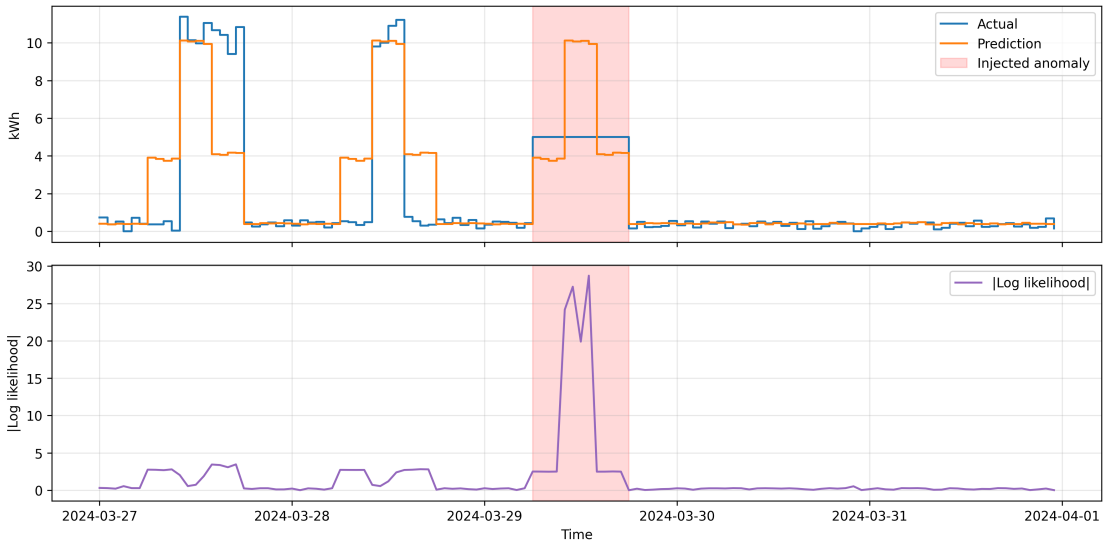



Figure 4.7: Absolute log-likelihood trace for the single-Gaussian predictor. The stuck-at-$5$ anomaly aligns with the high-likelihood center, suppressing the score.

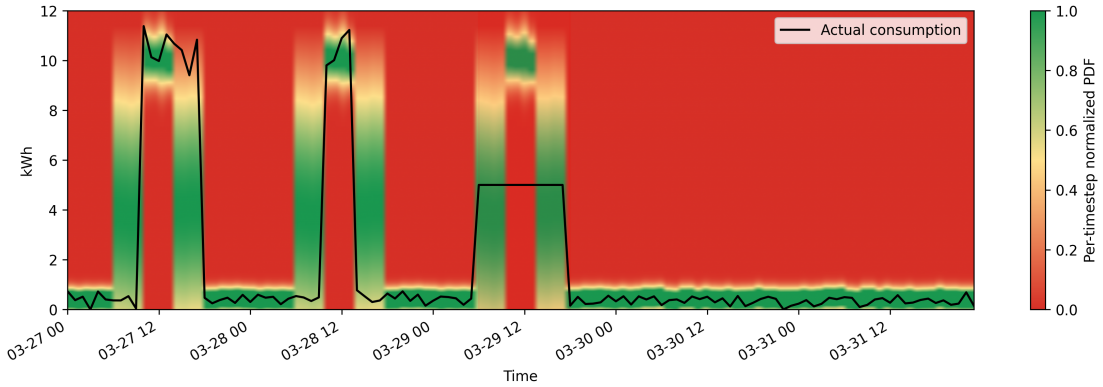



Figure 4.8: Per-timestep normalized PDF for the Gaussian model. High probability mass accumulates between the actual clusters, illustrating the variance-stretching paradox.

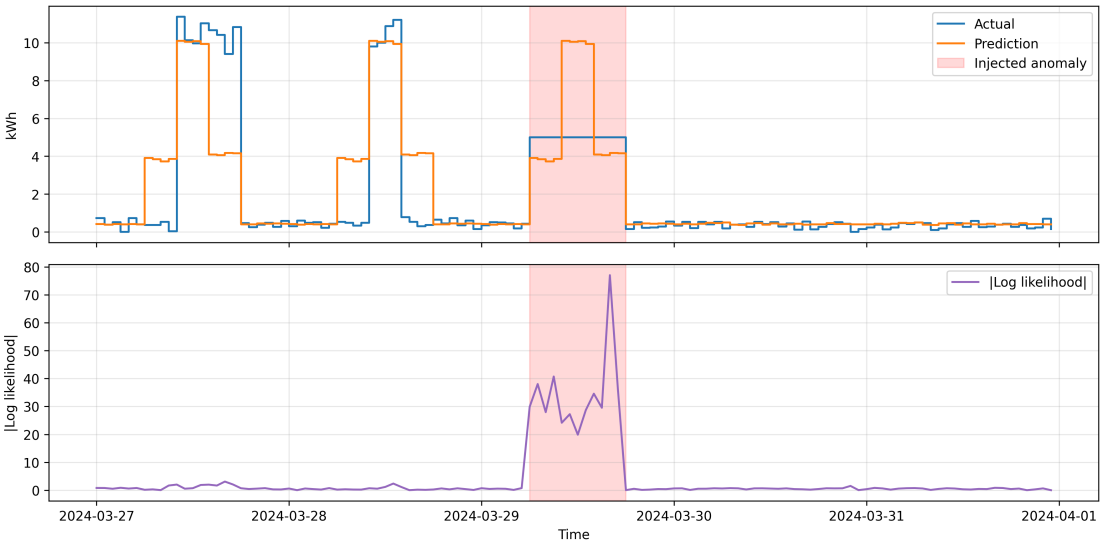



Figure 4.9: MDN absolute log-likelihood trace. The stuck-at-5 anomaly triggers a sustained spike because the value resides in a low-probability region between mixture components.

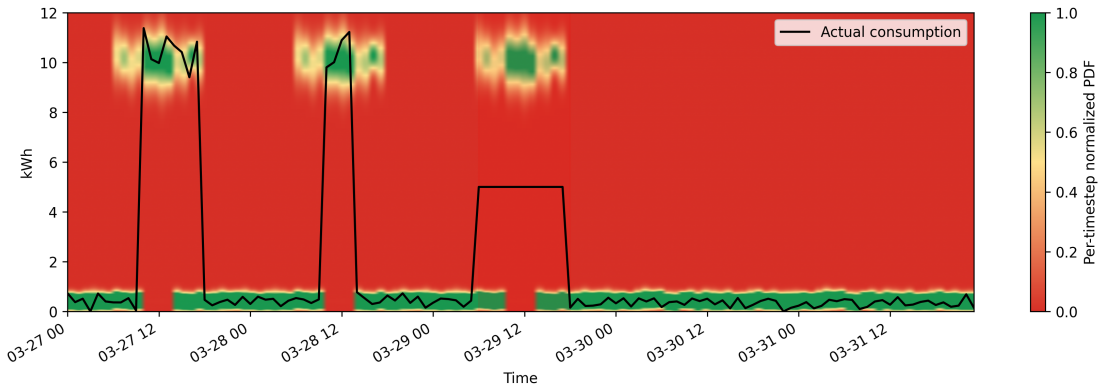



Figure 4.10: MDN normalized PDF heatmap. Two distinct high-probability ridges align with the real operating modes, while the middle band remains improbable.

## 4.6. Distribution-Aware Anomaly Scoring for Mixture Density Models

Building on Section 4.5, which details the Variable Shift dataset and its bimodal operating regimes, we now analyze how distribution-aware anomaly scores behave when the predictive density itself is multimodal. The deterministic residual failures from Section 4.4 are amplified in this setting because no single "expected value" exists, making deviation from the mean a misleading proxy for abnormality.

Figure 4.11 extends the MDN perspective by overlaying several candidate scores derived from the same mixture distribution: mean residuals, PIT, negative log-likelihood, and the proposed Density–Quantile (DQ) family.

### 4.6.1. Mean Residual: Failure Under Multimodality

The most common anomaly score is the absolute residual between the observation $x_t$ and the predicted mean $\mu_t$:

$$s_t^{\text{mean}} = |x_t - \mu_t|. \tag{4.1}$$

In multimodal settings, the mean of the predictive distribution often lies in a region of *low probability mass*. As shown in Figure 4.11, the MDN mean converges to an intermediate value between the two legitimate modes. Consequently, observations that are perfectly normal but belong to either mode exhibit large residuals and are falsely flagged as anomalous. Conversely, the injected anomaly—located near the mean but inside a low-density valley—produces a small residual and is incorrectly classified as normal.

This demonstrates that residual magnitude is not a valid proxy for abnormality when the expected behaviour cannot be represented by a single point estimate.

### 4.6.2. Probability Integral Transform (PIT)

A distribution-aware alternative is the Probability Integral Transform (PIT), which maps each observation to its cumulative probability under the predicted distribution:

$$\text{PIT}_t = F_t(x_t) = \int_{-\infty}^{x_t} p_t(y)\,\mathrm{d}y, \tag{4.2}$$


Figure 4.11: Comparison of anomaly scoring methods derived from a Mixture Density Network under a bimodal operating regime with an injected intermediate anomaly. The top panel shows the predicted probability density together with the actual observation and the MDN mean. Subsequent panels compare mean residuals, PIT-based scores, negative log-likelihood, and the proposed Density–Quantile (DQ) scores and severities.

where $p_t(y)$ denotes the MDN predictive density at time $t$. A symmetric anomaly score can be defined as:

$$s_t^{\text{PIT}} = 1 - \text{PIT}_t. \tag{4.3}$$

PIT correctly identifies observations in the extreme tails of the distribution. However, it remains insensitive to *low-density regions between modes*. In Figure 4.11, the injected anomaly lies near the median of the distribution and therefore yields a moderate PIT value, despite being highly unlikely. PIT thus fails to detect anomalies that occupy density valleys rather than tails.

### 4.6.3. Negative Log-Likelihood and Its Limitations

Another principled score is the negative log-likelihood (NLL):

$$s_t^{\text{NLL}} = -\log p_t(x_t). \tag{4.4}$$

NLL correctly assigns high anomaly scores to observations in low-density regions, including the valley between modes. As shown in Figure 4.11, it robustly detects the injected anomaly.

However, NLL values are *not comparable across time*. Each timestamp $t$ corresponds to a different predictive distribution with different entropy, variance, and scale. As a result, absolute NLL magnitudes cannot be meaningfully thresholded or aggregated over time, limiting their use for persistence analysis, severity ranking, and financial quantification.

### 4.6.4. Density–Quantile (DQ) Probability

To obtain a score that is both distribution-aware and comparable across time, this work introduces the Density–Quantile (DQ) probability. Instead of evaluating the density at a single point, DQ measures the proportion of probability mass that is *less likely* than the observed value:

$$\text{DQ}_t = \int_{\{y : p_t(y) \le p_t(x_t)\}} p_t(y)\,\mathrm{d}y. \tag{4.5}$$

By construction, $\text{DQ}_t \in (0, 1]$ and is invariant to the shape, scale, and entropy of the underlying distribution. Observations in high-density regions yield large DQ values, while points located in tails or low-density valleys yield small values.

An anomaly score can therefore be defined as:

$$s_t^{\text{DQ}} = 1 - \text{DQ}_t. \tag{4.6}$$

As shown in Figure 4.11, this score simultaneously suppresses false positives for legitimate operating modes and sharply highlights the injected anomaly located between the modes.

### 4.6.5. Density–Quantile Severity Scaling

While $1 - \mathrm{DQ}_t$ provides a normalized anomaly score, it does not reflect the *relative improbability* of extreme events. For example, the difference between $\mathrm{DQ} = 0.99$ and $\mathrm{DQ} = 0.98$ corresponds to a doubling of unlikeliness, yet both values are close on a linear scale.

To address this, a severity transformation is introduced:

$$\mathrm{Severity}_t = \min\left(1,\ \frac{p_{\min}}{\mathrm{DQ}_t}\right), \tag{4.7}$$

where $p_{\min}$ defines the minimum reference quantile that maps to maximum severity.

This transformation preserves the ordering induced by DQ while amplifying differences in the extreme low-probability regime. By selecting $p_{\min}$, the sensitivity of the detector can be explicitly controlled, as illustrated in Figure 4.11 for $p_{\min} = 10^{-2}$ and $p_{\min} = 10^{-4}$.

### 4.6.6. Summary

Density–Quantile scoring combines the strengths of likelihood-based detection with the comparability of quantile methods. Unlike residuals, it respects multimodality; unlike PIT, it captures density valleys; and unlike NLL, it produces normalized, time-comparable scores suitable for persistence tracking, severity ranking, and downstream financial impact estimation. For these reasons, DQ-based scoring forms the core anomaly quantification mechanism in this work.

## 4.7. Benchmark Data Generation and Composition

To compare the proposed methodology against established baselines, a comprehensive benchmark dataset was synthesized with the Building Optimization Performance Test Framework (BOPTEST)[**BOPTEST**]. BOPTEST provides high-fidelity building simulations that include weather, occupancy, and HVAC subsystems, offering a controlled

sandbox for evaluating diagnostic strategies without legacy noise.

### 4.7.1. Simulation Environment and Baseline Construction

The “Multizone Office Complex Air” test case was selected as it emulates a large office with coupled thermal zones, AHUs, and realistic schedules.

**Data fidelity** The simulator injects geographically consistent weather files, dynamic occupancy, and holiday calendars, ensuring the external drivers mirror commercial buildings.

**Integrity** Because all sensors are generated virtually, the baseline year is free of telemetry dropouts, frozen devices, or undocumented setpoint overrides, giving a clean reference profile for “healthy” operation.

**Temporal scope** A continuous year of data (15-minute granularity) was exported to capture seasonal shifts and enable out-of-season evaluation.

### 4.7.2. Feature Selection and Control Layer Logic

The raw export was decomposed into contextual features and consumption targets while deliberately excluding control-loop variables that could leak fault signatures.

**Leakage mitigation** Control-layer signals (e.g., valve positions, supply setpoints) were omitted. If a stuck valve drives excess consumption and its command signal is provided to the model, the anomaly becomes “explainable” by the feature set and disappears in the residual.

**Included features** Exogenous drivers such as outdoor temperature, global horizontal irradiance, wind speed, occupancy counts, calendar flags (weekday/weekend, holidays), and cyclical encodings (hour-of-day, day-of-year).

**Target variables** Seventeen metered electricity channels including the aggregate main meter, chiller and boiler production meters, AHU fan feeds, pump circuits, and lighting zones. This enables both whole-building financial attribution and subsystem diagnostics.

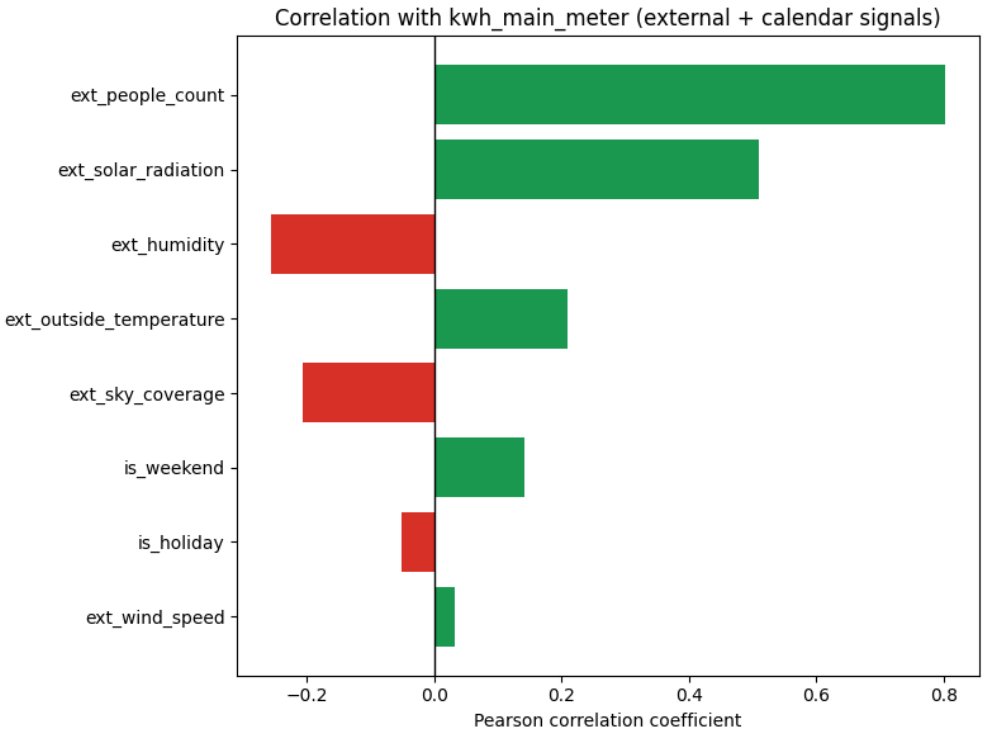



Figure 4.12: Empirical Pearson correlation of contextual and external drivers with the main electricity meter over the synthetic 2007 benchmark year.

## 4.8. Experimental Data Segmentation and Anomaly Injection

To probe generalization and sample efficiency, the baseline year was sliced into multiple training/testing regimes and augmented with a bounded taxonomy of synthetic faults. Anomalous points never exceed 5% of any derived dataset to preserve realism on the aggregated main meter.

### 4.8.1. Segmentation Strategy

- **Long-term:** Two non-overlapping six-month windows to analyze seasonal translation (e.g., winter learning, summer detection).
- **Medium-term:** Four quarterly (three-month) windows to test model freshness requirements.
- **Short-term:** Four two-week slices representing spring, summer, fall, and winter for low-data scenarios.

### 4.8.2. Anomaly Taxonomy and Labeling

Custom perturbations were injected per segment, spanning operational, control, and contextual deviations. Table 4.1 summarizes the categories applied to the main meter and selected subsystems.

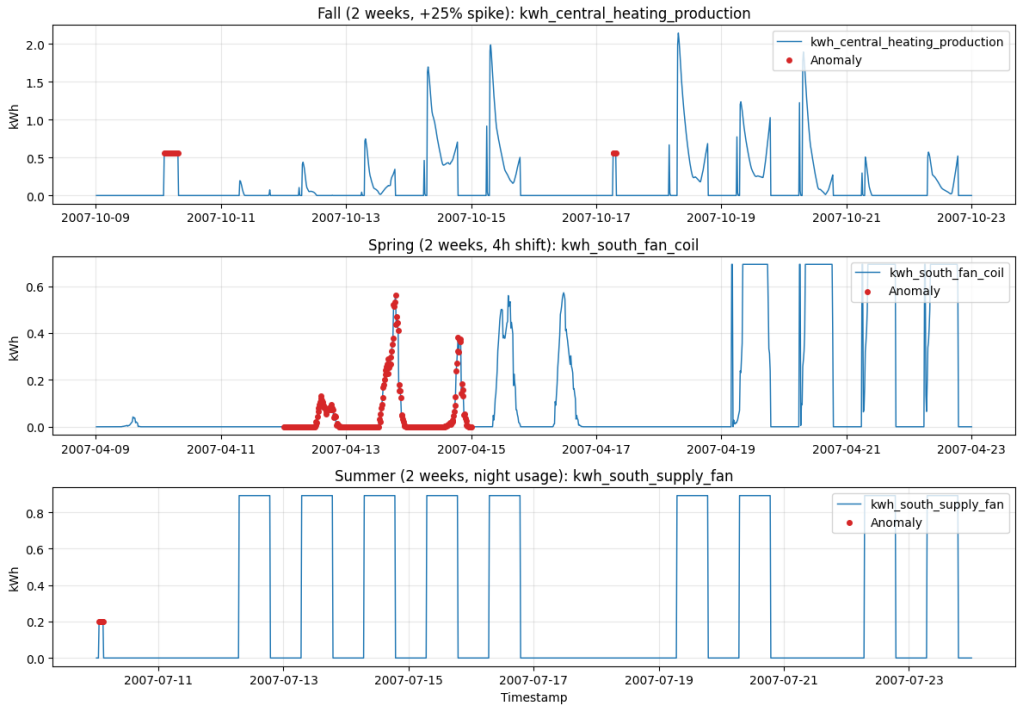



Figure 4.13: Representative sub-meter excerpts from the curated benchmark slices (fall spike, spring pattern shift, summer off-hours). Red markers indicate anomaly windows where the targeted device deviates from its baseline regime.

Each injected anomaly is labeled with its category, onset, and magnitude, enabling fine-grained evaluation of detection latency, duration coverage, and financial attribution across the benchmark suites described above.

## 4.9. Evaluation Constraints and Benchmark Limitations

While the BOPTEST-based benchmark enables controlled and reproducible evaluation, several practical constraints affect the interpretation of the resulting performance metrics. These limitations do not invalidate the benchmark, but they do influence the comparability and robustness of certain method classes.

Table 4.1: Synthetic anomaly taxonomy used in the BOPTEST benchmark datasets.

| extbfCategory | **Description** |
|---|---|
| Device failures | Full outage (0 kWh) and degraded operation ($50\%$ load) for selected meters. |
| Stuck states | Forced high-load and low-load plateaus irrespective of schedule. |
| Drift profiles | Exponential and linear drifts (10%–60%) applied over multi-day horizons. |
| Extended spikes | Sustained spikes at 10%, 25%, 50%, and 100% above baseline for varying durations. |
| Contextual/off-hours | Night or weekend consumption persisting at abnormal levels (high/low variants). |
| Pattern shifts | Phase shifts of daily load shapes by 30 minutes up to 4 hours. |
| Point anomalies | Single-timestep spikes from 10% to 500% of nominal demand. |

### 4.9.1. Hyperparameter Sensitivity and Training Stability

Several evaluated methods require extensive hyperparameter tuning on a per-dataset or per-meter basis. In particular, mixture-density-based models exhibit high sensitivity to initialization, learning rates, and regularization settings. On some meters, training converged reliably, while on others the optimization process diverged due to exploding losses.

As a direct consequence of this instability, benchmark coverage differs substantially between methods: while some models successfully completed nearly the full benchmark suite, others produced valid results for only a limited subset of meters. Aggregate performance metrics for such methods are therefore computed over a reduced and potentially favorable sample and must be interpreted cautiously. This asymmetry introduces a selection bias in aggregate results, as stable methods with low tuning requirements are naturally overrepresented relative to more expressive but fragile architectures.

Consequently, reported benchmark scores reflect not only detection capability, but also training robustness under limited tuning budgets.

### 4.9.2. Comparability Between Trainable Models and Foundation Models

A further limitation concerns the comparability between trainable models and time-series foundation models. Trainable baselines were fitted using fixed-length training windows (e.g., two-week seasonal slices) and evaluated on the same temporal context augmented with injected anomalies.

In contrast, foundation models require historical context preceding the evaluation window. Consequently, they were provided with additional pre-anomaly data that was not available to trainable models, leading to unequal informational priors at inference time. This discrepancy is particularly pronounced in short-window evaluations and seasonal transfer experiments.

The only configuration in which both model classes operate under closely comparable conditions is the year-long training setup followed by winter evaluation, where both approaches have access to nearly the same historical context. Even in this case, foundation models remain constrained by maximum context lengths (e.g., three months for Chronos), preventing full utilization of the available annual history.

### 4.9.3. Implications for Result Interpretation

These constraints imply that benchmark metrics should be interpreted as indicators of practical deployability under realistic engineering constraints rather than as absolute measures of algorithmic superiority. In particular, strong performance by stable models reflects robustness and ease of deployment, while underperformance by more expressive architectures may be attributable to tuning sensitivity rather than fundamental modeling limitations.

All benchmark results presented in the following section are therefore discussed in light of these constraints, with emphasis on qualitative behavior, persistence tracking, and financial interpretability rather than raw aggregate scores alone.

## 4.10. Comparative Model Performance and Structural Evaluation

The following section details the quantitative findings of the benchmark evaluation. It analyzes the detection performance of the implemented models across the generated datasets and validates the methodological hypotheses regarding feature selection and stochastic modeling.

The first evaluation phase comprised a maximum of 16,979 individual runs per model to ensure statistical significance across the diverse benchmark slices. A critical finding involves the performance of the standard CNN (Convolutional Neural Network) implementation, which was adapted from the TSB-AD (Time-Series Benchmark for Anomaly Detection) library. Although this architecture demonstrated the highest accuracy in the multivariate category of the original TSB-AD benchmark, it yielded suboptimal results when applied to the building-energy datasets.

The observed performance degradation in the standard CNN supports the hypothesis that the inclusion of historical consumption values in the input window negatively affects detection stability in non-stationary environments. To address this, a CNN Feature-Only model—which utilizes solely exogenous contextual drivers—was implemented. This architecture achieved significantly higher detection scores, confirming that contextual features provide a more reliable normative baseline than autoregressive dependencies for building energy telemetry.

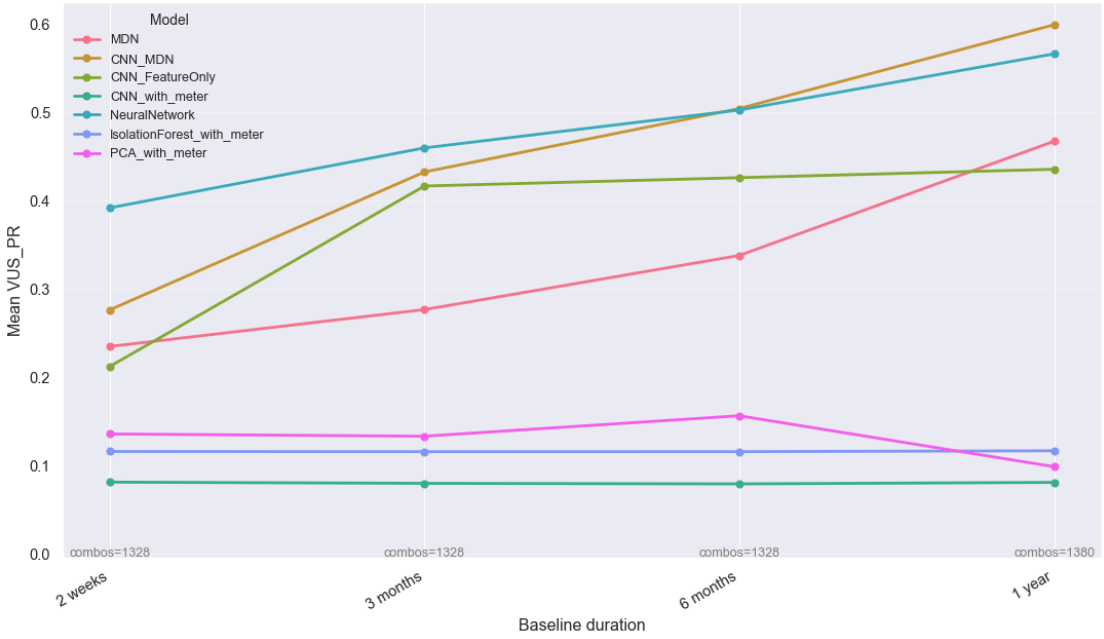



Figure 4.14: Benchmark comparison of neural baselines (dense, CNN, CNN Feature-Only) and mixture-density-based variants across the evaluated meters and dataset slices.

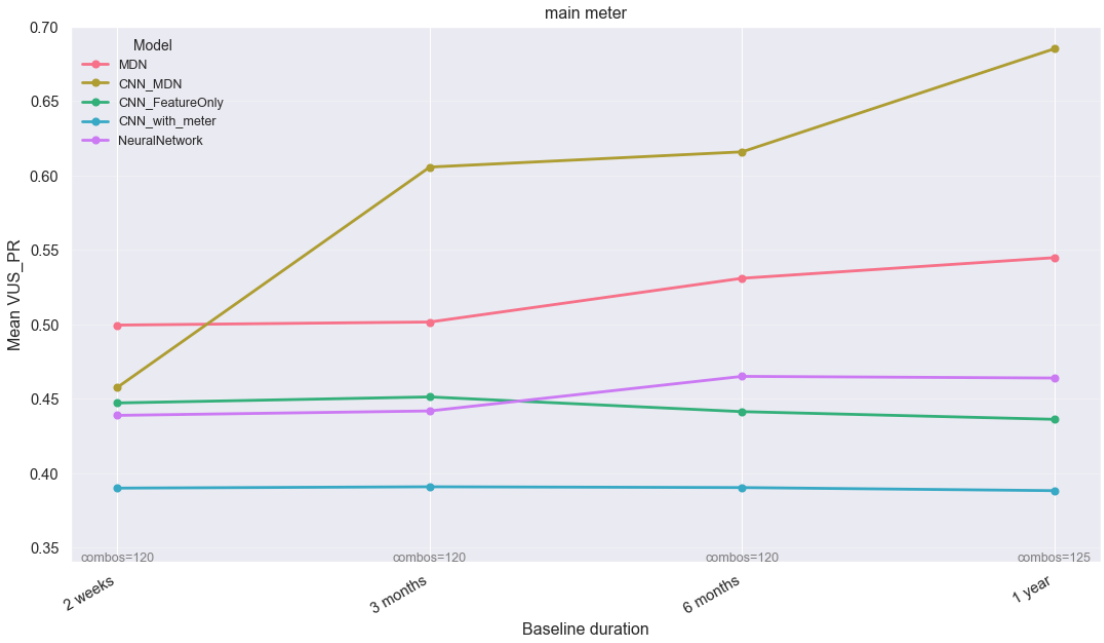



Figure 4.15: Benchmark comparison on the building main meter, highlighting the effect of feature-only modeling and stochastic output layers under long-horizon baselines.

### 4.10.1. Analysis of Stochastic and Hybrid Architectures

The evaluation further examined the efficacy of MDN (Mixture Density Networks) and hybrid models. The MDN model alone demonstrated lower aggregate performance than the CNN Feature-Only variant. However, the CNN-MDN—a hybrid architecture utilizing a convolutional feature extractor with an MDN output layer—achieved the highest VUS-PR (Volume Under the Surface - Precision-Recall) scores on the one-year dataset. For shorter training intervals, a standard Neural Dense Network with Residual Connections (bypass paths to improve gradient flow) proved most effective.

A consistent trend was observed where detection accuracy increased proportionally with the volume of available baseline data. Models trained on the one-year slices exhibited superior generalization compared to those limited to shorter temporal windows.

### 4.10.2. Training Stability and Baseline Comparisons

The discrepancy between the performance of the CNN-MDN and the standard neural network was attributed to the training stability of the probability layers. It was determined that the MDN training process is sensitive to initialization and requires precise Hyperparameter Tuning—the optimization of a model's internal configuration settings—to achieve convergence. While the results on the building's main meter indicate that the CNN-MDN is the most effective architecture, these findings require critical assessment due to the observed stochastic instability on specific sub-meters.

Finally, the evaluation included classical statistical methods, specifically PCA (Principal Component Analysis) and Isolation Forest, also sourced from the TSB-AD library. These methods yielded the lowest performance scores in the entire benchmark. This confirms that linear and tree-based statistical approaches are incapable of resolving the complex, non-linear dependencies inherent in multivariate building energy telemetry.

Due to its size, the consolidated training history visualization across all meters is provided in Appendix A.1 (Figure A.1).

### 4.10.3. Per-category performance on season-matched 3-month baselines

To reduce confounding effects from seasonal regime shifts, the following analysis restricts evaluation to the *3-month, season-matched* baseline configuration. This setting is the closest approximation for comparing trainable models and time-series foundation models under a shared seasonal context, although a residual asymmetry remains: foundation models infer anomalies from a two-week window conditioned on up to three

months of preceding context, whereas trainable models are fitted on the full three-month period (including the two-week slice, where the anomalous period is treated as nominal during training). In addition, the plotted means are computed over the intersection of runs for which *all* compared methods produced valid outputs; therefore, failures or crashes can shift aggregate scores through implicit subsampling.

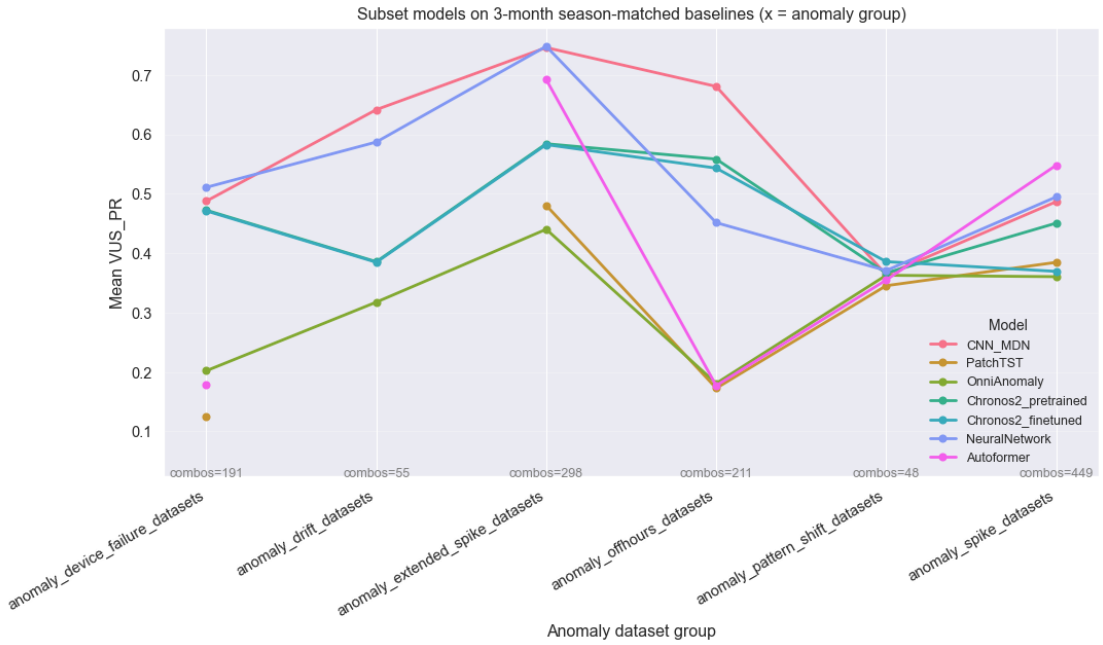



Figure 4.16: Comparison of Chronos and `CNN_MDN` performance under the season-matched three-month baseline configuration.

Across anomaly groups, the `CNN_MDN` achieves the strongest mean VUS-PR on most categories, with the deterministic neural network baseline remaining competitive on device-failure, drift, and extended-spike scenarios. Chronos-2 generally trails the strongest trainable models in this configuration, and fine-tuning yields a noticeable improvement primarily for the off-hours anomaly group, where it performs comparatively well. PatchTST and OmniAnomaly underperform on most categories, while Autoformer exhibits strong performance on spike-like anomalies and remains competitive on extended spikes.

### 4.10.4. Seasonal translation sensitivity

The seasonal translation experiment evaluates how detection performance changes when the anomaly window is evaluated against baselines from the same season, one season apart, and two seasons apart. As expected, mean VUS-PR degrades as the seasonal distance increases, indicating that out-of-season baselines reduce the fidelity

of the learned normative band. The largest performance drops are observed for CNN_MDN and PatchTST, suggesting limited robustness under season shifts. In contrast, Autoformer remains comparatively stable across seasonal distances, showing weaker sensitivity to season mismatch in this benchmark. Overall, these results align with the broader observation that comparatively simple or lightweight architectures can outperform large foundation-model approaches for anomaly detection under constrained and domain-shifted conditions.

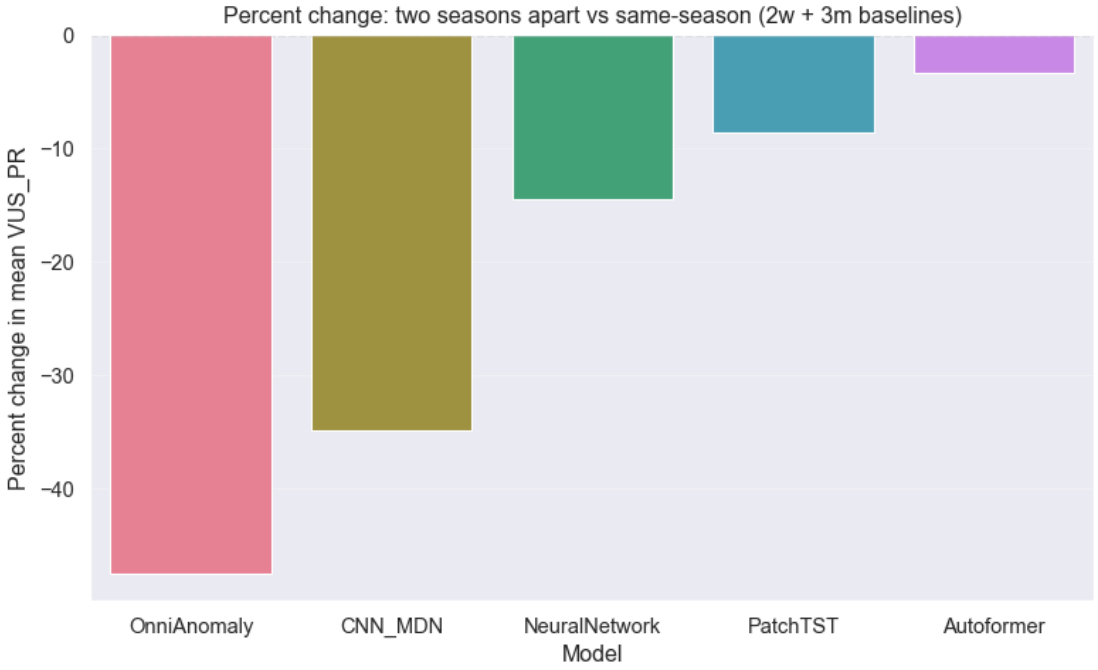



Figure 4.17: Seasonal translation sensitivity of mean VUS-PR as a function of seasonal distance between training baseline and evaluation window.

### 4.10.5. Model selection rationale

Although CNN_MDN attains the highest mean VUS-PR in most benchmark configurations, Chronos-2 is selected as the primary deployment model due to its fundamentally superior operational characteristics.

All non-foundation approaches evaluated in this study require explicit model fitting per meter (or per homogeneous meter cluster). In large-scale building portfolios this implies maintaining and retraining millions of individual models, introducing substantial engineering overhead, delayed adaptation, and fragile retraining pipelines. In addition, these methods require scheduled retraining to track distributional drift, typically on sliding windows (e.g., bi-weekly or monthly), which further amplifies system complexity.

Chronos-2 operates in a zero-shot regime and adapts online by conditioning its inference on the most recent historical context. This removes the need for explicit retraining, per-meter model management, and drift-triggered pipeline orchestration. As a result, structural changes—such as HVAC retrofits, occupancy regime shifts, or equipment replacements—are absorbed immediately into the conditioning context without any manual intervention, while trainable models would require explicit detection of the regime change, selective retraining on post-change data, and a warm-up phase with reduced reliability.

From a systems perspective, Chronos-2 therefore provides a scalable, maintenance-minimal and drift-robust solution that remains competitive in detection accuracy and operationally superior in large-portfolio deployments. This makes Chronos-2 the preferred model for real-world building-scale anomaly detection despite `CNN_MDN` achieving marginally higher benchmark scores.

Chronos-2 does not explicitly model a multimodal predictive density, in contrast to mixture-based approaches such as `CNN_MDN`. However, it produces direct probabilistic quantile bounds that are not constrained to any parametric distributional form. This enables anomaly scoring via distribution-free probability integral transform (PIT)–based formulations using predicted quantile envelopes. Consequently, Chronos-2 preserves calibrated uncertainty estimates under non-Gaussian and heavy-tailed regimes while avoiding the instability and overfitting tendencies commonly observed in explicit density mixture models.

# 5

# Implementation

The previous chapter established the methodological framework and the operational requirements for the anomaly detection system. This chapter details the technical realization of these concepts, focusing on the software stack, the integration into the Azure ecosystem, and the internal orchestration logic of the Scala microservice and the Python analytics endpoint.

## 5.1. Integrated System Architecture and Technology Stack

The implementation utilizes a polyglot approach to leverage the specific strengths of different programming paradigms. While the orchestration and data processing are handled by a Scala microservice to ensure high-performance parallelism, the predictive modeling is realized through a Python endpoint optimized for machine learning.

### 5.1.1. Deployment and Cluster Integration

The system is designed as a modular Docker container that operates within the same Azure Kubernetes Services (AKS) cluster as the core Eliona microservices. This co-location ensures low-latency communication between the detection logic and the primary data storage. The architecture remains environment-agnostic; for on-premise deployments, the entire stack, including the analytics endpoint, can be executed locally as a suite of interconnected containers.

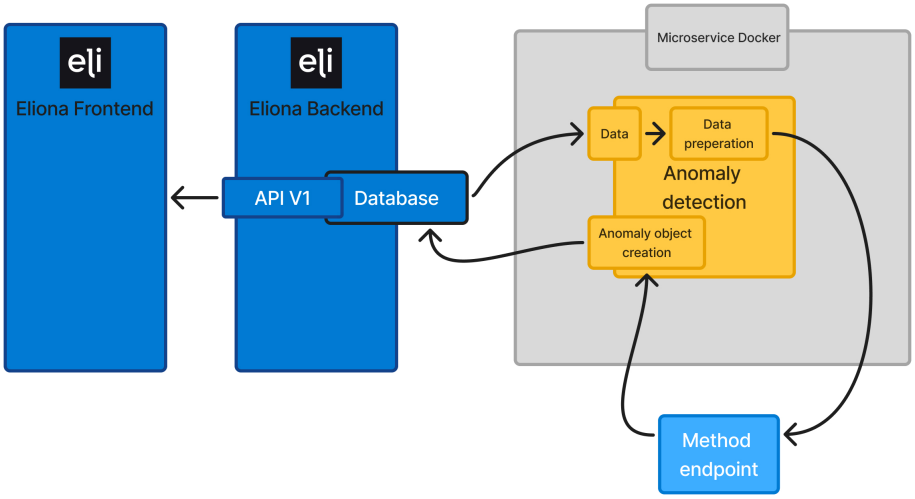



Figure 5.1: High-level deployment of the anomaly detection microservice and Python analytics endpoint within the Eliona and Azure ecosystems.

### 5.1.2. Data Orchestration and Persistence

The microservice establishes a closed-loop data pipeline with the Eliona backend:

**Ingestion** Telemetry data is retrieved directly from the centralized PostgreSQL/TimescaleDB instance.

**Analytics loop** After performing data preparation, the microservice transmits multivariate tensors to the method endpoint.

**Synthesis and storage** If the returned predictions indicate a deviation, the service creates an anomaly object. These objects are persisted in the database, where they become accessible to the Eliona frontend via API V1 for visualization and reporting.

## 5.2. Python Analytics Endpoint: Chronos-2 Integration

The analytical core of the system is realized as a specialized Python service. This choice is motivated by the maturity of the Python ecosystem regarding transformer-

based time-series models and the availability of managed deployment infrastructures.

### 5.2.1. Predictive Logic and Model Hosting

The endpoint hosts the Chronos-2 model, which is utilized in a batch prediction mode to process multiple time series simultaneously. It receives JSON payloads containing historical context and exogenous covariates, returning probabilistic forecasts and specific quantiles that are used to establish the normative operational band.

### 5.2.2. Managed Online Endpoints in Azure ML

For cloud-based production environments, the service is deployed as an Azure Machine Learning (AML) managed online endpoint. This infrastructure provides several quantifiable advantages:

**Automated scaling** The platform manages horizontal scaling and load balancing to handle variable request volumes from the Scala backend.

**MLOps integration** Azure ML provides a robust environment for creating training pipelines and registering versioned models, ensuring the non-stationary adaptation requirement is fulfilled.

**Flexibility** Despite being managed in the cloud, the service is fully containerized, allowing for a seamless transition to local Docker environments when on-premise capability is required.

## 5.3. Scala Microservice: Multi-Tenant Orchestration

The orchestration of the anomaly detection system is managed by an autonomous Scala microservice, which serves as the primary computational engine for data retrieval, transformation, and multi-tenant isolation. While the predictive logic is delegated to a Python-based endpoint, the Scala service functions as the “brain” of the architecture, coordinating complex asynchronous I/O operations and ensuring thread-safe execution across concurrent processing loops. The selection of Scala is motivated by the requirement for high-performance parallelism and strong static typing, which facilitates the management of the intricate building ontology and multivariate telemetry tensors. By executing on the Java Virtual Machine (JVM), the service achieves industrial-grade

scalability and resilience, allowing the system to process data from thousands of assets simultaneously without the execution bottlenecks typical of interpreted languages.

### 5.3.1. Multi-Tenant Lifecycle Management

The implementation ensures strict separation between different organizations through a dedicated multi-tenant management layer. Multi-tenancy is defined here as an architectural paradigm where a single instance of the software serves multiple distinct customers, known as tenants, while maintaining logical isolation of their data and configurations.

The lifecycle of these tenants is managed by the `TenantService`, which identifies licensed entities via the `DatabaseTenantRepository`. To maintain synchronization with the current platform state, the `TenantRegistry` performs a reconciliation sweep every 15 minutes. During this cycle, the registry identifies new licenses to instantiate corresponding `TenantWorkerLoop` instances, restarts workers that have encountered critical failures, and terminates processes for tenants whose licenses have expired or were removed.

Each tenant is assigned an independent `ADWorker`, ensuring that processing tasks for one organization do not interfere with the resource allocation of another. This isolation is further reinforced at the database level by a central registry that maps tenant identifiers to specific data scopes, ensuring that the `AttributeDataProcessor` only retrieves telemetry and hierarchical metadata belonging to the respective tenant. The registration of shutdown hooks ensures that all active workers, schedulers, and database connection pools are closed gracefully upon service termination, preventing data corruption or leaked resources.

## 5.4. Data Acquisition and Processing Pipeline

The transformation of raw building telemetry into structured inputs for the Chronos-2 model requires a multi-stage pipeline. This process ensures that the detection logic focuses on relevant energy signals while maintaining high data quality through automated cleaning and contextual enrichment.

### 5.4.1. Attribute Filtering and Hierarchical Selection

The system utilizes the Eliona attribute metadata to identify relevant telemetry points. To focus on electrical energy consumption, the `AttributeDataProcessor` queries all attributes assigned to the "Energy" type with *kWh* as the defined unit. To optimize computational resources and ensure operational relevance, the system applies two primary filters:

**Relevance thresholding** The processor analyzes the historical peak daily consumption for each meter. If the highest recorded energy consumption over a 24-hour period represents a financial value lower than a configurable threshold (standardized at $1.00), the attribute is excluded from both the detection pipeline and the Root Cause Analysis (RCA).

**Depth-limited detection** While all eligible attributes are organized into a tree structure based on their physical location, anomaly detection is strictly performed on the first two layers (typically the main meter and floor meters). The remaining descendant nodes in the hierarchy are utilized exclusively for diagnostic purposes during the RCA phase.

### 5.4.2. Contextual Enrichment: Site-Localized Weather

The system integrates exogenous environmental data to capture the impact of external drivers on building energy demand. Each meter tree is associated with a specific site, which provides precise geographical coordinates (latitude and longitude). The `WeatherProvider` utilizes these coordinates to fetch localized weather telemetry from the Open-Meteo API. This data is persisted in the database to serve as covariates for the `ADPipeline`, ensuring that consumption spikes caused by extreme weather are not incorrectly flagged as technical faults.

### 5.4.3. Data Cleaning and Gap-Resilience Logic

The microservice retrieves aggregated telemetry from the database for five distinct timeframes: 15-minute, 1-hour, 4-hour, 1-day, and 1-week buckets. To ensure data quality resilience, the following preparation steps are applied to the raw signal:

**Gap filling and spike redistribution** Missing data buckets are filled with zero values. If a transmission gap is followed by a recovery spike—a common occurrence in

IoT gateways—the accumulated energy value is distributed equally across the duration of the preceding gap to prevent false-positive alerts.

**Integrity flagging** For every data point, the system generates a binary feature that is set to 1 if a gap was present and 0 if the data was received normally. This provides the model with explicit context regarding signal reliability.

**Recursive imputation** To prevent known anomalies from "poisoning" the sliding input window of the Chronos model, any value previously identified as anomalous is replaced with its corresponding predicted median value before being used for subsequent forecasts.

### 5.4.4. Reactive Data Fetching

The pipeline operates on a reactive schedule to synchronize with the database's aggregation cycles. Every 15 minutes, a new fetch operation is triggered; however, data is only retrieved for completed aggregation buckets. For example, 15-minute buckets are refreshed every quarter-hour, whereas 1-day buckets are only updated every 24 hours once the full daily period has concluded. This ensures that the model always operates on finalized, consistent telemetry.

## 5.5. Stochastic Inference and Anomaly Quantification

The core of the detection logic resides in how data is transmitted to the Chronos-2 endpoint and how the resulting stochastic forecasts are interpreted to identify and quantify anomalies.

### 5.5.1. Feature-Driven Prediction Strategy

To fulfill the methodological requirement of reducing dependence on autoregressive lags, the system utilizes a specific inference strategy. As established in Section 4.4, standard sequential models often "adapt" to anomalies because the anomalous value becomes a feature for the next prediction. To mitigate this, the microservice does not send the actual value of the timestamp currently being evaluated for an anomaly.

Instead, the service transmits a historical buffer along with contextual features (weather, occupancy, and temporal indicators) and requests a prediction horizon that extends up

to the point of interest. By providing the model with "future" known covariates while withholding the actual consumption at the target timestamp, the model is forced to rely on the established contextual relationships rather than the most recent (potentially anomalous) observation.

### 5.5.2. Batch Processing and Quantile Requests

The `ADPipeline` batches requests for multiple attributes and timeframes into single JSON payloads to maximize the throughput of the Azure ML endpoint. For each attribute, the service requests specific quantiles from the Chronos-2 model:

- **99th percentile** ($q_{0.99}$): used as the high-sensitivity threshold for detecting extreme deviations.
- **68th percentile** ($q_{0.68}$): used as the normative baseline for financial impact calculations in high-consumption scenarios.
- **50th percentile** ($q_{0.50}$): the median, serving as the central tendency of the prediction distribution.

### 5.5.3. Detection Logic and Financial Quantification

An anomaly is formally triggered if the actual value $x_t$ falls outside an extreme prediction interval defined by the 99th percentile:

$$x_t > q_{0.99,\text{upper}} \quad \text{or} \quad x_t < q_{0.01,\text{lower}}. \tag{5.1}$$

Once an anomaly is confirmed, the system calculates the financial impact. A critical challenge in multi-modal building data is that the mean often represents an improbable "middle ground" between an *On* and *Off* state. To prevent inflated impact values, the system calculates the residual distance from the 68th percentile rather than from the mean.

**Negative impact (waste)** If the consumption is higher than the upper bound, the wasted energy is quantified as the distance to the $q_{0.68}$ quantile. This ensures that if the normal behaviour at that time could have been a high-load state, the anomaly is measured against that high-state boundary rather than a lower global average.

**Positive impact (savings)** If the consumption is lower than the lower bound, the savings are measured relative to the corresponding lower quantile.

These residuals are then multiplied by the tenant-specific `energyCostPerKwh` to generate a signed monetary value, which is persisted as part of the anomaly object. To avoid cluttering the system with events that correspond to only a few cents of effect, any candidate anomaly whose absolute financial impact falls below a configurable minimum threshold is discarded and not promoted to a persisted anomaly object.

## 5.6. Hierarchical Root Cause Analysis (RCA)

When an anomaly is confirmed on a high-level meter, such as the main meter, the `ADWorker` initiates a targeted diagnostic sweep of the descendant hierarchy. The service identifies all sub-meters within the functional tree and retrieves their corresponding time-series trends.

### 5.6.1. Diagnostic Attribution

These sub-meter trends are processed through the `ADPipeline` in a specialized diagnostic mode. The system calculates the individual financial impact for each sub-component to determine their relative contribution to the primary anomaly. The resulting diagnostic data includes:

**Asset-specific impact** A breakdown of financial loss attributed to individual assets (for example, “Light 1: -$3.00”, “HVAC Pump: +$1.00”).

**Asset type aggregation** Contributions grouped by category to identify systemic issues, such as a specific percentage of the total impact being caused by “Lighting” or “Plug Loads”.

### 5.6.2. Localization and Weather Context

The diagnostic summary is enriched with localized metadata and environmental context. The system utilizes the `WeatherProvider` to retrieve the meteorological conditions at the exact time of the event, such as the outside temperature and sky coverage. This information is formatted into a standardized diagnostic string, for example: “Unusual high energy consumption on Monday at 04:30. Outside temperature 12 °C, Weather: Clear Sky.”

## 5.7. Temporal Collapse and Persistence

The system monitors telemetry across five distinct timeframes (from 15-minute to 1-week buckets), which frequently results in overlapping detection events. To prevent redundant alerting and maintain data integrity, the `ADPipeline` implements a temporal collapse logic:

**Hierarchical merging** Anomalies detected on shorter timeframes are absorbed into higher-timeframe events as they mature. For example, four individual 15-minute anomalies are deleted once a corresponding 1-hour anomaly is confirmed, with their diagnostic data and timestamps merged into the 1-hour object.

**Impact optimization** To ensure the most realistic representation of waste, the system retains the highest financial impact value between the aggregated lower-timeframe residuals and the single higher-timeframe calculation.

## 5.8. AI Synthesis and Recommended Actions

For anomalies exceeding a defined priority or financial threshold, the diagnostic payload is transmitted to the `AnomalyExplainer`. This component utilizes a Large Language Model (LLM) to synthesize the raw diagnostic data into operational insights.

### 5.8.1. Scenario A: Behavioral Fault (Lighting)

If the root cause analysis identifies that approximately 90% of a nighttime anomaly was caused by lighting assets, the LLM generates a targeted explanation:

> Possible explanation: Technical staff or occupants likely left the lighting systems active during non-operational hours. Recommended action: Manually deactivate the identified lighting circuits and implement an automated “All-Off” logic using the platform’s rule chains.

### 5.8.2. Scenario B: Technical Fault or Misuse (Plug Loads)

In cases where a plug-load asset exhibits an extreme, sustained spike during a weekend, the AI identifies potential electricity theft or equipment malfunction:

Possible explanation: The sustained consumption on a specific plug load suggests the unauthorized use of high-power external devices or potential electricity theft. Recommended action: Inspect the physical location of the asset for unauthorized hardware and configure a real-time rule-engine alert for future weekend consumption spikes on this circuit.

The finalized anomaly object, including the diagnostic summary, financial impact, and AI-generated guidance, is persisted in the central anomalies table for frontend visualization.

## 5.9. Tenant-Specific Configuration and Parameterization

To ensure the system remains adaptable to different operational requirements and economic conditions, the implementation includes a dedicated configuration layer for tenant-specific parameters. These settings are persisted in a centralized `anomaly_config` table and are refreshed dynamically by the `TenantRegistry` to govern the behavior of the `ADWorker` loops.

The configuration table allows for granular control over the detection sensitivity and the financial logic applied to each organization. This structure ensures that the system can be tailored to the specific risk tolerance and cost structures of diverse tenants. Key parameters include:

**Sensitivity ($q$)** This parameter defines the quantile at which an anomaly is triggered. While the standard value is $0.99$, a tenant may adjust this to increase or decrease the width of the normative band.

**Check interval (minutes)** This setting determines the frequency of the `TenantWorkerLoop` execution. While the default is 15 minutes, a tenant may increase this interval if they are primarily interested in higher-level temporal aggregations, such as daily reports, rather than real-time quarter-hourly monitoring.

**Financial impact threshold** This parameter governs the `AttributeDataProcessor` filtering logic. If set to a value such as 1.00, any meter whose peak daily consumption represents less than $1.00 in potential waste is excluded from the monitoring and RCA processes to optimize computational resources.

**Financial impact alert threshold** This value serves as a final filter for the `ADPipeline` before an event is persisted. For example, if a tenant sets this to 10.00, the system

will only trigger a formal anomaly alert if the calculated financial impact exceeds $10.00, thereby reducing alert fatigue caused by negligible deviations.

**Energy cost per kWh** Each tenant specifies their current electricity rate here. This value is the primary scalar used to convert energy residuals into the monetary impacts stored in the anomalies table.

**Audit metadata** Each configuration record includes `modified_by` and `modified_at` fields to maintain a traceable history of administrative changes to the detection parameters.

## 5.10. Frontend Visualization and User Interaction

The results of the backend orchestration and AI-driven diagnostics are presented through a multi-tenant frontend interface. This interface acts as the visualization layer for the anomalies persisted in the database, allowing facility managers to monitor energy health and interact with detected faults. The application layer utilizes a widget-based system to display real-time telemetry alongside anomaly indicators.

### 5.10.1. The Anomalies Table Interface

The implementation introduces an anomalies list as a core functional component within the platform's alert center. This module is designed to mirror the existing alarms list to ensure a consistent user experience while providing specialized tools for energy-centric data management. The interface is extended with an "Anomalies" tab located adjacent to the existing "Alarms" section, providing a unified environment for system health monitoring.

The data grid presents a structured overview of all identified deviations to facilitate rapid triage and operational decision-making. Each entry in the table displays critical diagnostic metrics, including the severity, the source asset, the financial impact, and the comparison between predicted and actual values. Columns for timeframes and tags enable the filtering and sorting of anomalies based on specific operational scopes.

### 5.10.2. User Feedback and Status Management

A central feature of the frontend is the capacity for human operators to validate the findings of the autonomous detection system. This human-in-the-loop interaction, in

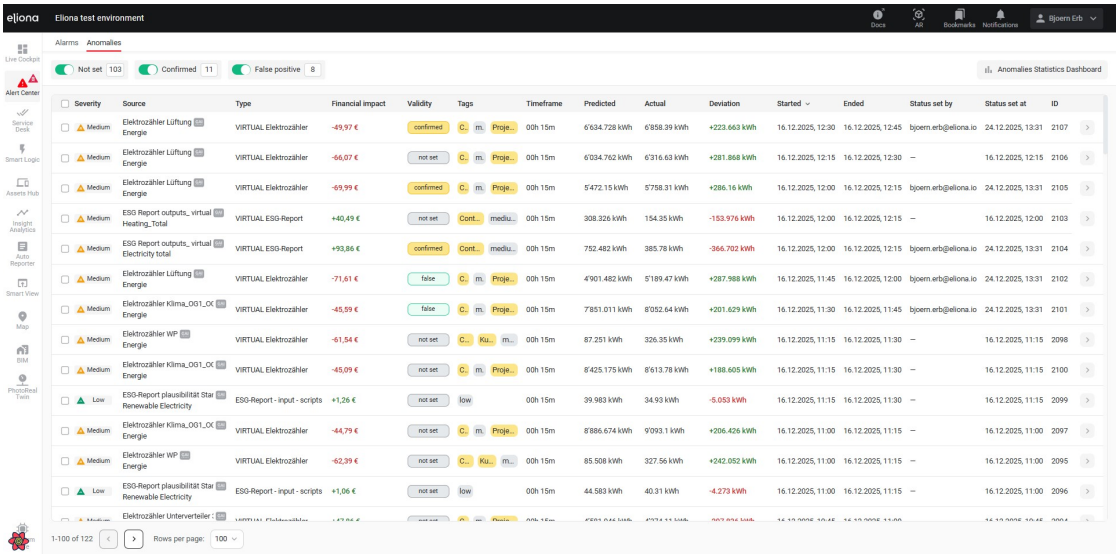



Figure 5.2: Anomalies list interface in the Eliona frontend, showing key diagnostic metrics, financial impact, and filtering options.

which human feedback is integrated into an automated process, is required to maintain model accuracy and long-term system trust.

Operators utilize a dynamic status control to select one or multiple anomalies and assign a status of “Not set”, “Confirmed”, or “False positive”. A false positive is defined as a result that indicates a condition is present when it is not. The “Add comment” function allows facility managers to record observations or remediation steps directly within the anomaly record, creating a traceable history for maintenance teams.

Marking an anomaly as a false positive triggers a data-cleansing process. This information is utilized to exclude the flagged period from future training cycles. As a result, non-anomalous events, such as authorized maintenance or unique operational shifts, are correctly classified as normal behaviour in future baselines.

### 5.10.3. Integrated Anomaly Analytics and Visualization

The implementation extends the platform’s analytical capabilities by introducing a specialized anomaly-detection analytic type. This component integrates directly into the existing insight analytics framework, allowing users to visualize stochastic detection results across dashboards and automated reports. It is designed to provide a cohesive visual bridge between raw telemetry and the probabilistic outputs of the Chronos-2 model.

### Dynamic Chart Overlays

When the anomaly-detection analytic is active, the platform provides two distinct layers of visual context to the consumption charts. First, the system generates a red background overlay for the entire asset profile whenever a deviation is identified. This highlight persists across all associated attributes of the asset, regardless of the selected aggregation level, enabling operators to observe how a failure in one sub-meter propagates through the energy consumption of the larger system.

Second, for specific energy meters, the interface provides a detailed granular view when the timeframe and aggregation type match the detection parameters. It displays red dots to mark individual anomalous data points alongside a green expected range. This green band represents the stochastic normative profile established by the model's quantiles, allowing users to visually quantify the magnitude of the deviation relative to the predicted operational baseline.

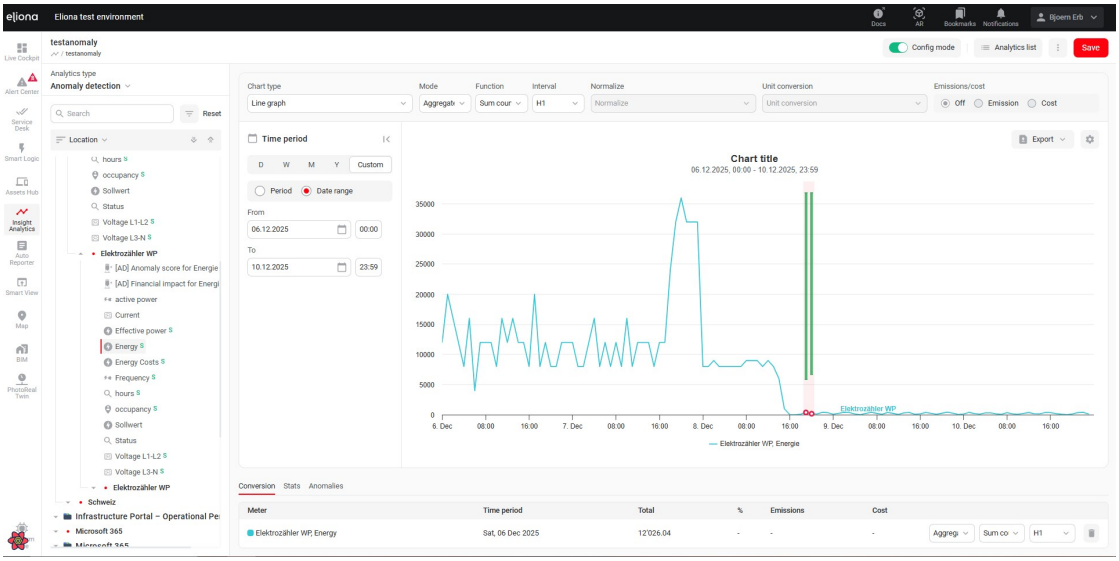



Figure 5.3: Integrated anomaly-detection analytic within the insight analytics framework, highlighting anomalous periods on the consumption charts.

### Interactive Diagnostics and Tooltips

To facilitate immediate root-cause identification, the system includes interactive tooltips that appear upon hovering over an anomalous data point. These tooltips provide a concentrated summary of the anomaly object, including:

**Quantitative metrics** Real-time data regarding the financial impact, the predicted versus actual consumption, and the severity level.

**AI-synthesized context** The AI explanation regarding the likely origin of the fault (for example, "extended equipment runtime or a control issue") and the corresponding

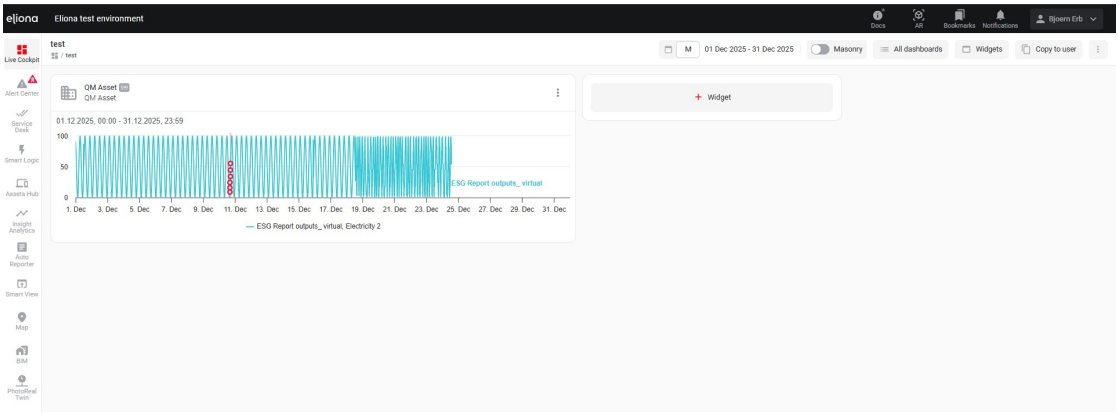



Figure 5.4: Anomaly-detection analytic embedded in a customizable dashboard, combining stochastic overlays with other operational widgets.

recommended action (for example, “review the affected equipment settings and operating schedule”).

**Validation status** The current validity status (for example, “Confirmed”), which reflects the user feedback provided in the anomalies list.

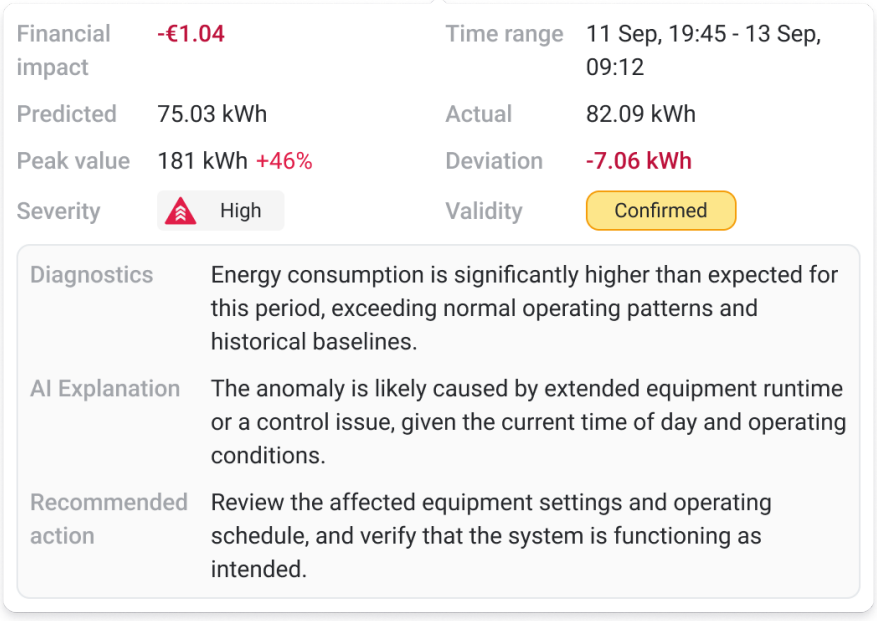



Figure 5.5: Interactive tooltip attached to an anomalous data point, summarizing financial impact, AI explanation, and validation status.

Because this functionality is implemented as a standard analytic type, it is fully compatible with the platform’s existing reporting engine and customizable dashboards. This ensures that the detection results are not isolated within a separate module but are

integrated into the daily operational workflows of the facility management team.

### 5.10.4. Anomaly Detail View and Operational Synthesis

The anomaly detail view provides a specialized environment for the deep-dive analysis of individual detection events. Users access this page by selecting the navigational arrow corresponding to a specific entry in the anomalies list. This view consolidates all relevant quantitative and qualitative data into a single operational summary, facilitating a transition from detection to remediation.

The detail page is structured to provide immediate clarity regarding the severity and context of the fault. The header section displays the core metadata, including the financial impact, the severity level, and the validation status. A dedicated validation panel allows the user to update the status to “Confirmed” or “False” and provides a text area for recording technical comments or observations.

The primary analytical component of this view is a high-resolution analytic chart. This visualization is automatically centered on the anomalous period, allowing the user to observe the consumption profile immediately before and after the identified deviation. This spatial centering provides critical temporal context, helping the operator determine whether the event was a singular spike or the onset of a sustained operational shift.

The lower section of the page presents the diagnostics and AI-synthesis results. The diagnostics string summarizes the technical conditions, such as the outside temperature and weather at the time of the event, and identifies the likely contributing submeters. Below this, the AI explanation provides a natural-language interpretation of the data, for instance by identifying a likely “data acquisition failure” or “unexpected shutdown of major loads”. The recommended actions then list specific investigative steps, such as checking metering infrastructure for outages or verifying scheduled maintenance activities.

By combining these interactive elements, the anomaly detail view transforms raw telemetry into a structured maintenance task, directly supporting the actionable-insights requirement established in the system’s methodology.

### 5.10.5. Anomaly Statistics and Macro-Level Reporting

To complement the detailed, asset-specific views, the platform includes an anomaly-statistics dashboard designed for macro-level analysis and executive reporting. This interface aggregates data across entire tenants or individual sites, providing facility managers with a high-level overview of system health and financial performance over

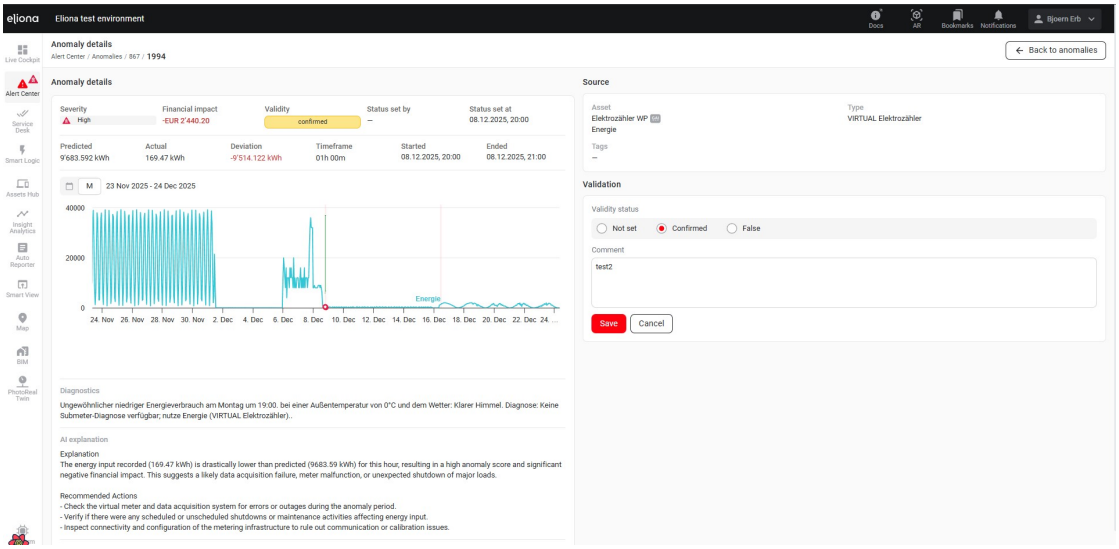



Figure 5.6: Anomaly detail view aggregating financial impact, validation status, analytic chart, diagnostics, and AI-synthesized recommended actions.

time.

The scope of the data presented is controlled through a unified timeframe selector. Users can toggle between standard periods, such as the current month or the current fiscal year, which instantly recalculates all metrics on the dashboard. The primary key performance indicators (KPIs) focus on the aggregated financial impact. A prominent widget displays the total monetary value of detected energy waste or savings within the selected period, alongside a percentage-based comparison to the previous equivalent timeframe. This immediate context allows stakeholders to quickly determine if operational performance is trending positively or negatively.

Beneath the top-level KPIs, the dashboard provides categorical breakdowns to identify systemic issues. A visualization of the most impact by asset type highlights which categories of equipment, such as HVAC or lighting, are contributing most significantly to financial losses. Temporal trends are visualized through an overspend-versus-saved bar chart, allowing users to visually track the effectiveness of remediation efforts over successive periods. A parallel bar chart provides a count of anomalies per asset type, distinguishing between high-frequency, low-impact events and rare, high-cost failures. For multi-site tenants, a site-aggregation list details the anomaly count and total impact for each location. Selecting a specific site in this list redirects the user to a filtered version of the statistics page dedicated solely to that location's data.

A critical analytical tool within this dashboard is the financial-impact heatmap, designed to reveal temporal patterns in energy waste. The configurations of its axes adapt dynamically based on the selected timeframe. For shorter periods, the x-axis represents the date and the y-axis represents the time of day. The cell colour intensity

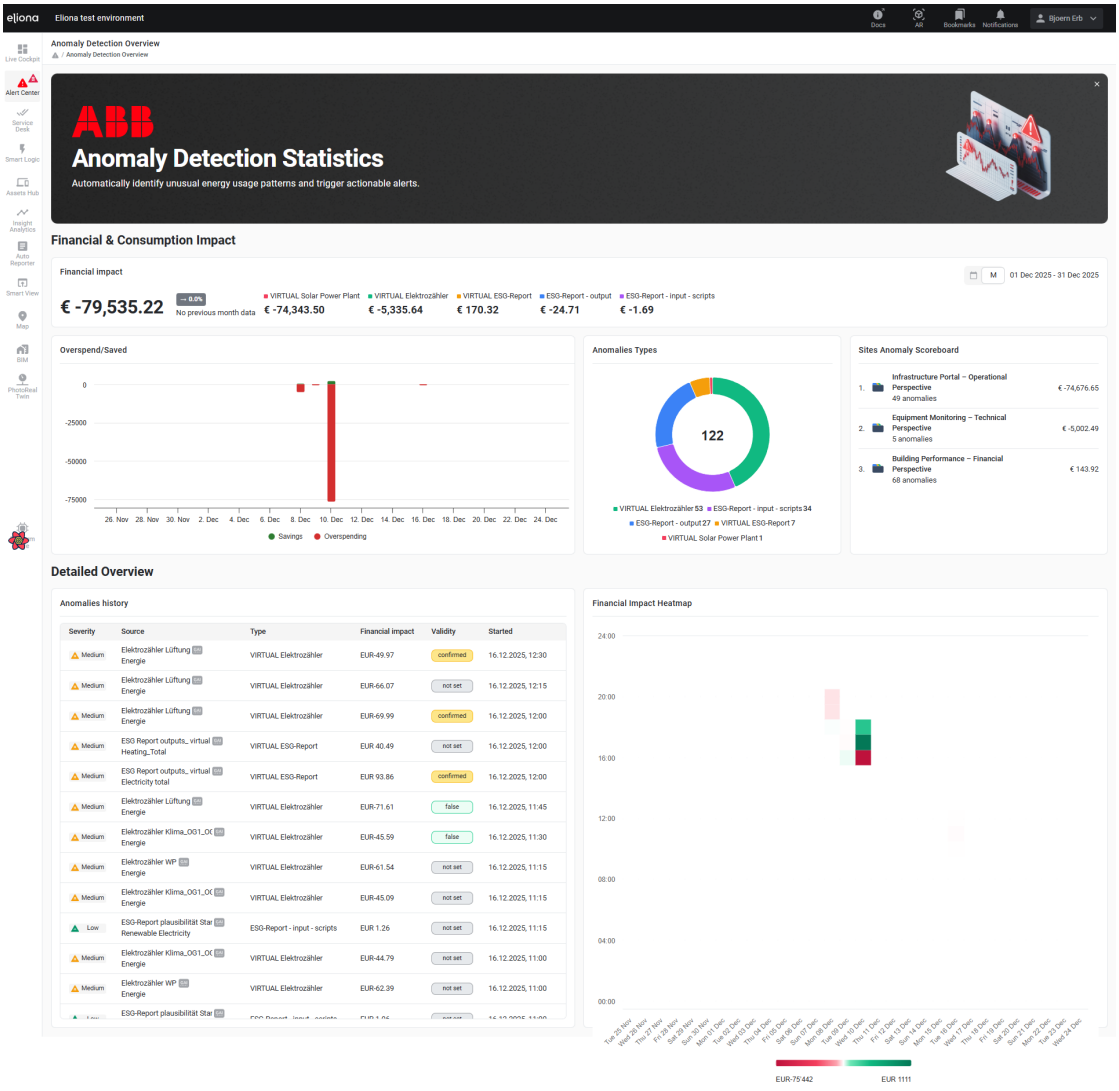



Figure 5.7: Anomaly-statistics dashboard providing macro-level KPIs, categorical breakdowns, temporal trends, and site-level aggregation for executive reporting.

indicates the magnitude of financial impact, making it immediately apparent if specific times of day consistently experience costly anomalies. For longer timeframes, such as a full year, the heatmap aggregates data to show day of week versus time of day. This view is particularly valuable for identifying systemic operational faults, such as recurring high-impact anomalies on Monday mornings that may indicate faulty equipment start-up sequences after weekend shutdowns.

### 5.10.6. Asset-Specific Anomaly Integration

The platform's asset-detail view is expanded to include a dedicated "Anomalies" tab, facilitating a seamless transition between general asset monitoring and specialized anomaly investigation. This view is filtered to display only the deviations and financial impacts associated with the meters assigned to the currently selected asset.

The interface provides a localized summary that includes the date of the most recent anomaly and the total financial impact accumulated by the asset's meters within the selected timeframe. A donut chart visualizes the distribution of anomaly types, while a localized "Anomalies on chart" widget highlights the specific timestamps where deviations occurred relative to the asset's consumption profile. An "Anomalies history" table lists the individual events, including their severity and financial impact, providing a direct link to the anomaly detail view for further investigation.

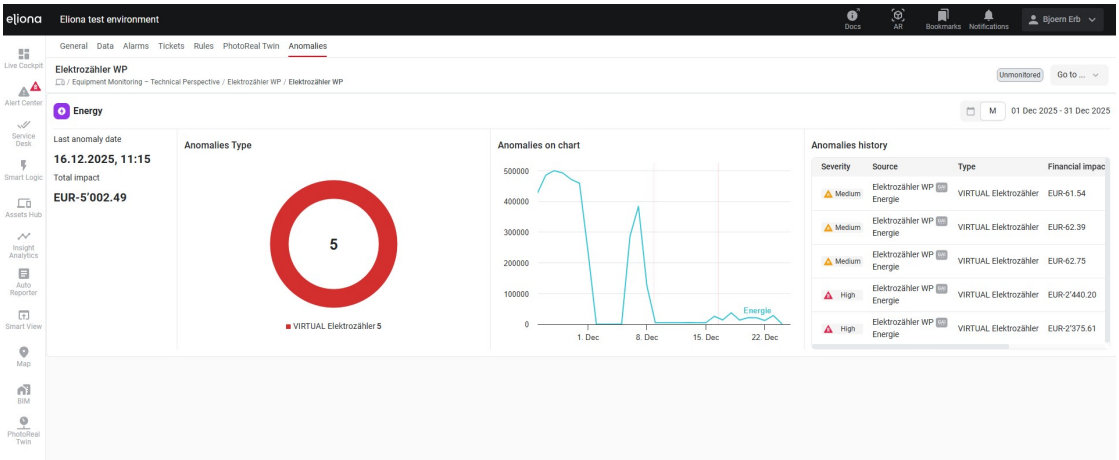



Figure 5.8: Asset-detail view with integrated anomalies tab, showing localized statistics, anomaly-type distribution, and historical events for the selected asset.

# 6

# Discussion and Future Work

The evaluation of the integrated system demonstrates the efficacy of combining stochastic forecasting with hierarchical root-cause analysis. However, the transition from a synthetic benchmark to diverse industrial environments reveals specific areas where the methodology can be refined to enhance diagnostic depth and modelling flexibility.

## 6.1. Critical Reflection on System Design

The current implementation of root-cause analysis (RCA)—the process of identifying the origin of a fault—relies primarily on the statistical attribution of financial impact across sub-meters. While this identifies *where* an anomaly occurs, it does not fully explain *why* the deviation was triggered within the control layer. Future iterations could enhance the RCA by integrating the operational states of high-consumption assets. By analysing control signals, such as valve positions or modulation frequencies, the system could verify whether a consumption spike is a legitimate response to a manual override or a failure of the underlying control logic.

Furthermore, the feature set utilized in the `ADPipeline` is currently restricted to meteorological data and temporal indicators. Empirical evidence from the BOPTEST correlation analysis (see Figure 4.12) indicates that occupancy—represented by people count—is one of the most significant drivers of energy consumption. The absence of real-time occupancy telemetry in many tenant buildings represents a significant information gap. The system should be expanded to allow tenants to select specific attributes, such as $CO_2$ levels or access-control logs, to be included as exogenous

features for customized detection models.

## 6.2. Data Integrity and User-Centric Baseline Selection

The current modelling approach assumes that all historical data preceding the activation of an anomaly-detection licence represents a healthy operational state. This assumption is often violated in real-world facilities where persistent faults may already be present. To address this, an interface for manual baseline configuration is proposed. This would enable facility managers to designate specific historical intervals as *gold-standard* periods.

Implementing this with the current Chronos-2 architecture presents a technical challenge, as the model requires a continuous temporal sequence immediately preceding the target timestamp. To utilize a non-contiguous healthy baseline from a previous year, the system would require a mechanism to translate and align historical timestamps to the current prediction window.

Additionally, the reliability of the RCA is heavily dependent on the metering density of the building. In facilities with low sub-metering granularity, the system's ability to attribute faults remains limited to large-scale aggregates, highlighting the need for further evaluation on diverse, real-world datasets.

## 6.3. Future Architecture: The Universal Energy Feature Forecaster

The utilization of Chronos-2 in the current system represents an adaptation of a sequential foundation model for feature-based anomaly detection. While effective, this approach does not fully leverage the model's internal probability distributions for anomaly scoring in the same way a mixture-density network (MDN) does. A significant limitation is the reliance on fixed quantile bounds (for example, $q_{0.99}$), which simplifies the complex multimodal output of the transformer into a binary threshold.

### 6.3.1. In-Context Zero-Shot Modelling

A proposed architectural shift involves the development of a specialized foundation model designed as a universal energy feature forecaster. Instead of predicting a sequence based on recent history, this model would utilize in-context learning (ICL). In

this paradigm, the model is provided with a set of baseline features and their corresponding target values as context, regardless of their temporal proximity to the current timestamp. This would allow the model to ingest healthy baseline data from a different season or a different year as a direct reference for the current prediction task.

### 6.3.2. Probabilistic Anomaly Scoring

The proposed model would retain the token-based probability output of the Chronos architecture but utilize the full distribution to calculate an anomaly score. By computing the negative log-likelihood (NLL) of an observed value across all predicted tokens, the system would achieve a sensitivity comparable to the MDN while maintaining the zero-shot generalization of a foundation model.

This architecture would also enable comparative baseline analysis without retraining. For example, an operator could predict March consumption using both January and February baselines to quantify the resulting energy savings from efficiency measures implemented in February. In doing so, the anomaly-detection system would evolve from a pure fault detector into a broader decision-support tool for evaluating building-decarbonization strategies.

## 6.4. Reflections on Energy Anomaly Benchmarking

The experimental evaluation conducted within this research represents a foundational step towards a standardized benchmarking framework for energy-specific anomaly detection. It highlights the necessity for datasets that prioritize **contextual anomalies**—deviations that are only anomalous relative to external variables such as weather or occupancy—over simple point deviations. By utilizing the **volume under the surface of the precision-recall curve (VUS-PR)**, the benchmark addresses the inherent temporal characteristics of industrial energy faults, which frequently persist over extended durations.

However, the current benchmarking methodology reveals several opportunities for improvement. While the results provide a comparative overview, the absence of exhaustive **hyperparameter tuning**—the process of optimizing the internal parameters of a model to achieve peak performance—for many baseline methods potentially masks their true detection capabilities. Furthermore, the protocol for comparing zero-shot **foundation models**—large-scale models pre-trained on diverse datasets to perform tasks without site-specific training—against traditional supervised methods requires further formalization.

Future benchmarking efforts must ensure that all models are evaluated under optimal configurations to support a scientifically robust comparison. Such a refined framework will serve as a vital tool for the objective validation of emerging architectures in the building-automation sector.

# 7

# Conclusion

The research conducted in this thesis successfully established a robust methodology and technical framework for the automated detection and quantification of energy anomalies in building environments. By addressing the statistical complexities of building telemetry—specifically its non-stationary and multi-modal nature—the project provided a solution that transcends the limitations of traditional deterministic forecasting.

The investigation into predictive modelling paradigms revealed that standard sequential forecasting is susceptible to error propagation and the adaptation paradox. It was shown that autoregressive models often incorporate anomalous data into their internal state, which leads to signal instability and the masking of sustained faults. To resolve these issues, a stochastic approach was developed that utilizes the **Chronos-2** foundation model within a feature-driven inference strategy. This methodology establishes a probabilistic normative operational band, allowing for the reliable identification of **Multivariate Context Point Anomalies (MCPA)**—deviations that are only identifiable through the joint analysis of consumption and exogenous drivers such as weather and occupancy.

The technical realization was achieved through a distributed, polyglot architecture. A high-performance **Scala** microservice managed multi-tenant isolation and data orchestration, while a **Python** endpoint hosted on **Azure Machine Learning** provided the required predictive capacity. The integration of a hierarchical **Root Cause Analysis (RCA)** and **Generative AI** synthesis transformed raw detection results into actionable operational intelligence. It was demonstrated that by attributing financial impact to specific assets and generating natural-language remediation steps, the system provides facility managers with a transparent tool for energy waste mitigation.

The implementation of specialized frontend modules—including the anomalies list,

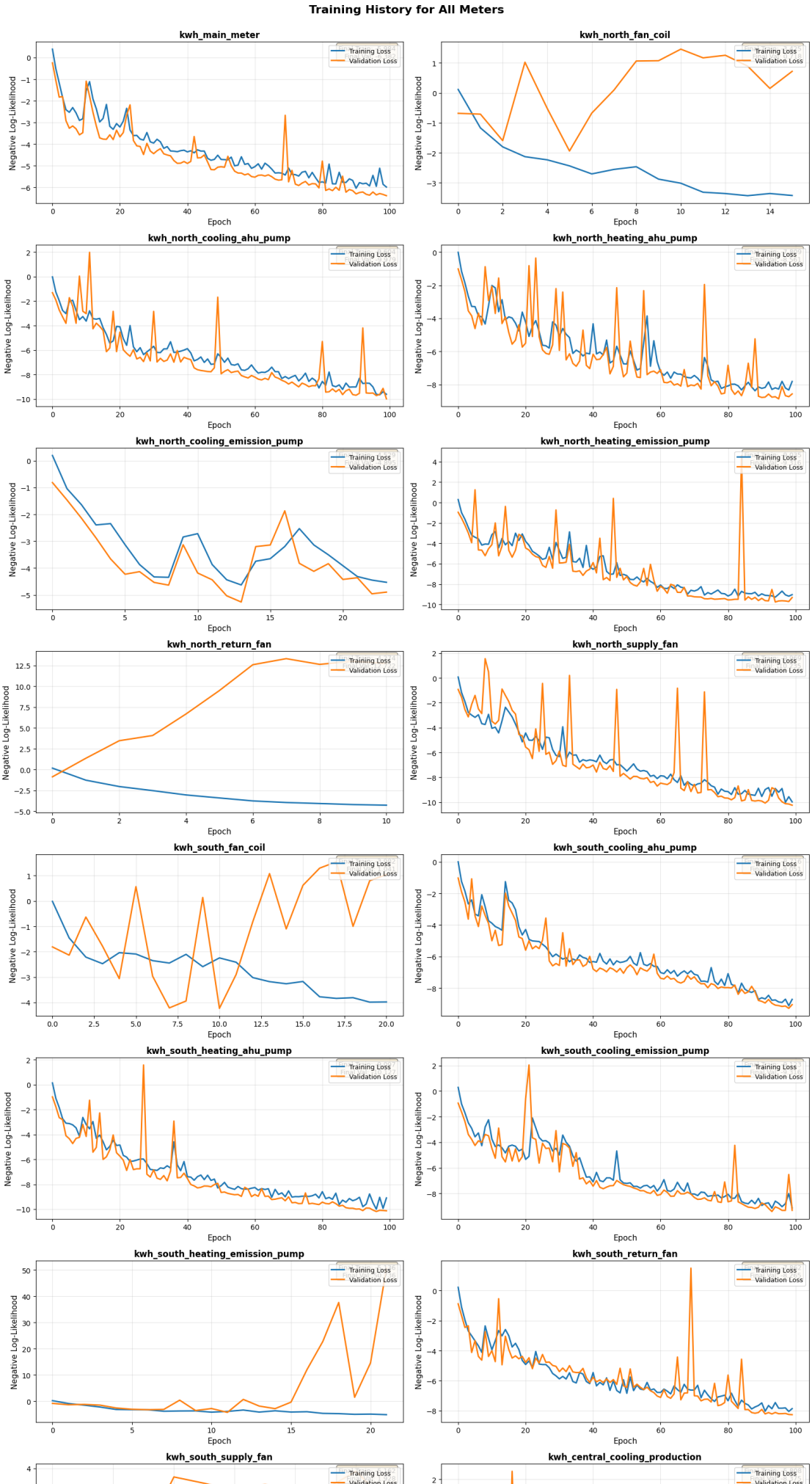
status management, and temporal heatmaps—ensures that detection events are integrated into standard maintenance workflows. The capacity for human-in-the-loop feedback, where users validate anomalies as confirmed or false, establishes a mechanism for continuous data cleansing and model refinement. This ensures that the system remains accurate as building characteristics evolve over time.

Ultimately, this research moved the **Eliona** platform from a state of reactive alarm management to proactive, intelligence-driven energy monitoring. The shift from point-based detection to stochastic distribution modeling enables a precise calculation of financial residuals, providing organizations with quantifiable data to support their sustainability and decarbonization objectives. While future work remains regarding the inclusion of control-layer states and the development of in-context forecasters, this project provides a scientifically validated foundation for the next generation of energy management systems in the building-automation sector.

# A

# Additional Figures

## A.1. Training History Across All Meters


Training History for All Meters
kwh_main_meter
kwh_north_fan_coil
kwh_north_cooling_ahu_pump
kwh_north_heating_ahu_pump
kwh_north_cooling_emission_pump
kwh_north_heating_emission_pump
kwh_north_return_fan
kwh_north_supply_fan
kwh_south_fan_coil
kwh_south_cooling_ahu_pump
kwh_south_heating_ahu_pump
kwh_south_cooling_emission_pump
kwh_south_heating_emission_pump
kwh_south_return_fan
kwh_south_supply_fan
kwh_central_cooling_production
Training Loss
Validation Loss
Negative Log-Likelihood
Epoch

# References


[LGW04] Ningyun Lu, Furong Gao, and Fuli Wang. “Sub-PCA modeling and on-line monitoring strategy for batch processes”. In: *AIChE Journal* 50.1 (2004), pp. 255–259.

[Rot+04] Kurt W Roth et al. “The energy impact of faults in US commercial buildings”. In: (2004).

[Ant09] Pedro Antmann. “Reducing technical and non-technical losses in the power sector”. In: (2009).

[MM09] Patrick McDaniel and Stephen McLaughlin. “Security and privacy challenges in the smart grid”. In: *IEEE security & privacy* 7.3 (2009), pp. 75–77.

[LN16] Xiufeng Liu and Per Sieverts Nielsen. “Regression-based online anomaly detection for smart grid data”. In: *arXiv preprint arXiv:1606.05781* (2016).

[Peñ+16] Manuel Peña et al. “Rule-based system to detect energy efficiency anomalies in smart buildings, a data mining approach”. In: *Expert Systems with Applications* 56 (2016), pp. 242–255.

[Wu17] Jianxin Wu. “Introduction to convolutional neural networks”. In: *National Key Lab for Novel Software Technology. Nanjing University. China* 5.23 (2017), p. 495.

[Su+19] Ya Su et al. “Robust anomaly detection for multivariate time series through stochastic recurrent neural network”. In: *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 2019, pp. 2828–2837.

[Fu22] Chun Fu. *Summary of 1st Place Solution — Large-scale Energy Anomaly Detection (LEAD)*. Kaggle competition writeup. 2022. URL: https://www.kaggle.com/competitions/energy-anomaly-detection/writeups/chun-fu-summary-of-1st-place-solution (visited on 12/25/2025).

[GA22] Manoj Gulati and Pandarasamy Arjunan. “LEAD1. 0: a large-scale annotated dataset for energy anomaly detection in commercial buildings”. In: *Proceedings of the thirteenth ACM international conference on future energy systems*. 2022, pp. 485–488.

[Pap+22] John Paparrizos et al. “TSB-UAD: an end-to-end benchmark suite for univariate time-series anomaly detection”. In: *Proceedings of the VLDB Endowment* 15.8 (2022), pp. 1697–1711.

[GCM23] Azul Garza, Cristian Challu, and Max Mergenthaler-Canseco. “TimeGPT-1”. In: *arXiv preprint arXiv:2310.03589* (2023).

[Alš24] Oskaras Alšauskas. “World energy outlook 2024”. In: *International Energy Agency: Paris, France* (2024).

[Edi24] Edison Foundation Institute for Electric Innovation. *120 million smart meters in US in 2022*. Online article. 2024. URL: https://www.enlit.world/library/120-million-smart-meters-in-us-in-2022 (visited on 12/24/2025).

[Gos+24] Mononito Goswami et al. “Moment: A family of open time-series foundation models”. In: *arXiv preprint arXiv:2402.03885* (2024).

[IoT24] IoT Analytics. *Global Smart Electricity Meter Adoption 2024 by Region*. Online graphic. 2024. URL: https://iot-analytics.com/wp-content/uploads/2024/02/Global-Smart-Electricity-Meter-Adoption-2024-by-Region-vweb.png (visited on 12/25/2025).

[LP24] Qinghua Liu and John Paparrizos. “The elephant in the room: Towards a reliable time-series anomaly detection benchmark”. In: *Advances in Neural Information Processing Systems* 37 (2024), pp. 108231–108261.

[Ans+25] Abdul Fatir Ansari et al. “Chronos-2: From univariate to universal forecasting”. In: *arXiv preprint arXiv:2510.15821* (2025).

[Azz+25] Davide Azzalini et al. “An empirical evaluation of deep autoencoders for anomaly detection in the electricity consumption of buildings”. In: *Energy and Buildings* 327 (2025), p. 115069. ISSN: 0378-7788. DOI: https://doi.org/10.1016/j.enbuild.2024.115069. URL: https://www.sciencedirect.com/science/article/pii/S037877882401185X.

[Eli25a] Eliona IoT Platform. *Asset Modeling – Create Templates*. Online documentation. 2025. URL: https://doc.eliona.io/collection/eliona-english/documentation/assets/asset-modeling-create-templates (visited on 12/23/2025).

[Eli25b] Eliona IoT Platform. *Assets*. Online documentation. 2025. URL: https://doc.eliona.io/collection/eliona-english/documentation/assets (visited on 12/23/2025).

[Eli25c] Eliona IoT Platform. *Introduction to Ontologies*. Online documentation. 2025. URL: https://doc.eliona.io/collection/eliona-english/academy/introduction-to-ontologies (visited on 12/20/2025).

[Eli25d] Eliona IoT Platform. *Rule Chains*. Online documentation. 2025. URL: https://doc.eliona.io/collection/eliona-english/documentation/alarms-rules-and-escalations/rule-chains (visited on 12/23/2025).

[Eli25e] Eliona IoT Platform. *Rules*. Online documentation. 2025. URL: https://doc.eliona.io/collection/eliona-english/documentation/alarms-rules-and-escalations/rules (visited on 12/23/2025).

[Eli25f] Eliona IoT Platform. *Structuring Assets*. Online documentation. 2025. URL: https://doc.eliona.io/collection/eliona-english/documentation/assets/structuring-assets (visited on 12/23/2025).

[HHA25] Basu Hela, Praveen Prasad Handigol, and Pandarasamy Arjunan. "Are Time Series Foundation models good for Energy Anomaly Detection?" In: *Proceedings of the 16th ACM International Conference on Future and Sustainable Energy Systems*. E-Energy '25. Association for Computing Machinery, 2025, pp. 656–665. ISBN: 9798400711251. DOI: 10.1145/3679240.3734633. URL: https://doi.org/10.1145/3679240.3734633.

[MM25] Roya Morshedi and S. Mojtaba Matinkhah. "A Comprehensive Review of Deep Learning Techniques for Anomaly Detection in IoT Networks: Methods, Challenges, and Datasets". In: *Engineering Reports* 7.9 (2025), e70415. DOI: https://doi.org/10.1002/eng2.70415. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/eng2.70415. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/eng2.70415.