GPH-International Journal of Applied Science
(Online ISSN 3050-9653)
Volume 09 | Issue 09 | 2026
Published by: Global Publication House
DOI: https://doi.org/10.5281/zenodo.23617262
Available online at: www.gphjournal.org
Journal homepage: https://gphjournal.org/index.php/as/index

Statistical Detection of Governance Failure in Production AI

Estimators, Sequential Tests, and Power Analysis over the Governed Decision Record

Luca Andrea Bonetti[1*] and Priya Ramesh Ramachandran[2]
[1, 2] *Politecnico di Torino, Italy*
* Corresponding author

Abstract

A governance failure in a production AI system is usually discovered after it has already changed the state of the world, and is then described in prose rather than measured. This paper argues that the control obligations established by Mesbaul Haque Sazu in the Governed Decision-Intelligence (GDI) framework [14], together with the enterprise governance framework that carries them into production [13], already emit, as a by-product of their operation, a structured event stream from which governance failure can be detected statistically and in near real time. The GDI framework makes the individual decision the unit of assurance and binds a Decision Record at generation time [14]; the governed-execution layer captures an Execution Record at commit and logs the propensity of every routing choice [13]. We take those two records as a joined observation and treat each governed decision as one sample in a monitored stochastic process. On that foundation we develop a detection layer: point and interval estimators for four quantities that these invariants make observable but that the frameworks do not themselves cast as random variables, specifically the recorded-refusal rate, the freshness-failure rate at commit, the calibration drift of the routing signal, and the propensity shift that signals drift-under-action. For each we give an estimator, its sampling distribution, a sequential test with a controlled false-alarm rate, and a power analysis that states how many governed decisions must accrue before a deviation of a given size is detectable. A stream of 60,000 governed decisions drawn from publicly available data in open-source platforms illustrates the operating characteristics of the tests; it shows how the detectors behave rather than standing as a validated field deployment. The contribution is a measurement discipline that sits on top of the governance control structure, not a replacement for it; every quantity we estimate is defined only because these frameworks require the evidence that makes it computable.

Keywords:

AI governance; Governed Decision-Intelligence; statistical process control; sequential testing; calibration drift; drift-under-action; power analysis; decision records; production AI; model risk management.

© 2026 Author(s). Published by Global Publication House (GPH).
This is an Open Access article distributed under the terms of the CC BY 4.0 License.

Highlights

- Two governance frameworks, GDI [14] and the enterprise governance framework that carries it into production [13], already produce a decision-indexed evidence stream; this paper reads that stream as a monitored stochastic process.
- Four governance quantities that these frameworks' invariants make observable (recorded-refusal rate, commit-time freshness-failure rate, routing-signal calibration drift, and automation-propensity shift) are cast as random variables.
- Each quantity receives a boundary-aware estimator, a cumulative-sum test with a governed false-alarm rate, and a power analysis expressed in governed decisions.
- A 60,000-decision stream drawn from publicly available data in open-source platforms shows the ordering and relative latency of the detectors.
- The method is strictly additive: every estimate exists only because the underlying frameworks require the evidence behind it.

1 Introduction

1.1 From governed decisions to a monitored process

The frameworks that Sazu published reorganised the governance of production AI around a single idea: assurance belongs to the decision, not to the model [14]. The Governed Decision-Intelligence (GDI) framework states the conditions under which an individual decision may be trusted [14], and the enterprise governance framework that extends it states the conditions under which a runtime may safely act on that decision [13]. Between them, the two frameworks specify a complete control structure, running from source and lineage registration through veracity gating, independent validation, generation-time evidence binding, confidence-gated oversight, capture-at-commit, and a closed outcome loop [14], [13].

Those frameworks are, by their own framing, normative and architectural: they say what must be true and where each obligation is enforced [14], [13]. What they do not claim to be is a statistical monitoring theory. Yet a system that satisfies their invariants is, almost incidentally, instrumented to an unusually high standard. Every acted-upon decision leaves a DecisionRecord bound at generation time; every commit leaves an ExecutionRecord joined by a shared identifier; every routing choice logs its propensity; every refused commit is recorded as a first-class event rather than silently dropped [13]. The output of a governed system is therefore not only a sequence of decisions but a sequence of fully attributed observations.

It is worth dwelling on why this matters in ordinary operational terms. Most teams running AI in production are rich in logs and poor in answers, a pattern well documented in surveys of machine-learning deployment [12]. They can tell you what every service did last Tuesday, but they struggle to tell you whether the system as a whole is still behaving the way it was signed off to behave. The reason is that their logs were designed to debug software, not to defend decisions. The governed records are different in kind: they were designed from the start to make a decision reconstructable and defensible [14]. That design choice is what makes statistical monitoring possible, and it is why this paper could not have been written against an ordinary logging stack.

This paper takes that observation seriously and asks a question these frameworks invite but do not answer: if a governed system emits a structured, decision-indexed event stream, what can

be estimated and tested from it, and with what statistical guarantees? We treat the governed decision stream as a monitored stochastic process and build a detection layer on top of the governance control layer.

1.2 Why detection, and why now

The frameworks are explicit that a system can satisfy every clause of their admissibility conditions and still act badly, because a threshold can be set against the wrong cost model, a validation population can fail to represent the deployed population, and reviewers can ratify escalations under load rather than adjudicating them [14], [13]. These are not failures of the control structure; they are failures that occur inside a formally satisfied control. They do not announce themselves. They show up as slow drift in a rate, a gradual loss of calibration, or a quiet shift in the propensity with which the system routes to automation.

A useful way to picture this is to think of a thermostat whose sensor is slowly going out of calibration. Nothing in the wiring is broken, every component reports that it is working, and yet the room is drifting colder by a fraction of a degree each week. No single reading looks wrong. Only the trend reveals the problem, and only if someone is watching the trend with the right instrument. Governance failure in a governed AI system has the same character. The controls are all in place and all reporting success, while the thing they protect is slipping. Catching that requires measuring the trend, not inspecting the components.

Drift of this kind is exactly what statistical process control was built to catch [8], and the concept-drift literature has catalogued the many forms it takes in deployed learning systems [5], provided the process is observable. Supervisory model-risk guidance [3] and the NIST AI Risk Management Framework [10] both expect deployed models to be monitored on an ongoing basis, but neither specifies the statistical form that monitoring should take for a system that acts. The governance frameworks make the process observable [14], [13]. The missing piece is a set of estimators and tests tuned to the specific quantities their invariants expose, together with an honest account of how much data each test needs before it can distinguish a real deviation from sampling noise. Supplying that piece is the purpose of this paper.

1.3 Relationship to the foundation and contributions

This paper is additive and strictly dependent. It proposes no new governance vocabulary and reuses the established terms exactly: named decision, veracity gating, evidence binding, capture-at-commit, logged propensity, recorded refusal, and drift-under-action all keep the meanings assigned to them in the GDI framework [14] and the enterprise governance framework [13]. The detection layer is meaningful only because those controls produce the evidence it consumes. Against that foundation the paper makes four contributions.

- **C1.** It casts four of these observable governance quantities, namely the recorded-refusal rate, the commit-time freshness-failure rate, the routing-signal calibration drift, and the automation-propensity shift, as random variables over the governed decision stream, with an explicit sampling model for each (Section 4).
- **C2.** It gives a point estimator, a variance, and a confidence interval for each quantity, choosing interval forms that behave correctly near the boundaries where governance rates typically sit (Section 5).

- **C3.** It specifies a sequential test for each quantity with a controlled false-alarm rate, so that drift is flagged as early as the data allow rather than on a fixed reporting cadence (Section 6).
- **C4.** It supplies a power analysis that converts a tolerable deviation into a required number of governed decisions, making the monitoring programme costable in advance (Section 7), and illustrates the operating characteristics on a stream drawn from publicly available data in open-source platforms (Section 8).

1.4 Scope

This is a methods paper. No production deployment is reported, and the illustration of Section 8 draws on publicly available data in open-source platforms, with its sources and processing specified in full in Appendix A. Nothing here demonstrates that governed systems drift less than ungoverned ones; it shows only that, when a system is governed to the standard these frameworks set [14], [13], drift becomes a measurable and testable quantity. The empirical question, which is whether these tests catch real governance failures earlier than existing review cadences, is the field programme that Section 9 is written to make obtainable.

2 The Foundation: The Governed Decision Record

2.1 Two records, one observation

Sazu's GDI framework binds a DecisionRecord at generation time, carrying the decision identifier, the model and policy versions, the prediction, the calibrated confidence, and a reference to the deterministic explanation [14]. The governed-execution layer binds an ExecutionRecord at commit, carrying the selected action, the policy verdicts, the validity signal, the human-in-the-loop state, and the commit timestamp [13]. The two share an identifier, so for each governed decision we observe a joined tuple. It is this tuple, accruing one row per decision, that we treat as the monitored process. No field in it is introduced by this paper; each is mandated by one of the two frameworks.

In less formal terms, the first record answers the question of why this decision was trusted, and the second answers the question of how it was carried out. Because they are stamped with the same identifier, they can always be read side by side. That pairing is the single raw material this paper works with. Everything that follows is a way of reading many such pairs in sequence and noticing when the pattern changes.

2.2 Logged propensity and recorded refusal as statistical primitives

Two of these obligations are doing more statistical work than their names suggest. Logged propensity, which means recording for every routing choice the probability with which the system would have taken each branch [13], is the standing obligation that makes off-policy estimation feasible at all, a point the reinforcement-learning and machine-learning safety literature has long emphasised [1], [7]. Recorded refusal, the requirement that a commit blocked by the admissibility check be logged as a first-class event with its failing condition, is what converts the refusal rate from an unobservable into an observable. The governance framework notes that a system with no record of its own refusals cannot report its deviation count, and that a deviation count of zero is then uninformative rather than reassuring [13]. We take that remark as the starting point for Section 4: a quantity is only estimable if the control that produces it is in place.

2.3 Decision quality rather than model accuracy

The third invariant makes decision quality, rather than model performance, the assurance target [14]. The quantities monitored here inherit that stance directly: none of them is a ranking metric, and a model can improve its AUC while its refusal rate collapses or its routing signal decalibrates. Keeping the monitored quantities separate from model metrics is therefore not a stylistic choice but a requirement carried over from that foundation, and Section 4 preserves it. The practical warning for a reader is simple: a rising accuracy score is not evidence that governance is healthy, and a team that watches only model metrics is watching the wrong dial.

3 Statistical Background

3.1 Control charting and sequential analysis

The machinery we apply is classical. Shewhart-style control charting and the exponentially-weighted approaches to small persistent shifts are standard in statistical quality control [8]; the cumulative-sum scheme of Page [11] and Wald's sequential probability ratio test [15] provide the sequential detection scaffolding. None of this is new as statistics. What is new is its attachment to the governance control boundaries [14], [13], so that each chart monitors a quantity the governance framework has already defined and made auditable, rather than an arbitrary operational metric. Statistical quality control was invented to keep physical manufacturing lines within tolerance. Applying it to a stream of governed decisions treats the production of decisions as a process to be kept in tolerance, which is close to how the frameworks already frame the problem.

3.2 Interval estimation for rare proportions

Governance rates are proportions, and they tend to sit close to zero or one. The score interval introduced by Wilson [16] and the comparative analysis of Brown, Cai, and DasGupta [4] establish that the textbook Wald interval loses its nominal coverage in exactly that region, while the Wilson interval retains it. We adopt the Wilson form throughout for that reason.

3.3 Measuring calibration

Binned calibration error, in the form popularised for modern neural networks by Guo et al. [6] and building on the Bayesian-binning approach of Naeini et al. [9], is the measurement that GDI's calibration obligation calls for [14]. Distribution-free methods such as conformal prediction [2] offer a complementary route to uncertainty statements and are compatible with the detection layer, but we do not rely on them here.

4 The Governed Decision Stream as an Observation Model

4.1 Notation

Let decisions arrive indexed by k = 1, 2, ..., each producing a joined record from the DecisionRecord and ExecutionRecord of Section 2.1. For decision k write v_k for the calibrated routing risk signal, $g_k \in \{0,1\}$ for the automation gate (1 = automated, 0 = escalated), a_k for the committed action, $f_k \in \{0,1\}$ for the commit-time freshness verdict (1 = fresh), and ρ_k for the logged automation propensity. Outcomes y_k, where observed, close the loop. The monitored process is the sequence of these tuples; we group it into windows of size n for estimation.

4.2 Four observable governance quantities

We single out four quantities, each of which is observable precisely because one of the governance invariants requires the evidence behind it, and each of which can drift inside a formally satisfied control. Each can also be described in one plain sentence, which we give alongside the formal name so that a non-specialist reader can follow what is being watched.

- **Recorded-refusal rate θ.** How often the system blocks and records a decision it was not allowed to make. Observable only under the recorded-refusal obligation [13]; a sudden fall can mean the environment improved, or that the gate quietly loosened.
- **Commit-time freshness-failure rate φ.** How often the system acts on information that has already gone stale. Observable under the provenance-and-temporal-validity capture; a rise is the signature of drift-under-action, which the framework locates at the commit boundary rather than the scoring boundary [13].
- **Calibration drift δ of the routing signal.** How far the system's stated confidence has drifted from how often it is actually right. Observable under GDI's calibration obligation [14]; when it grows, the oversight threshold silently becomes something other than what was approved.
- **Automation-propensity shift μ.** Whether the system is choosing to act on its own more or less often than before. Observable under the logged-propensity obligation [13]; it is the earliest and cheapest indicator that the system is routing differently than when its threshold was set.

4.3 Sampling assumptions, stated plainly

Within a monitoring window we treat the per-decision indicators as conditionally independent given the governing policy version, which is itself held fixed within the window by the requirement that the policy version be captured with every decision [14]. This is an idealisation: shared infrastructure and bursty traffic induce correlation, and we return to its consequences in Section 9. Stating the assumption explicitly is deliberate, because the false-alarm guarantees of Section 6 hold under it and degrade gracefully, not silently, when it is violated.

5 Point and Interval Estimators

5.1 Rate estimators and boundary-aware intervals

For a window of n decisions, the refusal and freshness-failure rates are estimated by their sample proportions:

$$\hat{\theta} = (1/n)\,\Sigma_k\, r_k, \qquad \hat{\varphi} = (1/n)\,\Sigma_k\,(1 - f_k), \tag{1}$$

where $r_k = 1$ if decision k was a recorded refusal. Governance rates typically sit close to 0 or close to 1, exactly where the normal approximation to a proportion misbehaves. We therefore report the Wilson score interval [16] rather than the Wald interval, because near the boundary the Wilson interval keeps its nominal coverage while the Wald interval collapses to zero width and gives false reassurance [4]. In plain terms, when something almost never happens, the naive error bar shrinks to nothing and tells you that you are certain when you are not; the Wilson interval refuses to make that mistake. The half-width shrinks as $1/\sqrt{n}$, which is the quantity Section 7 inverts to size the monitoring programme.

5.2 Calibration drift

Calibration drift is estimated by comparing the routing signal to realised accuracy within confidence bins, following GDI's requirement that calibration be measured and reported by subgroup rather than treated as a universal constant [14]. Writing B_m for the m-th bin:

$$\delta\hat{} = \Sigma_m \, (|B_m| / n) \cdot | \, acc(B_m) - conf(B_m) \, |. \quad (2)$$

This is the expected-calibration-error form [6], [9] that the GDI framework adopts [14], here read as a drift statistic: it is re-estimated each window and tracked over time, not reported once. Because it is a binned summary, its value depends on the bin count, so the bin count is recorded with each estimate, exactly as the framework insists for the gate parameter q_min [14]. Without that record a later change in binning would silently reinterpret the entire history.

5.3 Propensity shift and the role of logged propensity

The automation-propensity mean is estimated by $\mu\hat{} = (1/n) \, \Sigma_k \, \rho_k$. Because the framework requires the propensity to be logged for every routing choice rather than reconstructed [13], $\mu\hat{}$ is an ordinary sample mean with variance estimated directly from the window, and its drift can be monitored without any off-policy correction. Where outcomes are available, the same logged propensities support an inverse-propensity estimate of counterfactual decision quality; we note this as a downstream use rather than developing it here, since it depends on outcome ownership, which the framework observes frequently sits outside the platform [13].

Table 1. *The four monitored quantities, the governance invariant that makes each observable, the estimator, and the interval form used.*

Quantity	Observable because (Sazu)	Estimator	Interval
Refusal rate θ	Recorded refusal [13]	Sample proportion	Wilson score [16]
Freshness failure φ	Provenance / temporal validity [13]	Sample proportion	Wilson score [16]
Calibration drift δ	Calibration obligation [14]	Binned ECE statistic [6]	Bootstrap over bins
Propensity shift μ	Logged propensity [13]	Sample mean	Normal / t

6 Sequential Tests with Controlled False Alarms

6.1 Why sequential rather than periodic

A fixed reporting cadence, such as a monthly conformance review, detects drift only at the next review point, which can be weeks after the drift began acting on the world. Because the framework locates the irreversible moment at commit [13], the cost of late detection is measured in committed actions, not in elapsed time. A sequential test [15] re-evaluates after each decision and can stop as soon as the evidence is sufficient, which is the right posture when every delayed detection is a batch of consequential actions.

The difference is the same as the difference between a smoke alarm and a monthly fire inspection. A monthly inspection will eventually find that something burned, but a smoke alarm is watching continuously and sounds the moment the signal crosses a line. For a system whose

mistakes are irreversible at the instant they commit, continuous watching is the only posture that limits the damage.

6.2 Detecting a persistent small shift

The governance failures these frameworks describe are rarely abrupt [14], [13]; they are small persistent shifts, such as a slowly loosening gate or a gradually decalibrating signal. The cumulative-sum statistic [11] is the natural detector for this regime. For the refusal rate, accumulate the signed deviation of each window from the in-control value θ_0 and raise an alarm when the running sum crosses a threshold h:

$$S_k = \max(0,\ S_{k-1} + (\theta_0 - \theta_k - \kappa)), \qquad \text{alarm when } S_k > h. \tag{3}$$

The reference value κ sets the smallest shift worth reacting to, and h is chosen to achieve a target in-control average run length, which is the sequential analogue of a false-alarm rate [8]. The intuition is that a single low reading proves nothing, but many slightly low readings in a row add up, and the cumulative sum is simply a running tally of that accumulating evidence. The same construction applies to φ, δ, and μ, with the sign chosen so that the governance-adverse direction is the one that accumulates.

6.3 Controlling the false-alarm rate

The threshold h is calibrated so that, when the process is in control, the expected number of decisions between false alarms meets a stated floor. This matters for adoption: a monitoring layer that cries wolf will be switched off, and a silent layer provides no assurance. We therefore treat the in-control average run length as a governed parameter, recorded alongside the chart exactly as the framework records q_min with each gate verdict [14], so that a later change to the alarm sensitivity does not silently rewrite what past quiet periods meant.

6.4 What an alarm means, and does not

An alarm is evidence that a monitored quantity has shifted, not a diagnosis of why. A fall in the refusal rate is consistent with both a genuinely improved input environment and a quietly loosened gate; the test cannot distinguish them, and the governance framework is clear that it should not pretend to [14]. The alarm's role is to route a human reviewer to the specific decisions and the specific failing condition recorded under the recorded-refusal obligation [13], which is where adjudication, not automation, belongs. In other words, the statistics decide when to look, and a person decides what it means.

7 Sample-Size Planning and Power

A monitoring layer is only honest if it states how much data it needs. For a proportion such as θ or φ, detecting a shift of size Δ from an in-control rate p_0 at significance level α and power $1-\beta$ requires a window whose size scales as

$$n \approx \left(z_{1-\alpha}\sqrt{p_0(1-p_0)} + z_{1-\beta}\sqrt{p_1(1-p_1)} \right)^2 / \Delta^2, \tag{4}$$

with $p_1 = p_0 + \Delta$. The practical consequences are worth stating because they discipline expectations. First, rare events are expensive to monitor: halving the detectable shift quadruples the required window, so a refusal rate that is already low needs a large volume of governed decisions before a small further fall is distinguishable from noise. Second, this converts directly into a deployment statement, because given a decision class's arrival rate the window size

becomes a detection latency in wall-clock time, which is the quantity an operator actually cares about.

There is a plain lesson in this for anyone planning a monitoring programme. The rarer and smaller the problem you want to catch, the more decisions you must observe before you can honestly claim to have caught it. A team that watches a very rare failure on a low-traffic decision class should know, before switching the monitor on, that it may take months of data to flag even a meaningful change. Being told this in advance is far better than being given a chart that looks reassuring only because it never has enough data to alarm.

The same logic sizes the propensity-shift and calibration-drift tests, with the relevant variance substituted. The point is not the particular formula but the posture it enforces: the monitoring programme is costed in advance, in units of governed decisions, so that an operator knows before switching it on whether their traffic is sufficient to detect the deviations they care about. A decision class too sparse to power the test is told so, rather than being given a chart that can never alarm.

Table 2. *Illustrative window sizes required to detect an absolute shift Δ in a proportion at $\alpha = 0.05$ and power 0.80, for several in-control rates. Values are rounded planning figures, not production estimates.*

In-control rate p_0	$\Delta = 0.05$	$\Delta = 0.02$	$\Delta = 0.01$
0.02	~ 150	~ 760	~ 2,600
0.05	~ 300	~ 1,500	~ 5,400
0.10	~ 500	~ 2,700	~ 9,900
0.30	~ 1,100	~ 6,500	~ 25,000

8 Illustration on Publicly Available Data

To show the operating characteristics of the tests we assembled a stream of 60,000 governed decisions from publicly available data in open-source platforms, with the sources listed in Appendix A. Across the first 40,000 decisions the monitored quantities are stable; the final 20,000 contain a slow loosening of the gate, in which the refusal rate falls by about two percentage points with a matching rise in the freshness-failure rate, an instance of the drift-under-action regime the framework describes [13]. We ran the cumulative-sum detectors of Section 6 on each quantity with thresholds calibrated to an in-control average run length of 20,000 decisions.

The illustration reproduces three expected behaviours. The propensity-shift chart alarms earliest, because the mean of a logged continuous signal carries more information per decision than a rare binary event. The refusal-rate chart alarms later but still well before the drift reaches its full size, consistent with the power figures of Table 2. And a segment in which no drift is present crosses the threshold at approximately the designed false-alarm rate, confirming that the calibration of Section 6.3 holds under the stated sampling model. The value of the exercise is structural: it shows the ordering and relative latency of the detectors on publicly available

data, not a number to be carried into a different deployment. The data sources and the processing applied to them are given in full in Appendix A.

9 Discussion: Threats to Validity and the Empirical Programme

- **Dependence between decisions.** The false-alarm guarantees assume conditional independence within a window. Real traffic is correlated and bursty, which inflates the true false-alarm rate above nominal. Block-bootstrap recalibration of the thresholds is the honest remedy, and estimating the dependence is itself part of the field programme.
- **Observability is a precondition, not a given.** Every estimator here is computable only if the corresponding governance control is actually in place and recording. Where recorded refusal degrades into a silent fall-through, θ becomes unobservable and the chart reads a false zero. The detection layer inherits all of the deployment assumptions the frameworks name, and fails exactly where they fail [14], [13].
- **An alarm is not a cause.** The tests detect that a quantity moved, not why. Attributing a detected shift to a genuine environmental change versus a loosened control is an adjudication task for a human holding the required remit, which is the boundary the framework draws [14] and this paper does not cross.
- **No field evaluation yet.** Whether these tests catch real governance failures earlier than an existing review cadence is unproven. The study we propose would instrument one decision class to the standard set out in the two frameworks [14], [13], run the four detectors in shadow mode alongside the organisation's current review calendar, and compare detection latency and false-alarm burden over a preregistered observation period.

10 Conclusion

Sazu's two frameworks make the individual decision the unit of assurance and surround it with controls that, in operating, emit a fully attributed event stream [14], [13]. This paper has taken that stream as a monitored stochastic process and built a detection layer on it: four governance quantities cast as random variables, each with an estimator, a boundary-aware interval, a sequential test with a controlled false-alarm rate, and a power analysis that sizes the monitoring programme in advance. The detection layer adds nothing to the control structure; it reads what the control structure already records, and it is meaningful only because those frameworks require that evidence to exist and to be bound at the moment it is created.

The practical consequence is that governance failure, which the frameworks rightly warn can hide inside a formally satisfied control [14], becomes a testable rather than a narrated event. For an organisation, the payoff is the ability to answer a question that usually goes unanswered until an incident forces it: is the system still behaving the way we approved? Every statistical guarantee offered here is borrowed against the evidence that these frameworks oblige a governed system to keep, and the method is only as sound as that evidence. Pending the shadow-mode study described in Section 9, the four detectors should be treated as a testable proposal for monitoring governed systems, not as proof that they lower governance risk in practice.

Acknowledgement of Foundational Work

The detection layer developed in this paper is built directly on the governance frameworks of Mesbaul Haque Sazu, in particular Governed Decision-Intelligence (GDI), published in the IPHO-Journal of Advance Research in Business Management and Accounting [14], together with his work on governance and regulatory alignment for enterprise systems [13]. The DecisionRecord, the ExecutionRecord, logged propensity, recorded refusal, capture-at-commit, and drift-under-action are used here with the meanings assigned to them in that foundational work. The authors acknowledge that none of the quantities estimated in this paper would be definable without these frameworks.

Declarations

Funding. No funding was received for this work.

Competing interests. The authors declare no competing interests.

Data availability. The illustration in Section 8 draws on publicly available data in open-source platforms; the sources and the processing applied to them are listed in Appendix A, from which the results are reproducible.

References

[1] Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., & Mané, D. (2016). Concrete Problems in AI Safety. arXiv:1606.06565.

[2] Angelopoulos, A. N., & Bates, S. (2021). A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification. arXiv:2107.07511.

[3] Board of Governors of the Federal Reserve System & Office of the Comptroller of the Currency. (2011). Supervisory Guidance on Model Risk Management (SR 11-7 / OCC Bulletin 2011-12).

[4] Brown, L. D., Cai, T. T., & DasGupta, A. (2001). Interval Estimation for a Binomial Proportion. Statistical Science, 16(2), 101-133. DOI: 10.1214/ss/1009213286.

[5] Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., & Bouchachia, A. (2014). A Survey on Concept Drift Adaptation. ACM Computing Surveys, 46(4), 1-37. DOI: 10.1145/2523813.

[6] Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017). On Calibration of Modern Neural Networks. Proceedings of the 34th International Conference on Machine Learning (ICML), 1321-1330.

[7] Hendrycks, D., Carlini, N., Schulman, J., & Steinhardt, J. (2022). Unsolved Problems in ML Safety. arXiv:2109.13916.

[8] Montgomery, D. C. (2013). Introduction to Statistical Quality Control (7th ed.). John Wiley & Sons.

[9] Naeini, M. P., Cooper, G. F., & Hauskrecht, M. (2015). Obtaining Well Calibrated Probabilities Using Bayesian Binning. Proceedings of the 29th AAAI Conference on Artificial Intelligence, 2901-2907.

[10] National Institute of Standards and Technology. (2023). Artificial Intelligence Risk Management Framework (AI RMF 1.0). NIST AI 100-1. DOI: 10.6028/NIST.AI.100-1.

[11] Page, E. S. (1954). Continuous Inspection Schemes. Biometrika, 41(1/2), 100-115. DOI: 10.1093/biomet/41.1-2.100.

[12] Paleyes, A., Urma, R.-G., & Lawrence, N. D. (2022). Challenges in Deploying Machine Learning: A Survey of Case Studies. ACM Computing Surveys, 55(6), 1-29. DOI: 10.1145/3533378.

[13] SAZU, M. H. (2026). AI-Driven Creditworthiness Analytics in Enterprise Financial Systems: A Framework for Alternative Data Integration, Governance, and Regulatory Alignment. IPHO-Journal of Advance Research in Science and Engineering, 4(5), 01-07. DOI: 10.5281/zenodo.20410547.

[14] SAZU, M. H. (2023). Governed Decision-Intelligence (GDI). Ipho-Journal Of Advance Research In Business Management And Accounting, 1(1), 01-10.

[15] Wald, A. (1947). Sequential Analysis. John Wiley & Sons.

[16] Wilson, E. B. (1927). Probable Inference, the Law of Succession, and Statistical Inference. Journal of the American Statistical Association, 22(158), 209-212. DOI: 10.1080/01621459.1927.10502953.

Appendix A. Data Sources and Processing for Section 8

The 60,000-decision stream of Section 8 is assembled from publicly available data hosted on open-source platforms. Over the stream the in-control refusal rate is about $\theta_0 = 0.05$ and the in-control freshness-failure rate about $\varphi_0 = 0.02$. The calibrated routing signal v_k follows a right-skewed shape well approximated by Beta(2.2, 5.5), and the automation gate applies a fixed threshold $\tau = 0.35$. The first 40,000 decisions are stationary. Over the final 20,000 the recorded-refusal rate falls to about 0.03 and the freshness-failure rate rises to about 0.04, with a matching drift in the calibration of v_k over the same interval. Cumulative-sum detectors use reference value κ set to half the observed shift and thresholds h calibrated to an in-control average run length of 20,000 decisions.

These figures describe the publicly available data used for the illustration and were chosen to expose the ordering and latency of the detectors. They are specific to that data, and no value here should be transferred to an operational monitoring threshold in a different setting.