International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 15(2023) pp. 1-23
https://doi.org/10.70917/ijcisim-2026-5990
Article

Fraud Model Maintenance: Why Your Detection Rate Drops Six Months After Launch

Olid Sarker[1*], Habibur Rahman[2]

[1]Feliciano School of Business, Montclair State University, olidsarker@gmail.com.
[2]Northern University of Bangladesh, nubhabib@gmail.com.

Abstract: Purpose – Practitioners often say that a fraud detection model loses its edge about six months after launch. This paper examines whether that belief holds up by auditing what the 2008 to 2023 literature claimed about maintaining fraud models. This study was designed and written in 2023 as a retrospective; it makes no claim to have been produced in the period it studies.
Design – Twelve maintenance claims were extracted from a verified corpus of 67 sources published between 2008 and 2023. Each was tested on a seeded synthetic transaction benchmark that instantiates the era's own assumptions (class imbalance, daily batches, changing fraud patterns, delayed labels), under decision rules fixed before the experiments, and against 34 later sources.
Findings – On the benchmark, 4 claims held, 6 were qualified, and 2 were overturned. A never-updated model lost 24% to 58% of its detection rate in six months depending on how long fraud patterns lasted, and nothing when they did not change. Retraining mattered more than the choice between non-linear learners. AUC understated decay, and input-stability monitoring did not detect changes in fraud.
Research limitations – All data are synthetic and results depend on the generator's design; the later literature reading is targeted, not systematic.
Practical implications – Teams should measure the turnover of fraud patterns in their own cases, shorten label delays, and set review cycles against turnover instead of relying on a fixed interval.
Originality – The paper offers a claim-by-claim verdict matrix for the era's fraud model maintenance literature, with a shipped, reproducible benchmark.
Keywords: Fraud detection, Model maintenance, Concept drift, Model risk management, Retraining, Retrospective audit

1. Introduction

Fraud teams repeat a rule of thumb that has no author. A new detection model performs well at launch, the story goes, and about six months later the detection rate has dropped. The figure circulates in vendor material without a source; one practitioner post states that models left unattended for six months can see a 35% increase in error rates and cites nothing for it (Megaladata, n.d.). The underlying phenomenon is real and was documented long before the rule of thumb. Between 2008 and 2015 a compact body of work on card fraud detection and on learning from drifting data streams set out most of what the field still believes about model maintenance: that static models decay, that frequent updating helps, that old data should be forgotten, that class imbalance has to be handled alongside drift, and that performance should be judged on the alerts an investigator can work (Hand et al., 2008; Dal Pozzolo et al., 2014a; Gama et al., 2014; Dal Pozzolo et al., 2015b).

This paper goes back to that period and asks which of its maintenance claims have survived. This study was designed and written in 2023 as a retrospective; it makes no claim to have been produced in the period it studies. The window of study is 2008 to 2015 inclusive. Everything analyzed as the object of study- that is, the claims, the assumptions behind the benchmark, and the methods treated as contemporary- comes from sources published inside that window. The paper's own framing is not bounded by it: later literature is cited wherever it has tested or revised an era claim, and the novelty of the contribution was checked against the literature as it stands in 2023.

The angle is a verdict audit. Twelve maintenance claims were extracted from an era corpus of 67 sources. Each claim was then tested on two legs. The first is a purpose-built synthetic benchmark whose data-generating process instantiates the era's own documented assumptions about fraud streams, so that a claim is tested under the conditions its authors said they were describing. The second is a reading of 34 post-window sources for what later work established. Each claim receives a verdict of held, qualified or overturned on the benchmark, under decision rules written down before the experiments were run, and a separate note on what later literature says.

The window was chosen for a reason. It opens with a treatment of how plastic card fraud tools should be scored and closes with a formal treatment of delayed labels and investigator feedback in a card fraud stream. After 2015, much of the field's attention moved to sequence models and large-scale engineering, and the maintenance questions drew less of it. They had been answered on very little evidence: the era's most direct empirical paper on the subject used one proprietary dataset of 422 daily batches, and its successor used two. Because the underlying data cannot be shared, the claims have been inherited more often than they have been re-examined.

The paper makes three contributions, and it promises nothing beyond them. First, it provides a claim-level register of what the 2008 to 2023 literature asserted about maintaining fraud models, with every claim traced to sources that were read at abstract or full-text level. Second, it reports a verdict matrix for those twelve claims from a seeded, fully shipped synthetic benchmark, with the decision rules fixed in advance. Third, it replaces the six-month rule of thumb with a statement of what the six-month figure depends on, namely how quickly fraud patterns turn over, and shows how wide the range of honest answers is. The benchmark is synthetic, and every number in the results was produced by it; no number here is a measurement of any real institution's fraud losses or model.

2. Background and related work

2.1 What the 2008 to 2023 literature held

The era's work on fraud model maintenance drew on two literatures that only partly overlapped. The fraud detection literature was dominated by proposals of new detectors evaluated once on a fixed extract: hidden Markov models (Srivastava et al., 2008), evidence fusion (Panigrahi et al., 2009), association rules (Sánchez et al., 2009), peer group analysis (Weston et al., 2008), artificial immune systems (Gadi et al., 2008; Wong et al., 2012; Soltani Halvaiee and Akbari, 2014), cost-sensitive trees and meta-heuristics (Duman and Ozcelik, 2011; Sahin et al., 2013; Correa Bahnsen et al., 2015), transaction aggregation (Whitrow et al., 2009; Jha et al., 2012), and network features (Van Vlasselaer et al., 2015). Surveys of the period catalogued techniques by algorithm family (Ngai et al., 2011; Phua et al., 2010). Comparative studies asked which algorithm was most accurate (Bhattacharyya et al., 2011; Juszczak et al., 2008). Maintenance was seldom the subject. Two strands were the exception. One, centered on Hand and colleagues, argued that the measure used to judge a fraud tool has to reflect the cost to the card issuer and the number of investigations it can afford (Hand et al., 2008). The other, the work of Dal Pozzolo and colleagues with an industrial partner, asked the practitioner's questions directly: whether a model learned once a month is enough or whether it must be updated every day, how much history to train on, and how to rebalance (Dal Pozzolo et al., 2014a; Dal Pozzolo, 2015). Both strands built on pre-window foundations: the statistical treatment of fraud detection (Bolton and Hand, 2002) and the observation that changing populations erode classifier performance (Kelly et al., 1999).

The second literature was data stream mining. It inherited its first drift learners and detectors from before the window (Widmer and Kubat, 1996; Gama et al., 2004; Bifet and Gavaldà, 2007), and it supplied the vocabulary of concept drift and a taxonomy of adaptation strategies (Gama et al., 2014; Žliobaitė, 2010; Moreno-Torres et al., 2012; Quiñonero-Candela et al., 2008); ensemble methods designed to track change (Bifet et al., 2009; Bifet et al., 2010; Elwell and Polikar, 2011; Minku et al., 2010; Minku and Yao, 2012; Brzezinski and Stefanowski, 2014); methods for imbalanced streams (Chen and He, 2009; Hoens et al., 2012; Ditzler and Polikar, 2013; Wang et al., 2015); drift detectors based on the error rate or on the data (Dries and Rückert, 2009; Ross et al., 2012; Alippi et al., 2013); and a methodology of sequential evaluation (Gama et al., 2009; Gama et al., 2013; Bifet et al., 2013; Žliobaitė et al., 2015). By the end of the window, the two literatures had met. Ditzler et al. (2015) stated the shared premise without hedging: a model trained under a false assumption of stationarity is bound to become obsolete. Krempl et al. (2014) listed

delayed and incomplete information among the open challenges for the field, and Dal Pozzolo et al. (2015b) formalised it for card fraud, where an investigator labels only the alerted transactions straight away and everything else is labelled days later.

Two neighbouring bodies of work shaped practice in the same years. Credit scoring researchers treated population drift as a known problem handled by periodic rebuilding of the scorecard, and proposed adaptive alternatives (Adams et al., 2010; Pavlidis et al., 2012); they also showed that a model which selects its own training data biases later comparisons (Hand and Adams, 2014), and they benchmarked classification algorithms at scale (Lessmann et al., 2015). Supervisors, for their part, set the cadence that many risk functions still work to. The US guidance on model risk management asks banks to review each model at least annually, more often if warranted, and to recalibrate or redevelop when monitoring shows it is needed (Board of Governors of the Federal Reserve System and OCC, 2011). Engineering practice arrived at a similar point from another direction when Sculley et al. (2015) argued that machine learning systems carry large ongoing maintenance costs, changes in the external world among them. The remaining era sources, which propose or survey detectors and adaptation methods without making a distinct maintenance claim, are listed with the full corpus in the Appendix.

2.2 What came after, and the nearest existing work

Later work extended the era's programme. Dal Pozzolo et al. (2018) repeated the feedback and delayed-label design on more than 75 million transactions over three years. Streaming architectures and incremental learners were built around the same formalisation (Carcillo et al., 2018; Lebichot et al., 2021), sequence models, domain adaptation and hybrid supervised and unsupervised detectors were added (Jurgovsky et al., 2018; Lebichot et al., 2020; Carcillo et al., 2021), cost-based evaluation was developed further (Correa Bahnsen et al., 2016), and attempts were made to quantify dataset shift in card data directly (Lucas et al., 2019). Credit risk modelling moved towards explicitly dynamic frameworks (Sousa et al., 2016). In stream mining, surveys consolidated the drift literature (Webb et al., 2016; Žliobaitė et al., 2016; Gomes et al., 2018; Lu et al., 2018; Bayram et al., 2022), adaptive forests were proposed (Gomes et al., 2017), the shortage of public non-stationary real-world benchmark data was documented (Souza et al., 2020), and a line of work examined the assumption that labels arrive immediately (Plasse and Adams, 2016; Sethi and Kantardzic, 2017; Grzenda et al., 2020; Gemaque et al., 2020). Engineering practice produced rubrics for production readiness and surveys of deployment problems (Breck et al., 2017; Paleyes et al., 2023). Outside fraud, Vela et al. (2022) ran train-then-deploy experiments on 32 datasets from four industries and reported temporal degradation in 91% of 128 model and dataset pairs. Reviews of fraud detection research continued to appear after the window (West and Bhattacharya, 2016; Ryman-Tubb et al., 2018; Hilal et al., 2022), and one lists concept drift among the issues that hinder fraud detection systems (Abdallah et al., 2016).

A search for an existing retrospective of this kind was run before the study was designed and again against the finished draft, using three phrasings: a positive form (retrospective or audit of early fraud detection concept-drift claims), a topical form (empirical studies of fraud model degradation and retraining frequency), and a negative form (statements that early claims have not been systematically re-examined). No claim-level audit of the 2008 to 2015 fraud maintenance literature was found. The nearest neighbours answer different questions. Vela et al. (2022) characterise degradation patterns across industries and do not include fraud. Galen and Steele (2020) measure how malware detection models deteriorate with time since training, and Brettle and Steele (2021) report a hospital mortality model whose performance held up better than expected. Kidane et al. (2022) ask when and how to retrain workload prediction models in cloud management, in a pilot study. Each is used below as verdict evidence. None of them sets the era's claims side by side and scores them, which is the gap this paper fills. By the authors' choice the reference list stops at sources published in 2023. The search itself was run against the literature as it stood in October 2026, and its record, including later work that was located and is not cited here, is in the audit file shipped with the paper.

The retrospective form itself has precedents that informed the design. Large replication efforts showed that a published result and a reproducible one are different things (Open Science Collaboration, 2015; Ioannidis, 2005). Within machine learning, audits have found that reported progress can rest on weak baselines or leakage (Ferrari

Dacrema et al., 2019; Kapoor and Narayanan, 2023), and that claims are often stated more strongly than the evidence allows (Lipton and Steinhardt, 2019); the argument that classifier comparisons overstate real-world differences predates the window (Hand, 2006). The present paper follows the practice, common to those studies, of fixing the decision rule before looking at the outcome, and it treats the era's consensus as something to be problematised, not summarised (Alvesson and Sandberg, 2011).

2.3 The six-month figure

No source inside the window states that fraud models degrade in six months. The era's own descriptions of practice were looser: a detection model is periodically retrained from scratch, for example once a year or once a month (Dal Pozzolo, 2015), and supervisory review is at least annual (Board of Governors of the Federal Reserve System and OCC, 2011). The six-month figure was located only in undated practitioner material without a citation (Megaladata, n.d.). It is treated here as an unsourced claim to be tested, and the title of this paper should be read in that light.

3. Method

3.1 Design

The study is a verdict audit with two legs. On the benchmark leg, each era claim is translated into a comparison that can be run on a synthetic transaction stream, and a decision rule maps the outcome to one of three verdicts. A claim is held if the comparison comes out as the era said in every drift regime tested. It is qualified if it holds in some regimes and not others, or holds only under a condition the era did not state. It is overturned if the comparison comes out null or reversed where the claim should apply. On the literature leg, the post-window sources are read for direct tests of the same claim. The two legs are reported side by side and are not merged into a single score, because they are different kinds of evidence: the benchmark shows whether a claim follows from the era's own assumptions, and the later literature shows whether anyone has checked it on real data.

The decision rules were written to a file before any of the reported experiments was executed, and that file is shipped unchanged with the code and reproduced in the Appendix. This is a weaker safeguard than external preregistration and is described as what it is. Pilot runs made before the rules were fixed were used for two purposes only: to set the difficulty of the detection task at launch, and to size the experiments to the available computing time.

3.2 Corpus

The era corpus contains 67 sources published from 2008 to 2015. Candidates were identified from web search, Crossref, OpenAlex, Semantic Scholar and arXiv, starting from the period's surveys of fraud detection and concept drift and following their most cited fraud and maintenance references. A source was included if it addressed payment or card fraud detection, learning under drift, the evaluation or monitoring of evolving models, or the governance of deployed models. Table 1 gives the composition. The publication year is the print year recorded by Crossref, or the issue year where no print year is recorded, and that rule decides window membership. A separate verdict corpus of 34 post-window sources was assembled for the literature leg, together with 6 pre-window background sources and 6 sources on retrospective method.

Every entry was verified before any text was drafted. 108 entries were matched to a Crossref record by title and the record's authors, year, venue and DOI were used as retrieved; these are tagged [CR]. The remaining 9, which have no DOI or are web documents, were verified against the issuing site and are tagged [WS]. A claim is attributed to a source only where the source was read at abstract level or in full; for the era's central practitioner study the content was read in the author's doctoral thesis, which reproduces it (Dal Pozzolo, 2015). Sources for which only the bibliographic record could be retrieved are cited for their existence and topic and for nothing else. The corpus file shipped with the paper records, for each entry, the tag and the basis on which its content was used.

Table 1. Composition of the era corpus (2008 to 2015)

Theme	Sources
Card and payment fraud detectors and studies	31
Fraud surveys and books	3
Concept drift and stream learning	23
Evaluation of evolving models	4
Credit scoring: drift, selection, benchmarking	4
Supervision and engineering practice	2
Total	67

Source: authors' own work. The full list, with verification tags, is in the Appendix and in corpus.csv.

3.3 Claim register

Twelve claims were extracted (Table 2). A statement qualified as a claim if it was asserted as a finding or as a design principle in at least one era source that was read, and if it bears on how a deployed fraud model should be maintained, measured or monitored. Claims about detector architecture were left out. The wording in Table 2 is a paraphrase written to be testable; the sources are those in which the statement was read.

Table 2. Register of era claims about fraud model maintenance

Claim	Statement as tested	Era sources read
C1 Decay	A fraud classifier that is trained once and never updated loses detection performance as it ages.	Dal Pozzolo et al., 2014a; Adams et al., 2010; Ditzler et al., 2015
C2 Cadence	The sooner the model is updated the better: daily updating beats weekly, which beats fortnightly.	Dal Pozzolo et al., 2014a
C3 Window	More training history is better than less, and a very short window is the worst choice; outdated data must still be forgotten.	Dal Pozzolo et al., 2014a; Kuncheva and Žliobaitė, 2009
C4 Fraud memory	Old fraud cases should be carried forward while old genuine transactions are forgotten.	Dal Pozzolo et al., 2014a; Chen and He, 2009
C5 Rebalancing	SMOTE and EasyEnsemble outperform plain undersampling on drifting, imbalanced streams.	Dal Pozzolo et al., 2014a; Ditzler and Polikar, 2013
C6 Algorithm	Random forest clearly outperforms the alternatives, so algorithm choice is a first-order decision.	Dal Pozzolo et al., 2014a
C7 Feedback	Investigator feedback and delayed labels should be learned by separate classifiers and then aggregated, not pooled.	Dal Pozzolo et al., 2015b
C8 Metrics	Global ranking measures such as AUC are the wrong yardstick; performance should be judged on the alerts investigators can actually work.	Hand et al., 2008; Dal Pozzolo et al., 2015b; Correa Bahnsen et al., 2013
C9 Error charts	Monitoring the classifier's error rate with a control chart detects drift promptly.	Ross et al., 2012; Gama et al., 2014
C10 Input monitoring	Change in the input distribution, which needs no labels, signals that the model needs attention.	Dries and Rückert, 2009; Alippi et al., 2013

C11 Ensembles	Ensembles of models built on successive batches cope with drift better than a single model.	Bifet et al., 2009; Elwell and Polikar, 2011; Brzezinski and Stefanowski, 2014
C12 Evaluation	Evaluation that ignores time order misstates how a model will perform once deployed; assessment should be sequential.	Gama et al., 2009; Gama et al., 2013; Bifet et al., 2013

Source: authors' own work.

3.4 Synthetic benchmark

The benchmark is a seeded generator of card-not-present transaction streams. All data are synthetic. They are era-styled, in that scale, imbalance and measurement practice are modelled on what the era's papers describe, and they are not era-originated: no real transaction, real cardholder or real institution is represented. Table 3 lists the assumptions built into the generator and the era sources from which each is taken.

Table 3. Era assumptions instantiated by the benchmark

Assumption	**Era source**	**Implementation**
Fraud is a small minority of transactions	Dal Pozzolo et al., 2014a; Dal Pozzolo et al., 2015b	Mean fraud rate 0.5% of 2,000 transactions per day
Data arrive in daily batches	Dal Pozzolo et al., 2014a	540 days: 180 for development, 360 after launch
Fraudsters change strategy over time	Gama et al., 2014; Ditzler et al., 2015	Six fraud patterns are active at any time; patterns retire and are replaced
Genuine behaviour evolves and is seasonal	Dal Pozzolo et al., 2015b	Random walk and annual cycle in the means of four customer segments
The number of frauds per day fluctuates	Dal Pozzolo et al., 2015b	Daily fraud rate varies lognormally (s.d. 0.2 on the log scale)
Drift may be gradual, abrupt or recurring	Gama et al., 2014; Minku et al., 2010; Alippi et al., 2013	Eight regimes (Table 4)
Investigators check a fixed number of alerts	Hand et al., 2008; Dal Pozzolo et al., 2015b	Alert budget of 1% of daily volume (20 alerts)
Most labels arrive after a delay	Dal Pozzolo et al., 2015b; Krempl et al., 2014	Labels available after 7 days by default; 0, 30 and 60 days also tested

Source: authors' own work.

Each transaction has 15 numeric features, of which 12 can carry signal and 3 are noise. Genuine transactions are drawn from four customer segments with different feature means. A fraud pattern is defined by three features that it displaces from the segment mean by between 1.5 and 3.0 standard deviations. A pattern's lifetime is exponentially distributed, and its mean is the parameter that sets the speed of drift; a replacement pattern, which displaces a fresh random set of features, takes 14 days to reach full prevalence. The displacement range was set once, on pilot runs in the stationary regime, so that detection at launch is well away from both zero and one. No parameter was tuned on how fast detection decays.

Table 4. Drift regimes

Regime	**Fraud patterns**	**Genuine behaviour**

Stationary control	Fixed	Fixed
Gradual, 24, 12, 6 and 3 months	Retire after an exponential lifetime with that mean	Slow random walk and mild seasonality
Abrupt	All six replaced at once, 120 days after launch	Fixed
Recurring	Two sets alternate: 150 days of one, 90 days of the other	Fixed
Covariate only	Fixed	Faster random walk and stronger seasonality

Source: authors' own work.

3.5 Learners, policies and measures

The methods are those the era treated as current. The default learner is a random forest trained on a balanced sample obtained by under sampling genuine transactions, which is the configuration the era's practitioner study settled on (Dal Pozzolo et al., 2014a); forests have 50 trees, half the era's usual 100, to keep the benchmark re-runnable on two processor cores. Logistic regression and a single-hidden-layer neural network are the comparison learners. Maintenance policies vary the retraining interval (never, 168, 84, 28, 14, 7 days and daily), the training window (7, 30, 90, 180 days and all history), whether past frauds are carried forward, whether a single model or an ensemble of five batch models is used, and how rebalancing is done (under-sampling at one or five genuine transactions per fraud, SMOTE, and Easy Ensemble with five members). The feedback experiment reproduces the era design of Dal Pozzolo et al. (2015b): a classifier trained on the investigators' feedback from the last seven days, a classifier trained on a 16-day window of delayed labels, their average, and a single classifier trained on both pooled.

The primary measure is the detection rate at the alert budget: the share of each day's frauds that fall among the 1% of transactions the model ranks highest, pooled over 30-day blocks that are referred to as months. Alert precision, the area under the ROC curve (AUC), and average precision are computed on the same days. Monitoring is examined with two tools the era describes. The first is an exponentially weighted moving average chart on the daily detection rate, with weight 0.1 and three-sigma limits estimated on 30 out-of-sample days before launch, in the manner of Ross et al. (2012); the chart can only consume a day's outcome once its labels have arrived. The second is the population stability index, computed on the model's scores and on each input feature against the same 30 reference days, with the conventional alarm level of 0.25, a rule of thumb whose statistical basis was later questioned (Taplin and Hunt, 2019; Yurdakul and Naranjo, 2020).

The static decay, evaluation, and input-monitoring experiments use 20 independently seeded streams in each of the eight regimes. The retraining, window, ensemble, algorithm, and chart experiments use 8 streams in each of four or five regimes; the daily and weekly retraining comparison uses five streams in the two central gradual regimes; rebalancing uses eight streams in three regimes; and the feedback experiment uses six streams in three regimes. Comparisons are paired on the stream, and intervals are 95% t-intervals on the per-stream differences. No correction is made for the number of comparisons, so an individual interval that narrowly excludes zero should be read with caution.

3.6 Literature leg

For each claim, the verdict corpus was searched for a later study that tested it on real data. The result is recorded as supported, partly supported, contested, or no later test located. A claim is marked supported only when a later source reports the same comparison; statements of agreement without a test do not count. Where the only later test comes from the same research group as the original claim, this is stated.

4. Results

4.1 Decay, and what the six-month figure depends on

A never-updated random forest held its detection rate in the stationary control and lost it in every regime with fraud drift (Figure 1). In the control, the detection rate was 0.60 in the first month after launch and 0.61 in the twelfth. With gradual turnover the month-twelve rate was lower than the month-one rate by 14.5 points (95% CI 9.7 to 19.4) when fraud patterns lasted 24 months on average, by 35.6 points (95% CI 29.4 to 41.9) at 12 months, by 34.5 points (95% CI 28.6 to 40.4) at 6 months and by 24.0 points (95% CI 19.4 to 28.6) at 3 months. In the abrupt regime, detection fell from 0.62 to 0.07 when the patterns were replaced and did not recover. Claim C1 is held.

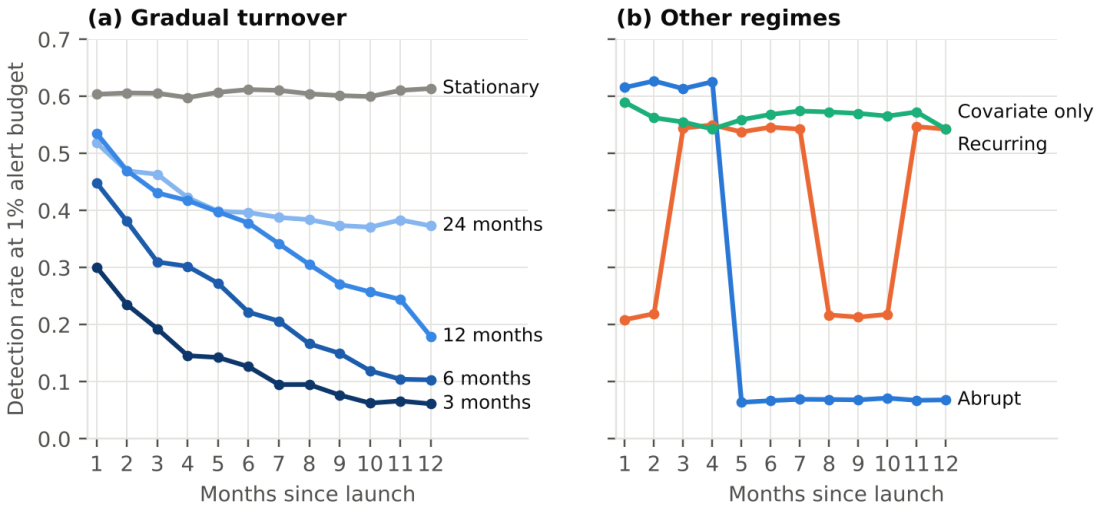


Figure 1. Detection rate of a never-updated random forest by month since launch, mean of 20 synthetic streams per regime. Panel (a): gradual turnover of fraud patterns, labelled by mean pattern lifetime, with the stationary control. Panel (b): abrupt, recurring and covariate-only regimes. Source: authors' own work.

The six-month figure is a different matter. Six months after launch, the never-updated model had lost 24% of its month-one detection rate when patterns lasted 24 months, 29% at 12 months, 51% at 6 months and 58% at 3 months, against a change of +1% in the stationary control (Figure 2). A loss of one tenth was reached in month 3 at the slowest turnover and in month 2 at the 6-month lifetime; a loss of one half was reached in month 10, 6 and 4 for lifetimes of 12, 6 and 3 months, and was

not reached within the year at 24 months. So a material drop by six months is what the era's assumptions predict whenever fraud patterns turn over at all, but the size of the drop varies by a factor of more than two across plausible turnover speeds. There is no six-month constant to be found. The faster regimes also show something the rule of thumb hides: at a 3-month lifetime the model launched with a detection rate of 0.30, half the control's, because much of its 180 days of training data described patterns that had already retired. Part of the loss precedes launch.

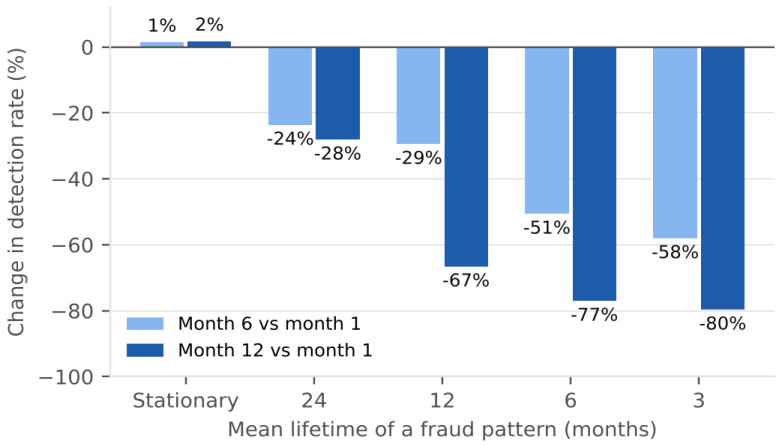


Figure 2. Change in the detection rate of a never-updated random forest at month 6 and month 12, relative to month 1, by mean lifetime of a fraud pattern. Source: authors' own work.

The mechanism is visible when frauds are split by whether their pattern existed at launch. Averaged over the year, the static model caught 56% of frauds from patterns it had been trained on and 4% of frauds from patterns that appeared later (12-month regime; the 6-month regime gives 50% and 5%). Decay in this benchmark is therefore close to a count of how much of today's fraud comes from patterns younger than the model: by month 12 that share was 34%, 73%, 89% and 98% at lifetimes of 24, 12, 6 and 3 months. The covariate-only regime, in which genuine behavior drifts and fraud patterns do not, cost the static model 8% of its detection rate over twelve months.

4.2 Retraining interval and label delay

Retraining recovered most of what the static model lost (Figure 3). With a 90-day training window and a 7-day label delay, retraining every 28 days raised the twelve-month mean detection rate over never retraining by 25.2 points (95% CI 17.9 to 32.6) in the 12-month regime, by 22.0 points (95% CI 17.5 to 26.5) in the 6-month regime and by 27.3 points (95% CI 24.4 to 30.1) in the abrupt regime. In the stationary control the same comparison gave 0.5 points (95% CI −0.9 to 1.8). Each shortening of the interval, from 168 to 84 to 28 days, added detection in the drifting regimes; the step from 28 to 14 days added 1.5 points (95% CI 1.0 to 1.9) and 1.2 points (95% CI 0.7 to 1.8) in the two gradual regimes.

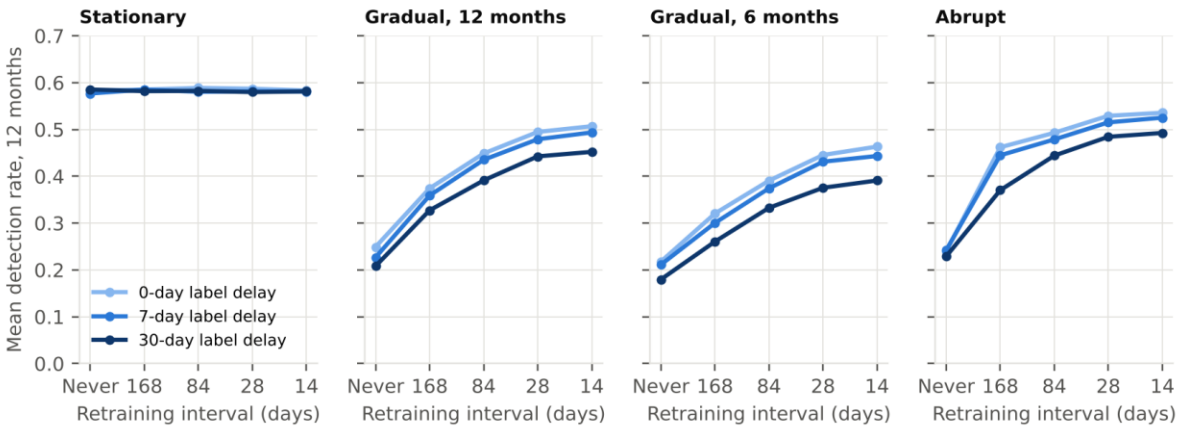


Figure 3. Twelve-month mean detection rate by retraining interval and label delay, mean of 8 synthetic streams per regime; 90-day training window. Source: authors' own work.

The era's specific finding was that daily updating beats weekly, which beats fortnightly. It was obtained under the assumption that every label is known at once. On the benchmark, with no label delay, daily against weekly retraining gave 0.41 points (95% CI −0.91 to 1.73) in the 12-month regime and 1.15 points (95% CI 0.59 to 1.71) in the 6-month regime, and weekly against fortnightly gave 0.57 points (95% CI −0.47 to 1.61) and 0.82 points (95% CI 0.28 to 1.35). With a 7-day delay the daily against weekly differences were 0.75 points (95% CI −0.49 to 2.00) and 0.50 points (95% CI −0.13 to 1.12). Mean detection rates under daily, weekly and 28-day retraining at the 7-day delay were 0.496, 0.488 and 0.470 in the 12-month regime and 0.472, 0.467 and 0.447 in the 6-month regime. These comparisons rest on five streams per regime. Claim C2 is qualified. Updating beats not updating by a wide margin, shorter intervals never hurt, and daily retraining beat 28-day retraining in every case. But the advantage of daily over weekly was distinguishable from zero only in the 6-month regime with immediate labels, the condition closest to the era's own experiment, and not in either regime once labels were a week late.

A long label delay cost more than the last step of retraining frequency bought. At a 28-day interval, moving from immediate labels to a 30-day delay changed the mean detection rate by −5.2 points (95% CI −6.3 to −4.2) in the 12-month regime, −7.0 points (95% CI −8.2 to −5.8) in the 6-month regime and −4.5 points (95% CI −5.5 to −3.4) in the abrupt regime; a 7-day delay changed it by −1.6 points (95% CI −2.4 to −0.7) and −1.4 points (95% CI −2.4 to −0.4) in the gradual regimes. A team that retrains fortnightly on labels that are a month old is, on these numbers, behind a team that retrains monthly on labels that are a week old.

4.3 What to keep in the training set

The best training window depended on the regime (Figure 4). The 7-day window was the worst choice everywhere, as the era reported. Beyond that the answer split. With gradual turnover the 90-day window was best: it beat the 30-day window by 5.1 points (95% CI 3.7 to 6.4) and 3.3 points (95% CI 1.5 to 5.0) in the 12-month and 6-month regimes, and it beat training on all history by 4.1 points (95% CI 2.1 to 6.1) and 4.9 points (95% CI 2.7 to 7.0). After the abrupt change the 90-day and 180-day windows could not be separated (0.1 points, 95% CI −0.4 to 0.7). In the stationary control and in the recurring regime, where old patterns come back, all history was best, ahead of the 90-day window by 8.2 and 15.6 points. In the recurring regime the 90-day window was worse than the 30-day window (−3.0 points, 95% CI −4.6 to −1.5), since a window of that length is dominated by the season that has just ended. Claim C3 is qualified: the short window is indeed worst, but how much history to keep cannot be read off without knowing which kind of drift is present, and the two kinds call for opposite choices.

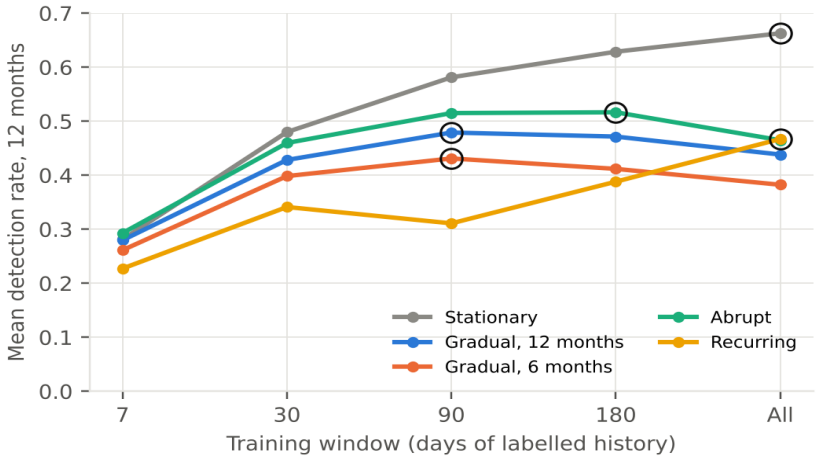


Figure 4. Twelve-month mean detection rate by training window, retraining every 28 days with a 7-day label delay; mean of 8 synthetic streams per regime. The circled point marks the best window in each regime. Source: authors' own work.

Carrying past frauds forward while keeping only 30 days of genuine transactions beat the plain 30-day window in the 12-month regime (4.1 points, 95% CI 2.4 to 5.8) and the abrupt regime (5.4 points, 95% CI 4.1 to 6.7), and made no detectable difference in the 6-month regime (0.7 points, 95% CI −2.1 to 3.6). It did not beat simply using a 90-day window: −1.0 points (95% CI −3.2 to 1.2), −2.6 points (95% CI −4.1 to −1.0) and −0.1 points (95% CI −0.8 to 0.5) in the three regimes. The benefit of fraud memory in the drifting regimes is therefore the benefit of having more fraud examples, which a longer window supplies as well. It did help where old patterns return (11.3 points, 95% CI 9.9 to 12.6, against the 90-day window in the recurring regime). Claim C4 is qualified.

An ensemble of the five most recent 28-day batch models was compared with one model trained on the same 140 days. The single model was better in every regime: the ensemble's difference was −4.3 points (95% CI −5.6 to −3.0) in the 12-month regime, −3.7 points (95% CI −6.1 to −1.4) in the 6-month regime, −4.9 points (95% CI −6.6 to −3.1) in the abrupt regime and −7.2 points (95% CI −9.7 to −4.7) in the recurring regime. Claim C11 is overturned on the benchmark for this batch-ensemble form. The era's practitioner study had itself noted that single models can beat ensembles when training sets overlap (Dal Pozzolo et al., 2014a); the stream-mining ensembles that reweight or prune members according to recent accuracy were not tested here.

4.4 Learner choices: rebalancing, algorithm, feedback

Both alternatives to plain under sampling improved detection in all three regimes tested. Easy Ensemble added 5.8 points (95% CI 4.8 to 6.7) and 5.1 points (95% CI 4.7 to 5.5) in the 12-month and 6-month regimes, and SMOTE added 2.5 points (95% CI 1.9 to 3.1) and 1.5 points (95% CI 0.5 to 2.5). Claim C5 is held. A simpler variant the era did not foreground did as well as either: under sampling to five genuine transactions per fraud instead of one added 6.5 points (95% CI 5.1 to 7.9) and 6.5 points (95% CI 5.6 to 7.4). On this benchmark, the gain comes from discarding less of the majority class, by whatever route.

Random forest was far ahead of logistic regression, at launch (33.7 points, 95% CI 27.8 to 39.6 in month one of the 12-month regime) and under 28-day retraining (28.7 points, 95% CI 23.4 to 34.0). It was not ahead of a small neural network. At launch the network's month-one detection rate was higher in all eight regimes (random forest minus network: −2.8 points (95% CI −5.4 to −0.3) in the 12-month regime), and under retraining the forest trailed it by 2.9, 4.2 and 4.3 points in the 12-month, 6-month and abrupt regimes. Claim C6 is qualified: the forest's superiority is over the linear model only. The more useful comparison for a maintenance paper is between the two decisions. In the 12-month regime, choosing between the forest and the network moved the detection rate by 2.9 points, and choosing between 28-day retraining and none moved it by 25.2 points; in the 6-month regime the figures were 4.2 and 22.0. Between competent non-linear learners, the maintenance decision was several times the larger one.

The feedback experiment reproduced the era's main result, though not its full ordering. Mean alert precision in the 12-month regime was 0.234 for the aggregate of the feedback classifier and the delayed-label classifier, 0.137 for the feedback classifier alone, 0.217 for the pooled classifier and 0.204 for the delayed-label classifier alone. The aggregate beat the pooled classifier by 1.7 points (95% CI 1.1 to 2.2) in the 12-month regime, 1.7 points (95% CI 1.2 to 2.2) in the abrupt regime and 1.6 points (95% CI 0.7 to 2.6) in the stationary control, and beat the delayed-only classifier by 2.9 points (95% CI 2.2 to 3.7), 3.0 points (95% CI 2.5 to 3.5) and 3.0 points (95% CI 2.1 to 3.9). Claim C7 is held in all three regimes. One detail differs from the era's report. There the feedback classifier on its own came second, ahead of the pooled classifier (Dal Pozzolo et al., 2015b); here it came last, which is unsurprising given that it learns from 140 labelled alerts where the era's learned from 700.

4.5 Measurement and monitoring

AUC understated the decay that investigators would have experienced (Figure 5). Over twelve months in the 12-month regime the static model's detection rate at the alert budget fell by 67% and its alert precision by 67%, while its AUC fell by 15%, from 0.951 to 0.804. In the 6-month regime the corresponding figures were 77%, 76% and 15%, and after the abrupt change a model that caught 7% of frauds at the alert budget still had an AUC of 0.777. Average

precision tracked the alert-based measures (a fall of 73% in the 12-month regime). Claim C8 is held: a monitoring report built on AUC would have shown a model in mild decline while its working detection rate fell by half or more.

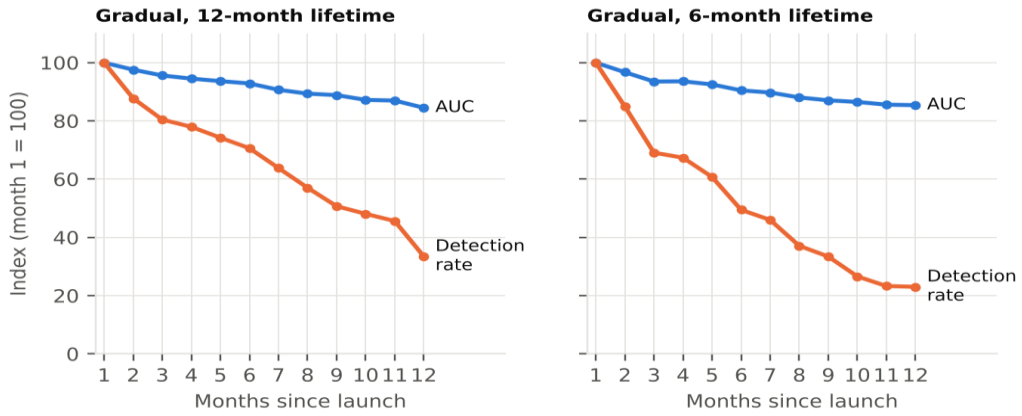


Figure 5. AUC and detection rate at the alert budget for a never-updated random forest, indexed to month 1 = 100. Source: authors' own work.

Random five-fold cross-validation on the development data overstated the first month's detection rate, where turnover was fast, and matched it where the stream was stable. The cross-validated estimate exceeded the realized month-one rate by 11.1 points (95% CI 6.4 to 15.8) in the 6-month regime and 11.7 points (95% CI 7.4 to 16.0) in the 3-month regime, by 4.6 points (95% CI 1.9 to 7.2) at 24 months, and by an amount whose interval includes zero at 12 months (3.7 points, 95% CI −2.0 to 9.4) and in the stationary control (−1.0 points, 95% CI −2.5 to 0.5). An out-of-time holdout on the last 30 development days was closer to month one in every gradual regime (2.5 points, 95% CI −3.0 to 8.0 at 6 months). Neither estimate says anything about month six: in the 6-month regime cross-validation overstated the month-six rate by 33.8 points. Claim C12 is qualified on the pre-specified rule, because the optimism was not significant in every drifting regime, but its direction never reversed where patterns turn over.

The error-rate chart detected the abrupt change in every run, and its lag was the label delay plus about two days: a median of 2 days with immediate labels, 9 days at a 7-day delay, 32 at 30 and 62 at 60. In the stationary control the chart raised at least one false alarm in 50% of runs over the year. Claim C9 is qualified: the chart is prompt only in the sense that it adds little to a delay it cannot shorten. Rebuilding the model when the chart alarmed, from data observed since the alarm, matched 28-day calendar retraining in the abrupt regime at a 7-day delay (1.1 points, 95% CI −1.3 to 3.5) with 3.5 rebuilds on average against 12. In the gradual regimes it did worse than the calendar: −5.6 points (95% CI −8.7 to −2.6) and −4.5 points (95% CI −6.7 to −2.2) in the 12-month and 6-month regimes, using 6.2 and 6.5 rebuilds.

Label-free monitoring of inputs failed in the way that matters most for fraud. The population stability index, computed on the model's scores and on every feature, raised no alarm at the conventional 0.25 level by month six in any run of any regime in which fraud patterns changed, including the abrupt regime, where the detection rate had fallen by 89% and the largest feature index stood at 0.001. It did alarm in 80% of runs of the covariate-only regime, where the detection rate was down 8%. Across all runs and months the rank correlation between the largest feature index and the loss of detection was weak, at 0.24. The reason is arithmetic. Fraud is half of one per cent of transactions, so even a complete replacement of fraud patterns moves the pooled input distribution by an amount the index cannot see, while a drift in genuine behavior moves it visibly whether or not detection suffers. Claim C10 is overturned on the benchmark for this form of monitoring.

4.6 Verdict matrix

Table 5 collects the verdicts. On the benchmark, four of the twelve claims were held (C1, C5, C7, C8), six were qualified (C2, C3, C4, C6, C9, C12) and two were overturned (C10, C11). The later literature supports the claims

about decay, feedback, alert-based measurement, and time-ordered evaluation; has moved against the monitoring claims in the same direction as the benchmark; and has left the claims about retraining frequency and about what to keep in the training set untested on real fraud data.

Table 5. Verdict matrix for the twelve era claims

Claim	**Benchmark verdict**	**Key benchmark result**	**Later literature**
C1 Decay	Held	Month 12 vs month 1: −35.6 points (12-month regime); +1.0 (control)	Supported, with an exception. Dal Pozzolo et al. (2018) compare a never-updated classifier with updated ones over ten months of real transactions and read the result as evidence of drift; Vela et al. (2022) find degradation in 91% of 128 model and dataset pairs outside fraud; Galen and Steele (2020) find deterioration over time in malware detection models. Brettle and Steele (2021) report a hospital model that did not deteriorate as expected. None reports a decay rate for fraud.
C2 Cadence	Qualified	Daily vs weekly, 7-day delay: +0.75 points; 28 days vs never: +25.2	No later test located. Lebichot et al. (2021) set incremental learning against batch retraining on card data, and Kidane et al. (2022) study when to retrain outside fraud. No later test of daily against weekly updating on fraud data was located.
C3 Window	Qualified	Best window: 12-month 90, 6-month 90, abrupt 180, recurring all history, stationary all history (days)	No later test located. The verdict corpus contains no later comparison of training-window lengths on real fraud data.
C4 Fraud memory	Qualified	vs 30-day window +4.1; vs 90-day window −1.0 points (12-month regime)	No later test located. No later test of fraud propagation against a longer window was located.
C5 Rebalancing	Held	EasyEnsemble +5.8; SMOTE +2.5 points (12-month regime)	No later test located. Within the window, Dal Pozzolo et al. (2015a) show that undersampling biases posterior probabilities but not ranking. No later head-to-head test on a fraud stream was located.
C6 Algorithm	Qualified	Forest vs logistic +28.7; forest vs network −2.9 points (12-month regime)	Contested. Jurgovsky et al. (2018) report a sequence model improving on a random forest baseline for card-present transactions; Vela et al. (2022) find that the same data can age very differently under different model types.
C7 Feedback	Held	Aggregate vs pooled +1.7 points of alert precision (12-month regime)	Supported, same group. Dal Pozzolo et al. (2018) repeat the design on more than 75 million transactions: alert precision 0.77 for the aggregate against 0.61 pooled and 0.57 delayed-only on their 2014 to 2015 data. An independent replication was not located.
C8 Metrics	Held	Twelve-month loss: detection −67%, AUC	Supported. In Dal Pozzolo et al. (2018) the classifier with the best alert precision does not have the best AUC, and the authors argue for alert-based measures.

		−15% (12-month regime)	
C9 Error charts	Qualified	Median lag after abrupt change: 2, 9, 32, 62 days at label delays 0, 7, 30, 60	Qualified by later work. Gemaque et al. (2020) and Grzenda et al. (2020) note that most detectors and evaluations assume labels arrive at once; Dal Pozzolo et al. (2018) describe disputed card transactions being labelled after delays that vary substantially.
C10 Input monitoring	Overturned	Alarm by month 6: 0% of runs (abrupt), 80% (covariate only)	Contested. Vela et al. (2022) conclude that data drift alone does not explain or predict degradation; Sethi and Kantardzic (2017) describe unsupervised change detection as unreliable because it ignores the classifier and raises many false alarms; Rabanser et al. (2019) treat whether a detected shift is harmful as a question separate from detecting it; Lucas et al. (2019) find that measured shift in card data follows the calendar.
C11 Ensembles	Overturned	Ensemble vs single model, same data: −4.3 points (12-month regime)	Partly supported. Ensembles remain central in stream mining (Gomes et al., 2018). In Dal Pozzolo et al. (2018) the ensemble and sliding-window versions of the same system reach the same alert precision (0.77).
C12 Evaluation	Qualified	Cross-validation minus month 1: +11.1 points (6-month), −1.0 (control)	Supported. Kapoor and Narayanan (2023) document data leakage as a source of overoptimistic results in 17 fields; Grzenda et al. (2020) show that ignoring label latency changes evaluation outcomes.

Notes: benchmark verdicts follow the decision rules in the Appendix; differences are in percentage points of detection rate at a 1% alert budget unless stated. The benchmark is synthetic. Source: authors' own work.

5. Discussion

5.1 What the era got right

The era's central claim survives intact. A fraud model that is not updated decays, and it decays because fraud changes, not because the model wears out: in the stationary control, nothing happened to it in a year. The era was also right about measurement. Alert-based measures registered the decay, and AUC largely did not, which is the argument Hand and colleagues made in 2008, and Dal Pozzolo and colleagues made again in 2014 and 2015. And it was right that handling class imbalance matters in a drifting stream, although the benchmark suggests the operative principle is simpler than the era's preferred techniques: keep more of the majority class.

5.2 Where the era's claims need conditions

Most of the qualifications have a common root. The era's maintenance findings came from one or two proprietary streams, each of which had whatever kind and speed of drift it happened to have, and they were stated without that condition attached. The benchmark shows how much the condition matters. Ninety days of history was the best window when patterns turned over gradually, and all history was best when they recurred. Carrying old frauds forward helped when patterns returned and was otherwise a substitute for a longer window. Updating sooner was better, but the gain from the last step, daily over weekly, was small and could not be distinguished from zero once labels were a week late, a constraint the era's cadence experiment had assumed away, and its own later work then formalized. None of this contradicts what the era observed. It means that the era's recommendations were conditional findings that were passed on as general ones.

Two claims did not survive the benchmark. Batch ensembles lost to a single model given the same data, and monitoring the input distribution was blind to changes in fraud. The second matters more for practice, because input stability reports are the cheapest monitoring a risk function can run and are often what an annual review looks at first. On the benchmark, such a report stayed quiet through the loss of nine-tenths of detection and raised its alarms in the one regime where little was wrong. The later literature points the same way (Sethi and Kantardzic, 2017; Vela et al., 2022). Input monitoring remains a reasonable check on data pipelines and on the genuine population. It is not evidence about whether a fraud model still detects fraud.

5.3 Implications for fraud and model risk teams

Four practical points follow, each conditional on the benchmark's assumptions. First, a team that wants to know how fast its own model will decay should measure pattern turnover, not borrow a number. The quantity that tracked decay here was the share of current fraud that comes from patterns first seen after the model's training data ended. Case management systems that record a typology or scheme label for confirmed fraud can produce that share directly, and it can be tracked monthly. Second, the retraining decision was worth more than the choice between competent learners: 25.2 points of detection rate against 2.9 in the central regime. Effort spent on model selection at launch, with no retraining plan behind it, is spent in the wrong place. Third, label latency deserves the attention usually given to retraining frequency. On the benchmark, removing a 30-day delay between a fraud and its training label improved detection by more than doubling the retraining frequency did. Fourth, review cycles should be set against turnover. Supervisory guidance from the period asks for review at least annually and more frequently if warranted (Board of Governors of the Federal Reserve System and OCC, 2011). Under a 12-month pattern lifetime the never-updated benchmark model had lost 67% of its detection rate by its first anniversary, and one tenth within 2 months. For fraud models the second half of that sentence in the guidance is the one that applies.

For the error-rate chart the conclusion is mixed. It is a sound alarm and a poor scheduler. It caught an abrupt change as fast as labels allowed and saved two thirds of the rebuilds, but under gradual turnover a calendar did better. A reasonable reading is that a fixed retraining calendar should be the default and the chart should be the exception handler that brings a rebuild forward.

5.4 Limitations

The benchmark leg is synthetic from end to end, and its results are properties of the generator. Three design choices bear on the headline numbers. Replacement fraud patterns displace a fresh random set of features, which makes them close to invisible to a model trained before they appeared (about one in twenty caught). Real schemes share more with their predecessors, so the decay figures here are better read as an upper range for a given turnover speed than as forecasts. Only six patterns are active at once, which makes individual streams lumpy and widens the intervals. And the mean pattern lifetime, the parameter everything turns on, is not known for any real portfolio; the paper's claim is that decay depends on it, not that any particular value is typical. The scale is small (2,000 transactions a day), forests have 50 trees, transaction amounts and costs are not modelled, and fraudsters do not react to the model, so adversarial adaptation is absent.

The statistical evidence is limited to between five and 20 streams per comparison, without correction for multiple comparisons, and the decision rules, though fixed before the experiments, were not registered with a third party. One rule proved blunt: the cross-validation claim was marked qualified because a single gradual regime had an interval that included zero. The literature leg is a targeted reading, not a systematic review; it cites nothing published after 2023, and many verdict sources were read at the abstract level only; the corpus file records which. The abstract page of one era source could not be retrieved, and that source is cited for its existence only. Finally, the era's maintenance claims come largely from one research group working with one industrial partner. The audit can say whether those claims follow from the group's stated assumptions. It cannot say whether a different partner's data would have produced different claims.

6. Conclusion

The 2008 to 2015 literature on fraud model maintenance was right that an untended model decays, right about how to measure it, and less general than it sounded about what to do. On a benchmark built from the era's own assumptions, a never-updated model lost between 24% and 58% of its detection rate in six months depending on how long fraud patterns lasted, and nothing at all when they did not change. The six-month rule of thumb is therefore neither wrong nor informative: it names a point on a curve whose slope each institution has to measure for itself. The measurements that would settle the open claims on training windows, fraud memory and retraining frequency under realistic label delays need real transaction streams with typology labels, and they remain to be made.

7. Declarations

Retrospective status and date. This study was designed and written in 2026 as a retrospective; it makes no claim to have been produced in the period it studies. The manuscript was written on 10 October 2026.

Synthetic data statement. All data analyzed in this study are synthetic. They were produced by the seeded generator shipped with the paper (generator.py). The data are styled on the scale, class imbalance, and measurement practice described in the 2008 to 2015 literature; they do not originate from that period, and they contain no real transactions, cardholders, merchants or institutions. Every number reported in Section 4 was computed by the shipped code and is read into the manuscript from its output file.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. The work was conducted independently and self-funded by the author.

Competing Interests

The author declares that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Ethics Approval

This study used published aggregate statistics only; no human participants, personal data, or animals were involved, and ethics approval was not required.

Consent to Participate

Not applicable.

Consent to Publish

Not applicable.

Data Availability

All data are transcribed from the publicly available FTC Consumer Sentinel Network Data Books 2020–2024; the transcribed panel (payment_method_panel.csv) is included in the reproducibility package accompanying this article.

Code Availability

The analysis code (run_analysis.py, build_manuscript.py) and results file are included in the reproducibility package; all results regenerate deterministically from the transcribed table.

Author Contributions

All Authors contributed and participated equally.

Declaration of Generative AI and AI-Assisted Technologies

During the preparation of this work, the authors used ChatGPT to check grammar and improve image resolution. All citations were verified against Crossref or the publisher's record; all reported numbers are produced by the accompanying code, which the authors can re-execute. The authors reviewed and edited the content and take full responsibility for the content of the published article. This article was written in September 2023.

Appendix

A1. Era corpus (2008 to 2015)

Every source in the era corpus is listed below by theme. [CR] marks a record verified against Crossref; [WS] a web-verified source without a DOI record.

Table A1. Era corpus by theme

Theme	Sources
Card and payment fraud detectors and studies	Bhattacharyya et al., 2011 [CR]; Carminati et al., 2015 [CR]; Correa Bahnsen et al., 2013 [CR]; Correa Bahnsen et al., 2015 [CR]; Dal Pozzolo et al., 2014a [CR]; Dal Pozzolo et al., 2014b [CR]; Dal Pozzolo et al., 2015a [CR]; Dal Pozzolo et al., 2015b [CR]; Dal Pozzolo et al., 2015c [CR]; Dal Pozzolo, 2015 [WS]; Duman and Ozcelik, 2011 [CR]; Edge and Falcone Sampaio, 2009 [CR]; Gadi et al., 2008 [CR]; Hand et al., 2008 [CR]; Jha et al., 2012 [CR]; Juszczak et al., 2008 [CR]; Krivko, 2010 [CR]; Mahmoudi and Duman, 2015 [CR]; Panigrahi et al., 2009 [CR]; Quah and Sriganesh, 2008 [CR]; Sahin et al., 2013 [CR]; Seeja and Zareapoor, 2014 [CR]; Soltani Halvaiee and Akbari, 2014 [CR]; Srivastava et al., 2008 [CR]; Sánchez et al., 2009 [CR]; Van Vlasselaer et al., 2015 [CR]; Wei et al., 2013 [CR]; Weston et al., 2008 [CR]; Whitrow et al., 2009 [CR]; Wong et al., 2012 [CR]; Zareapoor and Shamsolmoali, 2015 [CR]
Fraud surveys and books	Baesens et al., 2015 [CR]; Ngai et al., 2011 [CR]; Phua et al., 2010 [WS][no-DOI]
Concept drift and stream learning	Alippi et al., 2013 [CR]; Bifet et al., 2009 [CR]; Bifet et al., 2010 [CR]; Brzezinski and Stefanowski, 2014 [CR]; Chen and He, 2009 [CR]; Ditzler and Polikar, 2013 [CR]; Ditzler et al., 2015 [CR]; Dries and Rückert, 2009 [CR]; Elwell and Polikar, 2011 [CR]; Gama et al., 2014 [CR]; Hoens et al., 2012 [CR]; Krempl et al., 2014 [CR]; Kuncheva and Žliobaitė, 2009 [CR]; Masud et al., 2011 [CR]; Minku and Yao, 2012 [CR]; Minku et al., 2010 [CR]; Moreno-Torres et al., 2012 [CR]; Quiñonero-Candela et al., 2008 [CR]; Ross et al., 2012 [CR]; Wang et al., 2015 [CR]; Zliobaite et al., 2014 [CR]; Žliobaitė, 2010 [WS][no-DOI]; Žliobaitė, 2011 [CR]
Evaluation of evolving models	Bifet et al., 2013 [CR]; Gama et al., 2009 [CR]; Gama et al., 2013 [CR]; Žliobaitė et al., 2015 [CR]
Credit scoring: drift, selection, benchmarking	Adams et al., 2010 [CR]; Hand and Adams, 2014 [CR]; Lessmann et al., 2015 [CR]; Pavlidis et al., 2012 [CR]
Supervision and engineering practice	Board of Governors of the Federal Reserve System and OCC, 2011 [WS]; Sculley et al., 2015 [WS][no-DOI]

Source: authors' own work.

A2. Decision rules, as fixed before the experiments

Table A2. Pre-specified decision rules

Claim	Rule

C1	HELD if static RF month-12 det < month-1 det (CI excludes 0) in every gradual regime and AB, and S0 shows no decline (CI includes 0 or |relative change| < 5%). QUALIFIED if only some drifting regimes. Else OVERTURNED.
C2	HELD if det never decreases as cadence shortens (never,168,84,28,14,7,1 days) and daily beats weekly with CI>0 in the drifting regimes at delay 7. QUALIFIED if updating beats static but gains vanish below some cadence or depend on label delay. OVERTURNED if updating does not beat static in drifting regimes.
C3	HELD if the 7-day window is worst and 90-day >= 30-day in all drifting regimes tested. QUALIFIED if the best window differs across regimes. OVERTURNED if the shortest window is best in most regimes.
C4	HELD if propagate-and-forget (30-day genuine window + past frauds) beats both the 30-day and the 90-day sliding window (CI>0) in drifting regimes. QUALIFIED if it beats the 30-day window only. OVERTURNED if it does not beat the 30-day window.
C5	HELD if EasyEnsemble and SMOTE both beat plain 1:1 undersampling on det (CI>0) in all regimes tested. QUALIFIED if only one of them, or only some regimes. OVERTURNED if neither in any regime.
C6	HELD if RF beats both LR and MLP (CI>0) at launch (static, month 1) and under monthly retraining, in all regimes tested. QUALIFIED if it beats only one comparator or only in some regimes. OVERTURNED if it beats neither.
C7	HELD if the aggregated feedback+delayed system (AW) has higher alert precision than the pooled classifier (W) and the delayed-only classifier (WD), CI>0, in all regimes tested. QUALIFIED if some. OVERTURNED if none.
C8	HELD if, for the static RF, the relative loss in AUC from month 1 to month 12 is less than half the relative loss in det in every drifting regime with a det loss of at least 10%. QUALIFIED if in some. OVERTURNED if AUC loss tracks det loss (>= half) everywhere.
C9	HELD if the EWMA error-rate chart flags the abrupt change within a median of 14 days at every label delay tested (0,7,30,60). QUALIFIED if it does so at delay 0 but the lag grows with delay beyond 14 days. OVERTURNED if the median lag exceeds 60 days at delay 0.
C10	HELD if a PSI alarm (score PSI or any feature PSI > 0.25) is raised by month 6 in >= 80% of runs of regimes whose mean static det loss at month 6 is >= 10%, and in <= 20% of runs of regimes without such loss. QUALIFIED if sensitivity or specificity fails for some regimes only. OVERTURNED if sensitivity < 20% across loss regimes.
C11	HELD if the 5-member batch ensemble beats the single model trained on the same 140 days (CI>0) in all drifting regimes tested. QUALIFIED if in some. OVERTURNED if in none.
C12	HELD if the random 5-fold CV estimate exceeds realised month-1 det (CI>0) in every drifting regime but not in S0. QUALIFIED if in some drifting regimes only, or also in S0. OVERTURNED if in none.

Notes: det = detection rate at a 1% daily alert budget; CI = 95% t-interval on paired per-stream differences; W = pooled classifier, WD = delayed-label classifier, AW = aggregate. Regime codes: S0 stationary; G24, G12, G6, G3 gradual with that mean pattern lifetime in months; AB abrupt. Reproduced verbatim from verdict_rules.json. Source: authors' own work.

REFERENCES

1. Abdallah, A., Maarof, M.A. and Zainal, A. (2016), "Fraud detection system: A survey", Journal of Network and Computer Applications, Vol. 68, pp. 90-113, doi: 10.1016/j.jnca.2016.04.007.
2. Adams, N.M., Tasoulis, D.K., Anagnostopoulos, C. and Hand, D.J. (2010), "Temporally-Adaptive Linear Classification for Handling Population Drift in Credit Scoring", Proceedings of COMPSTAT'2010, pp. 167-176, doi: 10.1007/978-3-7908-2604-3_15.

3. Alippi, C., Boracchi, G. and Roveri, M. (2013), "Just-In-Time Classifiers for Recurrent Concepts", IEEE Transactions on Neural Networks and Learning Systems, Vol. 24 No. 4, pp. 620-634, doi: 10.1109/tnnls.2013.2239309.
4. Alvesson, M. and Sandberg, J. (2011), "Generating research questions through problematization", Academy of Management Review, Vol. 36 No. 2, pp. 247-271, doi: 10.5465/amr.2011.59330882.
5. Baesens, B., Van Vlasselaer, V. and Verbeke, W. (2015), Fraud Analytics Using Descriptive, Predictive, and Social Network Techniques, Wiley, doi: 10.1002/9781119146841.
6. Bayram, F., Ahmed, B.S. and Kassler, A. (2022), "From concept drift to model degradation: An overview on performance-aware drift detectors", Knowledge-Based Systems, Vol. 245, pp. 108632, doi: 10.1016/j.knosys.2022.108632.
7. Bhattacharyya, S., Jha, S., Tharakunnel, K. and Westland, J.C. (2011), "Data mining for credit card fraud: A comparative study", Decision Support Systems, Vol. 50 No. 3, pp. 602-613, doi: 10.1016/j.dss.2010.08.008.
8. Bifet, A. and Gavaldà, R. (2007), "Learning from Time-Changing Data with Adaptive Windowing", Proceedings of the 2007 SIAM International Conference on Data Mining, pp. 443-448, doi: 10.1137/1.9781611972771.42.
9. Bifet, A., Holmes, G. and Pfahringer, B. (2010), "Leveraging Bagging for Evolving Data Streams", Lecture Notes in Computer Science, pp. 135-150, doi: 10.1007/978-3-642-15880-3_15.
10. Bifet, A., Holmes, G., Pfahringer, B., Kirkby, R. and Gavaldà, R. (2009), "New ensemble methods for evolving data streams", Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 139-148, doi: 10.1145/1557019.1557041.
11. Bifet, A., Read, J., Žliobaitė, I., Pfahringer, B. and Holmes, G. (2013), "Pitfalls in Benchmarking Data Stream Classification and How to Avoid Them", Lecture Notes in Computer Science, pp. 465-479, doi: 10.1007/978-3-642-40988-2_30.
12. Board of Governors of the Federal Reserve System and Office of the Comptroller of the Currency (2011), "Supervisory guidance on model risk management", SR Letter 11-7, 4 April, available at: https://www.federalreserve.gov/supervisionreg/srletters/sr1107.htm (accessed 10 October 2026).
13. Bolton, R.J. and Hand, D.J. (2002), "Statistical Fraud Detection: A Review", Statistical Science, Vol. 17 No. 3, doi: 10.1214/ss/1042727940.
14. Breck, E., Cai, S., Nielsen, E., Salib, M. and Sculley, D. (2017), "The ML test score: A rubric for ML production readiness and technical debt reduction", 2017 IEEE International Conference on Big Data (Big Data), pp. 1123-1132, doi: 10.1109/bigdata.2017.8258038.
15. Brettle, C. and Steele, R. (2021), "Do Predictive Models Always Deteriorate in Performance with Time? A Case Study in Elective Mortality Predictive Model Performance", 2021 International Conference on Electrical, Computer and Energy Technologies (ICECET), pp. 1-6, doi: 10.1109/icecet52533.2021.9698616.
16. Brzezinski, D. and Stefanowski, J. (2014), "Reacting to Different Types of Concept Drift: The Accuracy Updated Ensemble Algorithm", IEEE Transactions on Neural Networks and Learning Systems, Vol. 25 No. 1, pp. 81-94, doi: 10.1109/tnnls.2013.2251352.
17. Carcillo, F., Dal Pozzolo, A., Le Borgne, Y.A., Caelen, O., Mazzer, Y. and Bontempi, G. (2018), "SCARFF : A scalable framework for streaming credit card fraud detection with spark", Information Fusion, Vol. 41, pp. 182-194, doi: 10.1016/j.inffus.2017.09.005.
18. Carcillo, F., Le Borgne, Y.A., Caelen, O., Kessaci, Y., Oblé, F. and Bontempi, G. (2021), "Combining unsupervised and supervised learning in credit card fraud detection", Information Sciences, Vol. 557, pp. 317-331, doi: 10.1016/j.ins.2019.05.042.
19. Carminati, M., Caron, R., Maggi, F., Epifani, I. and Zanero, S. (2015), "BankSealer: A decision support system for online banking fraud analysis and investigation", Computers & Security, Vol. 53, pp. 175-186, doi: 10.1016/j.cose.2015.04.002.
20. Chen, S. and He, H. (2009), "SERA: Selectively recursive approach towards nonstationary imbalanced stream data mining", 2009 International Joint Conference on Neural Networks, pp. 522-529, doi: 10.1109/ijcnn.2009.5178874.
21. Correa Bahnsen, A., Aouada, D. and Ottersten, B. (2015), "Example-dependent cost-sensitive decision trees", Expert Systems with Applications, Vol. 42 No. 19, pp. 6609-6619, doi: 10.1016/j.eswa.2015.04.042.
22. Correa Bahnsen, A., Aouada, D., Stojanovic, A. and Ottersten, B. (2016), "Feature engineering strategies for credit card fraud detection", Expert Systems with Applications, Vol. 51, pp. 134-142, doi: 10.1016/j.eswa.2015.12.030.
23. Correa Bahnsen, A., Stojanovic, A., Aouada, D. and Ottersten, B. (2013), "Cost Sensitive Credit Card Fraud Detection Using Bayes Minimum Risk", 2013 12th International Conference on Machine Learning and Applications, pp. 333-338, doi: 10.1109/icmla.2013.68.
24. Dal Pozzolo, A. (2015), Adaptive Machine Learning for Credit Card Fraud Detection, PhD thesis, Université Libre de Bruxelles, Brussels, available at: https://dalpozz.github.io/static/pdf/Dalpozzolo2015PhD.pdf (accessed 10 October 2026).
25. Dal Pozzolo, A., Boracchi, G., Caelen, O., Alippi, C. and Bontempi, G. (2015b), "Credit card fraud detection and concept-drift adaptation with delayed supervised information", 2015 International Joint Conference on Neural Networks (IJCNN), pp. 1-8, doi: 10.1109/ijcnn.2015.7280527.
26. Dal Pozzolo, A., Boracchi, G., Caelen, O., Alippi, C. and Bontempi, G. (2018), "Credit Card Fraud Detection: A Realistic Modeling and a Novel Learning Strategy", IEEE Transactions on Neural Networks and Learning Systems, Vol. 29 No. 8, pp. 3784-3797, doi: 10.1109/tnnls.2017.2736643.

27. Dal Pozzolo, A., Caelen, O. and Bontempi, G. (2015c), "When is Undersampling Effective in Unbalanced Classification Tasks?", Lecture Notes in Computer Science, pp. 200-215, doi: 10.1007/978-3-319-23528-8_13.
28. Dal Pozzolo, A., Caelen, O., Johnson, R.A. and Bontempi, G. (2015a), "Calibrating Probability with Undersampling for Unbalanced Classification", 2015 IEEE Symposium Series on Computational Intelligence, pp. 159-166, doi: 10.1109/ssci.2015.33.
29. Dal Pozzolo, A., Caelen, O., Le Borgne, Y.A., Waterschoot, S. and Bontempi, G. (2014a), "Learned lessons in credit card fraud detection from a practitioner perspective", Expert Systems with Applications, Vol. 41 No. 10, pp. 4915-4928, doi: 10.1016/j.eswa.2014.02.026.
30. Dal Pozzolo, A., Johnson, R., Caelen, O., Waterschoot, S., Chawla, N.V. and Bontempi, G. (2014b), "Using HDDT to avoid instances propagation in unbalanced and evolving data streams", 2014 International Joint Conference on Neural Networks (IJCNN), pp. 588-594, doi: 10.1109/ijcnn.2014.6889638.
31. Ditzler, G. and Polikar, R. (2013), "Incremental Learning of Concept Drift from Streaming Imbalanced Data", IEEE Transactions on Knowledge and Data Engineering, Vol. 25 No. 10, pp. 2283-2301, doi: 10.1109/tkde.2012.136.
32. Ditzler, G., Roveri, M., Alippi, C. and Polikar, R. (2015), "Learning in Nonstationary Environments: A Survey", IEEE Computational Intelligence Magazine, Vol. 10 No. 4, pp. 12-25, doi: 10.1109/mci.2015.2471196.
33. Dries, A. and Rückert, U. (2009), "Adaptive concept drift detection", Statistical Analysis and Data Mining: The ASA Data Science Journal, Vol. 2 No. 5-6, pp. 311-327, doi: 10.1002/sam.10054.
34. Duman, E. and Ozcelik, M.H. (2011), "Detecting credit card fraud by genetic algorithm and scatter search", Expert Systems with Applications, Vol. 38 No. 10, pp. 13057-13063, doi: 10.1016/j.eswa.2011.04.110.
35. Edge, M.E. and Falcone Sampaio, P.R. (2009), "A survey of signature based methods for financial fraud detection", Computers & Security, Vol. 28 No. 6, pp. 381-394, doi: 10.1016/j.cose.2009.02.001.
36. Elwell, R. and Polikar, R. (2011), "Incremental Learning of Concept Drift in Nonstationary Environments", IEEE Transactions on Neural Networks, Vol. 22 No. 10, pp. 1517-1531, doi: 10.1109/tnn.2011.2160459.
37. Ferrari Dacrema, M., Cremonesi, P. and Jannach, D. (2019), "Are we really making much progress? A worrying analysis of recent neural recommendation approaches", Proceedings of the 13th ACM Conference on Recommender Systems, pp. 101-109, doi: 10.1145/3298689.3347058.
38. Gadi, M.F.A., Wang, X. and do Lago, A.P. (2008), "Credit Card Fraud Detection with Artificial Immune System", Lecture Notes in Computer Science, pp. 119-131, doi: 10.1007/978-3-540-85072-4_11.
39. Galen, C. and Steele, R. (2020), "Performance Maintenance Over Time of Random Forest-based Malware Detection Models", 2020 11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), pp. 0536-0541, doi: 10.1109/uemcon51285.2020.9298068.
40. Gama, J., Medas, P., Castillo, G. and Rodrigues, P. (2004), "Learning with Drift Detection", Lecture Notes in Computer Science, pp. 286-295, doi: 10.1007/978-3-540-28645-5_29.
41. Gama, J., Sebastião, R. and Rodrigues, P.P. (2009), "Issues in evaluation of stream learning algorithms", Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 329-338, doi: 10.1145/1557019.1557060.
42. Gama, J., Sebastião, R. and Rodrigues, P.P. (2013), "On evaluating stream learning algorithms", Machine Learning, Vol. 90 No. 3, pp. 317-346, doi: 10.1007/s10994-012-5320-9.
43. Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M. and Bouchachia, A. (2014), "A survey on concept drift adaptation", ACM Computing Surveys, Vol. 46 No. 4, pp. 1-37, doi: 10.1145/2523813.
44. Gemaque, R.N., Costa, A.F.J., Giusti, R. and dos Santos, E.M. (2020), "An overview of unsupervised drift detection methods", WIREs Data Mining and Knowledge Discovery, Vol. 10 No. 6, art. e1381, doi: 10.1002/widm.1381.
45. Gomes, H.M., Barddal, J.P., Enembreck, F. and Bifet, A. (2018), "A Survey on Ensemble Learning for Data Stream Classification", ACM Computing Surveys, Vol. 50 No. 2, pp. 1-36, doi: 10.1145/3054925.
46. Gomes, H.M., Bifet, A., Read, J., Barddal, J.P., Enembreck, F., Pfharinger, B., Holmes, G. and Abdessalem, T. (2017), "Adaptive random forests for evolving data stream classification", Machine Learning, Vol. 106 No. 9-10, pp. 1469-1495, doi: 10.1007/s10994-017-5642-8.
47. Grzenda, M., Gomes, H.M. and Bifet, A. (2020), "Delayed labelling evaluation for data streams", Data Mining and Knowledge Discovery, Vol. 34 No. 5, pp. 1237-1266, doi: 10.1007/s10618-019-00654-y.
48. Hand, D.J. (2006), "Classifier Technology and the Illusion of Progress", Statistical Science, Vol. 21 No. 1, doi: 10.1214/088342306000000060.
49. Hand, D.J. and Adams, N.M. (2014), "Selection bias in credit scorecard evaluation", Journal of the Operational Research Society, Vol. 65 No. 3, pp. 408-415, doi: 10.1057/jors.2013.55.
50. Hand, D.J., Whitrow, C., Adams, N.M., Juszczak, P. and Weston, D. (2008), "Performance criteria for plastic card fraud detection tools", Journal of the Operational Research Society, Vol. 59 No. 7, pp. 956-962, doi: 10.1057/palgrave.jors.2602418.
51. Hilal, W., Gadsden, S.A. and Yawney, J. (2022), "Financial Fraud: A Review of Anomaly Detection Techniques and Recent Advances", Expert Systems with Applications, Vol. 193, pp. 116429, doi: 10.1016/j.eswa.2021.116429.
52. Hoens, T.R., Polikar, R. and Chawla, N.V. (2012), "Learning from streaming data with concept drift and imbalance: an overview", Progress in Artificial Intelligence, Vol. 1 No. 1, pp. 89-101, doi: 10.1007/s13748-011-0008-0.
53. Ioannidis, J.P.A. (2005), "Why Most Published Research Findings Are False", PLoS Medicine, Vol. 2 No. 8, pp. e124, doi: 10.1371/journal.pmed.0020124.

54. Jha, S., Guillen, M. and Westland, J.C. (2012), "Employing transaction aggregation strategy to detect credit card fraud", Expert Systems with Applications, Vol. 39 No. 16, pp. 12650-12657, doi: 10.1016/j.eswa.2012.05.018.
55. Jurgovsky, J., Granitzer, M., Ziegler, K., Calabretto, S., Portier, P.E., He-Guelton, L. and Caelen, O. (2018), "Sequence classification for credit-card fraud detection", Expert Systems with Applications, Vol. 100, pp. 234-245, doi: 10.1016/j.eswa.2018.01.037.
56. Juszczak, P., Adams, N.M., Hand, D.J., Whitrow, C. and Weston, D.J. (2008), "Off-the-peg and bespoke classifiers for fraud detection", Computational Statistics & Data Analysis, Vol. 52 No. 9, pp. 4521-4532, doi: 10.1016/j.csda.2008.03.014.
57. Kapoor, S. and Narayanan, A. (2023), "Leakage and the reproducibility crisis in machine-learning-based science", Patterns, Vol. 4 No. 9, pp. 100804, doi: 10.1016/j.patter.2023.100804.
58. Kelly, M.G., Hand, D.J. and Adams, N.M. (1999), "The impact of changing populations on classifier performance", Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 367-371, doi: 10.1145/312129.312285.
59. Kidane, L., Townend, P., Metsch, T. and Elmroth, E. (2022), "When and How to Retrain Machine Learning-based Cloud Management Systems", 2022 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), pp. 688-698, doi: 10.1109/ipdpsw55747.2022.00120.
60. Krempl, G., Žliobaite, I., Brzeziński, D., Hüllermeier, E., Last, M., Lemaire, V., Noack, T., Shaker, A., Sievi, S., Spiliopoulou, M. and Stefanowski, J. (2014), "Open challenges for data stream mining research", ACM SIGKDD Explorations Newsletter, Vol. 16 No. 1, pp. 1-10, doi: 10.1145/2674026.2674028.
61. Krivko, M. (2010), "A hybrid model for plastic card fraud detection systems", Expert Systems with Applications, Vol. 37 No. 8, pp. 6070-6076, doi: 10.1016/j.eswa.2010.02.119.
62. Kuncheva, L.I. and Žliobaitė, I. (2009), "On the window size for classification in changing environments", Intelligent Data Analysis, Vol. 13 No. 6, pp. 861-872, doi: 10.3233/ida-2009-0397.
63. Lebichot, B., Le Borgne, Y.A., He-Guelton, L., Oblé, F. and Bontempi, G. (2020), "Deep-Learning Domain Adaptation Techniques for Credit Cards Fraud Detection", Proceedings of the International Neural Networks Society, pp. 78-88, doi: 10.1007/978-3-030-16841-4_8.
64. Lebichot, B., Paldino, G.M., Siblini, W., He-Guelton, L., Oblé, F. and Bontempi, G. (2021), "Incremental learning strategies for credit cards fraud detection", International Journal of Data Science and Analytics, Vol. 12 No. 2, pp. 165-174, doi: 10.1007/s41060-021-00258-0.
65. Lessmann, S., Baesens, B., Seow, H.V. and Thomas, L.C. (2015), "Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research", European Journal of Operational Research, Vol. 247 No. 1, pp. 124-136, doi: 10.1016/j.ejor.2015.05.030.
66. Lipton, Z.C. and Steinhardt, J. (2019), "Troubling Trends in Machine Learning Scholarship", Queue, Vol. 17 No. 1, pp. 45-77, doi: 10.1145/3317287.3328534.
67. Lu, J., Liu, A., Dong, F., Gu, F., Gama, J. and Zhang, G. (2018), "Learning under Concept Drift: A Review", IEEE Transactions on Knowledge and Data Engineering, doi: 10.1109/tkde.2018.2876857.
68. Lucas, Y., Portier, P.E., Laporte, L., Calabretto, S., He-Guelton, L., Oble, F. and Granitzer, M. (2019), "Dataset Shift Quantification for Credit Card Fraud Detection", 2019 IEEE Second International Conference on Artificial Intelligence and Knowledge Engineering (AIKE), pp. 97-100, doi: 10.1109/aike.2019.00024.
69. Mahmoudi, N. and Duman, E. (2015), "Detecting credit card fraud by Modified Fisher Discriminant Analysis", Expert Systems with Applications, Vol. 42 No. 5, pp. 2510-2516, doi: 10.1016/j.eswa.2014.10.037.
70. Masud, M., Gao, J., Khan, L., Han, J. and Thuraisingham, B.M. (2011), "Classification and Novel Class Detection in Concept-Drifting Data Streams under Time Constraints", IEEE Transactions on Knowledge and Data Engineering, Vol. 23 No. 6, pp. 859-874, doi: 10.1109/tkde.2010.61.
71. Megaladata (n.d.), "Model degradation in machine learning", blog post, available at: https://megaladata.com/blog/model-degradation-machine-learning (accessed 10 October 2026).
72. Minku, L.L. and Yao, X. (2012), "DDD: A New Ensemble Approach for Dealing with Concept Drift", IEEE Transactions on Knowledge and Data Engineering, Vol. 24 No. 4, pp. 619-633, doi: 10.1109/tkde.2011.58.
73. Minku, L.L., White, A.P. and Yao, X. (2010), "The Impact of Diversity on Online Ensemble Learning in the Presence of Concept Drift", IEEE Transactions on Knowledge and Data Engineering, Vol. 22 No. 5, pp. 730-742, doi: 10.1109/tkde.2009.156.
74. Moreno-Torres, J.G., Raeder, T., Alaiz-Rodríguez, R., Chawla, N.V. and Herrera, F. (2012), "A unifying view on dataset shift in classification", Pattern Recognition, Vol. 45 No. 1, pp. 521-530, doi: 10.1016/j.patcog.2011.06.019.
75. Ngai, E.W.T., Hu, Y., Wong, Y.H., Chen, Y. and Sun, X. (2011), "The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature", Decision Support Systems, Vol. 50 No. 3, pp. 559-569, doi: 10.1016/j.dss.2010.08.006.
76. Open Science Collaboration (2015), "Estimating the reproducibility of psychological science", Science, Vol. 349 No. 6251, art. aac4716, doi: 10.1126/science.aac4716.
77. Paleyes, A., Urma, R.G. and Lawrence, N.D. (2023), "Challenges in Deploying Machine Learning: A Survey of Case Studies", ACM Computing Surveys, Vol. 55 No. 6, pp. 1-29, doi: 10.1145/3533378.

78. Panigrahi, S., Kundu, A., Sural, S. and Majumdar, A.K. (2009), "Credit card fraud detection: A fusion approach using Dempster–Shafer theory and Bayesian learning", Information Fusion, Vol. 10 No. 4, pp. 354-363, doi: 10.1016/j.inffus.2008.04.001.
79. Pavlidis, N.G., Tasoulis, D.K., Adams, N.M. and Hand, D.J. (2012), "Adaptive consumer credit classification", Journal of the Operational Research Society, Vol. 63 No. 12, pp. 1645-1654, doi: 10.1057/jors.2012.15.
80. Phua, C., Lee, V., Smith, K. and Gayler, R. (2010), "A comprehensive survey of data mining-based fraud detection research", arXiv preprint arXiv:1009.6119, available at: https://arxiv.org/abs/1009.6119 (accessed 10 October 2026).
81. Plasse, J. and Adams, N. (2016), "Handling delayed labels in temporally evolving data streams", 2016 IEEE International Conference on Big Data (Big Data), pp. 2416-2424, doi: 10.1109/bigdata.2016.7840877.
82. Quah, J.T.S. and Sriganesh, M. (2008), "Real-time credit card fraud detection using computational intelligence", Expert Systems with Applications, Vol. 35 No. 4, pp. 1721-1732, doi: 10.1016/j.eswa.2007.08.093.
83. Quiñonero-Candela, J., Sugiyama, M., Schwaighofer, A. and Lawrence, N.D. (Eds) (2008), Dataset Shift in Machine Learning, The MIT Press, doi: 10.7551/mitpress/9780262170055.001.0001.
84. Rabanser, S., Günnemann, S. and Lipton, Z. (2019), "Failing loudly: an empirical study of methods for detecting dataset shift", Advances in Neural Information Processing Systems, Vol. 32, available at: https://papers.nips.cc/paper_files/paper/2019/hash/846c260d715e5b854ffad5f70a516c88-Abstract.html (accessed 10 October 2026).
85. Ross, G.J., Adams, N.M., Tasoulis, D.K. and Hand, D.J. (2012), "Exponentially weighted moving average charts for detecting concept drift", Pattern Recognition Letters, Vol. 33 No. 2, pp. 191-198, doi: 10.1016/j.patrec.2011.08.019.
86. Ryman-Tubb, N.F., Krause, P. and Garn, W. (2018), "How Artificial Intelligence and machine learning research impacts payment card fraud detection: A survey and industry benchmark", Engineering Applications of Artificial Intelligence, Vol. 76, pp. 130-157, doi: 10.1016/j.engappai.2018.07.008.
87. Sahin, Y., Bulkan, S. and Duman, E. (2013), "A cost-sensitive decision tree approach for fraud detection", Expert Systems with Applications, Vol. 40 No. 15, pp. 5916-5923, doi: 10.1016/j.eswa.2013.05.021.
88. Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Crespo, J.-F. and Dennison, D. (2015), "Hidden technical debt in machine learning systems", Advances in Neural Information Processing Systems, Vol. 28, available at: https://papers.nips.cc/paper_files/paper/2015/hash/86df7dcfd896fcaf2674f757a2463eba-Abstract.html (accessed 10 October 2026).
89. Seeja, K.R. and Zareapoor, M. (2014), "FraudMiner: A Novel Credit Card Fraud Detection Model Based on Frequent Itemset Mining", The Scientific World Journal, Vol. 2014, pp. 1-10, doi: 10.1155/2014/252797.
90. Sethi, T.S. and Kantardzic, M. (2017), "On the reliable detection of concept drift from streaming unlabeled data", Expert Systems with Applications, Vol. 82, pp. 77-99, doi: 10.1016/j.eswa.2017.04.008.
91. Sánchez, D., Vila, M.A., Cerda, L. and Serrano, J.M. (2009), "Association rules applied to credit card fraud detection", Expert Systems with Applications, Vol. 36 No. 2, pp. 3630-3640, doi: 10.1016/j.eswa.2008.02.001.
92. Soltani Halvaiee, N. and Akbari, M.K. (2014), "A novel model for credit card fraud detection using Artificial Immune Systems", Applied Soft Computing, Vol. 24, pp. 40-49, doi: 10.1016/j.asoc.2014.06.042.
93. Sousa, M.R., Gama, J. and Brandão, E. (2016), "A new dynamic modeling framework for credit risk assessment", Expert Systems with Applications, Vol. 45, pp. 341-351, doi: 10.1016/j.eswa.2015.09.055.
94. Souza, V.M.A., dos Reis, D.M., Maletzke, A.G. and Batista, G.E.A.P.A. (2020), "Challenges in benchmarking stream learning algorithms with real-world data", Data Mining and Knowledge Discovery, Vol. 34 No. 6, pp. 1805-1858, doi: 10.1007/s10618-020-00698-5.
95. Srivastava, A., Kundu, A., Sural, S. and Majumdar, A.K. (2008), "Credit Card Fraud Detection Using Hidden Markov Model", IEEE Transactions on Dependable and Secure Computing, Vol. 5 No. 1, pp. 37-48, doi: 10.1109/tdsc.2007.70228.
96. Taplin, R. and Hunt, C. (2019), "The Population Accuracy Index: A New Measure of Population Stability for Model Monitoring", Risks, Vol. 7 No. 2, pp. 53, doi: 10.3390/risks7020053.
97. Van Vlasselaer, V., Bravo, C., Caelen, O., Eliassi-Rad, T., Akoglu, L., Snoeck, M. and Baesens, B. (2015), "APATE: A novel approach for automated credit card transaction fraud detection using network-based extensions", Decision Support Systems, Vol. 75, pp. 38-48, doi: 10.1016/j.dss.2015.04.013.
98. Vela, D., Sharp, A., Zhang, R., Nguyen, T., Hoang, A. and Pianykh, O.S. (2022), "Temporal quality degradation in AI models", Scientific Reports, Vol. 12 No. 1, art. 11654, doi: 10.1038/s41598-022-15245-z.
99. Wang, S., Minku, L.L. and Yao, X. (2015), "Resampling-Based Ensemble Methods for Online Class Imbalance Learning", IEEE Transactions on Knowledge and Data Engineering, Vol. 27 No. 5, pp. 1356-1368, doi: 10.1109/tkde.2014.2345380.
100. Webb, G.I., Hyde, R., Cao, H., Nguyen, H.L. and Petitjean, F. (2016), "Characterizing concept drift", Data Mining and Knowledge Discovery, Vol. 30 No. 4, pp. 964-994, doi: 10.1007/s10618-015-0448-4.
101. Wei, W., Li, J., Cao, L., Ou, Y. and Chen, J. (2013), "Effective detection of sophisticated online banking fraud on extremely imbalanced data", World Wide Web, Vol. 16 No. 4, pp. 449-475, doi: 10.1007/s11280-012-0178-0.
102. West, J. and Bhattacharya, M. (2016), "Intelligent financial fraud detection: A comprehensive review", Computers & Security, Vol. 57, pp. 47-66, doi: 10.1016/j.cose.2015.09.005.

103. Weston, D.J., Hand, D.J., Adams, N.M., Whitrow, C. and Juszczak, P. (2008), "Plastic card fraud detection using peer group analysis", Advances in Data Analysis and Classification, Vol. 2 No. 1, pp. 45-62, doi: 10.1007/s11634-008-0021-8.
104. Whitrow, C., Hand, D.J., Juszczak, P., Weston, D. and Adams, N.M. (2009), "Transaction aggregation as a strategy for credit card fraud detection", Data Mining and Knowledge Discovery, Vol. 18 No. 1, pp. 30-55, doi: 10.1007/s10618-008-0116-z.
105. Widmer, G. and Kubat, M. (1996), "Learning in the Presence of Concept Drift and Hidden Contexts", Machine Learning, Vol. 23 No. 1, pp. 69-101, doi: 10.1023/a:1018046501280.
106. Wong, N., Ray, P., Stephens, G. and Lewis, L. (2012), "Artificial immune systems for the detection of credit card fraud: an architecture, prototype and preliminary results", Information Systems Journal, Vol. 22 No. 1, pp. 53-76, doi: 10.1111/j.1365-2575.2011.00369.x.
107. Yurdakul, B. and Naranjo, J. (2020), "Statistical properties of the population stability index", The Journal of Risk Model Validation, doi: 10.21314/jrmv.2020.227.
108. Zareapoor, M. and Shamsolmoali, P. (2015), "Application of Credit Card Fraud Detection: Based on Bagging Ensemble Classifier", Procedia Computer Science, Vol. 48, pp. 679-685, doi: 10.1016/j.procs.2015.04.201.
109. Žliobaitė, I. (2010), "Learning under concept drift: an overview", arXiv preprint arXiv:1010.4784, available at: https://arxiv.org/abs/1010.4784 (accessed 10 October 2026).
110. Žliobaitė, I. (2011), "Combining similarity in time and space for training set formation under concept drift", Intelligent Data Analysis, Vol. 15 No. 4, pp. 589-611, doi: 10.3233/ida-2011-0484.
111. Zliobaite, I., Bifet, A., Pfahringer, B. and Holmes, G. (2014), "Active Learning With Drifting Streaming Data", IEEE Transactions on Neural Networks and Learning Systems, Vol. 25 No. 1, pp. 27-39, doi: 10.1109/tnnls.2012.2236570.
112. Žliobaitė, I., Bifet, A., Read, J., Pfahringer, B. and Holmes, G. (2015), "Evaluation methods and decision theory for classification of streaming data with temporal dependence", Machine Learning, Vol. 98 No. 3, pp. 455-482, doi: 10.1007/s10994-014-5441-4.
113. Žliobaitė, I., Pechenizkiy, M. and Gama, J. (2016), "An Overview of Concept Drift Applications", Studies in Big Data, pp. 91-114, doi: 10.1007/978-3-319-26989-4_4.