Temporal Drift in Financial Fraud Detection: A Leakage-Safe, Cost-Aware Evaluation and Monitoring Framework

Larsen Sørensen [1*], Bence Koffi [2]
[1] ETH Zurich, SWITZERLAND
[2] University of Bucharest, ROMANIA

ABSTRACT

Financial fraud detection is a rare-event prediction problem in which model performance can deteriorate as transaction behavior, fraud strategies, customer populations, and institutional processes change, so that a high score on a single held-out sample is insufficient to establish operational robustness. This article develops a leakage-safe, cost-aware evaluation and monitoring framework for temporal fraud detection and tests it empirically by reimplementing the chronological framework of Zuberi et al. on the public European credit-card benchmark (284,807 transactions, 492 frauds). The framework separates six functions that are often collapsed together in practice: temporal data design, training-only preprocessing and resampling, multi-metric model evaluation, distributional drift diagnosis, outcome-based performance monitoring, and governed intervention. The reimplementation reproduced the source partition exactly (384, 56 and 52 frauds in the training, validation and test periods) and a comparable level of performance: the best configurations reached a test F2 of 0.756 (soft-voting ensemble) and 0.754 (random forest with SMOTE), against 0.771 reported in the source study. Three further analyses follow from the framework. A deliberately leaky pipeline inflated F2 from 0.725 to 1.000, and a random split alone inflated it to 0.798. Every selected feature showed a statistically significant distributional shift between the training and test periods, and the largest shift (hour of day, PSI 4.08) reflected the time-of-day composition of the test window rather than changed behavior, while the model score distribution barely moved (PSI 0.008). Rolling evaluation showed recall varying from 0.33 to 1.00 across six test windows without any corresponding score-level alarm. The results support the article's central claim that a statistical drift alert requires operational interpretation and mature-label evidence before action; they come from a single 48-hour dataset and do not by themselves establish multi-institution or long-horizon effectiveness.

Keywords: Financial Fraud Detection; Temporal Drift; Concept Drift; Class Imbalance; Temporal Validation; Model Monitoring; Explainable AI; Model Governance

Introduction

Machine-learning systems are increasingly used to rank financial transactions by fraud risk, allocate investigations, and support payment authorization, because their value lies in detecting multivariate patterns that would be difficult to encode as fixed rules. Financial data, however, are generated by changing social, technological, and institutional processes: customer behavior shifts, payment channels evolve, fraud actors adapt deliberately to whatever controls they encounter, and changes in product design or policy alter the data a model is actually presented with at scoring time. The assumption that the statistical relationship a model learned during training will remain stable into deployment is therefore difficult to defend in a dynamic transaction environment, and treating a single historical evaluation as if it settled the question of production reliability is one of the more consequential methodological shortcuts available to a fraud-modeling team.

This problem has two connected dimensions that are easy to elide if they are not named separately. First, the distribution of incoming data may change on its own, independent of anything about fraud; second, and more consequentially, the relationship between observed features and the target outcome may itself change. The concept-drift literature distinguishes these situations explicitly and emphasizes that adaptive learning must

be evaluated with methods that reflect the timing and structure of change rather than with a single static test[1]. For fraud prevention specifically, this distinction has real operational stakes: a change in transaction amounts may simply reflect legitimate seasonality and call for no model response at all, while a change in the conditional relationship between transaction patterns and confirmed fraud can directly impair detection and call for a considered intervention.

A recent chronologically evaluated benchmark study offers a useful methodological case for grounding this discussion[2]. It combines chronological data splitting, train-only preprocessing, SHAP-based feature selection, multiple resampling strategies, comparison across model families, validation-tuned decision thresholds, and explicit drift metrics within a single leakage-controlled pipeline. Its benchmark dataset contains 284,807 transactions, of which 492 are labeled fraudulent, a prevalence of approximately 0.173 percent, and it reports a Matthews correlation coefficient of 0.8060 and a ROC-AUC of 0.9703 for a soft-voting ensemble evaluated under this protocol[2]. These results illustrate the value of disciplined leakage control and multi-metric evaluation applied to a genuinely difficult class-imbalance problem. They do not, however, by themselves establish long-term production resilience: the dataset spans approximately two days, which meaningfully limits what the study can say about seasonality, sustained behavioral change, or performance across the kind of multi-month or multi-year horizon that a deployed fraud system actually has to survive.

A further complication is that a strong ROC-AUC figure does not necessarily guarantee a useful, operationally manageable alert workload once a model is deployed against a highly imbalanced population. Saito and Rehmsmeier demonstrate why precision-recall plots can be substantially more informative than ROC plots specifically when positive cases are rare, because the ROC curve's apparent reassurance can mask an unmanageable absolute false-positive volume[3]. Similarly, a model explanation is not itself evidence of temporal stability: SHAP assigns feature-attribution values to individual predictions in a theoretically grounded way, but its interpretation must remain distinct from causal inference about why fraud behavior changed[4]. The Matthews correlation coefficient offers another useful summary precisely because it incorporates all four confusion-matrix outcomes into one balanced figure rather than privileging any single error type[5], which matters considerably once the positive class is as rare as it is in most fraud-detection settings.

This article accordingly develops a synthesis rather than simply restating or re-presenting the pipeline used in any one benchmark study. It asks how temporal integrity, class imbalance, predictive utility, drift diagnosis, interpretability, and governance can be combined into a single coherent evaluation protocol rather than treated as separate, loosely related concerns. Its central contribution is a decision-oriented framework that distinguishes the detection of a statistical change from evidence of genuine operational deterioration, and that specifies, as concretely as a conceptual synthesis can, when an institution should investigate further, recalibrate, retrain, or simply retain its current model.

Research Questions and Contribution

The article addresses four connected questions. The first concerns which design choices are necessary to obtain temporally credible fraud-detection estimates in the first place. The second concerns how performance should be evaluated when fraud is extremely rare and the two error types carry markedly asymmetric costs. The third concerns how input-distribution indicators should be combined with label-based performance monitoring, given that each is informative about a different part of the problem and neither is sufficient alone. The fourth concerns what governance process is actually needed to convert a drift signal into a defensible, documented intervention rather than an ad hoc reaction.

Three contributions follow. First, the article integrates methodological safeguards drawn from drift adaptation, imbalanced learning, explainability research, and fraud-detection practice into a single account rather than leaving a reader to reconcile them across separate literatures. Second, it formalizes a six-stage evaluation and monitoring workflow and provides explicit mathematical definitions for the core measures that workflow depends on. Third, it presents an operational decision matrix linking monitoring evidence to proportionate institutional responses, together with a set of testable propositions offered for future empirical evaluation. A fourth contribution is empirical: the framework is implemented on the public European credit-card benchmark, reproducing the source study's leakage-safe pipeline and then extending it with leakage, drift, calibration, cost and rolling-window analyses, so that each synthesis finding can be checked against data rather than asserted.

Related Work and Conceptual Foundations

Fraud Detection as Rare-Event Classification

Fraud detection is characterized by a low base rate, heterogeneous positive cases, and sharply asymmetric consequences of error. Accuracy alone can be actively misleading in this setting, because a classifier that predicts every transaction as legitimate can achieve a deceptively high accuracy score precisely when fraud is rare, while providing no useful fraud detection whatsoever. Precision captures the proportion of flagged transactions that are confirmed fraud, and recall captures the proportion of confirmed fraud that the model actually identifies; the appropriate balance between the two depends on an institution's financial exposure, the customer friction it is willing to tolerate, its analyst capacity, and the operational cost of investigating a flagged case.

The chronologically evaluated benchmark discussed above illustrates the severity of this imbalance concretely: only 492 of 284,807 transactions are labeled fraudulent[2]. A prevalence this low means that even a small false-positive rate, one that would look negligible as a percentage, can generate a large absolute number of alerts once it is applied to a realistic transaction volume. Saito and Rehmsmeier's analysis is directly relevant here: precision-recall analysis is particularly informative in imbalanced settings precisely because it represents the fraction of positive predictions that are correct, which is what an investigation team actually experiences, whereas ROC-AUC remains a useful discrimination measure but should never be interpreted in isolation from prevalence and the chosen operating threshold[3].

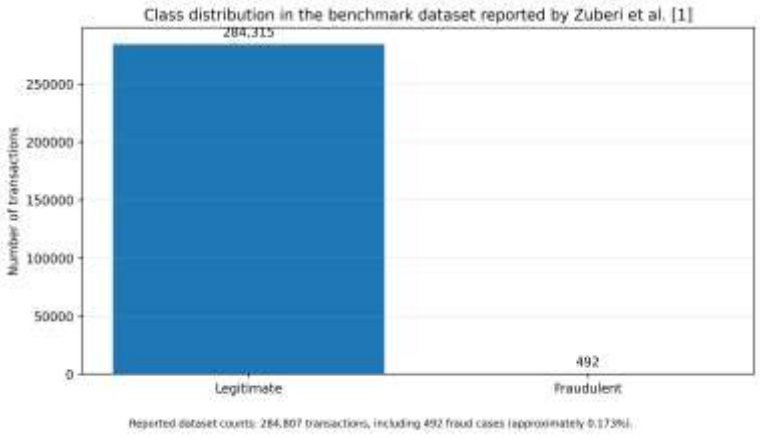

Figure 1. Class distribution in the benchmark dataset reported by Zuberi et al. Counts are reproduced from the cited study; no new dataset was analyzed for this article.

Temporal Validation and Leakage

A model intended to predict future transactions should, as a basic matter of evaluation logic, be evaluated on observations that occur chronologically after those used to train it. Random splitting can mix temporally adjacent observations across training and test partitions and can obscure genuine temporal dependence in the data, and it can create a subtler form of leakage whenever feature selection, scaling, imputation, or resampling is fitted before the chronological split is actually enforced. The chronological protocol used in the motivating benchmark addresses this concern directly by partitioning data into training, validation, and test periods and by confining every learned transformation strictly to the training period[2].

A single chronological split is nevertheless a limited instrument on its own. It provides an estimate for one particular future window and may be sensitive to that window's specific event mix, a concern that grows more serious the shorter the window is. Rolling-origin or forward-chaining evaluation, meaning training on an earlier interval and testing on the next, then advancing the cutoff and repeating the exercise, provides a considerably stronger view of how performance actually varies across time. The split design should also respect label maturity as a first-class constraint: fraud labels may be delayed, later revised, or simply unresolved at the moment a live monitoring decision has to be made, and an evaluation protocol that ignores this timing will systematically overstate what a real-time system could actually have known.

Concept Drift and Adaptation

Gama and colleagues define concept drift around changes in the relationship between inputs and the target variable and review a wide range of adaptation strategies developed for evolving data streams[1]. This literature cautions explicitly against treating drift as one uniform phenomenon. Covariate drift concerns a change in the distribution of the inputs themselves; prior-probability shift concerns a change in the base rate of the outcome; and concept drift proper concerns a change in the conditional relationship between inputs and outcome. These three changes can occur independently of one another, which matters a great deal for monitoring design: a system that monitors only marginal input distributions may entirely miss a changed decision boundary, while an input shift of real magnitude may occur without doing any damage to predictive performance at all.

The practical implication for fraud systems is that drift detection and drift adaptation should be evaluated as genuinely separate activities rather than as two names for the same action. Detection is a diagnostic task: it signals a change relative to a chosen reference distribution, nothing more. Adaptation is a decision task: it changes thresholds, parameters, features, or an entire model version, and it carries real costs and real risks of its own. A detection method can in principle generate a large volume of alerts while providing very little operational value, either because the alerts do not correspond to any meaningful performance deterioration or because the cost of responding to them exceeds whatever benefit the response would deliver.

Imbalance Handling and Threshold Choice

Oversampling and class weighting are the most common ways institutions try to improve learning from rare positive cases. Sampling of this kind, however, changes the effective class distribution the learner actually sees during training and can affect the calibration of its output probabilities in ways that are easy to overlook. Synthetic minority examples can genuinely help certain learners, but they can also create implausible feature combinations or distort local class geometry in ways that vary by model family; the benchmark study compares several oversampling methods across multiple classifiers and finds that their effects are model-dependent rather than uniformly beneficial, a finding that should discourage treating any single resampling recipe as a universal default[2].

For a valid evaluation, resampling must be conducted only within training data, including, critically, within each individual training fold during cross-validation rather than once over the whole training set. Validation and test data should retain the natural, unresampled prevalence throughout. Threshold selection should be performed on validation data against an explicit, stated objective, such as maximizing a particular Fβ score subject to a cap on false-positive volume, or minimizing an explicit expected-loss function; the threshold chosen this way must then be frozen before it is ever applied to the final test evaluation, precisely so that the test result remains an honest estimate of what the frozen, deployed system would actually do.

Explainability and Model Interpretation

Lundberg and Lee introduce SHapley Additive exPlanations as a unified approach to feature attribution, assigning each feature a quantified contribution to a given prediction under a specific set of formal consistency properties [4]. In fraud detection, feature attribution of this kind can usefully support case review, model debugging, and comparison of model behavior across different time periods. Attribution values, however, are conditional on the specific model, the data distribution in force at the time, and the particular explanation setup used to compute them; a high attribution value does not by itself establish causal influence on the outcome, and a change in attribution over time can arise just as easily from correlated features or from an ordinary distributional shift as from any genuine change in fraudster behavior.

SHAP should therefore be treated as one diagnostic layer among several rather than as a self-sufficient verdict on model behavior. Temporal explanation analysis is considerably more informative when it is paired with feature-distribution checks, data-pipeline audits, and outcome-based metrics computed once labels have matured. If a particular feature becomes markedly more influential at the same time that recall is declining, that joint pattern can reasonably motivate a targeted investigation; it is not, on its own, proof of the underlying mechanism responsible, and treating it as proof risks committing an institution to the wrong remedy.

Multi-Institution Validation and Governance

Rashid and Sattiraju propose a multi-institution validation architecture for drift-resilient early-warning models that covers both SME loan distress and transaction fraud[6]. Their analysis emphasizes that single-institution evaluation can easily confound genuine statistical generalization with institution-specific historical patterns that happen not to recur elsewhere, and it discusses candidly the practical and governance cost of validating a drift-monitoring architecture across independently operating sites with different systems, policies, and data-sharing constraints. Their contribution is architectural and proposal-oriented in character: it motivates cross-site validation as a necessary complement to within-institution testing, but it should not be read as proof that any particular system has already achieved generalizable performance across institutions, a distinction their own framing preserves carefully.

This perspective complements temporal validation rather than substituting for it. A model can generalize acceptably forward in time at one institution and still fail at another, because of differences in customer population, product mix, data definitions, or fraud-investigation processes that a purely temporal test at a single site would never surface. A complete validation programme should therefore test temporal transfer within institutions and institutional transfer across sites as two distinct exercises, while documenting, at each site, the differences in data access, label definitions, and governance obligations that a cross-site comparison has to account for if it is to be trusted.

Synthesis and Research Gap

Taken together, this literature identifies a set of distinct but genuinely interacting risks: leakage can inflate reported performance estimates; severe class imbalance can make headline metrics actively misleading; input drift need not equal concept drift and the two require different responses; feature attribution can be

misinterpreted as causal evidence when it is not; and single-site validation may not establish transferability to any other institution. The gap this article addresses is not the absence of individual tools for any one of these risks, each of which already has a substantial literature of its own, but the absence of a clearly articulated decision protocol that links their combined outputs to justified, proportionate operational responses. The framework proposed below integrates these tools while keeping each one's inferential limits explicit rather than implicit, so that a reader can see exactly what each diagnostic can and cannot establish on its own.

Methodological Approach

Article Design and Evidence Boundary

The article combines a critical synthesis of published work with an empirical implementation. Its core case is the chronologically evaluated benchmark framework described above[2], read alongside established work on concept drift[1], imbalanced evaluation[3], feature attribution[4], the Matthews correlation coefficient[5], and the multi-institution validation architecture discussed previously[6]. The framework is then implemented, using the source study's own protocol as its backbone, on the publicly available European credit-card dataset[7]. All code was written for this article and all figures reported in the empirical section were produced by it; numbers attributed to the source study are labelled as such. The empirical evidence is confined to one public dataset covering 48 hours, so it can test the internal logic of the framework but not its transfer across institutions or longer time spans.

Proposed Temporal Data Protocol

Let each transaction be represented as a triple of a feature vector, a fraud label, and a timestamp. Observations are ordered by timestamp and partitioned into non-overlapping chronological sets such that the training set precedes the validation set, which in turn precedes the test set, with no overlap or reordering permitted across the three. The training set is used for parameter estimation and for fitting every learned transformation; the validation set is used for model selection and threshold choice; and the test set is reserved exclusively for final evaluation and is never consulted before that point. A rolling-origin design repeats this chronological partitioning across several cutoffs where data volume permits, which meaningfully strengthens the evidence a single fixed split can provide. Feature definitions, label-maturity rules, and data-availability timestamps should all be documented explicitly, so that the evaluation genuinely reflects what would have been knowable at prediction time rather than what happens to be knowable in hindsight.

Leakage-Safe Preprocessing and Imbalance Treatment

All learned preprocessing, including imputation, scaling, feature selection, and resampling, must be fitted only on training data. During cross-validation specifically, each such operation must be refitted inside each individual training fold rather than fitted once over the full training set and then merely applied to each fold, since the latter practice reintroduces exactly the leakage the chronological split was meant to prevent. Test-set information must never be used, even implicitly, to select features, choose a model family, tune a decision threshold, or decide which evaluation window ultimately gets reported. The natural, unresampled class distribution should be retained throughout the validation and test sets at all times, so that reported performance reflects the prevalence the deployed model will actually face.

Performance Measures

For binary classification, true positives are fraudulent transactions correctly flagged, false positives are legitimate transactions incorrectly flagged, true negatives are legitimate transactions correctly left unflagged, and false negatives are fraudulent transactions the model misses. Precision is the share of flagged transactions that are genuinely fraudulent, TP/(TP+FP), and recall is the share of genuine fraud the model actually

identifies, TP/(TP+FN). The Fβ score permits a configurable emphasis on recall relative to precision, $F\beta = (1+\beta^2)\cdot(\text{Precision}\cdot\text{Recall})/(\beta^2\cdot\text{Precision}+\text{Recall})$, where values of β above one place greater weight on recall, which is frequently the more costly error type to tolerate in fraud prevention. The Matthews correlation coefficient incorporates all four confusion-matrix cells into a single figure, $MCC = (TP\cdot TN - FP\cdot FN) / \sqrt{[(TP+FP)(TP+FN)(TN+FP)(TN+FN)]}$, and is informative precisely under the kind of severe imbalance fraud detection presents, though it can become undefined if any denominator term is zero, a case any implementation should define and report explicitly rather than silently suppress[5]. ROC-AUC measures ranking discrimination across all thresholds simultaneously, whereas PR-AUC summarizes the precision-recall trade-off directly; in a rare-event setting, PR-AUC and threshold-specific alert volume should be reported alongside ROC-AUC rather than in its place[3].

Distributional Drift Measures

The population stability index compares proportions across defined bins between a reference period and a monitoring period, $PSI = \Sigma_b (q_b - p_b)\cdot\ln(q_b/p_b)$, where p_b and q_b denote the reference and monitoring proportions in bin b respectively. Bins should be defined using the reference data alone and then held fixed across subsequent monitoring periods, and any bin with a zero proportion requires a documented smoothing convention so the logarithm remains defined; PSI is a descriptive statistic, and its threshold for action is inherently context-dependent rather than universal, so it should never be treated as a self-sufficient test of model failure. For continuous features, the two-sample Kolmogorov-Smirnov statistic compares empirical cumulative distribution functions directly, $D_{KS} = \sup_x |F_{ref}(x) - F_{monitor}(x)|$; a large value indicates a genuine difference between the two empirical distributions, but its statistical significance is sensitive to sample size, so effect magnitude and operational relevance should always be examined alongside any reported p-value rather than in place of it. Jensen-Shannon divergence provides a further, symmetric and bounded divergence measure between two distributions, computed as the average of each distribution's Kullback-Leibler divergence from their mixture; the logarithm base and the handling of zero probabilities should be specified explicitly whenever it is used. These three indicators complement one another, but all three operate primarily on distributions, and none of them establishes a change in the conditional relationship between features and outcome without outcome information to corroborate it.

Calibration, Uncertainty, and Cost

Where a system outputs probabilities rather than bare labels, calibration should be evaluated explicitly, because a model can rank transactions correctly by relative risk while still systematically overstating or understating the absolute level of that risk. Reliability diagrams and proper scoring rules can usefully complement discrimination metrics in this respect. Uncertainty intervals should be reported for key metrics, especially when the test period contains very few fraud cases, as is routinely the case in this domain; bootstrap procedures must respect temporal dependence wherever it is material, and block-based resampling is generally preferable to independently resampling individual transactions, which would understate the true uncertainty. Threshold selection should reflect operational costs directly: expected loss can be represented in simplified form as a weighted sum of the cost of missed fraud, the cost of false positives, and the cost of the review workload the chosen threshold generates. This expression is a decision framework rather than a universal cost model, and institutions must define each cost term from their own context, taking care not to double-count losses that the three terms might otherwise capture more than once.

Analysis Procedure

The proposed analysis proceeds in six steps: auditing timestamps, label maturity, missingness, and feature provenance; creating chronological partitions and, where data volume permits, rolling-origin windows; fitting candidate models and every learned preprocessing step on training data only; selecting models and

thresholds on validation data against predefined criteria; reporting test performance together with uncertainty, calibration, and distributional shift; and finally linking the resulting monitoring signals to a documented decision matrix of the kind set out later in this article. The procedure is designed throughout to be reproducible and auditable, so that a reviewer or regulator can reconstruct exactly what evidence a given model-change decision rested on.

Proposed Integrated Framework

The framework proposed here comprises six connected stages: data and label integrity; leakage-safe temporal design; model development; multi-dimensional evaluation; drift diagnosis; and governed response. Its novelty lies specifically in the explicit separation of statistical detection from intervention, a separation many deployed systems collapse by design or by default. Monitoring results feed back into the next evaluation cycle once outcomes mature, but no single drift score, however large, automatically authorizes retraining on its own; a change in the deployed model is a governed decision, not a statistical reflex.



Figure 2. Proposed leakage-safe evaluation and monitoring framework, developed as a synthesis informed by the cited literature on drift, imbalance, explainability, and multi-institution validation.

Data and Label Integrity

The first stage verifies timestamps, data lineage, feature availability, and label maturity before anything else is attempted. It must distinguish a true shift in transaction behavior from changes caused by upstream schema modifications, missing fields, logging changes, or a revised fraud-confirmation process, any of which can produce a statistical signature that superficially resembles genuine drift. Label delay should be measured and reported explicitly, because an apparent recent decline in fraud prevalence may in fact reflect incomplete outcome observation rather than any real change in fraud activity, a confusion that can lead an institution to draw exactly the wrong conclusion from its own monitoring dashboard.

Temporal Design and Model Development

The second and third stages ensure, respectively, that future observations never influence model fitting and that imbalance treatment is evaluated fairly rather than in a way that flatters one model family over another. Candidate models should be compared under common time windows and under identical decision criteria, so that differences in reported performance reflect genuine differences between models rather than

differences in how generously each was evaluated. Where oversampling is used, the resampled training distribution must never be confused with the true operational prevalence the model will face in production. Feature selection should be repeated within the training process for each fold or cutoff, rather than computed once over the full dataset and then reused, since the latter shortcut reintroduces a subtle form of leakage.

Evaluation and Drift Diagnosis

The fourth stage reports discrimination, rare-event retrieval, calibration, uncertainty, and alert volume together, rather than any one of them in isolation. The fifth stage compares feature and score distributions first, and then checks outcome-based metrics once labels have matured enough to be trusted. A statistically significant input shift accompanied by stable downstream performance may warrant continued observation rather than immediate retraining, since acting on it immediately risks the kind of unnecessary disruption discussed earlier in connection with resampling and threshold choice. Conversely, a decline in recall with little accompanying marginal feature drift may warrant investigation for genuine concept drift, a label-quality problem, or an unmonitored variable that the feature-level indicators simply cannot see.

Governed Response

The final stage defines the available response options explicitly: retain the current model, investigate data quality, increase targeted human review, recalibrate output probabilities, adjust the decision threshold, retrain, or replace the model outright. Whichever option is chosen should be supported by a documented rationale and tested on a temporally later validation window before it is released into production, exactly as a brand-new model would be. Versioning, a documented approval step, an explicit rollback plan, and a post-deployment review together make the intervention auditable after the fact, which matters considerably if a regulator or internal risk committee later asks why a particular change was made and on what evidence.

Empirical Implementation on the European Credit-Card Benchmark

Data and Reimplemented Protocol

The dataset contains 284,807 card transactions made over two days in September 2013, of which 492 (0.173 percent) are fraudulent; 28 inputs are anonymised principal components (V1 to V28), and the remaining fields are the transaction amount and the elapsed time in seconds[7]. Following the source framework[2], transactions were sorted by time and split 70/15/15 into training, validation and test periods. Three features were added (sine and cosine of the hour of day and the log of the amount), giving 33 candidate inputs. The scaler, SHAP-guided feature selection and oversampling were fitted on the training period only and applied in transform-only mode to the later periods. The selection used a class-balanced gradient-boosting surrogate and kept the 15 features with the highest mean absolute SHAP value; decision thresholds were chosen on the validation period by maximising F2 over 300 candidate values, and all reported metrics come from the untouched test period.

The partition matched the source study exactly: 199,364 training transactions (384 frauds), 42,721 validation transactions (56 frauds) and 42,722 test transactions (52 frauds). The selected features were V4, V14, V8, V12, Hour_sin, V3, V18, V10, V7, V11, V5, V17, V26, V22, V6. V4 and V14 ranked first and second, as in the source study. When the surrogate was trained without class weighting, only 10 of the 15 selected features were the same, which shows that a "SHAP-selected" feature set depends on how the surrogate is configured and should be reported with that configuration.

Reproduction of the Source Framework

Table 1 reports test-period results for five classifiers and a soft-voting ensemble of the three strongest by validation PR-AUC (LightGBM, random forest and XGBoost)[8][9][10]. Confidence intervals come from 500 stratified bootstrap replicates of the 52-fraud test set.

Table 1. Test-period performance of the reimplemented leakage-safe pipeline (52 frauds; thresholds tuned on validation by F2).

Model	ROC-AUC	PR-AUC	Precision	Recall	F2	F2 95% CI	MCC
LogReg + SMOTE	0.981	0.635	0.379	0.750	0.627	0.53-0.70	0.532
XGBoost + BorderlineSMOTE	0.962	0.763	0.639	0.750	0.725	0.63-0.81	0.692
LightGBM + SMOTE	0.969	0.698	0.720	0.692	0.698	0.60-0.79	0.706
Random Forest + SMOTE	0.961	0.775	0.864	0.731	0.754	0.65-0.85	0.794
MLP 128-64-32 (no resampling)	0.948	0.739	0.731	0.731	0.731	0.62-0.83	0.730
Soft-voting ensemble	0.965	0.761	0.780	0.750	0.756	0.66-0.84	0.765

The best configurations (the ensemble at F2 0.756 and the random forest at 0.754) fall within the bootstrap uncertainty of the source study's best reported value of 0.771, so the reimplementation reproduces the framework's level of performance rather than an inflated one. Two points from Table 1 matter for the synthesis that follows. First, ROC-AUC ranks the models almost oppositely to the operational metrics: logistic regression has the highest ROC-AUC (0.981) but the lowest precision (0.379) and F2 (0.627). Second, with only 52 test frauds, a single additional true positive moves recall by almost two percentage points and the F2 intervals of all leading models overlap, so rankings among them should not be over-interpreted.

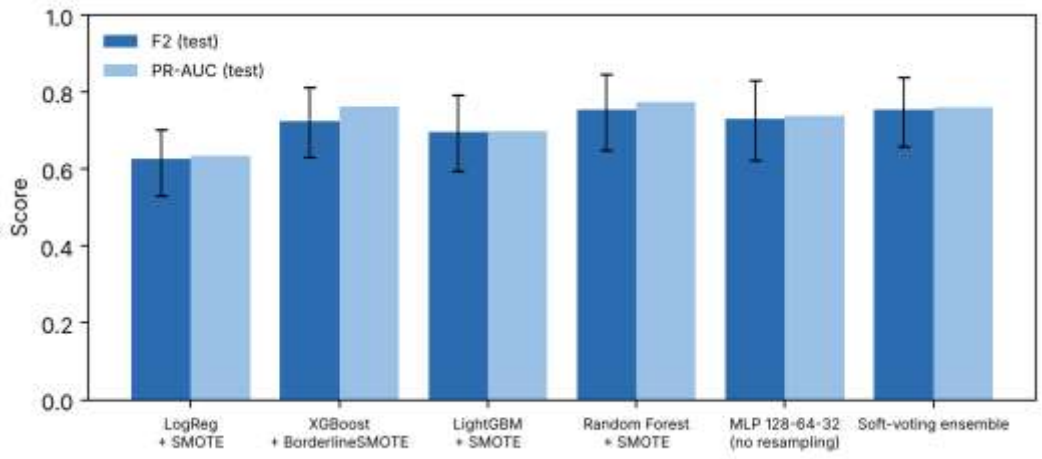

Figure 3. Test-period F2 (with 95% bootstrap interval) and PR-AUC of the reimplemented pipeline [authors' analysis].

How Much Leakage Inflates Performance

To test the framework's first premise, that temporal validity is a design property, the XGBoost configuration was re-estimated under two flawed designs. In the fully leaky design, scaling, SHAP selection and SMOTE were applied to the whole dataset before a random split, so synthetic minority cases derived from test-set frauds entered training. This design returned an F2 of 0.9996 and a PR-AUC of 1.0000, which are near-perfect scores with no operational meaning. With leakage-safe preprocessing but a random rather than chronological split, F2 was 0.798 and PR-AUC 0.817, against 0.725 and 0.763 under the chronological design. Ignoring time alone therefore overstated F2 by about 7 points even on a 48-hour dataset.

Distributional Drift Versus Performance

Every one of the 15 selected features differed significantly between training and test (Kolmogorov-Smirnov test, Bonferroni-corrected $p < 0.05$), and 8 of the 15 had a PSI of 0.10 or more. The prevalence of fraud also fell, from 0.193 percent in training to 0.122 percent in test (two-proportion $z = 3.14$, $p = 0.0017$). By the conventional reading of these statistics, the model faced major drift. Yet the largest shift, a PSI of 4.08 for the hour-of-day term, arose because the test period covers only the evening of the second day, a change in time-of-day composition rather than in fraud behaviour, and the distribution of model scores barely moved between validation and test (PSI 0.008). Figure 4 shows the per-feature values.

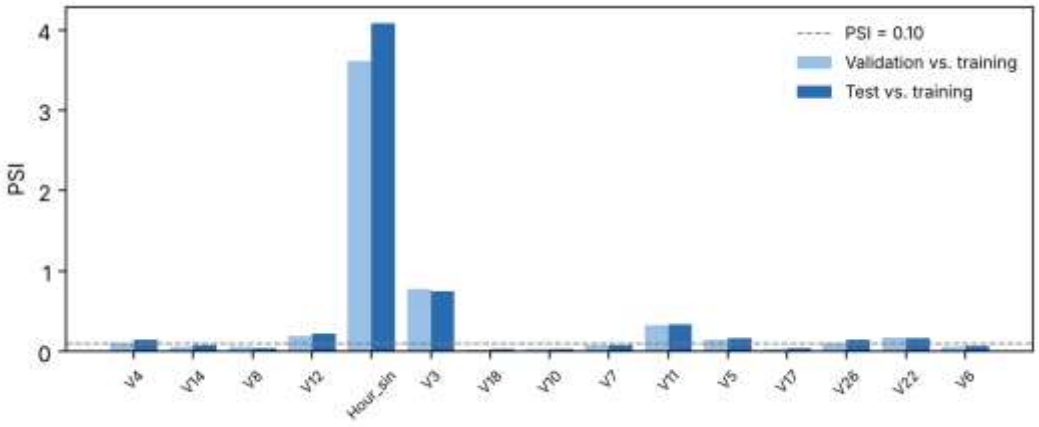

Figure 4. Population stability index of the 15 selected features, validation and test periods versus training; dashed line at 0.10 [authors' analysis].

Performance, by contrast, did vary. Splitting the test period into six consecutive windows of about 7,120 transactions, recall at the fixed threshold was 0.86, 0.69, 0.89, 1.00, 0.33, 0.80 and PR-AUC ranged from 0.38 to 1.00. The weakest window (recall 0.33 on 6 frauds) had a score PSI of only 0.002, so no label-free statistic would have flagged it. This is the pattern Synthesis Finding 2 anticipates: distributional statistics fired constantly for a benign reason and stayed silent when performance actually dipped.

Calibration and Cost

Probabilities from the oversampled XGBoost model were poorly scaled for decision use. Isotonic recalibration fitted on the validation period reduced the test Brier score from 0.000516 to 0.000424 and the expected calibration error from 0.000375 to 0.000203 without changing the ranking. For cost, each alert was assumed to cost EUR 5 of analyst time and each missed fraud its transaction amount; this cost structure is an assumption, since the dataset contains no investigation costs. At the F2 threshold the model raised 61 alerts, recovered EUR 3,796.48 of the EUR 6,168.88 fraud value in the test period, and reduced total cost from EUR

6,168.88 to EUR 2,677.40. The cost-optimal threshold chosen on validation (0.246) was almost identical to the F2 threshold (0.243), so in this dataset, with this cost assumption, the two criteria agree. That agreement is a property of the data, not a general rule, and the framework requires the comparison to be reported rather than assumed.

Results of the Literature-Based Analysis

This section relates the synthesis to both sources of evidence: numerical findings published by other authors, which are labelled as such, and the reimplementation reported above. The figure below reproduces, with attribution, two headline metric values reported for the motivating benchmark's soft-voting ensemble; the independent reimplementation in the previous section reached a comparable F2 and a ROC-AUC of 0.965 for the corresponding ensemble.

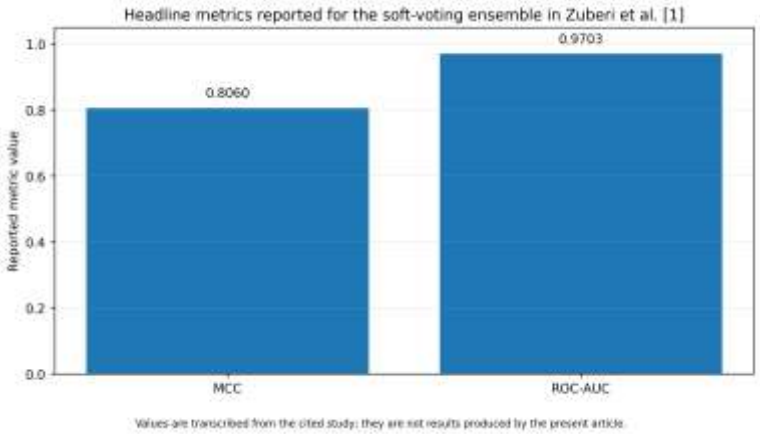

Figure 5. Matthews correlation coefficient and ROC-AUC reported for the soft-voting ensemble in the motivating benchmark study. These values are transcribed from the cited study and are not new results produced by this article.

The reported Matthews correlation coefficient of 0.8060 and ROC-AUC of 0.9703 indicate strong performance under the source study's own evaluation protocol[2]. ROC-AUC and the Matthews correlation coefficient nonetheless summarize different properties of a classifier, and neither alone establishes acceptable precision at a chosen production threshold, adequate probability calibration, stability across a longer run of time, or transferability to a different institution's data. Their interpretation should accordingly remain bounded by the specific data and evaluation design the source study actually used, rather than generalized beyond what that design can support. The class-distribution figure shown earlier demonstrates concretely why the choice of metric set matters so much here: with 492 fraud cases among 284,807 observations, the benchmark's positive prevalence is approximately 0.173 percent[2], and in a setting this imbalanced, even a seemingly small false-positive rate can translate into a very large number of alerts, which is exactly why precision-recall analysis and explicit workload measures belong alongside ROC-AUC rather than instead of a detailed accounting of alert volume[3].

Synthesis Finding 1: Temporal Validity Is a Design Property

The reviewed work supports the conclusion that temporal validity has to be designed into the full pipeline rather than achieved by a single step applied at the end. A chronological split is undermined the moment scaling, feature selection, oversampling, or threshold tuning is allowed to use any information from the test period, however indirectly. Conversely, a leakage-safe split does not by itself prove long-term robustness when the available timeline is short, as it is in the motivating benchmark. Evaluation quality in this domain

depends jointly on temporal ordering, strict pipeline isolation between periods, explicit handling of label maturity, and the diversity of the test windows actually examined, and a study that gets three of these four right while neglecting the fourth can still produce a materially misleading estimate.

Synthesis Finding 2: Drift Metrics Are Diagnostic, Not Dispositive

The population stability index, the Kolmogorov-Smirnov statistic, and Jensen-Shannon divergence each provide evidence of distributional change, but none of them directly measures a change in the conditional relationship between features and fraud, which is the relationship that actually determines predictive performance. The reimplementation illustrates the point directly: all 15 monitored features shifted significantly while the score distribution was stable and the weakest performance window produced no distributional signal. Their values should be interpreted in relation to sample size, the relevance of the feature being monitored, baseline variability in ordinary operation, and the operational consequences a given magnitude of shift would actually carry. Outcome-based metrics remain essential once labels are sufficiently mature to be trusted, and where labels are still delayed, a monitoring report should state explicitly that the effect on predictive performance is not yet known, rather than implying a conclusion the data cannot yet support.

Synthesis Finding 3: Model Selection Is Multi-Objective

Fraud detection is not reducible to maximizing any single metric, however attractive that would be operationally. Precision, recall, the Matthews correlation coefficient, PR-AUC, calibration quality, scoring latency, review capacity, and expected cost can each point toward a different preferred model, and a model that wins on one dimension may lose on several others. The preferred model should satisfy a predefined set of operational constraints rather than simply top a leaderboard on one metric, and it should be evaluated on the same temporal windows as every alternative under consideration. Thresholds, in turn, should be chosen on validation data and frozen before the final test evaluation is ever run, so that the comparison between candidate models remains fair and the final number reported is not itself the product of test-set tuning.

Synthesis Finding 4: Transferability Requires More Than Temporal Holdout

A model can generalize acceptably forward in time within one institution and still fail at another, because feature definitions, customer composition, product mix, and underlying fraud prevalence can all differ meaningfully across institutions even when each institution's own temporal holdout looks reassuring. The multi-institution architecture discussed earlier makes this generalization problem explicit rather than leaving it implicit[6]. Cross-site validation should therefore complement forward-in-time validation whenever deployment across more than one institution is intended, and the data-governance and secure-sharing arrangements such validation requires should be designed alongside the statistical protocol from the outset, not bolted on afterward once a cross-site pilot is already underway.

Operational Decision Matrix

Table 2 translates the synthesis above into concrete response guidance. The categories it presents are proposed governance rules rather than universal thresholds, and each institution applying them will need to calibrate the specific statistical triggers to its own data volume, risk appetite, and investigative capacity.

Table 2. Proposed interpretation of monitoring evidence and proportionate response.

Observed evidence	**Interpretation**	**Recommended response**
No material input shift; stable outcome metrics	No immediate evidence of deterioration	Continue routine monitoring; retain current version subject to review

Observed evidence	Interpretation	Recommended response
Input shift; outcome performance remains stable	Possible covariate drift without demonstrated failure	Inspect affected features; monitor persistence and segments
Input shift plus lower recall or precision	Distributional change accompanied by performance decline	Check labels, thresholds, data pipelines, and affected populations
Outcome performance declines but marginal inputs look stable	Possible concept drift, label issue, or unmonitored change	Investigate conditional relationships and label quality; increase review if warranted
SHAP attribution changes alongside a performance decline	Model reliance on features may have changed	Audit correlated features, data changes, and representative cases
Recent labels are incomplete	Performance impact cannot yet be established	Report uncertainty, use interim controls, reassess once labels mature

Discussion

Implications for Research

Researchers should report the full temporal design of their evaluation, including date boundaries, label-maturity rules, the order in which preprocessing steps were applied, the sampling strategy used, the method used to select a threshold, and whether the nominal test window was examined at any point during model development. Reporting only a random train-test split or a single summary score makes it genuinely difficult for a reader to assess how a model might behave under temporal change, which is, after all, the condition any deployed system will actually face. Evaluating multiple forward windows, rather than one, can reveal whether a reported result depends heavily on the particular period chosen, a dependency that a single-window study has no way of detecting on its own.

Comparisons across models or across studies should also distinguish ordinary performance variation from change that is both statistically and operationally meaningful. Confidence intervals and paired comparisons are useful tools here, but the resampling method used to construct them must respect temporal structure rather than treat transactions as independently exchangeable. If the test period contains very few fraud cases, as is common, uncertainty may remain large even when the point estimate looks strong, and researchers should state this limitation plainly rather than treating a single high score as conclusive evidence of anything beyond that one window.

Implications for Financial Institutions

Institutions should define monitoring ownership and escalation criteria before a system is ever deployed, not after the first alert arrives. Model owners, data engineers, fraud investigators, risk managers, and compliance teams typically observe different parts of the same underlying problem, and a shared record of data changes, drift alerts, label maturity, model versions, threshold changes, and operational outcomes can support genuinely coordinated decisions rather than each group acting on a partial picture. Where an alert would affect live customer transactions, proportionate human review and a working rollback mechanism are

important safeguards that should exist before they are needed, not be improvised once a problem has already surfaced.

Explainability and Governance

Explainability should be used to generate hypotheses about model behavior, not to substitute for empirical validation of those hypotheses. A change in SHAP values may usefully point an analyst toward a feature or customer segment worth investigating, but correlated variables and ordinary shifts in input distributions complicate that interpretation considerably[4]. Governance procedures should accordingly require an analyst to triangulate explanation-based evidence with data-quality checks and outcome-based metrics before treating an explanation shift as meaningful on its own. Material model changes, whatever evidence motivated them, should be approved, documented, validated on temporally later data, and monitored after release, exactly as any other production change of consequence would be.

Conclusion

Financial fraud detection requires evaluation methods that reflect the temporal and operational conditions of actual deployment, not merely the conditions of a single retrospective benchmark. Rare positive cases, evolving transaction behavior, delayed labels, and sharply asymmetric error costs together make a single historical score insufficient as evidence of operational robustness, however impressive that score may look in isolation. The literature reviewed here indicates that temporal partitioning, leakage prevention, imbalance-aware evaluation, explainability, and drift monitoring are complementary safeguards, but each has distinct and specific limits that the others do not automatically cover. This article has proposed a six-stage framework connecting data integrity, temporal design, model development, multi-dimensional evaluation, drift diagnosis, and governed response, and its central distinction throughout has been between detecting a statistical change and demonstrating that model performance has actually deteriorated in a way that matters operationally. Drift alerts, on this account, should trigger investigation and evidence-gathering; retraining or a threshold change should follow only once validated evidence and explicit operational objectives actually justify intervention, not as a reflexive response to a single statistic crossing a chosen line. Implemented on the European credit-card benchmark, the framework reproduced the source study's performance level (test F2 up to 0.754), showed that leakage can push F2 to 1.00, and showed that significant input drift on every monitored feature coexisted with a stable score distribution while a genuine dip in recall went unflagged. Future studies should apply it to longer-duration data and to multiple institutions, report uncertainty honestly, and evaluate both statistical and operational outcomes side by side. A useful next step would be a prospective study comparing single-threshold drift monitoring against a combined strategy that uses distributional indicators, mature-label performance, review capacity, and explicit governance controls together. Until such validation is available, the framework should be treated as empirically supported on one public benchmark rather than as a proven production solution.

References

[1] Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., & Bouchachia, A. (2014). A survey on concept drift adaptation. *ACM Computing Surveys, 46*(4), Article 44, 1-37.

[2] Zuberi, N., Rahman, M. M., Rashid, S. U., Ahsan, M. N., Rahman, M. D., & Mollah, M. H. O. R. (2026). A robust machine learning framework for detecting temporal drift in financial fraud prevention. *Scientific Reports*.

[3] Saito, T., & Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLOS ONE, 10*(3), e0118432.

[4] Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In Advances in Neural Information Processing Systems (Vol. 30, pp. 4765-4774). Curran Associates, Inc.

[5] Chicco, D., & Jurman, G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics, 21*, Article 6.

[6] Rashid, S. U., & Sattiraju, S. R. (2025). Multi-Institution Validation of Drift-Resilient Early-Warning Models for SME Loan Distress and Transaction Fraud: Evidence from US Community Financial Institution Pilots. *International Journal of Modern Computing*, *8*(1), 40-54.

[7] Dal Pozzolo, A., Caelen, O., Johnson, R. A., & Bontempi, G. (2015). Calibrating probability with undersampling for unbalanced classification. In *2015 IEEE Symposium Series on Computational Intelligence* (pp. 159-166). IEEE.

[8] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16, 321-357.

[9] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785-794).

[10] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems 30* (pp. 3146-3154).