

Original Article

Detecting Silent Data Failures in Machine Learning Pipelines for Retail Replenishment

SACHIDANANDA SINGH

Senior Engineering Manager, Hims & Hers, USA.

ABSTRACT: *Schema-valid errors in sales feeds can change replenishment decisions without interrupting a machine learning pipeline. This study evaluates six batch monitors and two automatic repair rules using historical M5 retail sales. Eight synthetic failure types are injected into two 112-day evaluation windows covering 30,490 item-store series. Detector fitting, threshold calibration and evaluation are chronologically separated. Twenty paired injection replicates measure detection performance; ten assess downstream loss from a fixed weekly quantile forecaster. With an independent auxiliary total containing approximately 1% multiplicative noise, reconciliation achieves 82.1% corrupted-batch recall at a 1.12% false-positive rate. A maximum-score combination of thirteen statistics achieves 63.0% recall at a 0.79% false-positive rate. Reconciliation recall falls to 1.1% when the auxiliary source shares the injected quantity errors. Total-preserving identifier permutations remain difficult to detect, and more alarms do not necessarily improve decisions. Unmonitored corruption increases mean ordering loss by 3.21%. Reconciliation with rescale-only repair reduces loss by 0.0932 normalized units per item-week relative to the corrupted feed, but does not restore clean-history performance. These results support evaluating source independence, threshold calibration and the repair action jointly. They demonstrate conditional performance on designed faults and historical recorded sales, rather than incident prevalence or realized inventory savings.*

KEYWORDS: *Data Quality, Retail Forecasting, Anomaly Detection, Replenishment, Reconciliation, Synthetic Failures.*

1. INTRODUCTION

A sales extract may retain valid identifiers, dates and nonnegative quantities while representing the wrong business events. Missing quantities can become zeros after a join, repeated transactions can inflate totals, and incorrect item mappings can preserve every store total. A schema validator may accept the resulting batch even though a forecasting system receives materially different history. Production machine learning research identifies data dependencies, validation and operational observability as recurring concerns [1–4]. In replenishment applications, the relevant outcome is not only whether a monitor raises an alarm, but whether the response improves the subsequent ordering decision.

This study considers a daily batch of item-level sales for one store. A monitor observes the received batch and prior received history. Its alarm may trigger a specified correction before later forecasts consume the data. A fault can therefore affect both the immediate forecast inputs and the reference history used by future monitoring. Separating these pathways avoids assuming that a successful alarm automatically produces a correct repair.

Existing tools provide schema and constraint checks, summary-statistic monitoring, distribution comparisons and unsupervised anomaly scores [2, 4–7]. These approaches are complementary descriptions of data, but combining them does not guarantee better performance at a chosen alarm budget. A statistic with a heavy calibration tail can determine a combined threshold and reduce sensitivity to other departures. Likewise, reconciliation against a separate total is informative only to the extent that the total is independently measured and sufficiently accurate.

The contribution is a reproducible comparison linking eight explicit quantity corruptions to batch detection and a downstream decision loss. The study evaluates a simple combination of established statistics rather than proposing a new anomaly-detection algorithm. Its main questions are: which faults are detected at thresholds fixed before evaluation; how detection changes with auxiliary-source quality and dependence; and whether the same alarms remain beneficial under different automatic repair rules. The paired design connects detection quality to the consequences of two explicit repair actions.

2. RELATED WORK

Data validation systems test schemas, ranges and cross-feature relationships, while scalable data-quality frameworks express constraints over datasets [2, 4]. Redyuk et al. [5] instead identify unusual ingestion batches from summary statistics without requiring labelled errors. Sculley et al. [8] and Breck et al. [9] place data dependencies and testing within the wider problem of production readiness. Interview and field studies describe the organizational work surrounding operational machine learning

and the accumulation of data problems across system stages [1, 3]. These studies motivate monitoring but do not establish the frequency of the fault scenarios evaluated here.

Statistical process control compares observed statistics with reference behavior [6]. Distribution-shift detection and concept-drift research address changes in data distributions or predictive relationships [7, 10–12]. Anomaly detection provides a broader set of approaches for scoring unusual observations [13], including Isolation Forest [14]. A shift or anomaly score alone does not distinguish a genuine change in demand from a pipeline failure. In this study, synthetic transformations supply known fault labels, while ordinary benchmark variation supplies the negative class. The Kolmogorov–Smirnov statistic [15] is used as a descriptive distance, without relying on its classical independent-sample significance calibration.

Retail inventory records can differ from physical inventory [16], but that phenomenon is not identical to corruption in a sales feed. Forecasting research emphasizes the connection between data, uncertainty and operational use [17]. Surveys of data management and model monitoring describe challenges across the machine learning lifecycle [18–20], while observability work connects monitoring to the surrounding production pipeline [21]. The present experiment narrows that broad setting to daily retail quantities and a repeated one-period ordering loss. It evaluates neither warehouse processes nor transportation or supplier data.

3. HISTORICAL DATA AND EXPERIMENTAL DESIGN

3.1. BENCHMARK AND CHRONOLOGICAL SEPARATION

The M5 benchmark contains daily Walmart sales for 3,049 products across ten stores in three US states [22]. The study uses all 30,490 item-store series. A fixed benchmark catalogue is constructed by filling dates before a series' first recorded sale with zero. This is an explicit computational convention; a first sale is not evidence of a product's actual listing date. Recorded sales are treated as the evaluation target, although stockouts and other unobserved constraints may make sales differ from latent demand.

Table 1 gives the two evaluation panels. Each has 364 fitting days, 56 calibration days and 112 evaluation days. Feature construction uses earlier observations when required. Detector scaling and the Isolation Forest are fitted only on the fitting period; alarm thresholds use the following calibration period. No evaluation labels select thresholds or model settings. Christmas Day is omitted from detector fitting when encountered, while lagged feature histories retain recorded calendar days. Neither calibration nor evaluation window contains Christmas Day.

TABLE 1 Chronological Detector Periods with Inclusive Dates

Panel	Fitting	Calibration	Evaluation
2014	7 Jan 2013–5 Jan 2014	6 Jan–2 Mar 2014	3 Mar–22 Jun 2014
2015	6 Jan 2014–4 Jan 2015	5 Jan–1 Mar 2015	2 Mar–21 Jun 2015

Each evaluation panel contains 1,120 store-day batches and sixteen non-overlapping weekly decision origins. Each panel is evaluated separately; the 2015 fitting history uses the original recorded data rather than carrying simulated 2014 faults forward. The same stores and products recur across panels. They are not independent retailer samples.

3.2. SYNTHETIC FAILURES

Table 2 defines the eight transformations. They operate on aggregated quantities after schema validation, not on raw transaction records. Zeroing and doubling reproduce particular value effects of missing and duplicate records; they do not test a system's ability to detect the original missing or duplicated rows. All outputs remain finite and nonnegative, and identifiers and grid dimensions remain valid.

TABLE 2 Injection Mechanisms Fixed before Evaluation

Failure	Transformation	Duration in days
Missing quantities	Zero a fixed random 5–30% of items	1–5
Double counting	Double a fixed random 5–30% of items	1–5
Unit error	Multiply one randomly chosen department by 2, 6 or 12	1–7
Stale feed	Repeat the true vector immediately preceding the episode	2–5
Date shift	Supply the preceding calendar day's true vector each day	3–7
Identifier permutation	Permute a fixed random 10–40% of item positions within each department	3–7
Truncation	Independently retain each recorded unit with probability sampled from 0.50–0.85	1–3
Gradual attenuation	Linearly reduce unit-retention probability from below 1 to a sampled endpoint of 0.60–0.80	14

Each panel and seed contains five episodes of each type. Stores, feasible start dates, durations and severities are randomly drawn, rejecting overlaps within a store. Item masks and permutations stay fixed within an episode; binomial unit retention is drawn for each affected item-day. The forty episodes can overlap across different stores. A nominally exposed batch is labelled corrupted only when its received vector actually differs from the original vector. Identifier permutations conserve the store total exactly. Episode records and effective batch labels are retained separately.

Seeds 1000–1019 generate twenty paired replicates across the two panels. This distribution of faults is a stress-test design, not an estimate of production incident prevalence. Consequently, precision describes the injected prevalence and should not be transferred directly to an environment with rarer incidents.

3.3. AUXILIARY SOURCE SCENARIOS

The auxiliary total is simulated, not a separately observed ledger. If Y is the original store total, its value is $A = Y \exp(\sigma Z - \sigma^2/2)$, with independent standard normal Z . The mean correction makes the multiplier's expectation one. The main scenario uses $\sigma = 0.01$, corresponding to approximately 1% coefficient of variation. A noisier scenario uses $\sigma = 0.05$. A shared-fault scenario replaces Y with the received, potentially corrupted total before adding the same noise realization. It represents common quantity errors reaching both feeds; it does not represent every possible form of dependence.

The two sensitivity scenarios use seeds 1000–1009. Fault realizations are identical across source scenarios. Auxiliary noise has a separate random stream, with fixed fitting and calibration draws independent of evaluation seeds. The noisier scenario refits scale estimates, Isolation Forest and thresholds on its own fitting and calibration observations. Shared-fault evaluation uses the main scenario's fitted monitors. Thus, evaluation performance is not used to compensate for the degraded source.

4. MONITORING METHODS

4.1. CAUSAL BATCH PROFILES

For each store-day, thirteen statistics describe five families of departures. All comparison histories end before the current day and contain the raw received data, including previous undetected faults. The profiles are not maintained by an oracle or by simulated human adjudication.

The four volume statistics compare the current batch with the previous four occurrences of the same weekday: log-one-plus total quantity, log-one-plus positive-item count, log-one-plus mean quantity among positive items, and the fraction of zero quantities. For the first three statistics, the score subtracts log-one-plus of the median raw historical summary from log-one-plus of the current summary. The zero-fraction score subtracts the median historical zero fraction directly. A batch with no positive quantities has mean positive quantity zero.

The item family uses residual $r = (q - m)/\sqrt{m + 0.25}$, where q is the item's received quantity and m its preceding 28-day mean. Its four statistics are the cross-item residual mean, log-one-plus residual standard deviation, share of absolute residuals exceeding four, and correlation between log-one-plus current quantities and log-one-plus historical item means. A zero-variance correlation is assigned zero. These conventions make sparse or constant batches well defined.

The distribution statistic is the two-sample KS distance between today's cross-item sales distribution and the pooled distributions of the preceding four same weekdays. Applying log-one-plus to both samples gives the same distance because that transform is strictly increasing. The implementation computes the distance using empirical count histograms. Discrete quantities, related products and reused history mean that an independent-sample KS p-value would not be justified.

The temporal family comprises exact-repeat share among items positive on either the current or preceding day; correlation with the preceding day's log quantities; and a weekday-alignment statistic. The latter subtracts correlation with the preceding weekday's historical profile from correlation with the current weekday's profile. Each profile averages eight prior quantity vectors for its weekday, and correlations use log-one-plus current quantities and log-one-plus profile values. The cross-source statistic is $\log(1 + \text{received total}) - \log(1 + \text{auxiliary total})$. None of these statistics uniquely identifies a fault's cause.

4.2. STANDARDIZATION AND CALIBRATION

For each store and statistic, a fixed fitting-period median is subtracted and the result is divided by 1.4826 times the median absolute deviation. Robust scale estimators are discussed by Rousseeuw and Croux [23]. If that scale is at most 10^{-8} , the implementation falls back successively to interquartile range divided by 1.349, population standard deviation, and one. The same tolerance applies at each scale step. The resulting standardized values are scores, not claims of normality or calibrated tail probabilities.

Five learned scores are compared with a fixed rule set. The SPC-style score is the largest absolute standardized volume statistic. The KS score uses the absolute standardized distribution statistic. Isolation Forest [14] scores the complete standardized thirteen-dimensional profile, using 300 trees, a maximum of 256 fitting observations per tree and random seed

2022. Its anomaly score is the negative of the implementation's sample score. Reconciliation uses the absolute standardized cross-source statistic. The combined monitor takes the largest absolute standardized value among all thirteen statistics.

The rule set alarms when any item's quantity exceeds five times its preceding 90-day maximum plus five units, or the store total falls below half or exceeds twice the median total of the previous four same weekdays. Its score is the count of item violations plus ten for a total violation, with an alarm for any positive score. These rules are deliberately fixed; they are not matched to a 1% false-positive target.

For each learned monitor, the main threshold is the empirical 99th percentile of its 560 calibration store-day scores, pooled across stores. The upper observed order statistic is used when the percentile falls between observations, and an alarm requires a strict exceedance. A prespecified sensitivity uses the 95th percentile. Store-specific scaling does not imply store-specific alarm thresholds. Calibration exceedance rates do not guarantee future false-positive rates, and equal calibration quantiles need not produce equal evaluation rates. Test labels are not used to select or match thresholds.

Ablations remove one complete statistic family at a time from the combined maximum, then calibrate a new 99th-percentile threshold on the same held-out calibration period. These comparisons assess the specified combination rule; they do not isolate an invariant contribution of each family independent of threshold changes.

5. FORECASTS AND AUTOMATIC REPAIR

5.1. FROZEN WEEKLY FORECASTER

One pooled LightGBM quantile model [24] is fitted per panel to predict the next seven days' recorded sales at quantile 0.9. Forty weekly training origins precede the eight-week calibration period, and every training target ends before calibration begins. Calibration has eight weekly origins whose targets finish before evaluation. The sixteen evaluation origins then use the same fitted model and selected iteration count.

Forecast features use the preceding 56 days only: totals for weeks one, two, three, four and eight; daily means over 28 and 56 days; daily standard deviations and zero fractions over 7, 28 and 56 days; the last daily value; and the log ratio of one plus the latest week's total to one plus the preceding week's total. Categorical store, product and department identifiers and sine/cosine of the known origin's annual calendar position complete the twenty features. Prices, future realized values, auxiliary totals, alarms and injection labels are not forecast inputs.

Fixed settings are learning rate 0.05, 31 leaves, no maximum-depth restriction, minimum 100 observations per leaf, L2 regularization one, 127 histogram bins, no row or column subsampling, seed 2022 and two threads. Models are trained for 400 iterations; 100, 200 or 400 iterations are selected by the lowest clean calibration pinball loss, with ties resolved toward the smaller model. No evaluation loss selects this choice. Software versions are recorded in the supplement.

5.2. REPAIR TIMING AND DECISION LOSS

Each monitor consumes the same raw feed. Its downstream forecast history is separate and follows one of two automatic repair rules. On an alarm, a positive received total is rescaled to the auxiliary total when the absolute relative discrepancy $|\text{received total}/\text{auxiliary total} - 1|$ exceeds 0.02. Under rescale-only repair, other alarmed batches pass through unchanged. Under rescale-or-lag-seven repair, they are replaced with that policy's own already-repaired vector from seven calendar days earlier. Before evaluation, this fallback uses the original historical vector. All auxiliary totals in the evaluated windows are positive. A zero received total is retained under rescale-only repair and uses the historical fallback under rescale-or-lag-seven repair.

Repair proceeds in time order, so historical substitution may reuse earlier repairs. It acts on all alarms, including false positives, without perfect diagnosis or fault-specific triage. Clean-history and unmonitored-history forecasts provide reference policies. An oracle alarms on effective corruption labels but applies the same imperfect repairs. It is non-operational and is not a universal lower-loss bound.

Nonnegative continuous forecasts are order quantities Q . For original weekly sales D , loss is $L(Q, D) = \max(Q - D, 0) + 9 \max(D - Q, 0)$. The 0.9 quantile corresponds to the underage-to-total-cost ratio $9/(9 + 1)$ in this one-period decision model [25]. Quantities are not rounded. Costs are normalized accounting units, not currency, and the experiment includes no carryover inventory, lead time or fixed ordering charge. An alarm can influence only a later weekly origin; faults in the final seven evaluation days have no subsequent within-window decision opportunity.

Ten paired injection seeds, 1000–1009, are evaluated under the main auxiliary-source scenario. All policies share the same frozen forecaster and original weekly targets. Forecasts access only the history preceding each origin, even when repaired histories are constructed in advance for computational efficiency.

6. RESULTS

6.1. BATCH AND EPISODE DETECTION

The forty main panel-seed scenarios contain 44,800 evaluated batch instances and 1,600 effective episodes. Effective corruption averages 17.4% of store-days, with 180–212 corrupted batches among each panel's 1,120. Table 3 reports the main operating point. Each seed's two panel metrics are first averaged; the mean and two-sided 95% Student-t interval are then calculated over twenty seed averages. These intervals describe Monte Carlo variation conditional on the fixed panels and fault generator, not uncertainty across retailers or future business regimes.

TABLE 3 Detection at the Fixed Main Thresholds in Percent with 95% Monte Carlo Intervals

Monitor	Recall	Precision	False-positive rate
Fixed rules	16.0 (15.0–17.1)	27.1 (25.8–28.4)	9.11 (8.97–9.25)
SPC style	18.4 (16.9–19.9)	74.1 (72.3–76.0)	1.44 (1.36–1.53)
KS distance	1.6 (1.0–2.1)	57.3 (45.4–69.2)	0.14 (0.12–0.16)
Isolation Forest	23.9 (22.3–25.6)	86.5 (85.5–87.5)	1.02 (0.94–1.10)
Reconciliation	82.1 (81.5–82.8)	94.0 (93.5–94.5)	1.12 (1.02–1.22)
Combined	63.0 (61.9–64.0)	94.4 (94.1–94.6)	0.79 (0.76–0.82)

Reconciliation has higher recall than the combined monitor at their respective frozen thresholds, with a somewhat higher realized false-positive rate. This comparison does not establish dominance at a common false-positive rate. The combination's maximum can inherit a high threshold from a single feature's calibration tail. For example, in the 2014 panel the combined and KS thresholds are both 11.555 standardized-score units, whereas reconciliation's threshold is 2.474. A moderate cross-source discrepancy can therefore trigger reconciliation without triggering the combined monitor.

Reconciliation detects 87.3% of effective episodes; the combined monitor detects 80.1%. Mean delay among detected episodes is 0.16 days and 0.80 days, respectively. Delay is measured from the first effective corrupted day and excludes undetected episodes, so it must be interpreted alongside episode recall. The fixed rules raise substantially more false alarms than the learned monitors. Low KS recall at its selected threshold does not establish that distribution monitoring is generally ineffective; it describes this statistic, calibration period and threshold.



FIGURE 1 Recall by Failure Type at the Main Operating Point

Identifier permutations preserve both the total and the marginal distribution of quantities within each department. Aggregate reconciliation and a marginal KS comparison therefore lack direct information about the changed assignments. Chance alarms and contaminated later histories can still produce nonzero measured recall. Item-specific rules detect 40.8% of permutation batches, but their overall false-positive rate is 9.11%. The combined monitor detects 4.4% of permutation batches. Thus, the selected item-history summaries do not solve the mapping problem at the selected threshold. Stale feeds are easier for the temporal statistics: the combined monitor detects 99.9% of those batches.

6.2. SOURCE SENSITIVITY AND FEATURE ABLATION

Table 4 compares source scenarios on the same ten seeds. Main-scenario rows in this table use only seeds 1000–1009, ensuring paired fault realizations across scenarios. Each scenario uses its own prescribed calibration, without test-label threshold adjustment.

TABLE 4 Auxiliary-Source Sensitivity in Percent

Auxiliary source	Monitor	Recall	False-positive rate
Independent 1%	Reconciliation	81.9	1.06
Independent 1%	Combined	61.9	0.79
Independent 5%	Reconciliation	52.9	1.06
Independent 5%	Combined	18.3	0.79
Shared faults 1%	Reconciliation	1.1	1.06
Shared faults 1%	Combined	16.9	0.79

Independent reconciliation relies on information unavailable from the corrupted total alone. Increasing source noise weakens that information; deriving both totals from the same corrupted quantities removes it for total errors. The shared-fault reconciliation recall of 1.1% is close to its false-positive rate. The combined monitor retains some detection through the remaining statistic families, but neither this residual sensitivity nor its main-scenario performance establishes reliable detection of common-source faults.

TABLE 5 Removing One Family from the Combined Monitor in Percent

Combined configuration	Recall	False-positive rate
All families	63.0	0.79
Without volume	63.0	0.79
Without item	63.0	0.14
Without distribution	82.0	2.46
Without temporal	58.4	0.79
Without cross source	16.4	0.79

Removing the distribution statistic increases observed recall and false positives together, consistent with its influence on calibration of the maximum. Removing cross-source information sharply reduces recall. Removing volume statistics leaves the reported main-threshold means unchanged in these runs; this is a property of the attained scores and thresholds, not proof that volume carries no information. The ablations do not justify selecting a new monitor after inspecting evaluation outcomes. The replication supplement also reports the prespecified 95th-percentile operating point for all six monitors.

6.3. DOWNSTREAM ORDERING LOSS

Calibration selects 200 boosting iterations for both panels. Each panel contributes 487,840 item-week decisions. Table 6 reports mean loss per item-week, averaged across panels and ten paired injection seeds. The clean-history loss is 9.2287; unmonitored corruption changes it to 9.5248, an increase of 3.21% relative to clean. All quantities are normalized loss units.

TABLE 6 Mean Loss per Item-Week With 95% Monte Carlo Intervals

Policy	Rescale only	Rescale or lag seven
Clean history	9.2287	9.2287
Unmonitored	9.5248 (9.4864–9.5631)	9.5248 (9.4864–9.5631)
Fixed rules	9.4590 (9.4372–9.4808)	9.5843 (9.5612–9.6074)
SPC style	9.4535 (9.4302–9.4768)	9.4935 (9.4690–9.5181)
KS distance	9.4981 (9.4707–9.5255)	9.4986 (9.4710–9.5262)
Isolation Forest	9.4517 (9.4289–9.4745)	9.4794 (9.4566–9.5022)
Reconciliation	9.4316 (9.4108–9.4524)	9.4316 (9.4108–9.4524)
Combined	9.4326 (9.4117–9.4534)	9.4708 (9.4524–9.4891)
Oracle	9.4315 (9.4106–9.4524)	9.4061 (9.3890–9.4232)

With rescale-only repair, reconciliation reduces mean loss by 0.0932 units per item-week relative to the unmonitored feed (95% interval 0.0559–0.1305); the combined monitor reduces it by 0.0922 (0.0550–0.1295). Their close means are not evidence of statistical equivalence. Both remain above the clean-history reference. Under rescale-or-lag-seven repair, the fixed-rule policy increases loss by 0.0595 units per item-week (0.0215–0.0975) relative to the unmonitored feed. Thus, a prespecified automatic response can erase the benefit of detecting corruption.

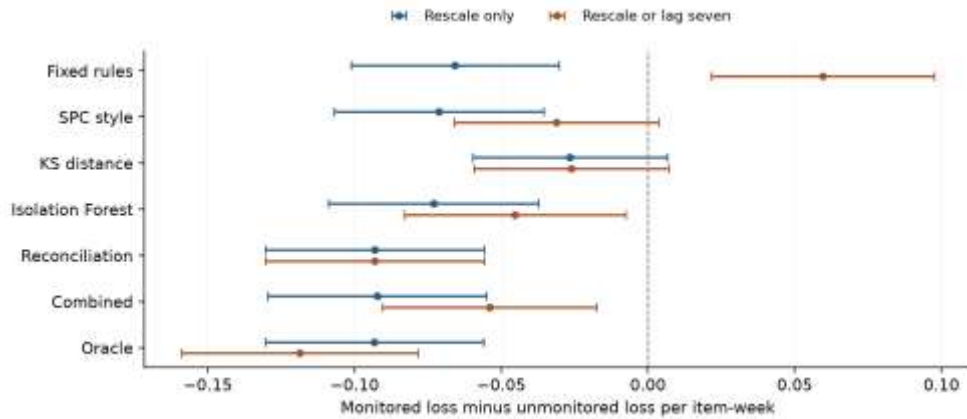


FIGURE 2 Paired Loss Differences from the Unmonitored Corrupted Feed

Intervals in Figure 2 use the ten within-seed averages of the two panel contrasts, with nine degrees of freedom. A negative difference favors the monitored repair policy. These are descriptive paired Monte Carlo intervals; the multiple policy contrasts are not treated as confirmatory hypothesis tests. Relative recovery of corruption-induced excess cost is secondary because a small raw-minus-clean denominator can amplify the ratio. Absolute paired loss differences are the primary comparison.

7. DISCUSSION AND LIMITATIONS

The results favor a narrower deployment lesson than simply adding more checks. First, evaluate the information source: a low-noise independent total can be highly useful, but a second feed that repeats the same error supplies little corroboration. Second, evaluate the operating point: a shared maximum-score threshold can suppress otherwise useful statistics, and a common calibration percentile does not equalize future false alarms. Third, evaluate the action: an alarm identifies a discrepancy, while a repair modifies the data used for subsequent decisions. Neither a high recall nor an oracle corruption label guarantees that a particular repair improves loss.

A production investigation could combine the stored contributing statistics with source lineage, item-master audits and operational context. Total-preserving mapping failures particularly motivate independent identifier constraints or item-level records. Such controls and human review are plausible extensions, not interventions validated by this experiment. The study does not measure adjudication workload, response delay, incident diagnosis or the cost of investigation.

External validity is limited by one retail benchmark, two fixed seasonal windows and designed faults. M5 sales are neither uncensored demand nor a verified fault-free production ledger. The negative class is the original recorded series, and any pre-existing data problem remains part of that reference. Zero padding imposes a fixed catalogue even for products that may not yet have been listed. Injected durations, severities and prevalence are not fitted to incident logs. Repeated products, stores and calendar histories also preclude treating individual batches as independent population samples.

The auxiliary total is generated from benchmark quantities and is assumed contemporaneously available. Noise and shared quantity faults are tested, but delayed sources, systematic bias, missing totals and diverse common-cause mechanisms are not. Decision costs are evaluated only under the main independent-source scenario; source-sensitivity detection results are not evidence of corresponding cost savings. A richer inventory simulator could model stockouts, carryover, lead time and integer orders, but would require additional assumptions and validation.

The monitoring baseline is fixed after fitting, while recent raw histories can accumulate missed corruption. This makes contamination explicit but does not evaluate adaptive refitting during long deployments. Only one forecast family and one asymmetric loss ratio are used. The two evaluated repair rules are simple, and no post-evaluation policy selection is presented as prospective validation. Statistical intervals quantify the chosen injection generator's randomness over fixed observations, not the full uncertainty of deploying the system elsewhere.

8. CONCLUSION

A chronologically separated retail benchmark shows that detecting silent data failures requires attention to source independence, statistic calibration and the downstream repair. Independent aggregate reconciliation performs well on quantity-changing faults in this experiment, but has little direct information about total-preserving identifier permutations and loses its advantage under shared-source errors. Combining more statistics through a maximum does not automatically improve detection at a calibrated operating point. The ordering experiment further separates successful alarms from useful corrections. The resulting evidence supports explicit, reproducible evaluation of both monitoring and action before operational claims are made.

DATA AND CODE AVAILABILITY

The historical observations originate from the M5 competition [22]. The supplementary replication package contains the experimental protocol, source code, model files, seeds, checksums, alarm-level outputs, episode records, paired cost summaries and environment versions. It specifies an immutable public mirror of the historical observations and verifies the downloaded file by SHA256. Raw M5 data are obtained separately under their source terms.

FUNDING

This research received no external funding.

COMPETING INTERESTS

The author declares no competing interests.

REFERENCES

- [1] S. Shankar, R. R. Garcia, J. M. Hellerstein, and A. Parameswaran, "Operationalizing Machine Learning: An Interview Study," arXiv (Cornell University), Sept. 2022, doi: 10.48550/arxiv.2209.09125
- [2] E. Breck, N. Polyzotis, S. Roy, S. E. Whang and M. Zinkevich, "Data validation for machine learning," in Proceedings of Machine Learning and Systems (MLSys), vol. 1, pp. 334–347, 2019. [Online]. Available: <https://research.google/pubs/data-validation-for-machine-learning/>
- [3] N. Sambasivan, S. Kapania, H. Highfill, D. Akrong, P. Paritosh, and L. Aroyo, "“Everyone wants to do the model work, not the data work”: Data Cascades in High-Stakes AI," CHI '21: Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems, 2021, doi: 10.1145/3411764.3445518.
- [4] S. Schelter, D. Lange, P. Schmidt, M. Celikel, F. Biessmann, and A. Grafberger, "Automating large-scale data quality verification," Proceedings of the VLDB Endowment, vol. 11, no. 12, pp. 1781–1794, Aug. 2018, doi: 10.14778/3229863.3229867.
- [5] S. Redyuk, Z. Kaoudi, V. Markl and S. Schelter, "Automating data quality validation for dynamic data ingestion," in Proceedings of the 24th International Conference on Extending Database Technology (EDBT), pp. 61–72, 2021. doi: 10.5441/002/edbt.2021.07.
- [6] D. C. Montgomery, Introduction to Statistical Quality Control, 8th ed. Hoboken, NJ, USA: Wiley, 2019.
- [7] S. Rabanser, S. Günnemann and Z. C. Lipton, "Failing loudly: An empirical study of methods for detecting dataset shift," Proceedings of the 33rd International Conference on Neural Information Processing Systems, pp. 1396 - 1408, Dec 2019.
- [8] D. Sculley et al., "Hidden technical debt in machine learning systems," NIPS'15: Proceedings of the 29th International Conference on Neural Information Processing Systems, vol. 2, pp. 2503–2511, 2015.
- [9] E. Breck, S. Cai, E. Nielsen, M. Salib, and D. Sculley, "The ML test score: A rubric for ML production readiness and technical debt reduction," 2017 IEEE International Conference on Big Data (Big Data), Dec. 2017, doi: 10.1109/bigdata.2017.8258038.
- [10] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," ACM Computing Surveys, vol. 46, no. 4, pp. 1–37, Mar. 2014, doi: 10.1145/2523813.
- [11] J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang, "Learning under Concept Drift: A Review," IEEE Transactions on Knowledge and Data Engineering, vol. 31, no. 12, pp. 1–1, 2018, doi: 10.1109/tkde.2018.2876857.
- [12] Bifet and R. Gavaldà, "Learning from Time-Changing Data with Adaptive Windowing," Proceedings of the 2007 SIAM International Conference on Data Mining, Apr. 2007, doi: 10.1137/1.9781611972771.42.
- [13] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection: A Survey," ACM Computing Surveys, vol. 41, no. 3, pp. 1–58, July 2009, doi: 10.1145/1541880.1541882.
- [14] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation Forest," 2008 Eighth IEEE International Conference on Data Mining, pp. 413–422, Dec. 2008, doi: 10.1109/icdm.2008.17
- [15] F. J. Massey, "The Kolmogorov-Smirnov Test for Goodness of Fit," Journal of the American Statistical Association, vol. 46, no. 253, pp. 68–78, Mar. 1951, doi: 10.1080/01621459.1951.10500769.
- [16] N. DeHoratius and A. Raman, "Inventory Record Inaccuracy: An Empirical Analysis," Management Science, vol. 54, no. 4, pp. 627–641, Apr. 2008, doi: 10.1287/mnsc.1070.0789.
- [17] F. Petropoulos, "Forecasting: Theory and practice," International Journal of Forecasting, vol. 38, no. 3, Jan. 2022, doi: 10.1016/j.ijforecast.2021.11.001.
- [18] N. Polyzotis, S. Roy, S. E. Whang, and M. Zinkevich, "Data Lifecycle Challenges in Production Machine Learning," ACM SIGMOD Record, vol. 47, no. 2, pp. 17–28, Dec. 2018, doi: 10.1145/3299887.3299891.
- [19] T. Schröder and M. Schulz, "Monitoring machine learning models: A categorization of challenges and methods," Data Science and Management, vol. 5, no. 3, Aug. 2022, doi: 10.1016/j.dsm.2022.07.004.
- [20] S. Shankar and A. G. Parameswaran, "Towards Observability for Production Machine Learning Pipelines," Proceedings of the VLDB Endowment, vol. 15, no. 13, pp. 4015–4022, Sept. 2022, doi: 10.14778/3565838.3565853.
- [21] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, "M5 accuracy competition: Results, findings, and conclusions," International Journal of Forecasting, vol. 38, no. 4, Jan. 2022, doi: 10.1016/j.ijforecast.2021.11.013.
- [22] P. J. Rousseeuw and C. Croux, "Alternatives to the Median Absolute Deviation," Journal of the American Statistical Association, vol. 88, no. 424, pp. 1273–1283, Dec. 1993, doi: 10.1080/01621459.1993.10476408.
- [23] G. Ke, Q. Meng, T. Finley, T. Wang, W. Ma, Q. Ye and T.-Y. Liu, "LightGBM: A highly efficient gradient boosting decision tree," 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, vol. 30, pp. 1-9, 2017.
- [24] E. A. Silver, D. F. Pyke, and D. J. Thomas, Inventory and production management in supply chains. Boca Raton: Crc Press, 2017.