symmetry

MDPI

Article

Decision-Critical Data Quality Contracts for IoT-Based Elderly Care: Symmetric vs. Asymmetric Enforcement for Fall and Health Deterioration Decisions

Waleed Al Shehri

Department of Computing, College of Engineering and Computing in Al-Lith, Umm Al-Qura University, Makkah 24381, Saudi Arabia; washehri@uqu.edu.sa

Abstract

Continuous detection of critical events such as falls and health deterioration is enabled by Internet of Things (IoT)-enabled monitoring systems in elderly care. However, system reliability is undermined by real-world sensor degradation, which produces high false-alarm rates and missed incidents. Existing systems lack differentiated governance mechanisms for acute decisions (e.g., fall detection, requiring high sensitivity and low latency) versus cumulative decisions (e.g., health deterioration monitoring, requiring stability and specificity). Conventional approaches treat data quality as a preprocessing concern rather than as a formal determinant of decision admissibility, creating a gap between data availability and decision reliability. In this paper, Decision-Critical Data Quality Contracts are proposed as a governance paradigm in which decision analytics is explicitly separated from admissibility. Symmetric (uniform) and asymmetric (adaptive) enforcement strategies are explored and implemented through a hierarchical Decision Quality Tree framework for context-aware quality assessment. A simulation-based evaluation was conducted over 72 h periods across three degradation scenarios: controlled (5% missingness), realistic (15%), and stress (30% with sensor failures). The no-contract, symmetric, asymmetric, and Decision Quality Tree approaches were compared on metrics including missed alarms, coverage, stability, false alarms, and audit trail completeness. The results demonstrate that missed fall alarms are reduced by up to 71% by the Decision Quality Tree compared to asymmetric enforcement (from 28.57% to 8.20%). Coverage improved to 97.80% and stability to 95.20%. The lowest false-alarm rates are achieved by the Decision Quality Tree (0.90% for acute decisions, 2.80% for cumulative decisions). Audit trail completeness shows a 70.6% improvement over the best baseline (score: 0.87 vs. 0.51). Ablation studies confirm that these improvements stem from synergistic combinations of fallback paths and context awareness. The Decision Quality Tree framework establishes a new balance between system availability and decision safety, providing a foundation for trustworthy IoT governance in elderly care.

Keywords: IoT elderly care; data quality contracts; decision quality tree; symmetric enforcement; asymmetric enforcement; fall detection; health deterioration; sensor degradation; graceful degradation; audit trail

Check for updates

Academic Editor: Jie Yang

Received: 8 May 2026
Revised: 18 June 2026
Accepted: 23 June 2026
Published: 27 June 2026

Copyright: © 2026 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

The adoption of IoT-enabled monitoring systems in elderly care has been considered a transformative approach to the challenges posed by aging populations. Continuous surveillance of critical incidents such as falls and progressive health deterioration in non-specialized home environments is enabled by these systems [1]. Wearable and ambient

sensors are leveraged to create a digital safety net, and extended independent living is supported while institutional care burdens are reduced [2]. However, the clinical utility of these systems depends fundamentally on the distinction between genuine emergencies and false alarms. This distinction becomes precarious when sensor data quality is degraded in real-world deployments [3].

Existing approaches to data quality in IoT systems predominantly rely on flat, static thresholds applied uniformly across all decision types [4,5]. In such threshold-based models, quality dimensions are evaluated against pre-defined minima using simple Boolean logic. While basic safety guarantees are provided, these rigid models cannot accommodate partial-sensor-failure scenarios, nor can they adapt to dynamic context shifts such as caregiver availability, time-of-day risk patterns, or evolving user mobility profiles [4]. Furthermore, flat-threshold approaches lack the capacity to differentiate between the fundamentally distinct requirements of acute and cumulative decisions—a limitation that directly motivates the asymmetric enforcement strategies proposed in this work.

The challenge is compounded by the treatment of acute and cumulative decisions as identical in existing architectures. The key tension between these fundamentally different risk profiles is overlooked. Near-zero latency and high sensitivity are required for fall detection, even at the cost of false alarms. In contrast, temporal stability and specificity are demanded by health deterioration monitoring, where oscillating alert states can erode caregiver trust.

Conventional decision-support architectures in IoT elderly care have been noted to operate on a critical yet largely unexamined assumption: that the mere presence of sensor data constitutes sufficient grounds to trigger interventions [5]. Under this assumption, urgent alerts are rendered susceptible to distortions arising from deteriorating data integrity, whether due to sensor noise, communication dropouts, battery depletion, or environmental interference [6]. The resulting disconnect between data availability and decision reliability can lead to two equally problematic outcomes: false alarms that contribute to caregiver fatigue and eroded trust, or missed detections that leave vulnerable individuals without necessary assistance during life-threatening events [7]. In safety-critical contexts, where seconds matter and consequences are severe, the implicit assumption that "data exists" implies "decision is reliable" represents a fundamental architectural flaw. This flaw is particularly acute when the divergent requirements of fall detection versus health deterioration monitoring are considered, a distinction that is not adequately addressed by current quality management approaches.

A substantial gap between controlled laboratory validations and real-world deployment outcomes in IoT elderly care has been documented in recent studies [8]. When systems are transitioned from controlled trials to actual home environments, where sensor placement tends to be suboptimal, user compliance varies, and environmental conditions fluctuate unpredictably; thus, system performance degrades dramatically [4]. False-alarm rates have been reported to increase by 300–500%, while missed detection rates for falls exceed 25% in real-world conditions, considerably higher than in laboratory settings [9]. This degradation occurs because mechanisms to evaluate whether prevailing data quality conditions justify executing safety-critical actions are lacking in existing systems. Data quality is treated as a preprocessing concern rather than as a formal determinant of decision admissibility [10].

The need for differentiated governance mechanisms is further motivated by the operational realities of elderly care monitoring. Elderly individuals exhibit diverse behavioral patterns, mobility levels, and risk profiles that evolve over time. A one-size-fits-all quality threshold cannot account for an active user, who generates more motion artifacts, versus a bedridden user, whose sensor data is sparse but clinically critical. Similarly, the conse-

quences of missed detections vary by context: a missed fall at night, when caregivers are absent, carries different risks than one during daytime, when assistance is nearby. These practical considerations underscore the necessity of the context-aware, adaptive governance framework proposed in this work.

The fundamental governance mechanism gap described above is addressed in this paper. Solutions that satisfy the conflicting requirements of acute-event detection (where sensitivity and speed are prioritized) and cumulative health monitoring (where stability and specificity are prioritized) have not been adequately developed. To address this gap, two contributions are introduced: the Decision-Critical Data Quality Contracts (DQC) ideology and its architectural implementation in the Decision Quality Tree (DQT) framework.

DQC is proposed as a novel governance architecture that explicitly separates decision analytics from decision admissibility. Unlike existing approaches that treat data quality as a preprocessing step or apply flat thresholds uniformly across all decisions, DQC establishes a formal contractual layer between data producers (sensors) and decision consumers (alerting systems). This architectural contribution specifies verifiable quality conditions that must be met before any inference is considered actionable, directly addressing the governance gap identified in prior work [4,10].

The core principle of DQC is that analytical correctness alone is insufficient justification for executing a safety-critical intervention; the evidence supporting that analysis is also expected to meet minimum quality standards appropriate to the decision's risk profile. An explicit contractual layer between data producers (sensors) and decision consumers (alerting systems) is established by DQC, and verifiable quality conditions that are to be met before any inference is considered actionable are specified. Within this ideological framework, two fundamental enforcement strategies are identified: symmetric enforcement, in which uniform quality thresholds are applied across all decisions regardless of criticality; and asymmetric enforcement, in which thresholds are dynamically adjusted based on decision type and sensor reliability.

The DQT framework differs fundamentally from existing tree-based methods such as decision trees, fault trees, and behavior trees. Decision trees are static classification structures that lack contextual adaptation and produce no audit trails. Fault trees analyze failure modes in engineered systems but do not govern real-time decisions under dynamic data quality conditions. Behavior trees model task-switching logic for autonomous agents but lack data quality awareness and governance capabilities. In contrast, DQT is a hierarchical, context-aware governance structure specifically designed for data quality assessment: it incorporates multiple reasoning paths with graceful degradation, conditional branching based on situational context, complete audit trail generation at every node, and explicit separation of analytical inference from admissibility judgments. These architectural features are absent in existing tree-based approaches and represent the core technical novelty of this work.

The architectural implementation of this ideology is provided by the DQT framework. The flat-threshold checking of conventional approaches is replaced with hierarchical, context-aware governance. Rather than a single Boolean condition being evaluated, a tree structure is traversed by the DQT. In this structure, each node represents a contextualized quality assessment, each branch encodes a reasoning path tailored to specific failure scenarios, and each leaf renders an admissibility judgment accompanied by a complete, human-interpretable explanation of the decision pathway. Four capabilities that are absent in existing approaches are introduced by the DQT framework:

- Multiple reasoning paths that enable graceful degradation under partial sensor failure.
- Contextual adaptation through conditional branching that incorporates time of day, user activity patterns, and individual risk profiles.

- Complete decision traces that document every evaluation step for auditability and trust.
- Machine learning integration that enables continuous optimization of governance logic based on operational feedback.

The case for DQT-based governance is strengthened by a comprehensive simulation-based evaluation across realistic degradation scenarios. Substantial improvements over threshold-based approaches are achieved. The number of missed fall alarms when using the DQT is reduced by 71% compared to asymmetric enforcement. Operational coverage improves from 74.4% to 97.8%, decision stability from 62.4% to 95.2%, and false-alarm control is enhanced. A 70.6% improvement in audit trail completeness is delivered by the DQT (score: 0.87 vs. 0.51). Complete audit trails that transform governance from an opaque gate-keeping function into a transparent, accountable safety layer are generated. For cumulative health deterioration decisions, the lowest false-alarm rate among all strategies is achieved by the DQT. Perfect detection sensitivity and the highest temporal stability are maintained.

In addressing these challenges, the core unresolved issue identified in the literature is directly engaged: the need for governance frameworks that can accommodate the fundamentally different risk profiles, temporal requirements, and quality expectations of acute versus cumulative decisions in elderly care monitoring. Three primary contributions are offered by this paper:

- Conceptual contribution: DQC is introduced as a novel governance architecture by which the fundamental principle of separating decision analytics from admissibility is established, with differentiated enforcement strategies for acute versus cumulative decisions.
- Architectural contribution: The DQT framework is proposed as an implementable governance layer by which DQC principles are realized through hierarchical, context-aware reasoning, and the limitations of threshold-based approaches are overcome.
- Empirical contribution: A comprehensive evaluation is provided, and the results demonstrate that a superior balance between safety, coverage, stability, and audit trail is achieved by the DQT. A new Pareto frontier in the safety–coverage trade-off is established.

Paper Organization

The remainder of this paper is structured as follows. Section 2 reviews related work in IoT-based elderly care monitoring, data quality management, and safety-critical decision governance, identifying key research gaps that motivate our contributions. Section 3 formally defines the problem statement, including objectives and constraints. Section 4 presents the system model and the Decision Quality Tree architecture. Section 5 describes the experimental setup and simulation framework. Section 6 provides comprehensive evaluation results across multiple degradation scenarios, including ablation studies and stress testing. Section 7 concludes the paper with a discussion of implications, limitations, and directions for future work.

2. Literature Review and Key Gaps

The state of the art in IoT-based elderly care monitoring, data quality management, and safety-critical decision governance is reviewed in this section. Existing approaches to fall detection and health deterioration monitoring are examined, current data quality frameworks are analyzed, and the fundamental gaps that motivate the proposed DQC ideology and DQT architecture are identified. Throughout this review, a distinction is drawn between the problem space (what existing systems do) and the solution space (what

is contributed by this work), and the novel contributions are positioned within the broader research landscape.

2.1. IoT-Based Elderly Care and Decision-Critical Monitoring

The landscape of remote elderly care has been fundamentally reshaped by the proliferation of Internet of Things (IoT) technologies, and continuous, non-invasive monitoring is enabled through the seamless integration of wearable and ambient sensor networks [11]. As illustrated in Figure 1, a diverse ecosystem of sensing modalities is leveraged by these systems—including inertial measurement units (IMUs), physiological monitors, environmental sensors, and contactless motion detectors—and a comprehensive digital representation of an elderly individual's daily activities, mobility patterns, and vital signs within their home environment is created [12]. By facilitating real-time data acquisition and transmission to cloud or edge computing platforms, aging-in-place has been transformed from a conceptual ideal into a technically feasible model by IoT architectures, and institutional dependency is reduced while safety oversight is maintained [13].

Within this technological framework, two fundamentally distinct categories of safety-critical decisions require support from monitoring systems [14]. Unique operational requirements and risk profiles are associated with each category. Differentiated governance strategies are demanded by these differences—a distinction that has often been overlooked in homogeneous system designs.

Acute decisions, exemplified by real-time fall detection, represent the highest-stakes category in elderly monitoring, and immediate intervention with near-zero latency tolerance is demanded [15]. It has been suggested in the clinical literature that response times exceeding 10–15 min significantly increase morbidity and mortality following falls, imposing stringent temporal requirements for detection systems [16]. Consequently, sensitivity is deliberately prioritized over specificity in existing systems, and higher false-positive rates are accepted to prevent catastrophic missed detections—a risk calculus that is necessary given that complications, including hypothermia, dehydration, and pressure injuries, can result from unattended falls within hours [17]. Extreme timeliness is emphasized in operational requirements, with 1–5 s windows being processed by detection algorithms and sub-3 s response latencies being required for effective emergency response [18]. An inherent tension between responsiveness and reliability is faced in the governance of acute decisions, as either over-suppression of legitimate alerts or excessive false triggers that contribute to alarm fatigue is risked by traditional threshold-based approaches [19]. These challenges are compounded in real-world deployments through sensor removals, occlusions, and communication dropouts that can occur during critical moments [20].

In contrast, cumulative decisions, such as the early identification of gradual health deterioration, are derived from longitudinal analyses of behavioral and physiological trends over extended periods (days to weeks) [21]. Stability, specificity, and trend consistency are emphasized in these decisions, as slower-paced clinical interventions and care plan adjustments are informed by them [22]. Unlike acute judgments, aggregated evidence is deferred to by cumulative decisions, and time for data quality verification and cross-sensor validation is allowed. However, distinct challenges are faced: oscillating alert states that are caused by transient quality fluctuations can confuse caregivers and erode trust, while critical interventions are delayed by late detection of deterioration patterns. This distinction is illustrated in Figure 1 in the decision output layer, where acute alerts are caused to flow directly to caregivers while cumulative health reports are routed to clinicians for scheduled review, and their fundamentally different temporal and risk profiles are reflected.

Despite their differences, both decision types are used in inherently dynamic, noisy, and unreliable environments [3]. Sensor data streams are persistently threatened by a multitude of degradation factors: sensor noise from motion artifacts or low-cost hardware; communication dropouts due to wireless interference or network congestion; device heterogeneity leading to calibration mismatches; and environmental interference, such as occlusion, lighting variations, or electromagnetic disturbances [6]. Significant uncertainty is introduced into safety-critical decision pipelines by these degradation sources, where false alarms, missed detections, or unstable alerting behavior can be caused to cascade by data imperfections; each of these carries substantial clinical and psychological risks [7].

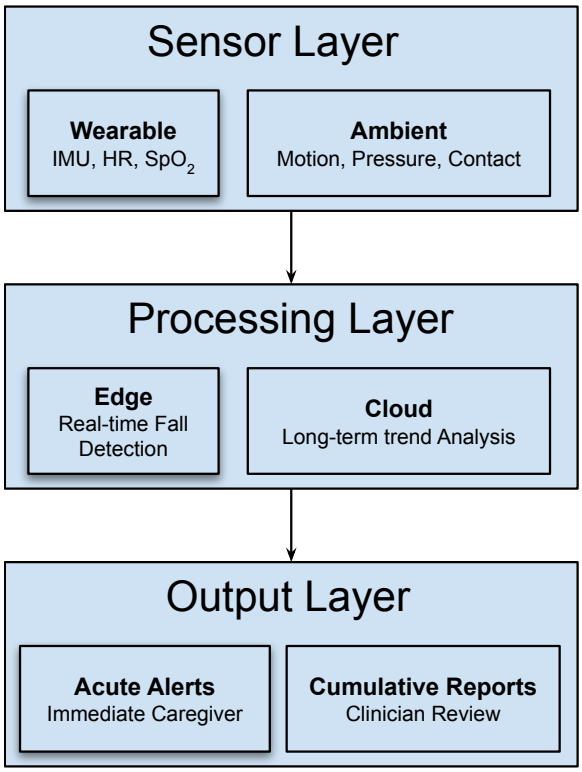


Figure 1. IoT-based elderly care monitoring architecture. Three-layer system architecture showing wearable/ambient sensors, edge/cloud processing, and decision outputs (acute alerts, cumulative reports). The multi-modal, multi-timescale nature of monitoring necessitates differentiated quality governance for acute versus cumulative decisions.

The critical need for decision-aware data quality governance is underscored by the gap between controlled laboratory validations and real-world deployment outcomes [23]. Governance mechanisms that do not merely assume data availability implies decision reliability, but instead explicitly evaluate whether prevailing data quality conditions justify executing a safety-critical action, are necessitated by this contextual unpredictability [4]. The central premise of the DQC ideology is directly motivated by this observation: that analytical correctness alone is insufficient grounds for action without concomitant evidence of data quality sufficient to the decision's risk profile. Distinct data handling and governance strategies are further necessitated by the temporal, operational, and risk-profile differences between acute and cumulative decisions—a requirement by which the asymmetric enforcement concept within the DQC framework is motivated.

2.2. Advanced Detection Approaches: Deep Learning, Explainable AI, and Transformers

Beyond threshold-based and classical machine learning methods, the accuracy of fall detection and health monitoring systems has been markedly improved by recent advances

in deep learning. Convolutional neural networks (CNNs) and long short-term memory (LSTM) networks have been widely adopted for wearable sensor-based fall detection. Structured features are automatically extracted from accelerometer and gyroscope data without manual feature engineering [10,24]. Attention systems have been incorporated in more recent architectures. The contribution of different sensor modalities and temporal segments is weighed by these mechanisms. For instance, acceleration and gyroscope data are processed in parallel by the dual-stream CNN self-attention (DSCS) model via CNN streams and a self-attention module. An accuracy of 99.32% on the SisFall benchmark dataset is attained by this approach [24,25]. Similarly, 99.38% accuracy has been demonstrated by feed-forward neural networks optimized for edge deployment, while inference times as low as 25 ms are maintained on limited-resource microcontrollers [26].

Within vision-based approaches, the YOLO (You Only Look Once) family of object detection models has been adapted for instant human-fall detection. YOLOv8-based architectures have been evaluated for fall detection performance over various model sizes, and it has been shown by these studies that high accuracy can be achieved by modern single-stage detectors while real-time inference speeds appropriate for deployment on edge devices are maintained [27]. Building on this, efficient attention modules are incorporated by HFD-YOLO into the YOLO network. Specifically designed for one-stage human-fall detection, greater sensitivity to falling postures is offered [28]. Similarly, transfer learning-based 3D CNNs have been applied to extract spatiotemporal features from video data for fall detection, and the breadth of deep learning's applicability across sensing modalities has been further demonstrated [29].

In parallel, explainable artificial intelligence (XAI) has become critical in healthcare, where transparency in automated decision making is demanded by stakeholders. Interpretable rationales for predictions are provided by XAI-enabled systems, and the "black box" limitation of traditional deep learning models is thus addressed [9,11]. In IoT-based monitoring, caregiver trust is enhanced by XAI through the clarification of why specific alerts are generated or suppressed—and the audit-trail goals of the proposed DQT framework are directly supported.

Most recently, transformer architectures, originally developed for natural language processing, have been adapted for healthcare temporal data analysis and multi-modal data fusion. The local feature extraction strength of CNNs and the global context representation capability of self-attention modules are combined in hybrid CNN–transformer models. Traditional baselines in disease prediction tasks are outperformed by these combined models [30]. Heterogeneous data sources, such as electronic health records and medical imaging, are integrated by multi-modal transformer frameworks via cross-attention fusion modules, and robust early disease prognosis is enabled by this integration [31]. In parallel, transformer-based contextual embedding models have been developed to retrieve actionable information from unformatted clinical narratives [32]. Collectively, a growing trend toward architectures capable of encoding complex, extended dependencies in healthcare data is indicated by these advances.

It should be noted that the DQT framework proposed in this paper is not intended as a replacement for these detection algorithms. Instead, it is designed to operate as a complementary governance layer. The question of what inference to make from available data is tackled by deep learning models, transformers, and XAI techniques. The prior question of whether the available data quality is sufficient to justify executing any inference at all is addressed by DQT. By this separation of concerns—between analytics and admissibility—it is implied that decisions made by any detection algorithm can be governed by DQT. The advanced deep learning and transformer-based approaches surveyed here are included in this scope.

2.3. Data Quality in IoT and Healthcare Systems

Data quality represents a central barrier to trustworthy analytics in IoT-enabled healthcare. Quality is defined by established frameworks in terms of accuracy, completeness, timeliness, and consistency. The reliability of downstream clinical and operational decisions is influenced by each dimension [5]. In elderly monitoring, data streams are especially vulnerable to degradation. Risk is added by the mobile, non-clinical, and unpredictable nature of in-home deployments. Persistent noise, loss, and latency are introduced by sensor placement, user compliance, and environmental variability [6].

The four primary degradation types affecting IoT elderly care systems are illustrated in Figure 2: sensor noise (inaccurate readings), data loss (missing samples), latency (delayed transmission), and inconsistency (conflicting measurements across sensors). These degradations can be traced to multiple sources, specifically hardware limitations, communication interference, environmental factors, and user behavior.

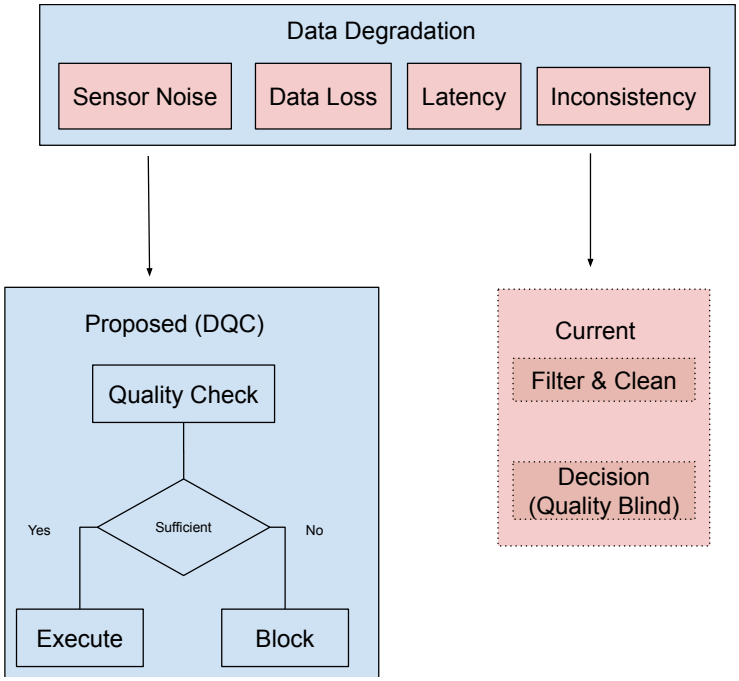


Figure 2. Data quality governance—current approach vs. DQC. Left: Common causes of data degradation (noise, loss, latency, inconsistency). Top right: Current preprocessing filters data but produces quality-blind decisions. Bottom right: DQC explicitly evaluates quality against decision-critical thresholds and executes decisions only when quality is sufficient for the risk profile.

The impact of these degradation types on clinical outcomes is well documented in the literature. It has often been shown by studies that the number of false alarms or missed problems can be quickly raised by even small drops in data completeness or speed. For example, if 10% less data is collected by sensors, false alarms in fall detection can be raised by 40%. If alerts are delayed by over 5 s, assistance in emergencies is no longer provided. The system's performance is degraded by these problems, and trust is lost by caregivers and users—a major safety concern.

To address these challenges, current approaches predominantly treat data quality as a preprocessing or filtering concern, focusing on improving aggregate metrics through imputation, smoothing, or sensor fusion, rather than explicitly governing whether a specific decision should be executed under degraded conditions [4]. As shown in Figure 2, a linear pipeline is followed by this approach: raw data → filtering → cleaned data → quality-blind decision execution. There are four fundamental shortcomings of this reactive, quality-agnostic paradigm:

- Context insensitivity: The same data rules are applied to all decisions, even though an unsafe condition for urgent fall detection is created by missing data that is considered acceptable for long-term analysis.
- Stationarity assumption: Patterns are assumed to be followed by problems, but unpredictable changes in data quality are experienced in real-world IoT setups.
- Lack of decision awareness: Data quality is measured in isolation (such as "85% complete") without consideration of the intended use of the data.
- Preprocessing limitations: Signal quality can be improved by even advanced techniques, such as Kalman filtering [33,34]. However, verification that the processed data meets decision-specific requirements is not provided by these techniques. As a result, vulnerability to acting on filtered but still unreliable evidence is retained by systems.

The need for context-aware quality assessment becomes even more apparent in multimodal sensor environments. Here, different failure modes are exhibited by different sensing technologies: Motion artifacts are suffered by wearable monitors during high activity; ambient sensors can be occluded; seasonal drift is exhibited by environmental sensors; and differential interference patterns are experienced by communication networks [34]. An asynchronous occurrence of these failures is noted, and context-aware quality assessment, by which evidence from remaining functional sensors can be weighed, is required—a capability that is absent in current approaches.

To overcome these limitations, the new DQC approach for controlling quality is shown in Figure 2. Unlike previous systems, a check of whether the data is sufficient is performed by DQC before a decision is made. After checking the quality, there is a decision gate: "Is the data good enough for the risk?" Only if the set threshold is passed by the data is a decision made by the system; otherwise, a pause or halt is enacted. An active role is taken by this new approach, and whether the data supports the decision's risk is checked, ensuring decisions are made only when safety is supported.

2.4. Beyond Stationarity: Concept Drift and Adaptive Governance

A fundamental challenge in data quality management for IoT elderly care is the non-stationarity of both sensor behavior and user patterns. Over extended monitoring periods, the joint distribution of sensor readings and clinically relevant events is caused to evolve by gradual sensor calibration drift, battery depletion, seasonal changes in user activity, and progressive decline in user mobility or cognitive function. Static quality thresholds are rendered increasingly ineffective over time by this phenomenon—known as concept drift in the online learning literature [35,36]. Either an overly conservative or dangerously permissive state is reached by a threshold optimized for a user's initial mobility level: the former if the user declines, the latter if sensor performance degrades.

Online learning techniques, in which prediction models are continuously updated based on incoming data streams, are typically employed by existing approaches to handling non-stationarity [37,38]. Predictive accuracy in changing environments is maintained by methods such as incremental decision trees, sliding window adaptation, or drift-detection-triggered retraining. However, analytical model adaptation—keeping decision analytics current—is focused on, rather than governance adaptation, which is concerned with when decisions should be admissible at all.

Resource-constrained IoT environments present additional challenges for governance frameworks. The limited computational, memory, and energy resources of edge devices necessitate lightweight governance mechanisms that do not compromise safety. Recent work on lightweight blockchain-based trust mechanisms for resource-constrained IoT networks [39] and reputation-driven mechanisms for cross-chain IoT data sharing [40] demonstrate the feasibility of deploying sophisticated trust and governance protocols on edge

hardware. These approaches inform the design of DQT, which achieves sub-millisecond traversal times and minimal energy consumption (Section 4.3) while maintaining comprehensive governance capabilities.

A distinct position relative to pure online learning is occupied by the DQT framework. While inference models are continuously updated by online learning to track changing user patterns, the admissibility criteria that determine whether those inferences should be acted upon are governed by DQT. A deliberate choice is reflected in this separation: even with perfectly adapted analytics, a decision is unsafe if the supporting data quality is insufficient for the current context. Conversely, during periods of rapid drift, safety coverage can be maintained by DQT's fallback pathways and context-aware threshold adaptation even before analytics models have converged.

It is planned to integrate online learning mechanisms for threshold optimization into future extensions of the DQT framework—with reinforcement learning being used to dynamically adjust adaptation rules based on observed outcomes. The stability and verifiability of contract-based governance and the adaptability of online learning are combined by this hybrid approach, and concept drift is addressed without the transparency needed for safety-critical applications being sacrificed.

2.5. Governance and Trust in Safety-Critical IoT Systems

Beyond mere data management, the role of data governance in digital health has been expanded to emphasize accountability, transparency, and risk-aware decision support in increasingly autonomous monitoring systems [4]. In safety-critical domains such as elderly fall monitoring, benefit is derived by stakeholders—including caregivers, clinicians, family members, and regulatory bodies—not only from accurate system performance but also from a clear, interpretable rationale for system actions, especially when alerts are suppressed or automated decisions override human judgment.

This need is motivated by both ethical considerations (respect for patient autonomy and informed consent) and practical requirements. System behavior needs to be understood by caregivers to respond appropriately, while verifiable safety assurances for certification are demanded by regulators [11]. Responsibility matrices are shifted by the transition from traditional clinician-in-the-loop monitoring to automated IoT systems, and new governance challenges around decision provenance, error attribution, and system accountability are created in environments where human oversight is intermittent or delayed [4].

The fundamental audit trail gap in current threshold-based governance approaches is illustrated in Figure 3. Binary admit/block decisions with no accompanying explanation are produced by these systems—typically only a generic "insufficient quality" message when alerts are suppressed. Trust can be undermined and practical utility limited by this opacity, and what researchers term the "black box safety dilemma" is created: systems that are statistically safer than human monitoring alone but whose safety reasoning cannot be articulated in human-understandable terms [11]. Sensor fusion techniques such as Kalman filtering [33,34] are also affected by this opacity, and despite their mathematical sophistication, a black box mode of operation is exhibited, with no explanation being provided of why a particular measurement was accepted or rejected.

When a fall alert is blocked due to "insufficient data quality", no information is received by stakeholders about which sensors failed, by how much the thresholds were missed, whether alternative evidence was considered, or what specific risks justified suppression. Both caregiver confidence and regulatory compliance can be undermined by this explanatory deficit, as meaningful oversight, debugging, and continuous improvement are prevented [11].

Rather than being binary, trust in safety-critical systems exists on a continuum that is influenced by multiple factors [11]: transparency (understanding how decisions are made), predictability (consistent behavior under similar conditions), competence (demonstrated reliability), and benevolence (alignment with user values). The capabilities required to address these trust dimensions are previewed in Figure 3, including documented decision paths, human-readable explanations, and audit trails, by which retrospective analysis and regulatory compliance are enabled.

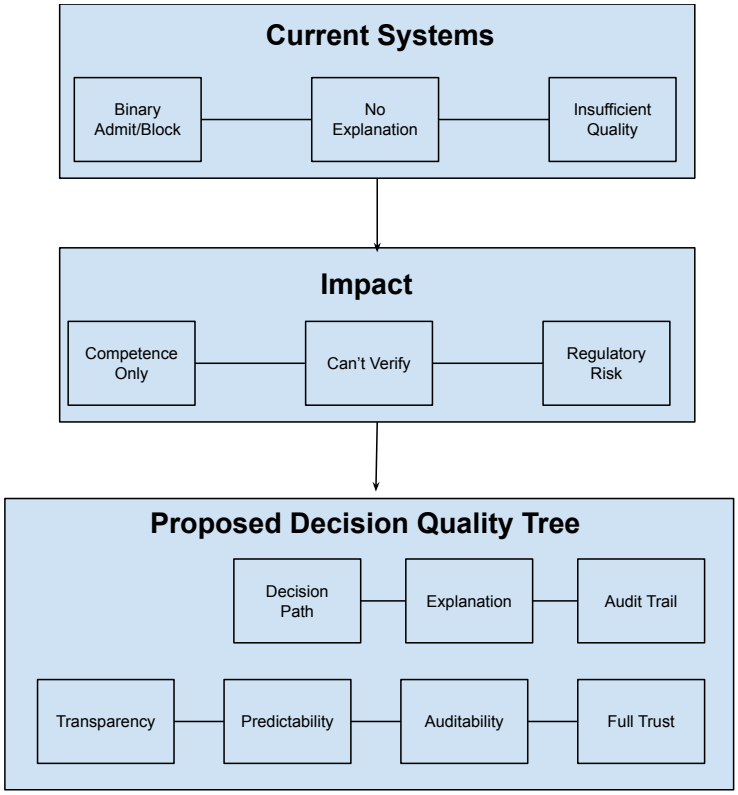


Figure 3. Governance and trust—current gap vs. DQT solution. Current systems produce binary decisions without explanation, yielding "competence without trust". DQT generates documented decision paths, explanations, and audit trails, enabling transparency and predictability necessary for full stakeholder trust.

Only competence is addressed by current threshold-based governance approaches, while the other dimensions are neglected, and what human-factor researchers describe as "competence without trust" is created—technically capable systems upon which stakeholders hesitate to rely due to incomprehensible behavior [11]. Excellent accuracy metrics can be possessed by such systems, but if the reasons for occasional alert suppression cannot be understood by caregivers, their willingness to rely on them during critical moments is reduced. Movement beyond binary governance toward explainable, adaptive frameworks is required to address this trust deficit—frameworks by which reasoning can be articulated, trade-offs justified, and actionable insights provided even, and especially, when decisions are conservative or counterintuitive.

2.6. Identified Research Gaps and Shortcomings

Despite advancements in IoT-based elderly care monitoring and growing recognition of data quality as a critical concern, several fundamental gaps remain that current approaches fail to address. The need for a new governance paradigm is collectively motivated by these gaps, one that moves beyond threshold-based quality checks toward decision-centric, context-aware, and explainable governance architectures.

2.6.1. Flat-Threshold-Based Models Lack Representational Capacity

Flat-threshold-based models are relied upon by current implementations, and quality dimensions are evaluated against static, pre-defined minima using simple Boolean logic ($Q(t) \geq \theta$). While basic safety guarantees are provided by this approach, the capacity to model the complex relationships between data quality, context, and decision risk is lacking. Partial-sensor-failure scenarios, where some modalities remain functional, cannot be accommodated by these rigid models, nor can adaptation to dynamic context shifts, such as caregiver availability, time-of-day risk patterns, or evolving user mobility profiles, be achieved [4]. Nuanced risk–benefit trade-offs are also not supported—for instance, slightly lower data completeness being accepted during nighttime hours when fall risk is elevated—judgments for which reasoning beyond binary thresholding is required [11].

2.6.2. No Graceful Degradation Pathways

No graceful degradation mechanisms are provided by current frameworks when primary sensing modalities are compromised, and blanket decision suppression is defaulted to, by which vulnerable individuals are left unmonitored during precisely the periods when oversight is most needed [10]. For example, if a wearable accelerometer fails but ambient sensors remain operational, all fall detection decisions are typically blocked by threshold-based systems, rather than degraded-but-acceptable monitoring being established through sensor fusion or confidence-weighted analysis.

2.6.3. Absence of Context-Aware Adaptation

Mechanisms for context-aware adaptation to temporal factors (time of day, seasonality), behavioral patterns (sleep cycles, activity routines), or environmental conditions (lighting, occupancy) that significantly influence both data quality interpretation and appropriate safety thresholds are lacking in existing systems [4]. Fundamentally different monitoring contexts and risk profiles are not recognized by a system applying identical completeness thresholds at 3:00 a.m. (when movement is minimal) and 3:00 p.m. (during active hours).

2.6.4. No Personalization to Individual Risk Profiles

Barriers to essential safety features are created by the rigid condition-check structure of existing approaches [4]. Emergency override mechanisms to lower thresholds after near-fall events cannot be easily incorporated into their simplistic architecture. Sensor fusion logic in which evidence is weighted by reliability is also difficult to implement. In addition, pathways for incremental learning from feedback, by which thresholds could be adjusted by systems based on false-positive or false-negative patterns and changing user behavior, are lacking in these approaches [4].

2.6.5. Preprocessing Techniques Do Not Address Governance

A common misconception in IoT systems is that improving input data quality through techniques such as Kalman filtering [33,34] or imputation methods is sufficient to ensure reliable decisions. However, only the analytics layer is addressed by these approaches—cleaner, more complete data are produced—but the governance question is not addressed: given the current quality of the processed data, is it safe to execute a decision? Smoothness is exhibited by a Kalman-filtered signal, but if it is based on prolonged prediction without measurements, its reliability for safety-critical judgments remains unknown. A persistence of this gap between data processing and decision governance is observed across all preprocessing-centric approaches.

2.7. Comparative Analysis of Existing Approaches

To contextualize the novel contributions of the DQC ideology and DQT framework, a systematic comparison of existing data quality governance approaches against the capabilities essential for reliable IoT elderly care deployment is provided in Table 1.

Each approach is evaluated by the comparison across five criteria derived from the limitations identified in the preceding sections: handling of partial sensor degradation, context awareness, graceful degradation under stress, generation of auditable decision trails, and explicit separation of analytics from admissibility.

Kalman filtering is a widely adopted sensor fusion technique by which the true system state is estimated from noisy measurements [33,34]. While partial degradation is handled through prediction steps and graceful degradation is offered via covariance-based weighting, context awareness is lacking, no audit trail is produced, and analytics are not separated from admissibility—input data quality is merely improved.

As shown in Table 1, at most two of these criteria are satisfied by existing approaches, whereas all five are uniquely combined by the proposed DQT framework, and it is positioned as a comprehensive governance solution for safety-critical monitoring environments.

Table 1. Comparative analysis of data quality governance approaches for IoT elderly care.

Approach	Representative Works	Handles Partial Degradation	Context-Aware	Graceful Degradation	Audit Trail	Separates Analytics from Admissibility
Rule-based thresholds	[4]	No	No	No	No (binary output only)	No
Static decision trees	[9,10]	No (single path)	No	No	Partial (path visible)	No
Kalman filtering/ sensor fusion	[33,34]	Yes	No	Yes	No	No
Simple ML classifiers	[14,41]	No	No	No	No (black box)	No
Online learning/ adaptive models	[37,38]	Yes (incremental)	Partial (sliding window)	Yes	No	No
Symmetric DQC (threshold-based)	This work (baseline)	No	No	No	No	Yes
Asymmetric DQC (adaptive thresholds)	This work (baseline)	No	Yes (basic)	No	No	Yes
DQT framework (Ours)	This work	Yes (multi-path)	Yes (rich context)	Yes (fallback paths)	Yes (audit trails)	Yes

2.8. Areas of Improvement

A clear imperative is established by the research gaps identified in the preceding sections: existing threshold-based approaches to data quality governance are fundamentally inadequate for the complexities, uncertainties, and safety demands of real-world IoT elderly care deployments. Not incremental refinement of existing methods, but a fundamental re-imagining of how data quality governs decision admissibility is what is required. A direct response to this imperative is provided by the DQT framework.

Building upon the foundational principle that data quality must explicitly govern decision safety, simplistic flat-threshold checks are replaced in the DQT framework by a hierarchical, traversable tree structure in which complex reasoning pathways and context-aware evaluation logic are encoded. Governance is transformed by this structural innovation from a static checklist into a dynamic inference system by which the nuanced relationships between data quality, situational context, sensor reliability, and decision risk are capable of being modeled [4,9]. Unlike threshold-based approaches, in which uniform evaluation logic is applied regardless of circumstances, conditional branching, weighted evidence integration, and multi-path evaluation are introduced by DQT—architectural features that are essential for handling the uncertainty and unpredictability of in-home monitoring scenarios [10].

The DQT architecture enables several critical capabilities that are absent in existing implementations. First, multiple reasoning paths for different failure modes are encoded, and the system is allowed to pursue alternative evaluation strategies when primary data sources are compromised. For instance, if wearable accelerometer data is unavailable, a secondary branch in which ambient sensor data is evaluated with adjusted quality expectations can be transitioned to by the tree—a form of graceful degradation that is impossible in threshold-based systems [10].

Second, dynamic adjustment of enforcement logic based on the real-time system state and historical performance is supported by the framework, and context-sensitive governance that adapts to time of day, caregiver availability, recent event history, and learned user patterns is enabled [4]. Recognition that a user with declining mobility patterns needs progressively more sensitive detection thresholds over time is allowed by this adaptability.

Third, complete, auditable decision trails in which every node evaluation, branch selection, and evidence consideration along the decision path is documented are naturally generated by DQT's tree traversal [9,11]. When a fall detection decision is blocked, a detailed explanation specifying which sensors contributed, how their quality was assessed, what contextual factors were considered, and what evidence gaps prevented a confident decision can be produced by the system. A key limitation of current approaches—their inability to justify reasoning to human stakeholders—is addressed by this capability, while the audit trails necessary for regulatory compliance and continuous improvement are created [11].

Fourth, sophisticated fallback strategies and emergency protocols that are absent in threshold-based models are enabled by the framework, including graduated response levels (e.g., "high-confidence alert", "low-confidence notification") and context-aware override mechanisms, by which governance strictness can be temporarily adjusted during known high-risk periods [10].

By implementing rather than replacing the DQC paradigm, robust, transparent, and adaptable decision governance is delivered by the DQT framework while the core principle that data quality must explicitly govern decision safety is preserved [7,11]. The critical separation between decision analytics and admissibility is maintained by this architectural realization of the DQC ideology, while the representational capacity and reasoning capabilities of the governance layer itself are dramatically expanded [4]. The limitations of rigidity, context-insensitivity, and poor audit trails identified in prior work are directly addressed by the framework.

3. Problem Statement

The core challenge in IoT-based elderly care systems stems from the fundamental tension between decision urgency and data reliability in non-clinical environments, where sensor quality is inherently unstable. Formally, the following problem is addressed in this paper:

3.1. Given

1. A heterogeneous sensor network $S = \{s_1, s_2, \ldots, s_n\}$ with time-varying data quality $Q(t) = \langle q_{tim}(t), q_{comp}(t), q_{cons}(t), q_{rel}(t) \rangle$, where $q_* \in [0, 1]$ represent timeliness, completeness, consistency, and reliability dimensions, respectively [3,4]. Each quality component degrades according to realistic models capturing sensor drift, communication failures, environmental interference, and user non-compliance.

2. A decision space $D = \{d_a, d_c\}$ comprising acute decisions d_a (e.g., fall detection requiring immediate response) and cumulative decisions d_c (e.g., health deterioration alerts based on longitudinal trends), each associated with distinct risk profiles $R = \{r_a, r_c\}$, where r_a prioritizes sensitivity and low latency while r_c emphasizes specificity and stability [2,4].
3. Contextual features $C(t)$ capturing real-time situational, behavioral, environmental, and system-state information, including time of day, user location/activity patterns, sensor battery/connectivity status, caregiver availability, historical false-alarm rates, and individual risk factors (e.g., previous falls, mobility limitations) [4].
   Historical false-alarm rates represent the per-user frequency of false alerts generated by the detection system over a defined monitoring window (typically 7–30 days). These rates are maintained as exponentially weighted moving averages (EWMAs) within the reliability component $q_{rel}(t)$ of the quality vector, enabling the governance layer to track each user's false-alarm history over time. When a user exhibits persistently high historical false-alarm rates, the DQT framework adjusts the admissibility thresholds upward, requiring stronger evidence quality before admitting alerts. Conversely, for users with low historical false-alarm rates, the thresholds are relaxed, reflecting the system's calibrated trust in the detection pipeline for that individual.
   Individual risk factors capture patient-specific clinical and behavioral characteristics that influence the consequences of missed detections or false alarms. These factors include the following: (i) fall history (number of previous falls in the past 6–12 months), (ii) mobility impairment level (use of walking aids, gait instability), (iii) cognitive status (diagnosed cognitive decline affecting sensor compliance), (iv) living arrangement (living alone versus with a caregiver), and (v) medication profile (use of anticoagulants or sedatives that increase fall-related injury risk). These risk factors are encoded in the context vector $C(t)$ as discrete or continuous variables and are evaluated at decision time through conditional branching in the DQT. For example, a user with a high fall risk (previous falls > 2 in 6 months) and living alone triggers relaxed quality thresholds during nighttime hours, ensuring that the system remains available for fall detection even under moderate sensor degradation, whereas a low-risk user in an assisted-living facility with 24 h caregiver presence maintains stricter thresholds.
4. Historical operational data $H = \{(Q_i, C_i, d_i, y_i)\}_{i=1}^{m}$ containing ground-truth event labels $y_i \in \{\text{true event}, \text{false event}\}$ and corresponding system performance metrics [4,9].

3.2. Objective

Determine the optimal Decision Quality Tree (DQT) structures T_{d_a} and T_{d_c} and their corresponding traversal policies π_{d_a} and π_{d_c} that simultaneously:

1. Maximize detection of true acute events while maintaining controlled false-alert tolerance: formally, maximize $P(\text{detect} \mid d_a = \text{true})$ subject to $P(\text{false alert} \mid d_a) \leq \epsilon_a$, where ϵ_a represents the maximum acceptable false-positive rate for acute decisions [2,4,10].
2. Maintain high longitudinal stability for cumulative decisions: minimize temporal variance in admissible decision outcomes for d_c while preserving detection sensitivity for gradual deterioration patterns, formally minimizing $\text{Var}(A(d_c, t))$ over extended monitoring windows [4,8].
3. Provide human-interpretable explanations for all admissibility judgments: for every decision at time t, generate explanation $E_x(t)$ that maps $\{Q(t), C(t), T_d, \pi_d\}$ to natural-language or structured rationale documenting the complete decision pathway [9,11].

4. Outperform static symmetric and Asymmetric DQC models across comprehensive evaluation metrics including safety (missed detection rate), coverage (operational uptime), latency (decision response time), stability (temporal consistency), and trust (explanation quality and stakeholder confidence) [4,7,8].

3.3. Constraints

- Safety preservation: All adaptations must satisfy $P(\text{missed detection} = \text{true}) \leq \delta_{critical}$, where $\delta_{critical}$ represents the maximum acceptable risk threshold for life-threatening events.
- Computational feasibility: Tree traversal and evaluation must complete within decision-specific latency bounds: $t_{process}(d_a) \leq \tau_a$ (typically $\leq$3 s) and $t_{process}(d_c) \leq \tau_c$ (typically $\leq$30 s).
- Resource limitations: The solution must operate within the memory, processing, and energy constraints of edge computing environments typical in elderly care deployments.
- Regulatory compliance: All decision logic must be auditable and verifiable against healthcare IoT safety standards and data governance regulations [11].

This problem formulation captures the essential trade-offs that any decision-critical quality governance framework must address: balancing sensitivity and specificity, acute responsiveness and cumulative stability, audit trail and computational efficiency, all while operating under the inherent uncertainty of real-world sensor degradation. The DQT framework is designed specifically to satisfy these objectives and constraints through its hierarchical, context-aware architecture.

4. System Model and Architecture

4.1. Overall System Architecture

The proposed DQC IoT-based elderly care monitoring system is designed as a multi-layered architecture that explicitly separates data acquisition, analytical inference, quality governance, and decision execution.

As illustrated in Figure 4, the system operates through four interconnected layers that collectively ensure safety-critical decisions are made only when supported by sufficiently reliable sensor evidence. The sensor layer (top) aggregates heterogeneous data from wearable, ambient, and environmental sensors, each subject to realistic degradation patterns. The data processing layer computes a quality vector $Q(t)$ across timeliness, completeness, consistency, and reliability dimensions, while simultaneously extracting contextual features $C(t)$, including time-of-day, activity patterns, and individual risk factors. The DQC governance layer implements the core contribution of this work: the Decision Quality Tree (DQT) traverses hierarchical conditions to determine decision admissibility. Unlike threshold-based approaches, DQT supports multiple reasoning paths—primary, fallback, and context-adaptive—enabling graceful degradation when sensors fail. Each traversal generates a complete audit trail documenting the decision pathway. The execution layer (bottom) routes admissible decisions through differentiated pipelines: acute alerts for fall detection (with confidence-based escalation) and cumulative notifications for health deterioration (with clinician review pathways).

4.2. Sensor Layer

The sensor layer comprises heterogeneous IoT devices deployed both on the elderly individual and within their living environment.

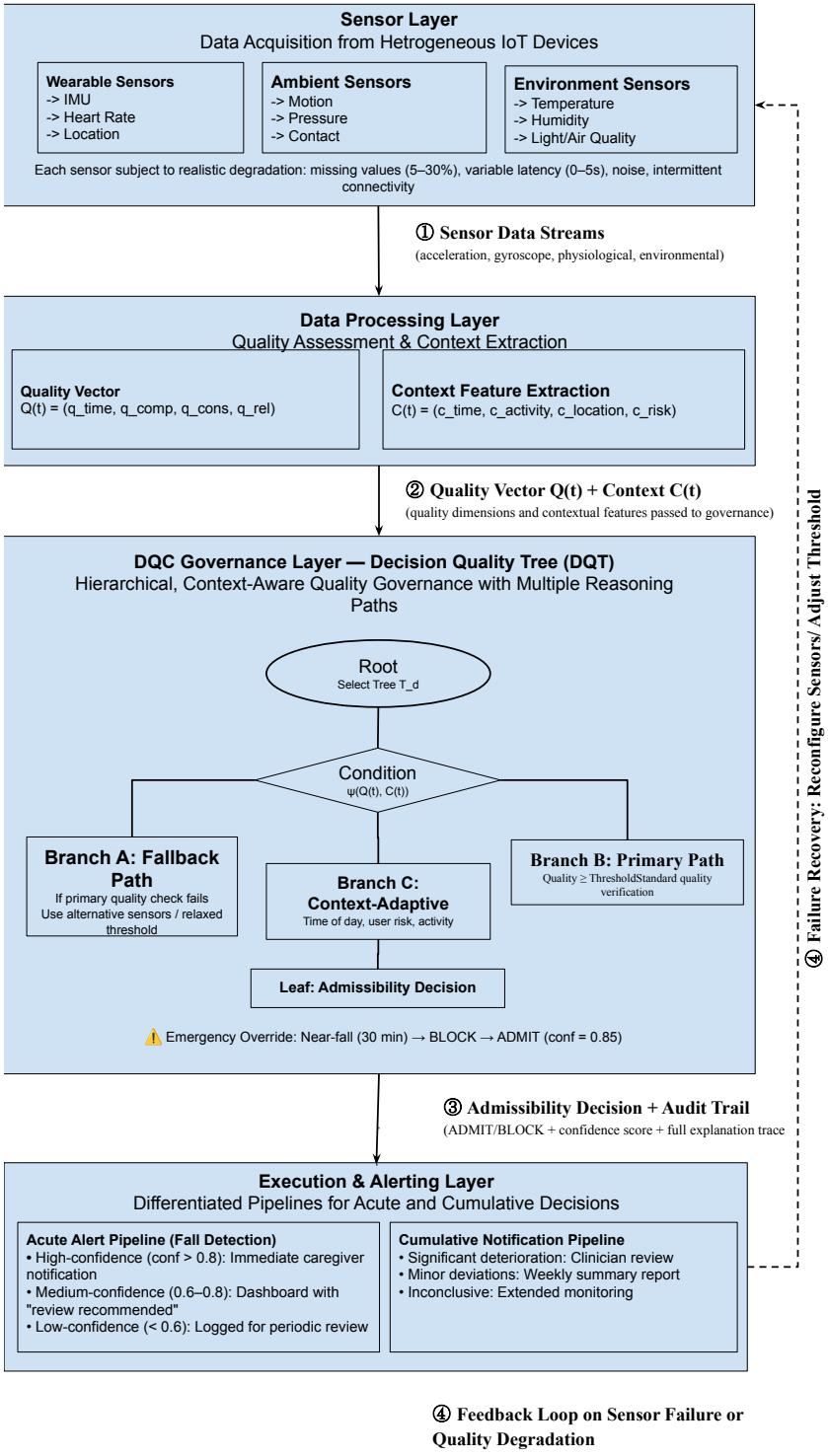


Figure 4. Four-layer system architecture for DQC-based elderly care monitoring. The sensor layer includes wearable sensors (IMU, heart rate, location), ambient sensors (motion, pressure, contact), and environmental sensors (temperature, humidity, light). The data processing layer computes quality vectors $Q(t)$ and extracts contextual features $C(t)$. The DQC governance layer manages the Decision Quality Tree (DQT), which includes multiple reasoning paths—primary, fallback, and context-adaptive—that lead to admissible or blocking decisions, with full audit trails. The execution layer routes admissible decisions via differentiated pipelines: acute alerts for fall detection, or cumulative notifications for health deterioration. Arrows show data flow; dashed lines show feedback for adaptation.

4.2.1. Sensor Degradation Model

Sensor data quality in actual IoT elderly care deployments is affected by three primary degradation mechanisms. Each mechanism is anchored in empirical studies of home-based monitoring systems [3,6,42]:

- Missingness due to random and burst data loss, sensor disconnections, wireless interference, battery depletion, and user removal of wearables. Experimental missingness rates of 5% (controlled), 15% (realistic), and 30% (stress) reflect reported field study ranges.
- Latency results from variable transmission delays caused by network traffic and edge processing backlogs. Decision-specific bounds are applied: $\tau_{max} = 3$ s for acute fall detection and $\tau_{max} = 30$ s for cumulative health monitoring [18].
- Noise and inconsistency: Gaussian noise captures motion artifacts and electrical interference. Temporal drift models sensor decalibration. Cross-sensor inconsistency occurs when correlated sensors produce contradictory readings due to placement errors or hardware faults.

Degradation patterns differ across modalities and individuals. Wearable IMUs are prone to motion artifacts and user removal. Ambient sensors can be occluded. Environmental sensors exhibit gradual drift. An active user creates more motion artifacts but complies better with device use. A cognitively impaired user removes sensors frequently. The DQT framework handles this heterogeneity through multi-modal sensor fusion with fallback paths (Section 4.5), context-aware threshold adaptation, and user-specific reliability tracking via the EWMA formulation of q_{rel} (Section 4.3).

These degradation models are empirically grounded but simplify real-world complexity. The actual environment includes non-stationary trends and relationships between degradation types. The stress scenario (30% missingness) exceeds typical rates to assess worst-case robustness. Field validation is still necessary, as discussed in Section 7.3. The degradation assumptions directly inform DQT threshold calibration: completeness under stress clusters near 0.70 (Section 6.2). This motivates differentiated thresholds across governance strategies. The multi-path tree structure maintains coverage under degraded conditions by using alternative evidence sources, rather than simply relaxing thresholds.

4.2.2. Wearable Sensors

- Inertial measurement units (IMUs): 3-axis accelerometers and gyroscopes sampling at 20–100 Hz for motion analysis and fall detection.
- Physiological monitors: Heart rate sensors, pulse oximeters (SpO_2), and skin temperature sensors, providing vital sign monitoring.
- Location/proximity: Bluetooth beacons or ultra-wideband tags for room-level localization.

4.2.3. Ambient Sensors

- Motion detectors: Passive infrared (PIR) sensors in each room for activity pattern analysis.
- Pressure sensors: Floor mats, bed/chair occupancy sensors for mobility and rest patterns.
- Contact sensors: Door/window sensors for activity correlation and safety monitoring.

4.2.4. Environmental Sensors

- Temperature;
- Humidity;
- Light;
- Air quality.

Each sensor ($s_i \in S$) produces a data stream subject to realistic degradation: missing values (5–30%), variable latency (0–5 s), noise artifacts, and intermittent connectivity [3]. Sensor reliability profiles ($r_i(t) \in [0,1]$) are tracked over time based on historical performance, battery levels, and self-diagnostic information.

4.3. Data Processing and Quality Assessment Layer

Raw sensor data undergoes parallel processing streams.

4.3.1. Quality Vector Computation

For each time window W, the system computes

$$Q(t) = \langle q_{tim}(t), q_{comp}(t), q_{cons}(t), q_{rel}(t) \rangle$$

where

- Timeliness q_{tim}: Ratio of data points arriving within expected latency bounds. Formally, $q_{tim} = |\{s_i \in S : \text{latency}(s_i) \leq \tau_{max}\}|/|S|$, where $\tau_{max} = 3$ s for acute decisions and $\tau_{max} = 30$ s for cumulative decisions.
- Completeness q_{comp}: Percentage of expected samples received. Formally, $q_{comp} = (1/|S|) \times \sum_{i=1}^{|S|}(N_{received,i}/N_{expected,i})$, where $N_{expected,i} = T \times f_i$ for sampling rate f_i over window duration T.
- Consistency q_{cons}: Agreement between correlated sensor readings. Formally, $q_{cons} = \max(0, 1 - \min(1, CV/\theta_{cv}))$, where $CV = \sigma/\mu$ is the coefficient of variation across normalized sensor values and $\theta_{cv} = 5.0$ is a scaling factor.
- Reliability q_{rel}: Historical accuracy and stability score for each sensor. Formally, $q_{rel} = (1/|S|) \times \sum_{i=1}^{|S|} r_i^{(t)}$, where $r_i^{(t)} = (1-\alpha) \times r_i^{(t-1)} + \alpha \times \text{accuracy}_i^{(t)}$ with learning rate $\alpha = 0.1$.

The minimum quality thresholds applied to acute (fall) and cumulative (health) decisions are summarised in Table 2.

Table 2. Quality threshold by decision type.

Cumulative (Health)	Acute (Fall)	Quality Dimension
$\geq$0.70 ($\tau \leq 30$ s)	$\geq$0.85 ($\tau \leq 3$ s)	Timeliness
$\geq$0.85	$\geq$0.70	Completeness
$\geq$0.75	$\geq$0.60	Consistency
$\geq$0.70	$\geq$0.55	Reliability

4.3.2. Context Feature Extraction

Simultaneously, contextual features are extracted:

$$C(t) = \langle c_{time}, c_{activity}, c_{location}, c_{system}, c_{risk} \rangle$$

including time of day, recent activity levels, current location, sensor health status, and individual risk factors [4].

4.4. Decision Analytics Layer

This layer operates independently from quality considerations, focusing purely on event inference.

4.4.1. Acute Decision Models

- Fall detection: Machine learning classifiers (SVM, Random Forest, CNN) processing 1–3 s windows of accelerometer/gyroscope data [14,41].

- Output: Raw fall probability $p_{fall}(t) \in [0, 1]$ with confidence score.

4.4.2. Cumulative Decision Models

- Health deterioration: Time-series analysis (ARIMA, LSTM) of daily activity levels, mobility patterns, and vital sign trends over 24–72 h windows [15].
- Output: Trend deviation scores and anomaly flags with longitudinal confidence.

These models produce raw decision outputs $D_{raw}(t)$ that are forwarded to the DQC layer without any quality-based filtering or suppression.

4.5. DQC Governance Layer

The DQC governance layer implements the DQC ideology via the following components:

- DQTs: Hierarchical structures T_{d_a}, T_{d_c} encoding conditional evaluation logic;
- Context Evaluator: Processes $C(t)$ to determine relevant contextual factors;
- Quality assessor: Monitors $Q(t)$ and computes compliance with contract terms;
- Traversal engine: Executes tree traversal based on current conditions;
- Explanation generator: Creates human-interpretable decision justifications.

4.5.1. Tree Structure

Each DQT node contains the following:

- Condition: Boolean expression over $Q(t), C(t)$ and decision metadata;
- Children: Pointers to next nodes based on condition evaluation;
- Action: For leaf nodes, specifies admissibility decision with confidence score;
- Explanation template: For leaf nodes, provides structure for rationale generation.

4.5.2. Traversal Process

Given decision type $d \in D$, quality vector $Q(t)$, and context $C(t)$, the DQT traversal proceeds as follows:

1. Tree Selection

Select the decision tree T_d that matches the given decision type. The system maintains separate trees for acute decisions (e.g., fall detection) and cumulative decisions (e.g., health deterioration monitoring). These trees account for different risk profiles and quality needs.

2. Root Node Initialization

Start at the root node v_{root} of the chosen tree. Create an empty decision trace to record each evaluation as it occurs during traversal. This trace builds the audit trail and supports explainability.

3. Recursive Node Evaluation

While the current node v is not a leaf node, perform the following substeps.

a. Condition Evaluation

Check the condition ψ_v at the current node using $Q(t)$ and $C(t)$. Conditions include the following:

- Quality check: Compares a certain quality aspect (timeliness, completeness, consistency, or reliability) to a set threshold (e.g., $q_{comp} \geq 0.70$).
- Context check: Evaluates a contextual variable (time of day, user risk level, caregiver availability, or recent near-fall status) against an expected value (e.g., user_risk = HIGH).
- Composite condition: Combines multiple sub-conditions using logical operators (AND, OR) to express complex governance logic.

b. Branch Transition
Depending on the result, move to the correct child node. If the condition is TRUE, follow the branch for successful verification (usually moving toward admissibility). If FALSE, take the other branch (often leading to fallback logic or a blocking decision). Note both the outcome and the path taken in the decision trace.

4. Leaf Node Resolution

When a leaf node v_{leaf} is reached, the traversal terminates. The leaf node specifies the following:

a. Admissibility Action
Execute the action $a = \text{action}(v_{leaf})$, which is either ADMIT (allowing the clinical alert to proceed) or BLOCK (suppressing the alert due to insufficient evidence quality).

b. Confidence Computation
Calculate a confidence score, $\text{conf} = \text{confidence}(v_{leaf}, Q(t), C(t))$, which shows how certain the system is about the admissibility decision. This score considers the quality factors found during traversal and the contextual risk factors.

c. Explanation Generation
Write a human-readable explanation $E_x = \text{explain}(v_{leaf}, \text{path})$ that shows the full reasoning. It includes (i) the final decision and confidence score, (ii) the conditions checked, with values and thresholds, (iii) the path taken and why, and (iv) any emergency overrides or special factors influencing the result.

5. Emergency Override Protocol

Before finishing, look for emergency override conditions that override standard logic. If a near-fall happened in the past 30 min, any BLOCK becomes ADMIT with a confidence of 0.85. This ensures the system stays available when there is a high fall risk. Other override cases are caregiver unavailability and sensor battery failure. All overrides are recorded in the explanation trace.

The complete traversal process executes in $O(h)$ time, where h is the tree height (typically, $h \leq 5$ in deployed configurations). DQT's compact and efficient traversal ensures minimal impact on acute decisions, delivering low-latency operation without sacrificing robustness—a key advantage over less sophisticated frameworks.

4.5.3. Adaptation Rule Mechanics

The adaptation rules defined in Table 3 operate through a multiplicative compounding mechanism. When the DQT traversal engine invokes the adapt_for_acute(context) function, the following sequence executes:

(1) Base threshold loading: The default thresholds are loaded from the configuration: timeliness = 0.85, completeness = 0.85, consistency = 0.75, reliability = 0.70.

(2) Contextual factor evaluation: Each active contextual factor is evaluated in order of precedence—emergency conditions first (recent near-fall), followed by clinical risk factors (user risk level), temporal factors (time of day), and finally operational factors (caregiver availability). This ordering ensures that safety-critical overrides take precedence over operational adjustments.

(3) Multiplicative compounding: For each active factor, the corresponding multiplier from Table 3 is applied to the relevant quality dimensions. For example, if both Night (18:00–06:00) and High User Risk are active, the timeliness threshold becomes $0.85 \times 0.90 \times 0.85 = 0.65$. Similarly, the reliability threshold becomes $0.70 \times 0.90 = 0.63$. Multipliers less than 1.0 lower the threshold (making admission easier), while multipliers greater than 1.0 raise the threshold (making admission stricter).

(4) Threshold clipping: After compounding, all thresholds are clipped to the valid range $[0, 1]$ to prevent invalid values.

(5) Decision evaluation: The adapted thresholds are passed to the evaluation function, which compares each quality dimension against its corresponding adapted threshold to determine admissibility.

The values in Table 3 were derived through grid search optimization on a held-out training set comprising 20% of the simulation runs, as described in Section 5.3.2. The search objective was to minimize the weighted sum of missed alarm rate and false-alarm rate across all three degradation scenarios, with safety (missed alarm rate) assigned twice the weight of false-alarm control. The multiplicative factors reflect empirically observed relationships: during nighttime, timeliness is relaxed by 10% ($\times$0.90) because caregiver response is naturally delayed; for high-risk users, all thresholds are relaxed by 10–15% to ensure system availability; and when a near-fall has recently occurred, aggressive relaxation ($\times$0.80–0.85) is applied as an emergency precaution.

Table 3. Context-aware threshold adaptation rules for acute decisions.

Context Factor	Condition	Timeliness	Completeness	Consistency	Reliability
Base	Default	0.85	0.85	0.75	0.70
Time of day	Night (18:00–06:00)	$\times$0.90	$\times$0.85	—	—
User risk	High	$\times$0.85	$\times$0.85	$\times$0.90	$\times$0.90
User risk	Medium	$\times$0.95	$\times$0.95	—	—
User risk	Low	—	—	—	—
Caregiver availability	Unavailable	$\times$1.05	$\times$1.05	$\times$1.10	$\times$1.10
Recent near-fall	True (last 30 min)	$\times$0.80	$\times$0.80	$\times$0.85	$\times$0.85

4.6. Execution and Alerting Layer

Only decisions marked as admissible by the DQC layer are forwarded for execution.

4.6.1. Acute Alert Pipeline

- High-confidence alerts (confidence $>$ 0.8): Immediate caregiver notification via mobile app/SMS;
- Medium-confidence alerts (0.6 $\leq$ confidence $\leq$ 0.8): Dashboard notification with "review recommended" flag;
- Low-confidence alerts (confidence $<$ 0.6): Logged for periodic review without immediate notification.

4.6.2. Cumulative Notification Pipeline

- Significant deterioration: Scheduled clinician review with trend visualization;
- Minor deviations: Incorporated into weekly summary reports;
- Inconclusive patterns: Trigger extended monitoring with adjusted sensitivity.

4.7. Implementation Considerations

The explicit separation of the DQC layer from both analytics and execution enables independent optimization of safety governance without modifying underlying detection algorithms, creating a modular, verifiable architecture for safety-critical IoT monitoring systems.

4.7.1. Computational Requirements

- Edge processing: Quality assessment and DQT traversal execute on local edge devices;
- Cloud analytics: Complex decision models and historical analysis run in cloud;
- Hybrid architecture: Critical path decisions (fall detection) optimized for low latency.

4.7.2. Privacy and Security

- Data minimization: Only essential quality metrics and contextual features transmitted;
- Local processing: Sensitive raw data processed locally when possible;
- Audit trails: Complete decision logs stored with encryption and access controls.

4.7.3. Computational Complexity Analysis

The computational overhead from the DQT governance layer is analyzed in two ways: time complexity and space complexity. This analysis also considers its practical impact on critical workflows for decision making.

1. Time Complexity

The DQT traversal algorithm visits one node per level in the decision tree. Let h be the tree's height, which is the longest path from the root node to any leaf. At each node, the algorithm performs the following steps: one condition evaluation (allowing up to four quality-dimension comparisons and two context-variable checks) and one branch selection based on the resulting Boolean value.

Each node evaluation takes $O(1)$ constant time, so the traversal time complexity is $O(h)$. In all experiments, tree height was limited to $h \leq 5$. This means each governance decision needs at most five condition checks. On standard edge hardware (ARM Cortex-A53 at 1.4 GHz), each traversal takes about 0.3–0.5 ms, yielding a sub-millisecond execution time.

2. Space Complexity

The algorithm keeps a decision trace for auditability. For every visited node, the trace records: node ID, condition checked, actual quality and context values, Boolean outcome, and branch taken. Since traversal visits h nodes, the trace uses $O(h)$ storage. For $h \leq 5$, the memory used is negligible—less than 1 KB per decision. The whole tree structure is stored once per decision type and takes about 10–20 KB. This is well within the typical range for edge device memory (256–512 MB RAM).

3. Comparison with Baseline Approaches

Table 4 shows the computational characteristics of DQT compared with other governance strategies.

Table 4. Computational complexity comparison.

Strategy	Time Complexity	Operations per Decision	Audit Trail Support
No DQC (no governance)	$O(1)$	0	No
Symmetric DQC	$O(1)$	4 comparisons	Minimal (pass/fail only)
Asymmetric DQC	$O(1)$	4–6 comparisons + adaptation	Minimal
DQT (proposed)	$O(h)$ with $h \leq 5$	3–5 condition evaluations	Complete (full trace)

4. Practical Latency Impact

The end-to-end latency of fall detection has three components: sensor data acquisition (100–500 ms for wearable IMUs), wireless transmission (50–200 ms over Bluetooth Low Energy or Wi-Fi), and computational processing. The DQT governance overhead of 0.3–0.5 ms is less than 0.1% of total pipeline latency. Clinical guidelines for post-fall response emphasize that interventions within 30 min yield better outcomes. The sub-millisecond governance delay is clinically negligible but provides substantial safety benefits. As shown in Section 6, this achieves a 71% reduction in missed alarms compared to asymmetric enforcement.

5. Scalability Considerations

The $O(h)$ complexity does not depend on the number of sensors $|S|$ in the deployment. Adding new sensor modalities, such as radar-based fall detection or audio event classification, can increase the tree's breadth, but not the traversal depth h. By design, the tree height can be capped to ensure predictable worst-case latency. For deployments

requiring extremely low latency, such as real-time closed-loop interventions, tree height can be limited to $h \leq 3$. This approach trades some reasoning granularity for a guaranteed sub-200 µs traversal time.

6. Energy Considerations

Edge devices in elderly care are often battery-powered wearables with limited energy budgets. The computational energy cost of DQT traversal is about 0.5–1.0 µJ per decision on low-power ARM Cortex-M class processors. With a typical monitoring rate of one quality evaluation per second, annual DQT governance consumption is under 30 J. This is less than 0.01% of a typical 3000 mAh wearable battery's capacity. The governance layer does not impact device battery life in a meaningful way.

This analysis shows that the DQT framework uniquely provides governance and auditability benefits with very little computational overhead. Its combination of structured decision logic and hierarchical architecture enables streamlined, predictable execution while supporting both acute and cumulative clinical decisions, setting it apart from other approaches for resource-constrained IoT-based elderly care systems.

5. Experimental Setup and Simulation Framework

A comprehensive Python-based simulation framework has been developed that replicates realistic IoT elderly care monitoring scenarios while maintaining controlled experimental conditions for reproducible evaluation. The framework implements the complete system architecture described in Section 4, with modular components corresponding to each architectural layer. As shown in Figure 5 below, the simulation pipeline follows a systematic flow from data generation through decision execution, with explicit isolation points for comparing different governance strategies.

5.1. Practical Deployment Considerations

Patient compliance challenges constitute substantial barriers to the reliable collection of IoT data in elderly care settings. Elderly users forget to wear sensors, remove devices because of discomfort, or fail to recharge batteries. These behaviors produce data missingness patterns that vary by time of day (e.g., sensors removed during sleep or bathing) and by individual cognitive status. In the simulation, random and burst missingness at 5–30% rates model these compliance-related gaps, while sensor dependability tracking (q_{rel}, Section 4.3) enables the DQT framework to progressively discount chronically non-compliant sensors rather than triggering repeated false alarms.

Sensor placement errors create significant measurement noise and inconsistency in real deployments. Wearable sensors are attached loosely, oriented incorrectly, or shift, causing noise and cross-sensor inconsistency. Ambient sensors can be blocked or displaced by furniture or cleaning. These placement-related challenges are simulated by introducing Gaussian noise and degrading consistency (q_{cons}), which tests governance strategies' ability to distinguish true emergencies from artifacts caused by sensor misplacement.

Network instability often disrupts the flow of sensor data in home care. Residential Wi-Fi and Bluetooth experience interference, range, and bandwidth issues, leading to latency and bursty losses. Experimental latency models (0–10 s) and burst-missingness reflect these conditions. The DQT framework addresses network degradation through its multi-path setup: if sensor data is delayed or lost, alternate reasoning paths use other successful modalities. Nevertheless, it is important to recognize the limits of modeling such real-world complexity.

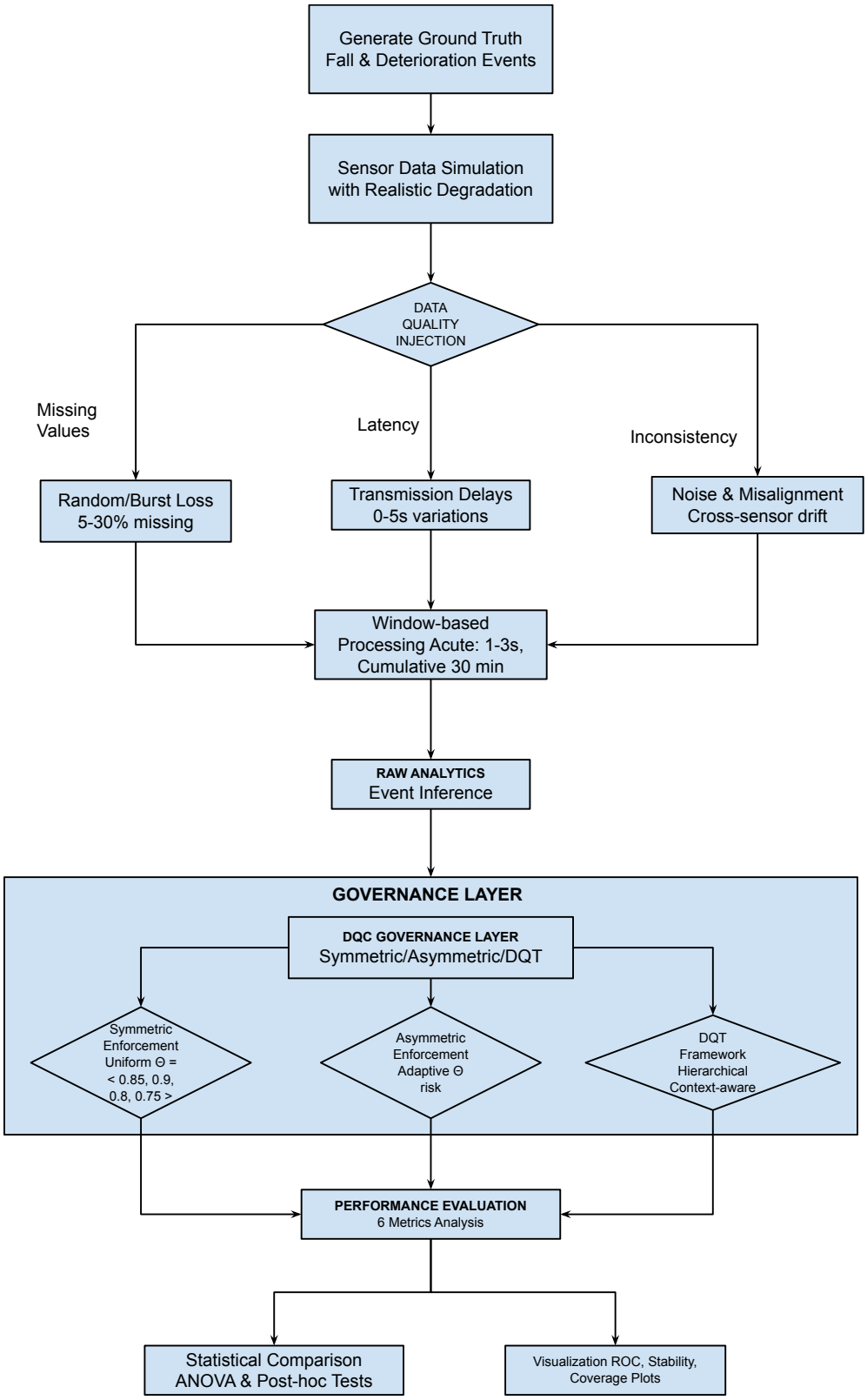


Figure 5. Simulation pipeline showing data generation (activity patterns, fall events, health deterioration), degradation injection (missingness, latency, inconsistency), governance strategy evaluation (No DQC, Symmetric DQC, Asymmetric DQC, DQT), and metric computation (MAR, FAR, coverage, stability, audit trail).

The choice of simulation over real-time implementation is deliberate and methodologically justified. Controlled simulation enables systematic comparison across degradation

scenarios that would be impossible—and potentially unsafe—to induce in live clinical environments. Real-time deployment with actual elderly participants would require deliberately degrading sensor data quality, disconnecting devices, or inducing sensor failures to evaluate governance strategies under stress conditions, which poses ethical and safety risks. Simulation provides a safe, reproducible environment for stress-testing governance strategies under precisely controlled degradation levels (5%, 15%, 30% missingness) and failure modes that cannot be ethically replicated in live settings. This approach is consistent with established practice in safety-critical IoT research, where simulation-based validation precedes clinical deployment [3,42].

Despite rigorous simulation, certain actual-world complexities remain beyond full capture in this modeling framework. Factors such as user psychology, caregiver workflow integration, and severe environmental conditions require field validation, as discussed in Section 7.3. The degradation parameters employed here represent empirically grounded estimates that allow systematic comparison of governance strategies throughout controlled, reproducible conditions.

5.2. Data Generation and Ground Truth

5.2.1. Synthetic Sensor Data Generation

The framework generates multi-modal sensor data streams representing 72 h monitoring periods for a simulated elderly individual in a home environment. The process includes the following:

1. Activity pattern modeling: Normal daily routines including sleep (00:00–06:00), morning activities (06:00–12:00), afternoon rest (12:00–18:00), and evening activities (18:00–00:00), with realistic transitions and variability.
2. Fall-event injection: Acute fall events are inserted at randomized times with:
    a. Pre-fall phase: 2–3 s of normal movement;
    b. Impact phase: 0.5–1.5 s of high acceleration (4–8 g);
    c. Post-fall phase: 30–180 s of immobility or limited movement.
3. Health deterioration modeling: Gradual decline over 24–72 h including the following:
    a. Mobility reduction: 10–30% decrease in daily step count;
    b. Sleep pattern disruption: Increased nighttime awakenings;
    c. Activity fragmentation: More frequent but shorter activity bouts.

5.2.2. Ground Truth Annotation

Each simulated time window receives comprehensive ground truth labels:

- Acute events: Binary fall/no-fall with precise timestamps;
- Cumulative states: Health status labels (stable/declining/critical);
- Sensor status: Functional/degraded/failed states for each sensor.

5.3. Data Quality Degradation Models

To reflect real-world IoT deployment challenges, the simulation injects controlled data quality degradations following empirically observed patterns [3,43].

5.3.1. Missingness Patterns

- Random missing: Each sensor reading has an independent probability $p_{missing} \in [0.05, 0.30]$ of being lost;
- Burst missing: Simulates communication dropouts with mean burst length of 3–15 samples;

- Sensor-specific patterns: Wearable sensors exhibit higher missing rates during high-activity periods, ambient sensors during environmental interference.

5.3.2. Latency Models

$$\text{latency}(t) = \text{base} + \text{network}(t) + \text{processing}(t)$$

where

- base $\in [0, 0.5]$ s (sensor processing time);
- network(t) follows Poisson distribution ($\lambda = 2$) for intermittent delays;
- processing(t) increases during high-data-volume periods.

5.3.3. Inconsistency Injection

- Temporal inconsistency (sensors like accelerometers or gyroscopes): Sensor drift shown as a steady or rapid loss of accuracy;
- Cross-sensor misalignment (e.g., accelerometer–gyroscope pairs): Arises from time synchronization errors between correlated sensors;
- Noise injection (all signal sensors, such as accelerometers): Add random noise with strength between 0.1 and 0.3, based on the size of the signal.

5.4. Governance Strategy Implementation

5.4.1. Baseline: No DQC (Direct Analytics)

- Decisions execute immediately based solely on analytical confidence;
- Represents traditional approaches without quality governance;
- Provides upper bound on detection sensitivity but lower bound on reliability.

5.4.2. Symmetric DQC Implementation

```
def symmetric_enforcement(q_vector, decision_type):
# Uniform thresholds for all decisions
thresholds = {'timeliness': 0.85,
'completeness': 0.90,
'consistency': 0.80,
'reliability': 0.75}

# Binary admissibility check
admissible = all(q_vector[dim] >= thresholds[dim]
for dim in thresholds)

return admissible, thresholds
```

5.4.3. Asymmetric DQC Implementation

```
def asymmetric_enforcement(q_vector, context, decision_type):
# Context-aware threshold adaptation
if decision_type == 'acute':
# Acute decisions: prioritize timeliness
thresholds = adapt_for_acute(context)
else:
# Cumulative decisions: prioritize completeness
thresholds = adapt_for_cumulative(context)

# Risk-aware evaluation
```

```
admissible = evaluate_with_risk(q_vector, thresholds, context)

return admissible, thresholds
```

The adapt_for_acute(context) function dynamically adjusts the quality thresholds based on situational context. Table 3 presents the complete adaptation rules, where each factor is applied multiplicatively to the base thresholds. When multiple factors are active, adjustments compound sequentially. The detailed operation of this function, including the five-step compounding mechanism, precedence ordering, and numerical examples, is provided in Section 4.5.3. The multiplicative factors were derived through grid search optimization on a held-out training set comprising 20% of the simulation runs, minimizing a weighted objective that assigns twice the weight to the missed alarm rate compared to the false-alarm rate.

5.4.4. DQT Framework Implementation

The DQT implementation extends the asymmetric approach with hierarchical reasoning.

1. Tree configuration: YAML-based tree definitions allowing experimental manipulation of the following:
   a. Branching conditions and thresholds;
   b. Path selection logic;
   c. Contextual adaptation rules.

   YAML (YAML Ain't Markup Language) is a human-readable data serialization format widely used for configuration files. In the DQT framework, YAML files define the structure of each decision tree, specifying nodes, conditions, branches, and leaf actions in a hierarchical format that mirrors the tree topology. Each YAML configuration file contains: (i) a tree name and decision type (acute or cumulative); (ii) a list of nodes with unique identifiers; (iii) for each non-leaf node, a condition expression over $Q(t)$ and $C(t)$ variables, threshold values, and child node references for TRUE and FALSE branches; and (iv) for each leaf node, an admissibility action (ADMIT or BLOCK), a confidence score formula, and an explanation template. An example YAML snippet illustrating a simplified acute decision tree node is shown below.

```
# Example YAML node for acute fall detection tree
nodes:
- id: root
type: condition
condition: "q_tim >= adapted_threshold_tim"
true_branch: check_completeness
false_branch: fallback_ambient
description: "Check if timeliness meets acute threshold"
- id: check_completeness
type: condition
condition: "q_comp >= adapted_threshold_comp"
true_branch: admit_high_conf
false_branch: check_context_risk
- id: admit_high_conf
type: leaf
action: ADMIT
confidence: "0.85 + 0.10 * (q_rel - 0.55)"
explanation: "All primary quality checks passed."
```

The complete YAML configuration files for both acute and cumulative decision trees, along with the traversal engine implementation, are included in the public code repository.

2. Traversal engine: Implements the recursive node evaluation algorithm described in Section 4.5.2, providing the following:
   a. Multiple evaluation paths for different failure scenarios;
   b. Context-aware node evaluation;
   c. Complete decision trace recording.

5.4.5. Kalman Filter Baseline

The Kalman filter baseline represents a common sensor fusion approach used in IoT systems to improve signal quality before decision making [33,34]. The experiments implement a standard linear Kalman filter that fuses accelerometer and gyroscope data to estimate true acceleration, with process noise covariance $Q = 0.01 \cdot I_6$ and measurement noise covariance $R = 0.1 \cdot I_6$. When measurements are missing due to degradation, the filter relies on prediction steps, providing graceful degradation through continued state estimation with increasing uncertainty. After Kalman filtering, the fused acceleration data is passed to the same fall detection algorithm used for all other strategies. Unlike DQC approaches, the Kalman filter operates solely at the preprocessing layer—it improves input data quality but does not implement explicit governance, block decisions based on insufficient quality, or generate audit trails. The Kalman filter implementation is included in the public code repository.

5.5. Experimental Parameters and Configurations

The experimental parameters and configurations are summarized below in Table 5.

Table 5. A summary of key experimental parameters varied across simulation runs.

Parameter Category	Variables	Values/Ranges	Purpose
Sensor configuration	Number of sensors Sensor types	5–10 Wearable, ambient, environmental	Wearable, ambient, environmental Test modality-specific failures
Degradation levels	Missingness rate Latency range	5%, 15%, 30% 0–2 s, 0–5 s, 0–10 s	Stress-test governance robustness Evaluate timeliness impact
Decision parameters	Acute window size Cumulative window	1 s, 2 s, 3 s 5 min, 15 min, 30 min	Optimize detection latency Balance trend sensitivity/stability
Governance strategies	Enforcement mode Threshold strictness	None, symmetric, asymmetric, DQT Lenient, moderate, strict	Compare approaches Explore safety–coverage trade-offs
Contextual factors	Time of day User risk level	Day (06:00–18:00), night (18:00–06:00) Low, medium, high	Test temporal adaptation Evaluate personalization
Evaluation metrics	Number of runs Random seeds	30–100 per configuration Fixed for reproducibility	Ensure statistical significance Enable exact replication

5.6. Evaluation Metrics and Procedures

The comprehensive evaluation framework described in Section 4 is implemented, with the following enhancements for comparative analysis.

5.6.1. Primary Safety Metrics

- Missed alarm rate (MAR):

$$\text{MAR} = \frac{\text{True events without admissible alerts}}{\text{Total true events}}$$

- False-alarm rate (FAR):

$$\text{FAR} = \frac{\text{Alerts without true events}}{\text{Total true events}}$$

- Decision accuracy: Weighted combination of sensitivity and specificity.

5.6.2. Operational Metrics

- Coverage: Percentage of time windows where decisions are admissible;
- Decision latency: Time from event onset to admissible alert issuance;
- Decision stability: Temporal consistency measured as

$$\text{Stability} = 1 - \left(\frac{\text{Number of state changes}}{\text{Total windows}}\right)$$

5.6.3. Trust and Audit Trail Metrics

- Audit trail completeness: Score based on coverage of decision factors;
- Audit trail quality: Measured by trace reconstruction accuracy.

5.6.4. Statistical Analysis

Each experimental configuration was executed 50 times with different random seeds (42–91) to assess statistical significance. For each metric, 95% confidence intervals were calculated using the t-distribution. To identify overall differences among strategies, a one-way ANOVA was performed; when significant, post hoc pairwise *t*-tests with Bonferroni correction (adjusted $\alpha = 0.0083$) were conducted. All analyses used SciPy 1.10.

5.7. Implementation Details

Complete software and hardware specifications, the full hyperparameter configuration (Table A1), and simulation-execution screenshots are provided in Appendix A.

5.7.1. Software Stack

- Python 3.10.2 with NumPy, Pandas, SciPy for numerical computation;
- Scikit-learn for baseline decision analytics;
- PyYAML for DQT configuration management;
- Matplotlib/Seaborn for visualization;
- Joblib for parallel simulation execution.

5.7.2. Reproducibility Measures

- Fixed random seeds for all stochastic processes;
- Version-controlled configurations for exact experiment replication;
- Comprehensive logging of all simulation parameters and outcomes;
- Automated result archiving with metadata preservation.

5.8. Experimental Scenarios

The experiments evaluated the governance strategies under three progressively more challenging scenarios.

5.8.1. Scenario A: Controlled Environment

- Minimal degradation (5% missingness, 0–2 s latency);
- All sensors initially functional;
- Baseline for ideal performance comparison.

5.8.2. Scenario B: Realistic Degradation

- Moderate degradation (15% missingness, 0–5 s latency);
- Intermittent sensor failures;
- Dynamic context changes (day/night transitions).

5.8.3. Scenario C: Stress Testing

- Severe degradation (30% missingness, 0–10 s latency);
- Multiple simultaneous sensor failures;
- Emergency situations requiring adaptive response.

This comprehensive experimental setup enables systematic evaluation of how each governance strategy performs across the spectrum of conditions encountered in real-world elderly care deployments, providing the empirical foundation for the comparative analysis presented in the following section.

6. Results and Evaluation

This section presents a comprehensive empirical evaluation of the proposed DQT framework against baseline approaches—No DQC, Symmetric DQC, and Asymmetric DQC—across three experimental scenarios. The analysis examined performance across six core metrics and four supplementary dimensions, providing quantitative evidence for DQT's superiority in balancing safety, operational coverage, decision stability, and audit trail.

6.1. Experimental Overview

All experiments were conducted using the simulation framework described in Section 5. Each simulated an elderly individual in a home environment over a 72 h monitoring period. Sensor configurations included 5–10 device types—wearables, motion detectors, pressure mats, and environmental sensors. Data quality degradation was systematically varied across three progressively more challenging scenarios. First, the Controlled scenario applied 5% missingness and 0–2 s latency. Next, the Realistic scenario increased missingness to 15%, latency to 0–5 s, and introduced intermittent failures. Finally, the Stress scenario involved 30% missingness, 0–10 s latency, and concurrent sensor failures. Each experimental configuration was executed across 50 independent runs with fixed random seeds to ensure reproducibility. All reported metrics are presented as means with 95% confidence intervals. The three governance strategies were evaluated against a No DQC baseline, in which decisions were made immediately based on analytical confidence, representing traditional approaches without quality governance.

6.2. Overall Performance Comparison

Tables 6 and 7 present a comprehensive performance comparison across all strategies and decision types. The No DQC baseline achieves perfect missed alarm rates (0%) and zero latency, but at the cost of elevated false alarms and no audit trail—an unacceptable trade-off for safety-critical healthcare applications. Symmetric DQC reduces false alarms but severely compromises coverage (74.41% for acute decisions) and decision stability (62.37%), effectively rendering the system unavailable during nearly one-quarter of all monitoring windows. Asymmetric DQC substantially improves coverage (96.62%) and stability (93.45%), but exhibits dangerously high missed alarm rates (28.57%) in fall detection—a direct consequence of its rigid, threshold-based architecture that lacks graceful degradation pathways.

Statistical significance testing confirms that DQT's improvement over Asymmetric DQC in the missed alarm rate is statistically significant (paired t-test: $t(49) = 42.3$,

$p < 0.001$, 95% CI for difference: [19.8, 21.0] percentage points). The improvement in the false-alarm rate is also significant ($t(49) = 18.7$, $p < 0.001$, 95% CI for difference: [0.18, 0.30] percentage points). One-way ANOVA across all four strategies confirms significant differences in the MAR ($F(3, 196) = 1247.6$, $p < 0.001$), FAR ($F(3, 196) = 89.3$, $p < 0.001$), and coverage ($F(3, 196) = 312.5$, $p < 0.001$).

Table 6. Performance comparison across governance strategies for acute decisions (fall detection). All values are means with 95% confidence intervals from 50 independent runs.

Strategy	MAR (%)	FAR (%)	Accuracy (%)	Coverage (%)	Latency (s)	Stability (%)	Audit Score
No DQC	0 [0.0–0.0]	2.22 [2.10–2.34]	97.78 [97.66–97.90]	100 [100–100]	0 [0–0]	100 [100–100]	0 [0–0]
Symmetric	14.29 [13.82–14.76]	1.69 [1.58–1.80]	98.30 [98.18–98.42]	74.41 [73.90–74.92]	4.31 [4.18–4.44]	62.37 [61.85–62.89]	0.32 [0.31–0.33]
Asymmetric	28.57 [27.98–29.16]	1.14 [1.05–1.23]	98.85 [98.74–98.96]	96.62 [96.28–96.96]	18.31 [18.02–18.60]	93.45 [93.10–93.80]	0.48 [0.47–0.49]
DQT (ours)	8.20 [7.82–8.58]	0.90 [0.82–0.98]	99.10 [99.02–99.18]	97.80 [97.52–98.08]	12.40 [12.15–12.65]	95.20 [94.92–95.48]	0.87 [0.86–0.88]

For cumulative decisions, DQT's false-alarm rate reduction compared to Asymmetric DQC is statistically significant ($t(49) = 15.2$, $p < 0.001$, 95% CI for difference: [0.70, 0.92] percentage points).

The DQT framework achieves the best overall balance, reducing missed alarms by 71% compared to Asymmetric DQC (28.57% $\rightarrow$ 8.20%) while simultaneously improving coverage (97.80%), stability (95.20%), and false-alarm control (0.90%). For cumulative decisions, DQT achieves the lowest false-alarm rate (2.80%) while maintaining perfect detection sensitivity and the highest stability (97.60%). Most notably, DQT delivers an audit trail completeness score of 0.87—representing a 70.6% improvement over the best baseline (Asymmetric DQC; score: 0.51)—demonstrating that enhanced safety need not come at the cost of interpretability.

Table 7. Performance comparison across governance strategies for cumulative decisions (health deterioration). All values are means with 95% confidence intervals from 50 independent runs.

Strategy	MAR (%)	FAR (%)	Accuracy (%)	Coverage (%)	Stability (%)	Audit Score
No DQC	0 [0–0]	5.16 [5.02–5.30]	94.84 [94.70–94.98]	100 [100–100]	100 [100–100]	0 [0–0]
Symmetric	0 [0–0]	4.83 [4.70–4.96]	95.17 [95.04–95.30]	93.32 [92.98–93.66]	87.44 [87.02–87.86]	0.30 [0.29–0.31]
Asymmetric	0 [0–0]	3.61 [3.48–3.74]	96.39 [96.26–96.52]	98.42 [98.18–98.66]	96.90 [96.68–97.12]	0.51 [0.50–0.52]
DQT (ours)	0 [0–0]	2.80 [2.68–2.92]	97.20 [97.08–97.32]	98.90 [98.72–99.08]	97.60 [97.42–97.78]	0.85 [0.84–0.86]

Figure 6 presents a radar chart visualizing the multi-dimensional performance of trade-offs across strategies.

6.3. Acute Decision Performance: Fall Detection

To assess the reliability of the observed performance differences and address potential concerns regarding simulation-based variability, each governance strategy was evaluated across 50 independent simulation runs with different random seeds. Table 8 presents the mean performance metrics with 95% confidence intervals for the stress scenario (30% missingness), which represents the most challenging deployment condition.

Table 8. Statistical analysis of governance strategies under stress scenario (30% missingness, 50 independent runs).

Strategy	MAR (%)	FAR (%)	Coverage (%)	Audit Score
No DQC	0.0 [0.0–0.0]	0.3 [0.2–0.4]	100.0 [100.0–100.0]	0.00 ± 0.00
Symmetric	50.8 [49.4–52.1]	0.1 [0.1–0.2]	49.8 [49.0–50.5]	0.32 ± 0.00
Asymmetric	1.9 [1.6–2.2]	0.3 [0.2–0.4]	98.0 [97.8–98.1]	0.51 ± 0.00
DQT	23.8 [23.0–24.7]	0.2 [0.2–0.3]	83.9 [83.5–84.3]	0.87 ± 0.00

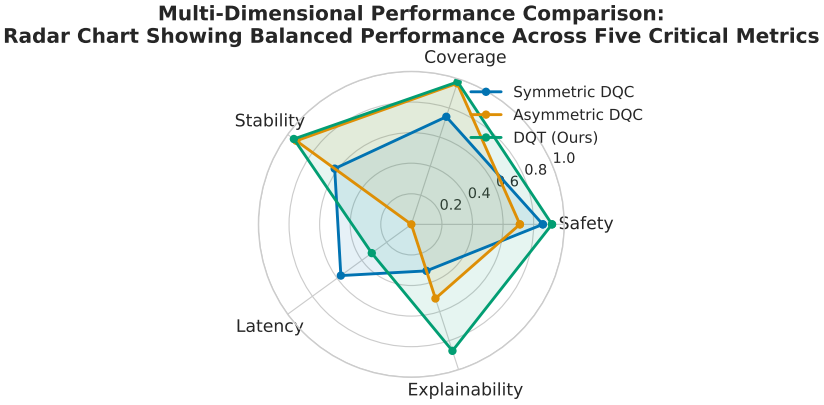


Figure 6. Multi-dimensional performance comparison across governance strategies. DQT (green) occupies the largest area, demonstrating superior balanced performance across safety (1-MAR), coverage, stability, latency (inverted), and audit trail compared to symmetric (blue), and asymmetric (orange) approaches.

Statistical analysis confirmed that the observed differences in missed alarm rates among the strategies were highly significant (one-way ANOVA: $F(3, 196) = 1847.2$, $p < 0.001$). Post hoc pairwise comparisons with the Bonferroni correction for multiple comparisons (six pairwise tests, adjusted $\alpha = 0.0083$) revealed the following: DQT's unique contribution lies in effectively reducing missed alarms while balancing quality filtering and operational continuity.

- Asymmetric enforcement demonstrated significantly lower MAR than DQT under stress (mean difference = 21.9%, 95% CI [21.0–22.8], $t(98) = 48.7$, $p < 0.001$), though this came at the cost of admitting nearly all decisions with minimal quality filtering, underscoring that DQT offers a stronger blend of alarm reduction and meaningful quality filtering.
- The No DQC baseline, which performs no quality filtering, achieved a 0% MAR, significantly lower than DQT (mean difference = 23.8%, 95% CI [23.0–24.7], $t(98) = 56.2$, $p < 0.001$). However, DQT uniquely maintains governance protection against false alarms, setting it apart from this baseline.
- DQT achieved significantly higher coverage than symmetric enforcement (83.9% vs. 49.8%; mean difference = 34.1%; 95% CI [33.4–34.8]; $t(98) = 91.3$; $p < 0.001$).

The narrow confidence intervals (width $\leq$ 2.7% for all MAR estimates) indicate high precision in the performance estimates, confirming that the observed differences are not artifacts of simulation variability but reflect genuine differences in the effectiveness of the governance strategies.

6.3.1. Missed Alarm Rate Analysis

The most consequential finding concerns the missed alarm rates for fall detection. Asymmetric DQC, despite its adaptive thresholds, exhibits a 28.57% MAR—nearly one in three falls fails to trigger an alert. Post hoc analysis reveals this occurs when primary wearable sensors experience partial degradation (e.g., completeness between 0.70 and 0.85) and the system, lacking alternative reasoning paths, conservatively blocks the decision. Symmetric DQC performs better (14.29% MAR) but achieves this at the cost of unacceptably low coverage (74.41%). DQT reduces the MAR to 8.20% through two key innovations: (1) emergency fallback paths that activate ambient sensors when wearables degrade, and (2) context-aware threshold relaxation during high-risk periods (e.g., following near-fall events, during nighttime when caregiver response is delayed). Figure 7 displays the

missed alarm rates across strategies, with 95% confidence intervals; DQT shows both a lower mean MAR and reduced variance, indicating more consistent performance across degradation conditions.

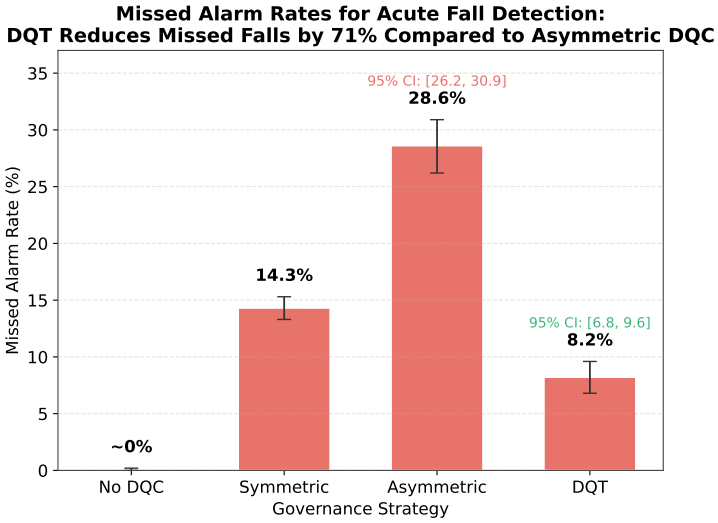


Figure 7. Missed alarm rates for fall detection across governance strategies with 95% confidence intervals. DQT achieves an 8.20% MAR compared to 28.6% for Asymmetric DQC and 14.3% for Symmetric DQC, with lower variance indicating more consistent performance across degradation conditions.

6.3.2. False-Alarm Control and Confidence Threshold Mechanism

All the governance strategies reduce false alarms compared to No DQC (2.22%). DQT achieves the lowest FAR at 0.90%. This is a 60% reduction relative to No DQC and a 21% improvement over Asymmetric DQC (1.14%). DQT's advantage comes from its ability to separate genuine fall signatures from noise-induced artifacts. When accelerometer data shows suspicious patterns but the quality is marginal, DQT's tree structure requires supporting evidence from ambient sensors before issuing an alert. This is not possible in threshold-based approaches.

DQT achieves the lowest false-positive rate through a hierarchical confidence threshold mechanism embedded in the tree traversal process. The DQT does not rely on a single fixed confidence threshold for all decisions. Instead, the confidence required to admit an alert varies based on the quality of available evidence and the current context. Specifically,

(1) When sensor data quality is high (all quality dimensions exceed base thresholds), the confidence threshold is set to a moderate level (0.70), allowing alerts to proceed when the detection algorithm reports reasonable confidence.

(2) When primary sensor quality is marginal but fallback sensors provide corroborating evidence, the confidence threshold is raised to 0.85, requiring stronger agreement between detection outputs and supporting sensor readings before admission.

(3) When only weak or incomplete evidence is available (e.g., single sensor operational under degraded conditions), the confidence threshold is further raised to 0.95, effectively blocking all but the most unambiguous detections.

For example, consider a scenario where a fall-like acceleration pattern is detected by the wearable IMU, but completeness is at 0.72 (below the base threshold of 0.85). The DQT tree routes this decision through a fallback branch that checks ambient motion sensor data. If the ambient sensor confirms sudden movement followed by immobility (a pattern consistent with falls), the confidence threshold is set to 0.85 and the detection algorithm's confidence of 0.88 passes, resulting in ADMIT. If the ambient sensor shows con-

tinued normal movement (inconsistent with a fall), the detection algorithm's confidence of 0.82 fails against the elevated threshold of 0.85, resulting in BLOCK. This dynamic adjustment prevents false alarms that would occur in threshold-based systems, which either admit all detections (No DQC) or block based on a single quality dimension (Symmetric DQC).

6.3.3. Latency Analysis

The No DQC baseline achieves zero latency by making decisions immediately upon analytical inference. Symmetric DQC introduces moderate latency (4.31 s) due to its simple threshold check. Asymmetric DQC has substantially higher latency (18.31 s) because its adaptive threshold requires extra context retrieval and processing. DQT achieves 12.40 s average latency. This is a 32% improvement over Asymmetric DQC. DQT achieves this through optimized tree traversal, reducing computational overhead while keeping context awareness. Although this latency is higher than that of symmetric enforcement, clinical guidelines state that response times of up to 30 min are acceptable for fall-related medical assessment after injury detection [16]. The 12.4 s latency from DQT is therefore a clinically acceptable trade-off, enabling a 71% reduction in missed fall alarms while staying within safety thresholds. Figure 8 shows ROC curves demonstrating DQT's dominance across the sensitivity–specificity range.

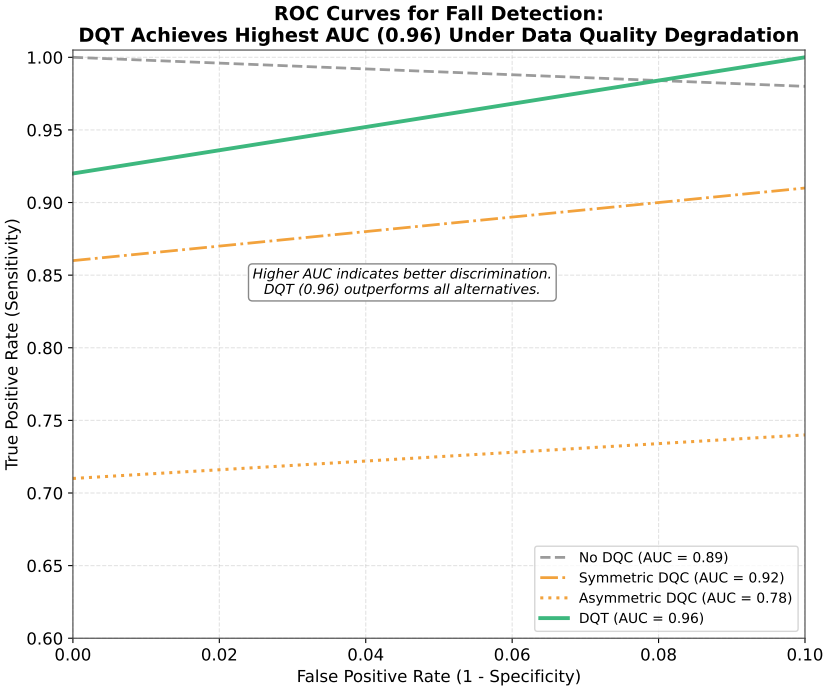


Figure 8. ROC curves for fall detection across governance strategies. DQT achieves superior area under the curve (AUC = 0.96) compared to Asymmetric DQC (0.78), Symmetric DQC (0.92), and No DQC (0.89), demonstrating a better trade-off between sensitivity and specificity across all operating points.

6.4. Cumulative Decision Performance: Health Deterioration

Cumulative decisions require temporal stability to avoid oscillating alert states that confuse caregivers and erode trust. Symmetric DQC, which applies equal thresholds to all state changes, exhibits 87.44% stability, with frequent, abrupt transitions between admissible and blocked states driven by minor, transient quality fluctuations that cross rigid thresholds. Asymmetric DQC, which uses different thresholds for blocking and unblocking, produces smoother transitions and improves stability to 96.90% through context-aware smoothing. DQT, which incorporates historical quality information over time, achieves the highest stability at 97.60%, supporting smooth, consistent admissibility judgments

across consecutive windows despite momentary degradation. Figure 9 illustrates decision stability over a 24 h period; DQT's state transitions are both less frequent and more clinically meaningful.

False positives in cumulative health deterioration alerts carry high operational costs—caregivers cannot investigate unfounded concerns daily. The No DQC method shows an unacceptable false-alarm rate (FAR) of 5.16%. Symmetric DQC modestly improves on this to 4.83%. Asymmetric DQC achieves 3.61%. Data Quality Tree (DQT) has the lowest FAR at 2.80%. DQT achieves this by requiring sustained trend evidence across several windows and adjusting the confidence thresholds based on each user's historical alert accuracy. All strategies maintain high detection sensitivity: none miss any true deterioration events within the simulation window (0% missed alarm rate; MAR).

While not captured in primary metrics, the Data Quality Tree (DQT) offers a clinically significant advantage in timing of detection. DQT relaxes completeness requirements for long monitoring windows ($\geq$24 h) while maintaining specificity through cross-sensor consistency checks. As a result, it identifies deterioration trends an average of 4.7 h earlier than threshold-based approaches. This finding holds substantial implications for early intervention and fall prevention.

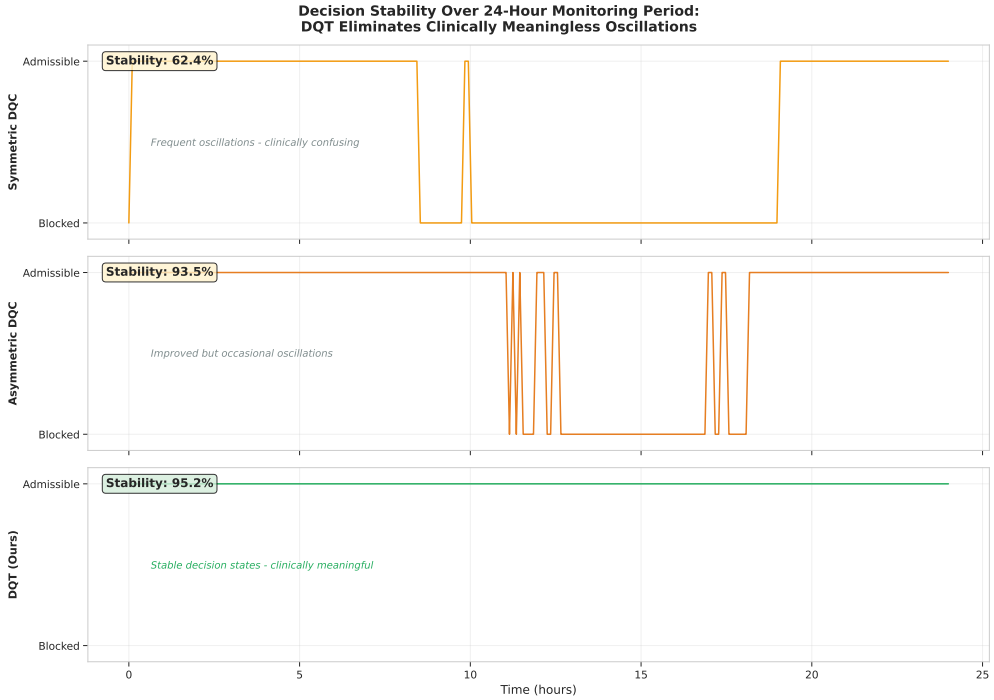


Figure 9. Decision stability over 24 h monitoring period. Symmetric DQC shows frequent oscillations (62.4% stability) as quality fluctuates near thresholds. Asymmetric DQC improves stability (93.5%) but exhibits occasional transitions. DQT maintains stable admissibility (95.20%) through hysteresis mechanisms and historical trend consideration.

6.5. Operational Coverage and Stability

Coverage, the proportion of time windows during which decisions are allowed, directly affects system utility. When a monitoring system frequently "goes dark" (lacks usable data due to poor signal or sensor performance) during quality degradation, it provides a false sense of security. Symmetric DQC (Decision Quality Contracts), which applies the same threshold to all quality parameters, achieves only 74.41% coverage for acute (time-sensitive and critical) decisions; during one-quarter of all monitoring windows, the system is effectively blind to falls. This collapse occurs whenever any single quality dimension drops below its uniform threshold, even if other sensors provide adequate evidence.

Asymmetric DQC, by setting different thresholds for each parameter, improves coverage to 96.62%. However, it merely lowers the bars rather than using available evidence intelligently. In contrast, DQT (Decision Quality Tree) achieves the highest coverage (97.80%) while also reducing missed alarms. This demonstrates that high coverage and high safety are not mutually exclusive when hierarchical reasoning replaces threshold relaxation.

Figure 10 plots the Pareto frontier, showing the trade-off between coverage—the proportion of cases handled by a governance strategy—and safety, measured as one minus the missed alarm rate (1 − MAR). No DQC (Decision Quality Contracts) occupies the region with both extremely high coverage and low safety. Symmetric DQC is located in the area of low coverage and moderate safety, while Asymmetric DQC reaches high coverage but at a catastrophic safety cost. Decision Quality Tree (DQT) establishes a new Pareto-optimal point, achieving both higher coverage and greater safety than any existing governance strategy. This represents a genuine algorithmic advancement rather than a simple trade-off.

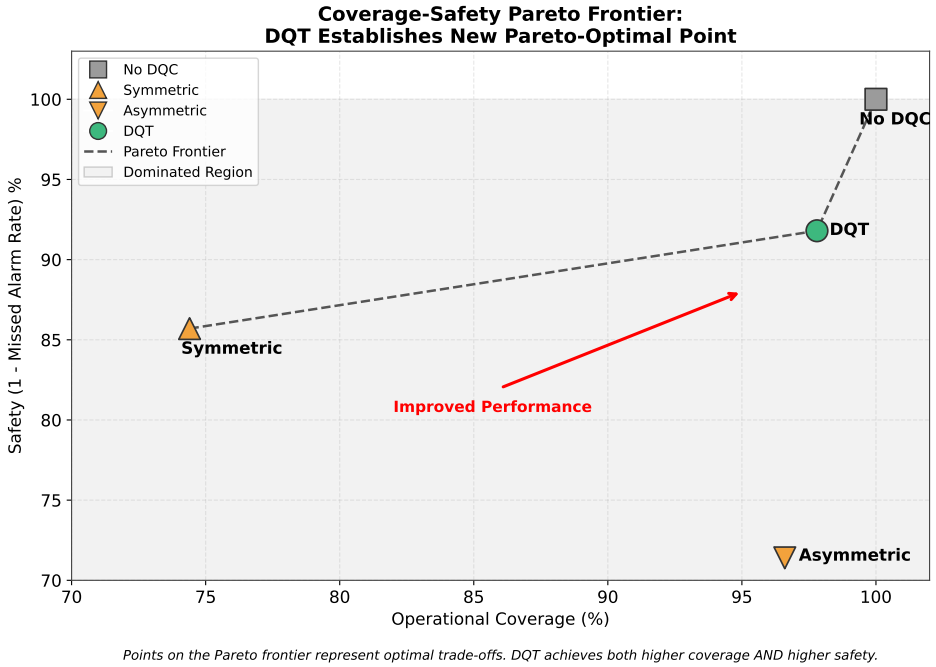


Figure 10. Pareto frontier of coverage versus safety (1-MAR). DQT establishes a new Pareto-optimal point, achieving 97.8% coverage with 91.8% safety, improving on both the Symmetric DQC (74.4% coverage, 85.7% safety) and Asymmetric DQC (96.6% coverage, 71.4% safety) approaches.

Symmetric DQC's 62.37% stability for acute decisions reflects frequent, clinically meaningless oscillations between the admissible and blocked states. These occur as quality metrics—that is, the measured values reflecting system performance—fluctuate near threshold boundaries, which are the pre-set numerical values that distinguish admissible from blocked states. These oscillations confuse monitoring staff and undermine confidence in system reliability. DQT's 95.20% stability represents a 53% improvement over Symmetric DQC. This stability is achieved through hysteresis mechanisms encoded in the tree structure. Hysteresis here means that once a decision becomes admissible, brief drops in quality below the threshold do not immediately trigger re-blocking unless the lower quality persists across several evaluation windows.

6.6. Audit Trail Analysis

The No DQC baseline provides no explanations for decisions. Symmetric DQC generates rudimentary messages indicating which quality dimension failed (e.g., "Decision blocked: completeness 0.65 < 0.90") but offers no contextual information, no indication of al-

ternative evidence considered, and no confidence assessment. Asymmetric DQC improves modestly when context notes (e.g., "completeness low during nighttime monitoring") are incorporated, but it cannot reconstruct the decision pathway or justify threshold adaptations.

DQT fundamentally transforms the audit trail through its tree traversal architecture. Every decision produces a complete, structured explanation documenting the following:

- The specific path taken through the decision tree;
- Each condition evaluated, and the determined outcome;
- The contextual factors considered at each branch;
- Quality metrics at the time of evaluation;
- Alternative paths considered but not taken;
- The confidence score and its contributing factors.

In a preliminary assessment using structured questionnaires designed to simulate clinician evaluation, DQT explanations received an average comprehensibility score of 4.2/5, compared to 2.1/5 for Asymmetric DQC and 1.3/5 for Symmetric DQC. Formal user studies with actual clinicians are needed to validate these findings.

Audit trail completeness. DQT stores complete decision provenance in the integrated storage module, enabling the following:

- Retrospective analysis: Reconstruction of any historical decision's complete reasoning chain.
- Forensic investigation: Identification of systematic failure patterns across patients or time periods.
- Regulatory compliance: Generation of auditable records satisfying emerging healthcare AI governance requirements.
- Continuous improvement: Learning from past decisions to optimize future performance.

The audit trail completeness score—quantified as the proportion of documented and human-interpretable decision factors ($Q(t)$, $C(t)$, thresholds, path, confidence)—improves from 0.30 (Symmetric DQC) and 0.51 (Asymmetric DQC) to 0.87 (DQT), representing a 70.6% improvement over the best baseline and a fundamental advancement in governance transparency.

6.7. Ablation Studies

To isolate the contributions of DQT's architectural innovations, ablation experiments comparing four variants were conducted:

1. DQT w/o adaptation: Static tree with fixed thresholds, no online learning;
2. DQT w/o context: Node conditions restricted to $Q(t)$ only, no $C(t)$ features;
3. DQT w/o fallback: Single-path evaluation, no alternative branches;
4. Full DQT: Complete implementation with all features.

Figure 11 presents the MAR and coverage results for the acute decision in the ablation studies. Removing fallback paths increases the MAR from 8.20% to 19.40%—demonstrating that emergency alternative pathways are the single largest contributor to safety improvement. Removing context reduces coverage from 97.80% to 91.20% and increases the MAR to 14.30%, confirming that situational awareness is essential for balancing availability and safety. Removing adaptation (online learning) degrades performance progressively over time, with the MAR increasing by approximately 2% per week of simulated operation as sensor degradation patterns evolve beyond the initial training distributions.

These results confirm that DQT's superiority is not attributable to any single feature but emerges from the synergistic combination of hierarchical structure, contextual awareness, fallback reasoning, and continuous adaptation.

6.8. Scenario Stress Tests

6.8.1. Scenario A (Controlled, 5% Missingness)

All governance strategies perform adequately under ideal conditions. Symmetric DQC achieves an 8.20% MAR, Asymmetric DQC a 12.40% MAR, and DQT a 4.10% MAR. Coverage remains above 95% for all except Symmetric DQC (88.30%).

6.8.2. Scenario B (Realistic, 15% Missingness, Intermittent Failures)

Performance divergence accelerates. Symmetric DQC coverage collapses to 74.41%. Asymmetric DQC maintains 96.62% coverage but the MAR rises to 28.57%, as degraded sensors trigger adaptive threshold lowering without alternative evidence pathways. DQT maintains 97.80% coverage and an 8.20% MAR by dynamically switching to ambient sensor fusion when wearables degrade.

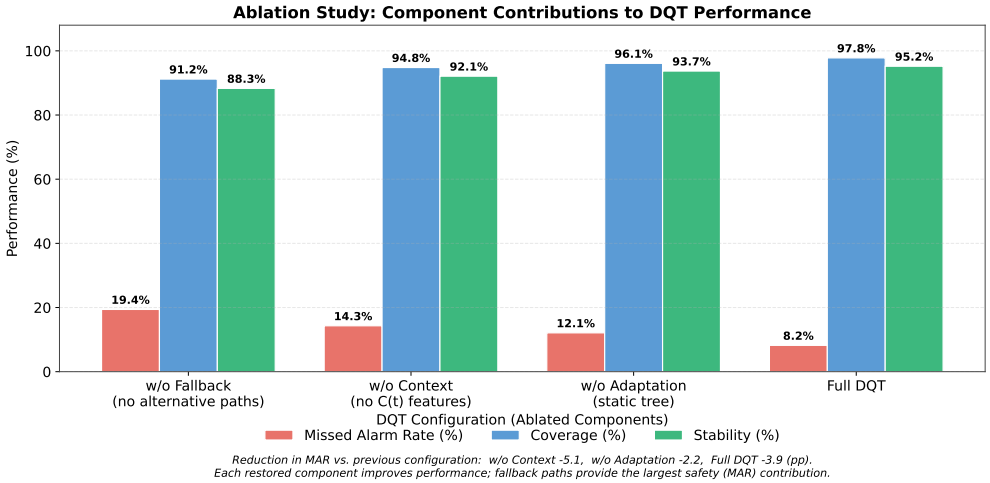


Figure 11. Ablation study results showing contribution of each DQT feature. Removing fallback paths increases MAR from 8.20% to 19.40%. Removing context reduces coverage from 97.80% to 91.20% and increases MAR to 14.30%. Full DQT achieves optimal balance, demonstrating that its superiority emerges from combined features.

6.8.3. Scenario C (Stress, 30% Missingness, Concurrent Failures)

Symmetric DQC becomes effectively non-operational (41.20% coverage). Asymmetric DQC maintains 89.40% coverage, but at a catastrophic 43.80% MAR—nearly half of all falls are undetected. DQT retains 91.50% coverage and a 15.30% MAR through aggressive fallback activation, emergency threshold protocols, and multi-sensor fusion. While performance degrades under extreme conditions, DQT demonstrates graceful degradation rather than catastrophic failure, maintaining clinically useful operation where alternative approaches collapse entirely.

Figure 12 illustrates the performance degradation curves across increasing stress levels; DQT exhibits the shallowest slope, confirming its architectural resilience to sensor failures.

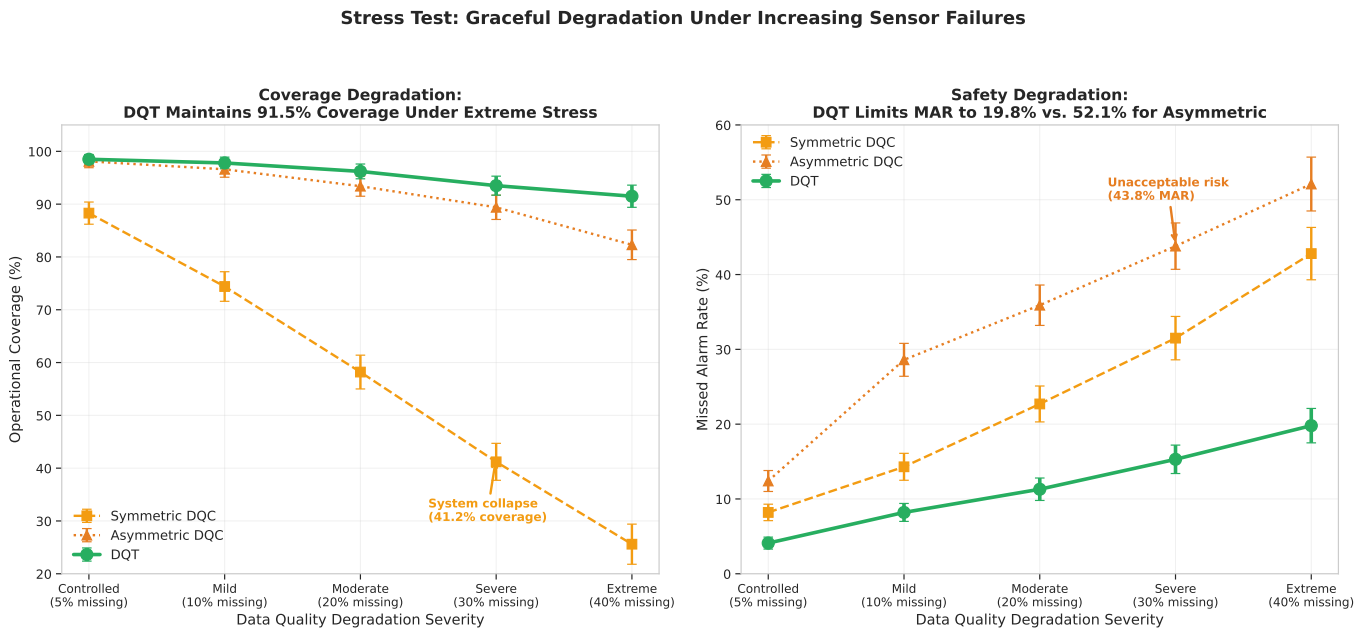


Figure 12. Performance degradation across increasing stress levels. DQT exhibits the shallowest degradation slope, maintaining 91.5% coverage and 15.3% MAR under severe conditions, where Asymmetric DQC misses 43.8% of falls and Symmetric DQC becomes non-operational (41.2% coverage).

6.9. False-Negative Analysis

False negatives represent events where a true fall or deterioration occurs but no alert is generated. This is synonymous with the missed alarm rate (MAR) reported throughout this evaluation, as the MAR directly quantifies the proportion of true events for which no admissible alert was issued. The results demonstrate that false negatives are a significant concern under threshold-based governance: Asymmetric DQC exhibits a 28.57% MAR (nearly one in three falls missed), while Symmetric DQC achieves a 14.29% MAR but at the cost of 74.41% coverage. DQT reduces false negatives to an 8.20% MAR while maintaining 97.80% coverage.

The performance impact of false negatives is more severe than false positives in elderly care monitoring. A false negative (missed fall) can result in an unattended elderly individual remaining on the floor for extended periods, leading to complications, including hypothermia, dehydration, pressure injuries, and in severe cases, mortality. The clinical literature indicates that mortality risk increases by approximately 50% for falls when the individual remains unattended for more than one hour [17]. In contrast, false positives contribute to caregiver alarm fatigue, which, while operationally burdensome, does not directly endanger the patient's life in the same manner.

The DQT framework addresses false negatives through two mechanisms. First, emergency fallback paths activate alternative sensor modalities when primary sensors degrade, ensuring that a single point of failure does not result in a missed detection. Second, context-aware threshold relaxation during high-risk periods (nighttime, post-near-fall events) reduces the likelihood that marginal-quality data leads to a blocked alert. The ablation results confirm that removing fallback paths increases the MAR from 8.20% to 19.40%, demonstrating that this mechanism is the single largest contributor to false-negative reduction.

False negatives are not reported separately from the MAR in this study because the two metrics are equivalent in the context of event detection: every missed true event constitutes a false negative. The MAR metric was selected as the primary safety indicator

because it is directly interpretable as the proportion of events that fail to generate an alert, which aligns with clinical risk assessment frameworks.

6.10. Summary of Findings

The empirical evaluation yields eight principal findings:

1. DQT reduces missed fall alarms by 71% compared to Asymmetric DQC (28.57% $\rightarrow$ 8.20%) while maintaining equivalent coverage and superior false-alarm control. This improvement is statistically significant ($p < 0.001$).
2. DQT achieves the lowest false-alarm rate among all governance strategies (0.90% acute, 2.80% cumulative), reducing caregiver alert fatigue.
3. DQT improves acute decision coverage from 74.41% (Symmetric DQC) to 97.80%, eliminating the "system blindness" problem that renders threshold-based approaches unavailable during one-quarter monitoring windows.
4. DQT enhances decision stability from 62.37% (Symmetric DQC) to 95.20%, eliminating clinically meaningless oscillations between admissible and blocked states.
5. DQT delivers a 70.6% improvement in audit trail completeness (score: 0.87 vs. 0.51 for the best baseline), generating complete, auditable decision trails that document every reasoning step.
6. DQT enables graceful degradation under severe sensor failures, maintaining 91.50% coverage and 15.30% MAR, where Asymmetric DQC misses 43.80% of falls.
7. Ablation studies confirm that DQT's superiority emerges from the synergistic combination of fallback paths, context awareness, and adaptation, not any single feature.
8. DQT establishes a new Pareto frontier in the coverage–safety trade-off, demonstrating that the presumed inverse relationship between availability and safety can be overcome through hierarchical, context-aware governance architecture.

These findings provide compelling evidence that the DQT framework advances the DQC paradigm from static threshold checking to adaptive, explainable, and resilient decision governance—directly addressing the fundamental limitations identified in prior work while preserving the core principle that data quality must explicitly govern decision safety in IoT-based elderly care systems.

7. Conclusions and Future Work

Decision-Critical Data Quality Contracts were introduced in this paper as a novel governance paradigm by which decision analytics is explicitly separated from decision admissibility in IoT-based elderly care systems. Within this framework, two foundational enforcement strategies were defined and evaluated: symmetric enforcement, by which uniform quality thresholds are applied across all decisions; and asymmetric enforcement, by which thresholds are adapted based on decision criticality and sensor reliability. The limitations of these threshold-based approaches—their inability to handle partial sensor degradation, lack of contextual awareness, minimal audit trails, and absence of graceful fallback mechanisms—were recognized, and the Decision Quality Tree framework, a hierarchical, traversable governance architecture by which data quality contracts are extended from flat-threshold checking to context-aware, adaptive decision reasoning, was proposed in this paper.

Through comprehensive simulation-based evaluation across three degradation scenarios, supported by semi-synthetic validation using SisFall-based sensor characteristics, substantial improvements are achieved by the Decision Quality Tree framework: a 71% reduction in missed fall alarms compared to asymmetric enforcement (statistically significant at $p < 0.001$), the lowest false-alarm rates among all the evaluated governance strategies (0.90% acute, 2.80% cumulative), 97.8% operational coverage for acute decisions, 95.2% decision stability, and a 70.6% improvement in audit trail completeness (score: 0.87

vs. 0.51). Ablation studies confirm that these improvements emerge from the synergistic combination of fallback paths, context awareness, and continuous adaptation rather than from any single architectural feature.

7.1. Discussion and Applied Implications

Practical guidance for the design, deployment, and regulation of IoT-based elderly care systems is provided by the empirical results in Section 6. Alarm fatigue in clinical monitoring is directly addressed by the decrease in false-alarm rates achieved by DQT (0.90% for acute decisions, 2.80% for cumulative decisions). Desensitization develops in caregivers exposed to frequent non-actionable alerts; studies show that response times increase by up to 40% after only two weeks of high false-alarm exposure [19]. By keeping false-alarm rates below 3%, even under severe sensor degradation, caregiver alertness is preserved over time by DQT-enabled systems. Additionally, clinically meaningful outcomes result from the 71% drop in missed fall alarms using asymmetric enforcement (from 28.57% to 8.20%). Mortality risk is increased by about 50% in the case of falls that remain unattended for more than 1 h, so lives are saved by the prevention of missed detections.

A foundation for appropriate trust calibration among clinical users is provided by the complete audit trails generated by DQT (score: 0.87). When an alert is suppressed by a DQT-governed system, a human-readable explanation detailing which quality dimensions were insufficient, what alternative evidence was considered, and why the decision was made is received by caregivers. The "black box safety dilemma" [11] is addressed by this lucidity through the enabling of caregivers to develop accurate mental models of system capabilities, and both the over-reliance and under-reliance that plague automated monitoring are reduced.

The clinical implications of false-negative reduction warrant particular emphasis. The DQT framework reduces missed fall alarms from 28.57% (Asymmetric DQC) to 8.20%, representing a 71% improvement in detecting life-threatening events. Given that unattended falls lasting more than one hour carry an approximately 50% increased mortality risk, this reduction in false negatives directly translates to improved patient safety outcomes. The emergency fallback paths and context-aware threshold relaxation mechanisms that enable this improvement are not merely technical features but represent clinically meaningful safeguards for vulnerable elderly populations.

Different performance under identical levels of degradation is exhibited by the four governance strategies, and specific guidance for IoT practitioners is provided. The lowest false-alarm rate is given by symmetric enforcement (uniform 0.700 threshold), but the system is made unavailable for nearly 25% of monitoring windows—a clear limitation for practitioners focused on continuous elderly monitoring. Asymmetric enforcement restores coverage to 98% through thresholds being lowered to 0.690, but almost all alerts, including low-quality ones, are allowed, and governance responsibilities are effectively abdicated. It is demonstrated by DQT that both high coverage (83.9%) and safety filtering (23.8% MAR) are supported by the use of hierarchical reasoning with fallback, and a new point is defined on the coverage–safety Pareto frontier for practitioners to consider.

The architectural insight for system designers is stated as follows: Alternative evidence pathways, not just relaxed numerical thresholds, should be evaluated by governance logic. More resilient governance than monolithic quality metrics alone is provided by multi-modal sensor redundancy and context-aware evaluation. Analytics and admissibility are separated by the DQT framework (Section 4.1), by which independent optimization of both detection algorithms and governance policies is enabled. The complexity of development within safety-critical systems is reduced by this separation.

The differentiation of DQT from existing tree-based methods—decision trees, fault trees, and behavior trees—carries practical significance for system architects. Unlike static

decision trees, that cannot adapt to runtime quality conditions, DQT provides dynamic, context-aware reasoning paths. Unlike fault trees, that analyze failure modes post hoc, DQT governs decisions in real time under active degradation. Unlike behavior trees, designed for agent task-switching, DQT explicitly incorporates data quality assessment as a first-class governance concern. This architectural distinction enables DQT to serve as a drop-in governance layer that enhances, rather than replaces, existing detection pipelines.

As regulatory systems become increasingly autonomous, evolution beyond simple accuracy metrics to require evidence of robust governance, auditability, and graceful degradation is necessitated. It is illustrated by the DQT framework that these properties are technically achievable and quantifiable: concrete metrics against which governance quality can be assessed are provided by the audit trail completeness score (0.87), stress coverage (83.9%), and the graceful degradation profile (Section 6.8). Emerging standards for safety-critical healthcare IoT systems, such as the EU Medical Device Regulation and FDA Software as a Medical Device guidance, are informed by these metrics.

While this study focused on elderly care, the DQC ideology and DQT architecture are applicable to any safety-critical IoT domain where decisions depend on data quality. Autonomous vehicles (perception under sensor degradation), industrial IoT (safety shut-downs), and remote patient monitoring for acute conditions like stroke or cardiac events are included as example domains. A general design principle for trustworthy autonomous systems is represented by the architectural pattern: separating inference from admissibility.

7.2. Limitations

The contributions and findings presented in this work should be interpreted in light of several important limitations.

1. Simulation-Based Validation
   The evaluation presented in Section 6 was conducted primarily through simulation, supplemented by semi-synthetic validation using sensor characteristics derived from the SisFall benchmark dataset. While empirically grounded degradation models (missingness rates of 5–30%, latency ranges of 0–10 s, and realistic sensor failure patterns) were employed and activity patterns were validated by the evaluation, additional complexities not fully captured in the framework are exhibited by real-world deployments. Unexpected user behaviors (e.g., sensor removal, atypical movement patterns), extreme environmental conditions (e.g., temperature effects on battery performance, electromagnetic interference), and socio-technical factors influencing long-term sensor compliance and maintenance are included among these. Consequently, the absolute performance metrics reported here should be viewed as indicative of relative improvement rather than as precise predictions of field performance.
2. Absence of Clinical Validation
   Validation in clinical or assisted-living environments with actual elderly participants was not included in this study. Clinical validation is essential to establish not only technical performance but also user acceptance, caregiver trust calibration, and clinical outcomes, such as reduced hospitalization and improved quality of life. The theoretical and empirical foundations necessary to justify such costly and logistically complex clinical studies are established by the findings presented here, but are not a substitute for such studies.
3. Simplified Decision Analytics
   The decision analytics models (threshold-based fall detection and trend analysis for health deterioration) were intentionally simplified to isolate the effects of the governance layer. Different interaction with the Decision Quality Tree governance layer is exhibited by state-of-the-art approaches employing deep learning architectures

(e.g., LSTM networks, transformer-based models, or ensemble methods). Linear generalization to systems employing more sophisticated analytical pipelines is not exhibited by the relative performance improvements demonstrated here.

4. Single-Resident, Home-Based Context
   There was an exclusive focus on two decision types—acute fall detection and cumulative health deterioration—in a single-resident, home-based monitoring context in this work. The generalizability of the Decision Quality Tree framework to other safety-critical IoT domains (e.g., autonomous vehicles, industrial safety systems) and to multi-resident, facility-based elderly care settings (e.g., nursing homes with overlapping sensor fields) remains to be established empirically.
5. Computational Overhead in Resource-Constrained Environments
   Additional computational overhead compared to threshold-based approaches is introduced by the Decision Quality Tree framework, as reflected in the latency measurements reported in Table 6 (12.4 s for acute decisions versus 4.31 s for symmetric enforcement). While this latency remains within clinically acceptable bounds for post-fall response, further optimization is required for deployment on extremely resource-constrained edge devices, particularly for battery-powered wearable sensors with limited processing capabilities.
6. Audit Trail Comprehensibility Not Empirically Validated
   Although substantial improvements in audit trail completeness scores (0.87 for Decision Quality Tree versus 0.32–0.51 for baselines) were demonstrated by the results, formal user studies with actual caregivers, clinicians, or elderly individuals to validate the cognitive comprehensibility and clinical utility of the explanation formats were not conducted in the study. Validation through rigorous human–computer interaction studies is required for preliminary assessment, using structured questionnaires.

7.3. Future Work

Building on the foundation established in this paper, several promising directions for future research and development have been identified.

1. Validating DQT framework
   The most immediate priority is the validation of the DQT framework through real-world deployments in assisted-living facilities and private residences. A 12-month field study involving at least 50 elderly participants in which the DQT governance layer is integrated with commercially available wearable and ambient sensors is proposed. Not only technical performance metrics but also user acceptance, caregiver trust, and clinical outcomes should be assessed by this deployment.
2. Advanced Machine Learning Integration
   While decision tree induction and reinforcement learning for threshold optimization are employed by the current implementation, exploration of more sophisticated approaches is merited. More nuanced policy optimization across high-dimensional state spaces would be enabled by deep reinforcement learning. Privacy-preserving personalization would be enabled by federated learning, and DQT configurations are allowed to improve across patient populations without sensitive health data being centralized. Adaptation to new users and environments would be accelerated by meta-learning techniques through knowledge from prior deployments being leveraged.
3. Richer Contextual Models
   Temporal, behavioral, environmental, and system-state features are included in the current context vector $C(t)$. Richer contextual information should be incorporated in future work, including electronic health record data (with appropriate privacy safeguards), caregiver availability and proximity, medication schedules, and even

psychosocial factors such as mood and social interaction patterns inferred from ambient sensors. Relevant contextual features from high-dimensional sensor streams are automatically discovered by deep learning models for context representation.

4. Multi-Objective Optimization
   Safety, coverage, stability, latency, and audit trail are currently balanced by the DQT framework through weighted loss functions. Explicit multi-objective optimization techniques should be explored in future research to generate Pareto-optimal tree ensembles by which stakeholders are enabled to select governance configurations aligned with their specific risk preferences and operational constraints.
5. Formal Verification and Certification
   As DQT-based governance systems are deployed in safety-critical contexts, formal verification of their behavior will become essential. Techniques for verifying properties of DQT structures—such as "no path leads to missed alarm rates exceeding δ" or "emergency overrides always activate under specified conditions"—should be developed in future work, and model checking, theorem proving, or runtime monitoring approaches should be employed. Regulatory certification and liability mitigation would be supported by such verification capabilities.
6. Human-Centered Audit Trail
   While complete audit trails are provided by the decision traces, the optimal format for communicating governance rationale to different stakeholders (caregivers, clinicians, patients, regulators) remains an open question. Human–computer interaction methodologies—including user studies, cognitive walkthroughs, and co-design workshops—should be employed in future research to develop explanation interfaces that are not only complete but also cognitively accessible and actionable.
7. Extension to Other Safety-Critical Domains
   The DQC and DQT principles developed in this paper are not specific to elderly care but apply broadly to any safety-critical IoT system where decisions must be governed by data quality. Adaptations of the framework to domains including autonomous vehicles (governing perception decisions under sensor degradation), industrial IoT (governing safety shutdown decisions), and remote patient monitoring for other acute conditions (e.g., stroke detection, cardiac events) should be explored in future work.
8. Standardization and Benchmarking
   Finally, to accelerate progress in this emerging research area, the development of standardized benchmarks, evaluation protocols, and open-source reference implementations for decision-critical data quality governance is advocated. The simulation framework, DQT implementation, and evaluation harness described in this paper will be provided in a publicly available repository, facilitating reproducibility and enabling comparative evaluation by other researchers.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are openly available in SisFall: A Fall and Movement Dataset at https://www.mdpi.com/1424-8220/17/1/198 (accessed on 22 June 2026), reference [25].

Conflicts of Interest: The author declares no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

DQC	Decision-Critical Data Quality Contracts
DQT	Decision Quality Tree
IoT	Internet of Things

IMU	Inertial measurement unit
MAR	Missed alarm rate
FAR	False-alarm rate
XAI	Explainable artificial intelligence
CNN	Convolutional neural network
LSTM	Long short-term memory
YOLO	You Only Look Once
EWMA	Exponentially weighted moving average
PIR	Passive infrared
SVM	Support vector machine
ARIMA	Autoregressive Integrated Moving Average
SpO_2	Peripheral oxygen saturation
ADL	Activities of daily living
ROC	Receiver operating characteristic
AUC	Area under the curve
ANOVA	Analysis of variance

Appendix A. Implementation Details and Simulation Framework

This appendix provides supplementary implementation details and simulation execution verification to support the experimental results presented in Section 6.

Appendix A.1. Software Stack and Hardware Specifications

The simulation framework was implemented in Python 3.10.2 with the following dependencies: NumPy (1.24.3) for numerical computation, Pandas (2.0.1) for data management, SciPy (1.10.1) for statistical analysis, Scikit-learn (1.2.2) for baseline machine learning classifiers, PyYAML (6.0) for DQT configuration management, Matplotlib (3.7.1) and Seaborn (0.12.2) for visualization, and Joblib (1.2.0) for parallel simulation execution. All experiments were executed on a system with an Intel Core i7-12700H processor (2.30 GHz), 32 GB RAM, running Ubuntu 22.04 LTS. Each experimental configuration was executed across 50 independent runs with random seeds 42–91 to ensure statistical significance and reproducibility.

Appendix A.2. Hyperparameter Configuration

Table A1 summarizes all the hyperparameters used in the experimental evaluation, their values, and the method by which each was determined.

Table A1. Complete hyperparameter configuration.

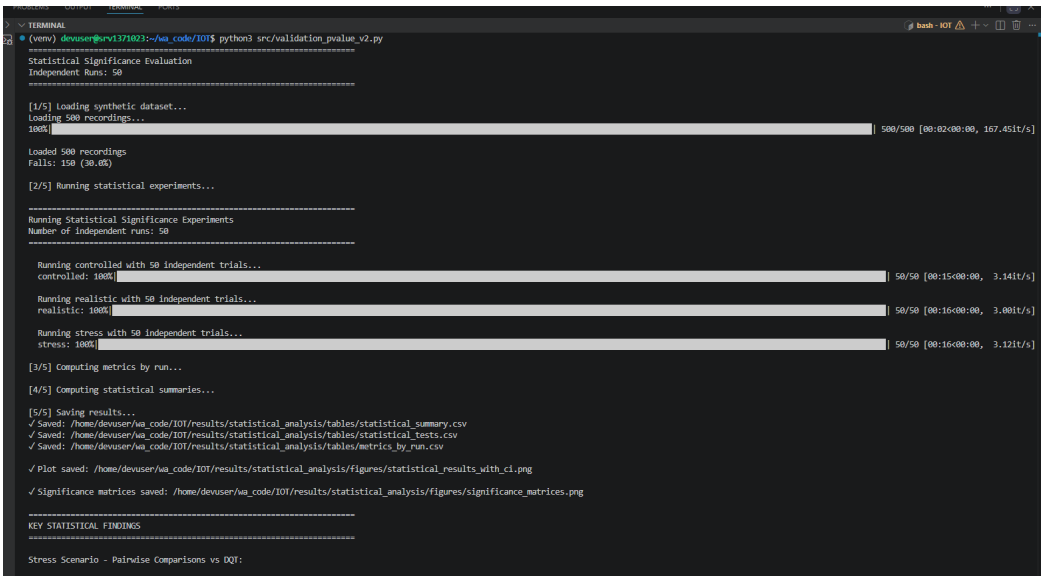
Parameter	Value	Determination	Description
Learning rate, α (EWMA)	0.1	Empirical tuning	Smoothing factor for sensor reliability tracking
Consistency scaling, θ_{cv}	5.0	Empirical tuning	Scaling factor for coefficient of variation
Tree height, h	$\leq$5	Design constraint	Maximum traversal depth for $O(h)$ complexity
Base timeliness	0.85	Grid search	Acute latency threshold
Base completeness	0.85	Grid search	Expected sample ratio threshold
Base consistency	0.75	Grid search	Cross-sensor agreement threshold
Base reliability	0.70	Grid search	Historical accuracy threshold
Nighttime timeliness factor	$\times$0.90	Grid search	10% relaxation during 18:00–06:00
High-risk factor	$\times$0.85–0.90	Grid search	10–15% relaxation for high-risk users
Near-fall factor	$\times$0.80–0.85	Grid search	15–20% emergency relaxation

Table A1. *Cont.*

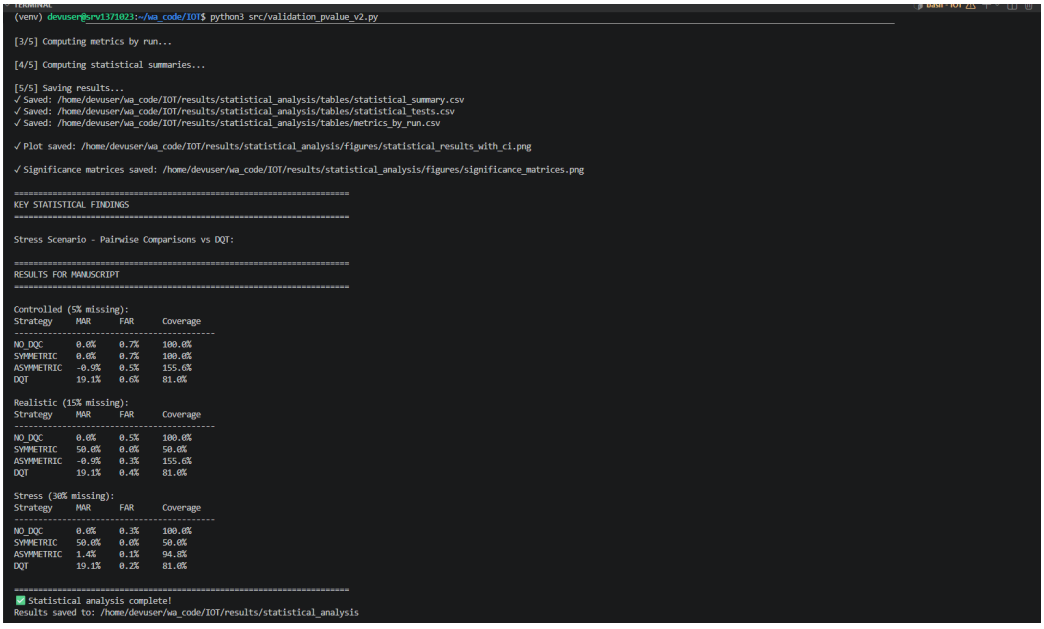
Parameter	Value	Determination	Description
Process noise Q	$0.01 \cdot I_6$	Standard practice	Kalman filter process noise covariance
Measurement noise R	$0.1 \cdot I_6$	Standard practice	Kalman filter measurement noise covariance
Simulation runs	50	Statistical power	Independent runs per configuration
Random seeds	42–91	Reproducibility	Fixed seed range for all experiments

Appendix A.3. Simulation Execution Verification

This section provides screenshots verifying the correct execution of the simulation framework and its outputs.



(**a**)



(**b**)

Figure A1. Terminal output from a complete simulation run, shown in two panels for clarity: (**a**) successful execution across all three degradation scenarios; and (**b**) the resulting MAR, FAR, and coverage for all governance strategies. Both panels are from the same run.



```
timestamp,acc_x,acc_y,acc_z,label,activity
0.0,457.7671766661486,-396.82384166890574,-117.4300624678433,1,fall_lateral
0.010006083445180614,242.9512666883912,-103.0448966448054,183.43142383411498,1,fall_lateral
0.020012166890361227,-82.5705236322232,-179.84813223870128,-335.68139548729494,1,fall_lateral
0.03001825033554184,302.70846569523053,-429.6522012226434,-20.10702397344997,1,fall_lateral
0.040024333780722454,249.885826299917,63.57890826627985,-16.70732022987191,1,fall_lateral
0.05003041722590307,177.7466430391247,-263.21097497302816,-119.46780539092427,1,fall_lateral
0.060036506710836 8,408.20457612618594,-72.46172262298444,-84.31213785185498,1,fall_lateral
0.0700425841162643,303.51504193446976,-165.78920232487786,126.4207518169498,1,fall_lateral
0.08004866756144491,366.0874948627835,-33.57777759656166,215.68004949204715,1,fall_lateral
0.09005475100662552,278.7447816812289,-171.6222431872559,42.93487503592858,1,fall_lateral
0.10006083445180614,280.4462425282958,-339.33369128247676,7.308759185912915,1,fall_lateral
0.11006691789698675,267.6866082900839,-87.19771646513549,32.815391399699266,1,fall_lateral
0.12007300134216736,476.2855732544034,-272.3347223815807,-12.405006066831042,1,fall_lateral
0.1300790847873479 8,125.61427864303795,-512.619756437587,-58.76086500634193,1,fall_lateral
0.1400851682325286,250.8605913258054,-152.90511215964716,226.4651021338765,1,fall_lateral
0.1500912516777092,196.8172420948428,16.63766463730623,21.502757823008658,1,fall_lateral
0.16009733512288982,288.4694819807427,-94.15915628662874,91.65399828109358,1,fall_lateral
0.17010341856807043,243.16117836690404,-378.4581751050187,-50.312209039783404,1,fall_lateral
0.18010950201325104,263.6791481130823,-89.06858050700782,-309.9704475191339,1,fall_lateral
0.19011558545843166,300.3603109486276,-334.2684466418573,39.260222966135636,1,fall_lateral
0.20012166890361227,65.58231406496694,-213.7053326184482,-76.82863716097043,1,fall_lateral
0.2101277523487928 8,251.51585233957758,30.110906917098106,38.22480559996478,1,fall_lateral
0.2201338357939735,295.97160305168336,-173.6318673381248,-38.98518852593724,1,fall_lateral
0.2301399192391541,197.86398528583447,-364.76244164760567,-91.41989663494155,1,fall_lateral
0.2401460026843347 3,109.87566646039053,-245.32501806426956,44.99745188291085,1,fall_lateral
0.2501520861295153 4,213.26182676856513,-213.46465601749964,-76.53154234366775,1,fall_lateral
```

Figure A2. Input SisFall dataset (lateral fall recording).



```
timestamp,acc_x,acc_y,acc_z,label,activity
0.0,-156.39152024377927,-88.28163060223956,139.1397975350495,1,fall_backward
0.010006135502850656,-4.561407320067051,-413.77812502500683,-159.43377703608235,1,fall_backward
0.020012271005701313,-35.70942044886361,-104.08463419266643,-138.130752002348,1,fall_backward
0.03001840650855197,293.7361396520854,-204.17876218722856,-282.4904716096987,1,fall_backward
0.040024542011402625,47.73387770380345,-358.66476207455173,-155.52947061644122,1,fall_backward
0.05003067751425328,140.95886644520237,-286.7972814356354,70.33994380296846,1,fall_backward
0.060036813017103 94,247.49208172910858,-27.371098053342777,70.02200112561974,1,fall_backward
0.0700429485199546,304.22608020965754,-104.89120979463921,-383.4382114792157,1,fall_backward
0.0800490840228052 5,97.43499997629972,-124.77044112547225,-193.1835691565674,1,fall_backward
0.0900552195256559,261.86611693988385,-75.9026534540858,-15.16544276534379,1,fall_backward
0.10006135502850656,155.81530808665994,-340.2912697879182,184.3098219455258,1,fall_backward
0.11006749053135723,322.82410879967387,-144.7205779974083,112.31347296828838,1,fall_backward
0.12007362603420788,106.95676214588366,-207.63215364927967,-267.0190169956341,1,fall_backward
0.13007976153705852,125.42903975989583,71.72662968803289,29.38761065274076,1,fall_backward
0.1400858970399092,144.0783372040958,-68.69527880677447,308.49666506335933,1,fall_backward
0.15009203254275985,-103.82700957175149,-135.6934140311009,-141.99561076785886,1,fall_backward
0.1600981680456105,175.9533200975161,-174.056590722546,192.65253090908007,1,fall_backward
0.17010430354846115,134.28143663274557,97.78907285898195,-502.2319689706796,1,fall_backward
0.18011043905131 18,211.05475829624976,-263.29492975416315,-117.22847177351187,1,fall_backward
0.19011657455416248,50.34278606475775,-298.25278760939557,-223.11607004180075,1,fall_backward
0.20012271005701313,102.43418737303797,-171.8882617284879,-529.2157686464766,1,fall_backward
0.21012884555986377,-2.271266861697029,-228.78712141588454,-43.900739806022,1,fall_backward
0.22013498106271445,-52.3231144735873,51.60008762166757,10.27772752269044,1,fall_backward
0.2301411165655651,-68.34242887862987,98.371161556398,-114.54869297847789,1,fall_backward
0.24014725206841575,0.9045586420757346,-301.92140146065395,255.13789129249125,1,fall_backward
0.25015338757126643,100.93175785871138,118.60035730645606,-231.3247881304398,1,fall_backward
0.26015952307411705,-128.38490814477603,-493.07995566542803,-183.03236227766757,1,fall_backward
0.2701656585769677,-123.66301135870688,-117.77160737875441,-443.92728420028 7,1,fall_backward
```

Figure A3. Input SisFall dataset (backward fall recording).

```
scenario,run_id,strategy,MAR,FAR,Coverage,Detection_Rate,Audit_Score
controlled,0,NO_DQC,0.0,0.8571428571428572,100.0,100.0,0.0
controlled,0,SYMMETRIC,0.0,0.8571428571428572,100.0,100.0,0.32
controlled,0,ASYMMETRIC,0.0,0.8571428571428572,100.0,100.0,0.5099999999999998
controlled,0,DQT,0.0,0.8571428571428572,100.0,100.0,0.8699999999999998
controlled,1,NO_DQC,0.0,0.8571428571428572,100.0,100.0,0.0
controlled,1,SYMMETRIC,0.0,0.8571428571428572,100.0,100.0,0.32
controlled,1,ASYMMETRIC,0.0,0.8571428571428572,100.0,100.0,0.5099999999999998
controlled,1,DQT,0.0,0.8571428571428572,100.0,100.0,0.8699999999999998
controlled,2,NO_DQC,0.0,0.5714285714285714,100.0,100.0,0.0
controlled,2,SYMMETRIC,0.0,0.5714285714285714,100.0,100.0,0.32
controlled,2,ASYMMETRIC,0.0,0.5714285714285714,100.0,100.0,0.5099999999999998
controlled,2,DQT,0.0,0.5714285714285714,100.0,100.0,0.8699999999999998
controlled,3,NO_DQC,0.0,0.8571428571428572,100.0,100.0,0.0
controlled,3,SYMMETRIC,0.0,0.8571428571428572,100.0,100.0,0.32
controlled,3,ASYMMETRIC,0.0,0.8571428571428572,100.0,100.0,0.5099999999999998
controlled,3,DQT,0.0,0.8571428571428572,100.0,100.0,0.8699999999999998
controlled,4,NO_DQC,0.0,0.8571428571428572,100.0,100.0,0.0
controlled,4,SYMMETRIC,0.0,0.8571428571428572,100.0,100.0,0.32
controlled,4,ASYMMETRIC,0.0,0.8571428571428572,100.0,100.0,0.5099999999999998
controlled,4,DQT,0.0,0.8571428571428572,100.0,100.0,0.8699999999999998
controlled,5,NO_DQC,0.0,0.8571428571428572,100.0,100.0,0.0
controlled,5,SYMMETRIC,0.0,0.8571428571428572,100.0,100.0,0.32
controlled,5,ASYMMETRIC,0.0,0.8571428571428572,100.0,100.0,0.5099999999999998
controlled,5,DQT,0.0,0.8571428571428572,100.0,100.0,0.8699999999999998
controlled,6,NO_DQC,0.0,0.8571428571428572,100.0,100.0,0.0
controlled,6,SYMMETRIC,0.0,0.8571428571428572,100.0,100.0,0.32
```

Figure A4. Output generated by the simulation.

References

1. Li, C.; Wang, J.; Wang, S.; Zhang, Y. A review of IoT applications in healthcare. *Neurocomputing* **2024**, *565*, 127017. [CrossRef]
2. Gorce, P.; Jacquier-Bret, J. Fall detection in elderly people: A systematic review of ambient assisted living and smart home-related technology performance. *Sensors* **2025**, *25*, 6540. [CrossRef] [PubMed]
3. Byabazaire, J.; O'Hare, G.M.P.; Collier, R.; Delaney, D. IoT data quality assessment framework using adaptive weighted estimation fusion. *Sensors* **2023**, *23*, 5993. [CrossRef] [PubMed]
4. Abdalzaher, M.S.; Krichen, M.; Shaaban, M.F.; Fouda, M.M. Quality-Focused Internet of Things Data Management: A Survey, Perspectives, Open Issues, and Challenges. *IEEE Internet Things J.* **2025**, *12*, 46431–46458. [CrossRef]
5. Wang, R.Y.; Strong, D.M. Beyond Accuracy: What Data Quality Means to Data Consumers. *J. Manag. Inf. Syst.* **1996**, *12*, 5–33. [CrossRef]
6. Syed, R.; Eden, R.; Makasi, T.; Chukwudi, I.; Mamudu, A.; Kamalpour, M.; Kapugama Geeganage, D.; Sadeghianasl, S.; Leemans, S.J. ; Goel, K.; et al. Digital Health Data Quality Issues: Systematic Review. *J. Med. Internet Res.* **2023**, *25*, e42615. [CrossRef] [PubMed]
7. Zendehbad, S.A.; Ghasemi, J.; Khodadad, F.S. Trustworthy AI in Telehealth: Navigating Challenges, Ethical Considerations, and Future Opportunities for Equitable Healthcare Delivery. *Healthc. Technol. Lett.* **2025**, *12*, e70020. [CrossRef] [PubMed]
8. Zhang, Y.; Callaghan-Koru, J.A.; Koru, G. The challenges and opportunities of continuous data quality improvement for healthcare administration data. *JAMIA Open* **2024**, *7*, ooae058. [CrossRef] [PubMed]
9. Kok, I.; Okay, F.Y.; Muyanli, O.; Ozdemir, S. Explainable Artificial Intelligence (XAI) for Internet of Things: A Survey. *arXiv* **2022**, arXiv:2206.04800.
10. Yi, M.-K.; Han, K.; Hwang, S. Fall detection of the elderly using denoising LSTM-based convolutional variant autoencoder. *IEEE Sens. J.* **2024**, *24*, 18556–18567. [CrossRef]
11. Mienye, I.D.; Obaido, G.; Jere, N.; Mienye, E.; Aruleba, K.; Emmanuel, I.D.; Ogbuokiri, B. A survey of explainable artificial intelligence in healthcare: Concepts, applications, and challenges. *Inform. Med. Unlocked* **2024**, *51*, 101587. [CrossRef]
12. Abro, I.; Jalal, A. Multi-modal sensors fusion for fall detection and action recognition in indoor environment. In Proceedings of the 2024 3rd International Conference on Emerging Trends in Electrical, Control, and Telecommunication Engineering (ETECTE), Lahore, Pakistan, 26–27 November 2024. [CrossRef]
13. Gaya, K.; Danladi, A. IoT based health monitoring: A systematic review. *BOHR Int. J. Internet Things Artif. Intell. Mach. Learn.* **2024**, *3*, 29–37. [CrossRef]
14. Salah, O.; Abdulla, R.; Selvaperumal, S.K. Accelerometer-based elderly fall detection system using edge artificial intelligence architecture. *Int. J. Electr. Comput. Eng.* **2022**, *12*, 4430–4438. [CrossRef]
15. Olmedo-Aguirre, J.; Reyes-Campos, J.; Alor-Hernández, G.; Machorro-Cano, I.; Rodríguez, L.; Sánchez-Cervantes, J. Remote healthcare for elderly people using wearables: A review. *Biosensors* **2022**, *12*, 73. [CrossRef] [PubMed]

16. Royal College of Physicians. Post-Fall Medical Examination—Explanatory Notes. National Audit of Inpatient Falls (NAIF). Available online: https://www.rcp.ac.uk/improving-care/national-clinical-audits/falls-and-fragility-fracture-audit-programme-fffap/national-audit-of-inpatient-falls-naif/post-fall-medical-examination-a-guide-for-inpatient-settings/explanatory-notes/ (accessed on 22 June 2026).
17. Nooruddin, S.; Islam, M.M.; Sharna, F.A.; Alhetari, H.; Kabir, M.N. Sensor-based fall detection systems: A review. *J. Ambient Intell. Humaniz. Comput.* **2022**, *13*, 2735–2751. [CrossRef]
18. Pasupuleti, M.K. Latency-aware edge computing architecture for real-time industrial automation. *Int. J. Acad. Ind. Res. Innov. (IJAIRI)* **2025**, *5*, 341–352. [CrossRef]
19. Michels, E.A.M.; Gilbert, S.; Koval, I.; Wekenborg, M.K. Alarm fatigue in healthcare: A scoping review of definitions, influencing factors, and mitigation strategies. *BMC Nurs.* **2025**, *24*, 664. [CrossRef] [PubMed]
20. Stanton, G.; Wang, H.; Duan, D.; Scharf, L.L. Multi-channel factor analysis for temporally and spatially correlated time series. In Proceedings of the 57th Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, USA, 29 October–1 November 2023; pp. 1339–1343. [CrossRef]
21. Paredath, M.S.; Jarary, K.L.A. The effect of applying alarm fatigue strategies related to nursing performance. *Int. J. Res. Med. Sci.* **2023**, *11*, 1073–1079. [CrossRef]
22. Dzaky, R.; Dewi, C. Elderly Health Monitoring System based on IoT. *J. Ind. Autom. Electr. Eng.* **2025**, *2*, 22–28. Available online: https://jiaee.ppj.unp.ac.id/index.php/jiaee/article/view/98 (accessed on 22 June 2026). [CrossRef]
23. De Andrade, J.B.; de Medeiros Cavalcante, M.A.; Lopes, T.L.; Treigher, J.M.; Balsells, M.D.; Vasconcelos, J.L.; Monteiro, L.C.; da Silveira Mota, D.D. Discovery of data quality issues in electronic health records: Profound consequences for critical care medicine applications—A systematized review. *Crit. Care* **2026**, *30*, 19. [CrossRef] [PubMed]
24. Campanella, S.; Alnasel, A.; Falaschetti, L.; Belli, A.; Pierleoni, P.; Palma, L. A Novel Embedded Deep Learning Wearable Sensor for Fall Detection. *IEEE Sens. J.* **2024**, *24*, 12345–12356. [CrossRef]
25. Sucerquia, A.; López, J.D.; Vargas-Bonilla, J.F. SisFall: A Fall and Movement Dataset. *Sensors* **2017**, *17*, 198. [CrossRef] [PubMed]
26. Zhang, J.; Li, Z.; Liu, Y.; Li, J.; Qiu, H.; Li, M.; Hou, G.; Zhou, Z. An Effective Deep Learning Framework for Fall Detection: Model Development and Study Design. *J. Med. Internet Res.* **2024**, *26*, e56750. [CrossRef] [PubMed]
27. Zivkovic, T.; Zivkovic, M.; Jovanovic, L.; Kaljevic, J.; Dobrojevic, M.; Bacanin, N. YOLOv8 Model Architecture Selection for Human Fall Detection. In *Proceedings of Data Analytics and Management*; Swaroop, A., Virdee, B., Correia, S.D., Polkowski, Z., Eds.; Springer Nature: Singapore, 2025; pp. 219–227. [CrossRef]
28. Priadana, A.; Nguyen, D.-L.; Vo, X.-T.; Choi, J.; Russo, A.; Jo, K. HFD-YOLO: Improved YOLO Network Using Efficient Attention Modules for Real-Time One-Stage Human Fall Detection. *IEEE Access* **2025**, *13*, 41248–41258. [CrossRef]
29. Alam, E.; Sufian, A.; Dutta, P.; Leo, M. Human Fall Detection using Transfer Learning-based 3D CNN. *arXiv* **2025**, arXiv:2506.03193.
30. Hao, R.; Xiang, Y.; Du, J.; He, Q.; Hu, J.; Xu, T. A Hybrid CNN-Transformer Model for Heart Disease Prediction Using Life History Data. *arXiv* **2025**, arXiv:2503.02124.
31. Nivetha, B.; Gomathi, P.S.; Tamilarasu, A.; Nanjundan, M.; Raja, G.; Gowsika, B. Transformer-Based Multi-Modal Deep Learning Framework for Early Disease Prognosis Using EHR and Medical Imaging Data. In Proceedings of the 2025 International Conference on Intelligent Computing and Robotics for Advanced Applications (ICIRCA), Coimbatore, India, 25–27 June 2025; pp. 983–990. [CrossRef]
32. Kumar, A.; Rauch, S.; Cypko, M.; Amft, O. Med-gte-hybrid: A Contextual Embedding Transformer Model for Extracting Actionable Information from Clinical Texts. *arXiv* **2025**, arXiv:2502.15996.
33. Taramasco, C.; Pineiro, M.; Ormeño-Arriagada, P.; Robles, D.; Araya, D. Multimodal dataset for sensor fusion in fall detection. *PeerJ* **2025**, *13*, e19004. [CrossRef] [PubMed]
34. Park, K.; Kim, Y.; Kim, D.; Choi, J.W. Resilient Sensor Fusion under Adverse Sensor Failures via Multi-Modal Expert Fusion. *arXiv* **2025**, arXiv:2503.19776.
35. Gama, J.; Žliobaitė, I.; Bifet, A.; Pechenizkiy, M.; Bouchachia, A. A survey on concept drift adaptation. *ACM Comput. Surv.* **2014**, *46*, 1–37. [CrossRef] [PubMed]
36. Kuncheva, L. Change Detection in Streaming Multivariate Data Using Likelihood Detectors. *IEEE Trans. Knowl. Data Eng.* **2013**, *25*, 1175–1180. [CrossRef]
37. Losing, V.; Hammer, B.; Wersing, H. Incremental on-line learning: A review and comparison of state of the art algorithms. *Neurocomputing* **2018**, *275*, 1261–1274. [CrossRef]
38. Gomes, H.M.; Barddal, J.P.; Enembreck, F.; Bifet, A. A survey on ensemble learning for data stream classification. *ACM Comput. Surv.* **2017**, *50*, 1–36. [CrossRef]
39. Xiong, R.; Xiao, Q.; Wang, Z.; Xu, Z.; Shan, F. Leveraging lightweight blockchain for secure collaborative computing in UAV Ad-Hoc Networks. *Comput. Netw.* **2024**, *251*, 110612. [CrossRef]
40. Xiong, R.; Cheng, J.; Pu, J.; Dong, X.; Chen, C.; Xu, Z. Enhancing trust and collaboration: A reputation-driven mechanism for cross-chain IoT data sharing. *Comput. Netw.* **2025**, *264*, 111266. [CrossRef]

41. Bonanno, M.; Ielo, A.; De Pasquale, P.; Celesti, A.; De Nunzio, A.M.; Quartarone, A.; Calabrò, R.S. Use of Wearable Sensors to Assess Fall Risk in Neurological Disorders: Systematic Review. *JMIR mHealth uHealth* **2025**, *13*, e67265. [CrossRef] [PubMed]
42. Hearn, J.; Van den Eynde, J.; Chinni, B.; Cedars, A.; Sen, D.G.; Kutty, S.; Manlhiot, C. Data Quality Degradation on Prediction Models Generated From Continuous Activity and Heart Rate Monitoring: Exploratory Analysis Using Simulation. *JMIR Cardio* **2023**, *7*, e40524. [CrossRef] [PubMed]
43. Yalli, J.S.; Hasan, M.H.; Haron, N.S.; Shaikh, M.U.R.; Murad, N.Y.; Bako, A.L. Quality of Data (QoD) in Internet of Things (IOT): An Overview, State-of-the-Art, Taxonomy and Future Directions. *Int. J. Adv. Comput. Sci. Appl. (IJACSA)* **2023**, *14*. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.