Natural Resources for Human Health
eISSN: 2583-1194

Original Research

Received 15 July 2026
Revised 18 Aug 2026
Accepted 01 Sep 2026

Natural Resources for Human Health 2026, Vol. 6 (19s): 256–283

KEYWORDS:
intelligent computing; intrusion detection; Internet of Things; federated learning; cloud–edge computing; credit scoring; explainable AI; algorithmic fairness; volatility forecasting; value-at-risk; digital transformation

Artificial Intelligence Driven Intelligent Computing: Integrating Cybersecurity, IoT, Cloud Computing, Financial Analytics, and Credit Risk Assessment for Digital Transformation

[1]Tariq Saeed Mian, [2]Abdelrahman Elsharif Karrar, [3]Rashiq Rafiq Marie, [4]Dr. Muhammad Farrukh, [5]Omar Almomani, [6]Syed Adil Abbas Rizvi

[1]Department of IS, College of CS & Engineering, Taibah University, Madinah Almunwarah, Saudi Arabia, tmian@taibahu.edu.sa
ORCID: 0000-0003-2666-9223
[2]College of Computer Science and Engineering
Taibah University, Medina, Saudi Arabia
ORCID: 0000-0003-1683-7668, akarrar@taibahu.edu.sa
[3]College of Computer Science and Engineering Taibah University, Madinah, Saudi Arabia
rmarie@taibahu.edu.sa
ORCID: 0009-0000-2682-8568
[4]School of Creative Arts,The University of Lahore.
muhammad.farrukh@soca.uol.edu.pk
[5]Department of Networks and Cybersecurity, Hourani Center for Applied Scientific Research, Al-Ahliyya Amman University, Amman, Jordan.
orcid.org/0000-0003-3160-6542 o.almomani@ammanu.edu.jo
[6]Senior Manager Business Process Analyst - IT Projects Mohammed Ali Jinnah University
Email: aabbas.5522@gmail.com

ABSTRACT: **Background**: Digital transformation couples cybersecurity, Internet of Things (IoT) and cloud infrastructure with data-driven financial decision-making, yet artificial intelligence (AI) in these domains is usually evaluated in isolation and under optimistic conditions.

Objective: To test, under one statistically controlled protocol, where AI adds measurable value across security, cloud–edge computing, credit risk and financial analytics.

Methods: We propose a four-layer Intelligent Computing for Digital Transformation (ICDT) framework and instantiate it on five public data sources. Eight intrusion detectors were trained on NSL-KDD and tested on its official shifted split; federated learning (FedAvg, FedProx) was simulated across ten IID or label-skewed clients; seven credit models were compared on two datasets with 30-fold repeated cross-validation; and machine-learning (ML) forecasts of the VIX and S&P 500 value-at-risk were benchmarked against HAR and GARCH-t models. Inference used bootstrap confidence intervals (CIs), McNemar, corrected resampled t, Friedman, Diebold–Mariano and

Natural Resources for Human Health
eISSN: 2583-1194

Kupiec/Christoffersen tests, with calibration, SHAP explanations and age-group fairness.

Results: Intrusion detectors scoring F1 of 0.97–0.999 in cross-validation fell to 0.74–0.83 on the shifted test; novel attacks were detected 38 percentage points (pp) less often than known ones; tree ensembles kept the best ranking (XGBoost AUC 0.964) but a multilayer perceptron gave the best F1 (0.834, 95% CI [0.829, 0.839]). Federated training came within 3.5 pp of centralised F1 under label skew and far exceeded isolated clients. Stacking improved credit AUC over logistic regression by only 0.017 (95% CI [0.005, 0.028]) and 0.018 ([0.009, 0.027]), without reducing expected misclassification cost, and was the worst-calibrated model on one dataset; all models approved creditworthy young applicants 7–14 pp less often. HAR beat random forest and LightGBM for VIX forecasting (Diebold–Mariano $p < 0.001$), and only GARCH-t passed all 5% VaR backtests.

Conclusions: The benefit of AI complexity is domain-dependent: large for stationary high-dimensional classification, small for credit scoring, and absent for persistent financial time series. A shared assurance layer for shift-aware validation, calibration, explainability and fairness is essential for trustworthy digital transformation

1. INTRODUCTION

Digital transformation has moved from the adoption of isolated information technologies to the re-engineering of entire value chains around data, connectivity and automated decision-making [1]. In financial services this transition is particularly consequential: lending, payments, trading and risk management are now delivered through cloud-hosted platforms, fed by streams of device and transaction data, and increasingly governed by machine-learning (ML) models [2,3]. Firms that develop an organisational capability for artificial intelligence (AI) report gains in creativity and performance [4], and AI-enabled finance has been argued to widen access to credit for previously excluded populations [5,6]. The same transformation, however, enlarges the attack surface. Internet of Things (IoT) endpoints, edge gateways and multi-tenant cloud services create new routes for intrusion, and cyber incidents at financial institutions can propagate through payment and liquidity networks [7,8]. Security, computing infrastructure and financial decision-making are therefore no longer separable engineering problems; they are coupled components of one socio-technical system.

Research on these components has nonetheless developed in parallel silos. The intrusion-detection literature has produced a large number of ML and deep-learning detectors, usually evaluated on public network-traffic benchmarks [9–11]. Edge and cloud computing research focuses on where models should run and how they should be trained across distributed devices [12,13], with federated learning (FL) proposed as the privacy-preserving default for IoT security [14,15]. Credit-risk research benchmarks classifiers on lending data and increasingly asks whether complex models are transparent, calibrated and fair [16–19]. Financial-forecasting research, in turn, debates whether flexible ML improves on parsimonious econometric baselines [20,21]. Because each strand uses its own datasets, metrics and validation conventions, it is difficult to answer a question that matters to an organisation pursuing digital transformation: *across the security, computing and financial functions it must modernise, where does AI add measurable value, how large is that value, and under what conditions does it disappear?*

Three methodological weaknesses make the question harder. First, many published evaluations are optimistic because training and test data are drawn from the same distribution, a pitfall that has been documented systematically for security ML [22,23]. Network-intrusion detectors that score near-perfect accuracy under random cross-validation can fail when attack behaviour changes [24,25]. Second, model comparisons are frequently reported as point estimates without the variance estimates, confidence intervals and multiple-comparison controls needed to separate genuine improvements from resampling noise [26,27]. Third, discrimination metrics such as accuracy or the area under the ROC curve (AUC) are often treated as sufficient, although deployed credit and risk models must also be well calibrated, explainable and non-discriminatory, obligations that are now codified in regulation; the European Union *Artificial Intelligence Act* classifies AI systems used to evaluate the

creditworthiness of natural persons as high-risk [28], and the U.S. NIST AI Risk Management Framework sets out governance expectations for trustworthy AI [29].

This study responds with an integrated, reproducible evaluation of AI across the four functions named in the title. We propose an *Intelligent Computing for Digital Transformation* (ICDT) framework that organises the functions into four interacting layers (edge/IoT security analytics; cloud–edge distributed learning; financial intelligence for credit and market risk; and a cross-cutting assurance layer for statistics, calibration, explainability and fairness), and we test it empirically on five public data sources under a common protocol. Rather than proposing yet another architecture tuned to a single benchmark, the study asks whether the *complexity premium* (the gain from moving from a transparent baseline to a more flexible AI model) is consistent across domains once evaluation is made rigorous.

Four pre-specified hypotheses guided the analysis:

- **H1 (cybersecurity/IoT).** Tree-ensemble detectors outperform linear baselines for network intrusion detection, but their performance falls significantly when evaluated on a test distribution containing attack types unseen in training.
- **H2 (cloud–edge computing).** Federated training approaches centralised performance when client data are independent and identically distributed (IID), whereas label-skewed (non-IID) clients slow convergence and reduce accuracy; a proximal correction (FedProx) partly mitigates this.
- **H3 (credit risk).** Ensemble learners improve discrimination over logistic regression, but the improvement is modest and dataset-dependent, and gains in AUC do not automatically translate into better calibration or fairness.
- **H4 (financial analytics).** For volatility forecasting and market-risk measurement, nonlinear ML does not reliably outperform parsimonious econometric benchmarks (HAR for implied volatility; GARCH with Student-*t* innovations for value-at-risk).

The contributions are fourfold. (i) We provide a single, statistically controlled benchmark that spans intrusion detection, federated cloud–edge training, credit scoring and market-risk analytics, using only public data so that every result can be reproduced. (ii) We quantify, with confidence intervals and effect sizes, how much of the apparent performance of AI detectors survives a distribution shift, and how much latency and memory each detector costs at the edge. (iii) We show that in credit scoring the ranking of models depends on whether one looks at discrimination, calibration, expected misclassification cost or group fairness. (iv) We document a domain-dependent "simplicity frontier": in financial time series the parsimonious econometric baselines were not beaten, whereas in high-dimensional tabular classification the ensembles were. Throughout, we separate what the experiments demonstrate from mechanisms that remain hypotheses.

1.1 Related work and research gap

AI for cybersecurity in IoT and cloud environments. Surveys converge on the view that supervised ML, especially tree ensembles and deep networks, achieves high detection rates on benchmark traffic [9,10,30]. IoT-specific reviews stress resource constraints, heterogeneous protocols and the scarcity of realistic datasets [11], motivating new testbeds such as TON_IoT and Edge-IIoTset [31,32]. Critical work has shown that benchmark results often fail to generalise: Arp et al. identified recurring pitfalls including sampling bias and data snooping [22], Sarhan et al. showed that detectors transfer poorly across datasets with different feature sets [25], and Andresini et al. demonstrated performance decay under concept drift [24]. Explainability has become a requirement for security operations centres [33].

Cloud–edge intelligence and federated learning. Edge intelligence moves inference, and sometimes training, close to data sources to reduce latency, bandwidth and privacy exposure [12,13]. FL trains a shared model by averaging locally computed updates [34]; its main obstacles are statistical heterogeneity across clients, communication cost and system heterogeneity [35,36]. FedProx adds a proximal term that limits client drift under heterogeneous data [37]. In IoT intrusion detection, FL has been compared with centralised and on-device learning [14] and reviewed against practical deployment requirements [15]. Operationalising such pipelines in production relies on MLOps practices for versioning, monitoring and retraining [38], and on zero-trust architectures that authenticate every request rather than trusting the network perimeter [39].

AI for credit risk. Large benchmarks found that heterogeneous ensembles generally outperform logistic regression [40], but later work qualifies this. Deep learning offered no consistent advantage over gradient boosting for credit scoring [16]; logistic regression augmented with tree-derived non-linear effects can close much of the gap while preserving interpretability [17]; and

broader evidence on tabular data favours tree ensembles over deep networks [41,42]. Explainability methods such as Shapley-value attributions are now applied to credit models [43,44], and audits of transparency and fairness are argued to be preconditions for deployment [18,19,45]. Evidence from mortgage markets shows that more accurate ML can redistribute credit unevenly across groups [46].

AI for financial analytics. ML has improved the measurement of equity risk premia [20], yet in volatility forecasting the heterogeneous autoregressive (HAR) model [47] remains a demanding benchmark: recent large-scale studies find that a well-specified HAR is hard to beat with tuned ML [21], including for the VIX index [48]. For market risk, ML-based quantile models have been proposed for value-at-risk (VaR) estimation [49], and are formally evaluated with unconditional and conditional coverage tests [50,51].

Table 1 summarises how these strands are typically evaluated and what the present study adds. The gap is not a lack of models but a lack of *comparable, uncertainty-quantified evidence across domains* that a digital-transformation programme must manage jointly.

Table 1. Representative recent studies by domain, their typical evaluation practice, and the gap addressed in this study.

Domain	Representative studies	Typical evaluation practice	Limitation addressed here
Cybersecurity / IoT	[9–11,32]	Accuracy/F1 on a single benchmark, often random splits	Official shifted test split; novel-attack analysis; bootstrap CIs; McNemar tests
Cloud–edge / FL	[14,15,36,37]	Convergence under selected partitions, few seeds	IID vs. Dirichlet non-IID; FedAvg vs. FedProx; partial participation; 5 seeds; communication cost
Credit risk	[16–19]	AUC on hold-out or k-fold, sometimes without variance correction	Corrected resampled t-tests, Friedman–Nemenyi, calibration, expected cost, fairness, SHAP
Financial analytics	[20,21,48,49]	Out-of-sample loss; ML vs. econometric baselines	Walk-forward VIX forecasts with Diebold–Mariano tests; VaR backtests with Kupiec/Christoffersen
Integration	[1,2,13]	Conceptual frameworks, rarely tested empirically	One protocol across all four functions; explicit separation of demonstrated vs. hypothesised mechanisms

2. MATERIALS AND METHODS

2.1 The ICDT framework and study design

The ICDT framework (Figure 1) is a layered reference model. Layer 1 (edge/IoT security analytics) ingests network-flow records at gateways and classifies them as benign or malicious. Layer 2 (cloud–edge learning) determines where models are trained and updated, contrasting centralised cloud training with federated training across edge clients. Layer 3 (financial intelligence) hosts credit-risk scoring and market-risk analytics. Layer 4 (assurance) is cross-cutting: every model in Layers 1–3 must pass the same statistical, calibration, explainability and fairness checks before it is considered deployable, in line with risk-management guidance [29] and MLOps practice [38]. The empirical study instantiates each layer with public data and a common evaluation protocol (Sections 2.2–2.7). The design is a comparative computational experiment; no human participants were involved.

Natural Resources for Human Health
eISSN: 2583-1194

Layer	Function	Inputs	Models evaluated	Outputs to next layer
L4 Assurance (cross-cutting)	Statistical validity, calibration, explainability, fairness, governance	Predictions and fold-level metrics from L1–L3	Bootstrap CI, McNemar, corrected t-test, Friedman–Nemenyi, DM, Kupiec/Christoffersen; ECE; TreeSHAP; DIR/EOD	Deploy / revise decision; audit trail
L3 Financial intelligence	Credit-risk scoring; volatility forecasting; value-at-risk	Applicant attributes; VIX and S&P 500 series	LR, DT, MLP, RF, XGBoost, LightGBM, stacking; HAR, RF, LightGBM; HS, EWMA, GARCH-*t*, quantile GBM	Risk scores, forecasts and limits
L2 Cloud–edge learning	Where and how models are trained	Partitioned traffic data on 10 clients	Centralised MLP; FedAvg (IID, non-IID, 50% participation); FedProx	Global model, communicatio n cost
L1 Edge/IoT security analytics	Detect intrusions at gateways	NSL-KDD network flows	LR, kNN, DT, RF, ExtraTrees, XGBoost, LightGBM, MLP	Alerts; latency and memory profile

Figure 1. Architecture of the Intelligent Computing for Digital Transformation (ICDT) framework and its empirical instantiation in this study (editable diagram). Information flows upward (L1→L2→L3); assurance feedback runs from L4 to every layer. The diagram is built as a native Word table so that it remains fully editable. Abbreviations are defined in Section 2.

2.2 Datasets

Five public data sources were used (Table 2). For cybersecurity, the NSL-KDD benchmark [52] was chosen because it provides an official training split (KDDTrain+, 125,973 records) and a deliberately harder test split (KDDTest+, 22,544 records) that contains 17 attack types absent from training (3,750 test records, 29.2% of test attacks). This property allows the effect of distribution shift to be measured directly, which random cross-validation cannot do [22]. Flows were mapped to five classes (Normal, DoS, Probe, R2L, U2R). NSL-KDD is old and its traffic is not native IoT traffic; it is used here as a controlled shift experiment, and newer IoT testbeds [31,32] are discussed as external-validity targets. For credit risk, two consumer-credit datasets with different sizes and feature sets were used: the German Credit data (1,000 applicants, 30.0% bad, 20 attributes, with a published 5:1 misclassification-cost convention) and a consumer credit-scoring dataset of 4,446 applications (28.1% bad, 15 attributes). For financial analytics, daily CBOE VIX closes (1990-01-02 to 2026-09-21) and the monthly S&P 500 composite price index from 1871 (Shiller series) were used.

Table 2. Datasets, sources and evaluation design.

Layer / task	Dataset	Size	Target / class balance	Evaluation design
L1 Intrusion detection	NSL-KDD [52]	125,973 train; 22,544 test; 41 features	Attack 46.5% (train), 56.9% (test)	Official split + 5-fold CV on train; bootstrap (B = 2,000)
L2 Federated IDS	NSL-KDD partitioned	10 clients	IID or Dirichlet (α = 0.3) label skew	40 rounds × 5 seeds; test on KDDTest+
L3 Credit risk	German Credit	1,000; 20 attributes	Bad 30.0%	10-fold × 3 repeats (30 folds)
L3 Credit risk	Consumer credit scoring	4,446; 15 attributes	Bad 28.1%	10-fold × 3 repeats (30 folds)

Layer / task	Dataset	Size	Target / class balance	Evaluation design
L3 Volatility forecasting	CBOE VIX daily	9,278 days	log VIX_{t+1}	Walk-forward, annual refit; test 2010-01-04–2026-09-21 (n = 4,237)
L3 Market risk	S&P 500 monthly (1871–2026)	1,868 monthly returns	5% and 1% VaR	Rolling/expanding; test 1950-01-01–2026-09-01 (n = 921)

Note: All data are publicly available (see Data Availability). Class balance refers to the positive (attack or bad-credit) class.

2.3 Pre-processing and feature engineering

All pre-processing was fitted inside each training fold or training window to avoid leakage [22]. For NSL-KDD, the three categorical fields (protocol, service, flag) were one-hot encoded with an "ignore unknown" rule so that unseen service values in the test split did not break inference; the heavy-tailed duration and byte counts were log-transformed ($\log(1 + x)$); a constant field (num_outbound_cmds) was removed; and features were standardised for distance- and gradient-based models. Credit attributes were one-hot encoded (categorical) and standardised (numeric) for logistic regression and the multilayer perceptron, and passed raw to tree models. For VIX forecasting, predictors were the HAR components (daily log VIX and its 5-day and 22-day means) [47] and, for ML models only, five lags, the daily log change, the log high–low range, a 22-day volatility of log changes, and the deviation from the 22-day mean. For VaR, monthly log returns (%) were computed from the price index; the quantile model used three return lags, the previous absolute return, 6-, 12- and 36-month rolling volatilities, 12-month momentum and the 12-month drawdown, all lagged by one month.

2.4 Models

Model families were chosen to span the transparency–flexibility spectrum (Table 3). Gradient boosting was implemented with XGBoost [53] and LightGBM [54]. To avoid optimistic bias from tuning on test folds, hyper-parameters were fixed *a priori* at conservative, widely used values rather than optimised per dataset; the consequence (a possible under-statement of the best achievable performance) is discussed in Section 4.7. The stacking ensemble combined logistic regression, random forest and XGBoost through a logistic meta-learner trained on internal 5-fold out-of-fold predictions. For federated learning, a one-hidden-layer perceptron (121 inputs, 64 ReLU units, sigmoid output; 7,873 parameters) was trained with mini-batch stochastic gradient descent (learning rate 0.05, batch 128, one local epoch per round). FedAvg weighted client updates by sample size [34]; FedProx added a proximal penalty $\mu\|w - w_{\text{global}}\|^2/2$ with $\mu = 0.01$ [37]. Non-IID partitions allocated each traffic class to clients with proportions drawn from a Dirichlet(0.3) distribution, producing strong label skew [36]. One deviation from the fixed settings was required: the five-class LightGBM detector with $\eta = 0.1$ diverged on the *training* data (training accuracy 0.553), so its learning rate was reduced to 0.05, which restored a training accuracy above 0.999. The decision was based on training-set diagnostics only, never on test results.

Table 3. Models and fixed hyper-parameters.

Model	Layer(s)	Key settings
Logistic regression (LR)	L1, L3	L2 penalty, C = 1; standardised inputs
k-nearest neighbours (kNN)	L1	k = 5, Euclidean distance on standardised inputs
Decision tree (DT)	L1, L3	L1: min leaf 5; L3: depth ≤ 5, min leaf 20
Random forest (RF)	L1, L3	L1: 200 trees; L3: 300 trees, min leaf 3; VIX: 300 trees, min leaf 20
Extremely randomised trees (ET)	L1	200 trees
XGBoost (XGB)	L1, L3	L1: 300 trees, depth 6, $\eta = 0.1$; L3: 300 trees, depth 3, $\eta = 0.05$,

Model	Layer(s)	Key settings
		subsample 0.8
LightGBM (LGBM)	L1, L3	L1: 300 trees, 31 leaves; L3: 300 trees, 15 leaves, $\eta = 0.05$; VIX: 400 trees, $\eta = 0.03$
Multilayer perceptron (MLP)	L1, L2, L3	L1: (64, 32), early stopping; L3: (32), $\alpha = 0.01$; L2: (64), SGD
Stacking (LR + RF + XGB → LR)	L3	Internal 5-fold out-of-fold meta-features
HAR [47]	L3	OLS on daily, weekly (5-day) and monthly (22-day) log-VIX components
Historical simulation (HS)	L3	Empirical quantile of previous 120 months
EWMA-normal	L3	$\lambda = 0.97$ on previous 240 months; Gaussian quantile
GARCH(1,1)-*t*	L3	Constant mean, Student-*t* innovations; re-estimated every 12 months, filtered monthly
Quantile gradient boosting (QGBM)	L3	LightGBM pinball loss at $\alpha = 0.05/0.01$; 300 trees, 7 leaves; refit every 12 months

2.5 Evaluation protocols

Intrusion detection. Each detector was trained once on KDDTrain+ and evaluated on KDDTest+, with the attack class as positive. Reported metrics are accuracy, precision, detection rate (DR, recall), false-positive rate (FPR), F1, Matthews correlation coefficient (MCC) and AUC. Detection rates were further broken down by attack category and by whether the attack type was *known* (present in training) or *novel*. As an in-distribution reference, six detectors were also evaluated by stratified 5-fold cross-validation on KDDTrain+. Five-class performance (macro- and weighted-F1, per-class recall) was assessed for the three strongest ensembles.

Edge profiling. For each fitted pipeline (including pre-processing) we measured the median and 95th-percentile latency of single-record inference over 300 calls, the amortised per-record time in a batch of 22,544 records, and the serialised model size. Measurements were taken on a two-core virtual CPU without hardware acceleration, which approximates a modest edge gateway; absolute values are hardware-specific, so the analysis focuses on relative differences.

Federated learning. Global models were evaluated on KDDTest+ after every communication round. Each configuration was run with five random seeds; we report mean ± standard deviation (SD) of the final-round F1 and AUC, the number of rounds needed to reach a target F1, and the communication volume (parameters × 4 bytes × 2 directions × participating clients × rounds). A centralised model trained for the same number of epochs and a *local-only* baseline (each client training alone) bracket the federated results.

Credit risk. Stratified 10-fold cross-validation repeated three times (30 folds) was used. Metrics were AUC, Brier score, Kolmogorov–Smirnov (KS) statistic, balanced accuracy and F1 for the bad class. For German Credit, the expected misclassification cost per applicant was computed with a cost of 5 for accepting a bad applicant and 1 for rejecting a good one, using the Bayes-optimal threshold $P(\text{bad}) \geq 1/6$. Calibration was summarised by the expected calibration error (ECE, 10 equal-width bins) on out-of-fold predictions. Global explanations used exact TreeSHAP attributions from XGBoost [44], aggregated over one-hot levels to original attributes. Group fairness by age (< 25 vs. ≥ 25 years) was assessed with the disparate-impact ratio (DIR, ratio of approval rates), the equal-opportunity difference (EOD, difference in approval rates among applicants who repaid) and the false-approval difference [19,45]; age was used as an illustrative protected attribute because it is recorded in both datasets.

Financial analytics. VIX forecasts were produced one day ahead in a walk-forward design: for each calendar year from 2010 onwards, models were re-estimated on all prior data and used to forecast every day of that year. Benchmarks were the random walk (RW, today's value) and HAR. Losses were evaluated on the log scale and in index points. VaR was forecast one month ahead for 921 months from 1950-01 to 2026-09 at the 5% and 1% levels.

2.6 Statistical analysis

Uncertainty and significance were quantified with procedures matched to each design (Table 4). For the single NSL-KDD test split, 95% confidence intervals (CIs) were obtained by non-parametric bootstrap (2,000 resamples) and paired detector comparisons used McNemar's test with continuity correction. For repeated cross-validation, fold-level metrics are not independent, so pairwise differences from logistic regression were tested with the corrected resampled t-test, which inflates the variance by $(1/J + n_{test}/n_{train})$ with $J = 30$ [26]; we report mean differences, corrected 95% CIs, p-values and the paired standardised effect size d_z (mean difference divided by SD of differences). Omnibus differences among the seven credit models were tested with the Friedman test, with Kendall's W as the effect size and the Nemenyi critical difference (CD) for post-hoc rank comparisons [27]. VIX forecast accuracy was compared with HAR using the Diebold–Mariano test on squared errors with the Harvey–Leybourne–Newbold small-sample correction [55,56]; CIs for root-mean-square error (RMSE) used a moving-block bootstrap (block length 20, 1,000 resamples). VaR forecasts were assessed with the Kupiec unconditional-coverage test, the Christoffersen independence and conditional-coverage tests [50,51] and the quantile (tick) loss. Two-sided $\alpha = 0.05$ was used throughout; because many tests are reported, p-values close to 0.05 are interpreted cautiously and the emphasis is on effect sizes and intervals.

Table 4. Statistical procedures by analysis.

Analysis	Design	Inference	Effect size / uncertainty
Intrusion detection	Single official test split	McNemar test (paired, continuity-corrected)	Bootstrap 95% CI (F1, AUC, MCC)
Federated learning	5 seeds × 40 rounds	Descriptive; seed variability	Mean ± SD; rounds-to-target; bytes
Credit risk	10-fold × 3 repeated CV	Corrected resampled t-test; Friedman; Nemenyi	ΔAUC, ΔBrier, Δcost with corrected 95% CI; d_z; Kendall's W
VIX forecasting	Walk-forward, annual refit	Diebold–Mariano (HLN-corrected)	RMSE/MAE; block-bootstrap 95% CI
Value-at-risk	Rolling/expanding windows	Kupiec (UC), Christoffersen (IND, CC)	Violation rate; tick loss

2.7 Reproducibility and computing environment

All experiments were implemented in Python 3.13 with scikit-learn 1.9, XGBoost, LightGBM, statsmodels 0.15 and arch; the federated simulation was written directly in NumPy so that every aggregation step is explicit. Random seeds were fixed for every split, model and bootstrap. Experiments ran on a two-core virtual machine with 7 GB of memory and no GPU. Scripts that regenerate every table and figure from the raw public data are available from the corresponding author (see Data Availability).

3. RESULTS

3.1 Layer 1: intrusion detection under distribution shift

Under in-distribution 5-fold cross-validation on KDDTrain+, every detector appeared almost perfect. Mean F1 ranged from 0.9725 (LR) to 0.9992 (LGBM), and the tree ensembles reached AUC ≥ 0.9999 with fold-to-fold SDs below 0.0003 (Table 5). Such figures are typical of benchmark reports [9,10]. On the official KDDTest+ split, however, F1 fell to between 0.740 and

0.834, an absolute loss of 16.4–24.7 percentage points (pp) for the six models evaluated both ways (Figure 2). The degradation was not uniform and it changed the ranking of the detectors.

Table 5. Binary intrusion-detection performance: in-distribution cross-validated F1 versus the official KDDTest+ split (attack = positive class; threshold 0.5).

Model	CV F1 (train)	Accuracy	Precision	DR	FPR	F1 [95% CI]	MCC	AUC [95% CI]	McNemar χ^2 vs. MLP (p)
LR	0.9725	0.751	0.915	0.621	0.077	0.740 [0.734, 0.747]	0.554	0.778 [0.771, 0.784]	1379.3 (< 0.001)
kNN	—	0.777	0.968	0.630	0.028	0.763 [0.757, 0.770]	0.617	0.818 [0.812, 0.824]	878.6 (< 0.001)
DT	0.9980	0.800	0.922	0.708	0.080	0.801 [0.796, 0.807]	0.628	0.855 [0.850, 0.860]	317.8 (< 0.001)
RF	0.9988	0.769	0.968	0.614	0.027	0.751 [0.745, 0.758]	0.606	0.961 [0.959, 0.964]	1241.7 (< 0.001)
ET	—	0.788	0.969	0.648	0.027	0.777 [0.771, 0.783]	0.633	0.960 [0.957, 0.963]	790.3 (< 0.001)
XGB	0.9991	0.790	0.968	0.652	0.028	0.780 [0.773, 0.786]	0.636	0.964 [0.962, 0.966]	805.6 (< 0.001)
LGBM	0.9992	0.781	0.968	0.637	0.028	0.769 [0.762, 0.775]	0.623	0.961 [0.959, 0.964]	1034.6 (< 0.001)
MLP	0.9979	0.834	0.969	0.732	0.031	0.834 [0.829, 0.839]	0.701	0.905 [0.901, 0.910]	(reference)

Note: CV F1 = mean of stratified 5-fold cross-validation on KDDTrain+ (kNN and ET were not cross-validated). DR = detection rate (recall); FPR = false-positive rate; MCC = Matthews correlation coefficient. Bracketed values are percentile-bootstrap 95% CIs (2,000 resamples). McNemar tests compare each model's correct/incorrect decisions with those of the highest-F1 model (MLP).

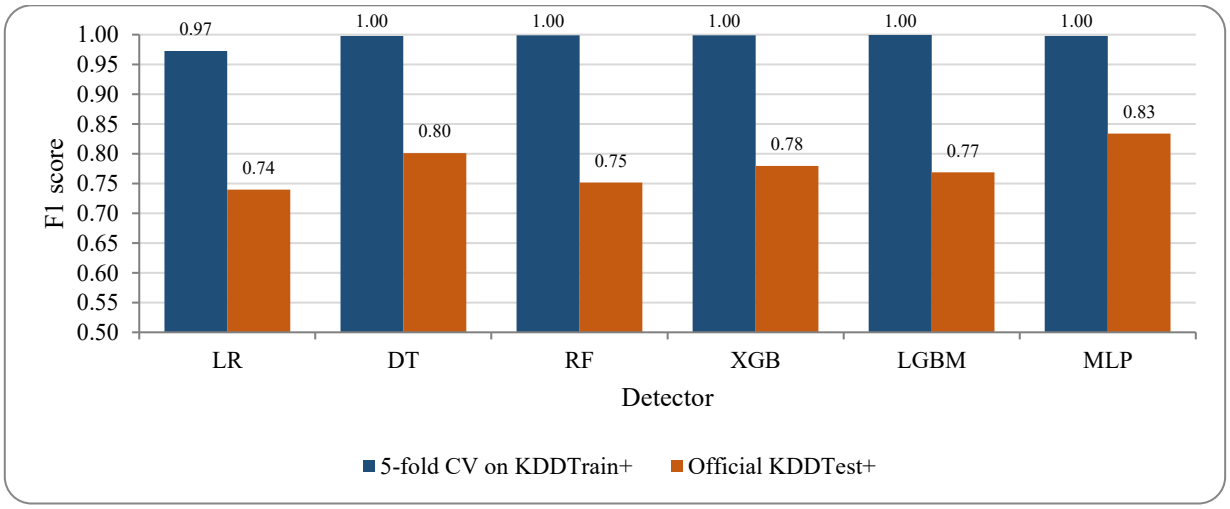


Figure 2. In-distribution versus shifted-test F1 for six intrusion detectors. The gap between the two bars is the performance lost when the detector meets the KDDTest+ distribution, which includes attack types absent from training.

On KDDTest+, the multilayer perceptron achieved the highest F1 (0.834, 95% CI [0.829, 0.839]) and MCC (0.701), followed by the single decision tree (F1 0.801); every other detector differed from the MLP significantly by McNemar's test (all $p < 0.001$). The tree ensembles nevertheless produced the best *ranking* of records: XGBoost had the highest AUC (0.964, 95% CI [0.962, 0.966]), and RF, ET and LightGBM reached 0.960–0.961, compared with 0.905 for the MLP. The apparent contradiction is resolved by the operating point. At the default threshold the ensembles were extremely conservative: precision ≈ 0.97 and FPR ≈ 0.03, but a detection rate of only 0.61–0.65. Their scores separated attacks from normal traffic well, but the threshold learned on the training distribution was mis-placed for the test distribution, in which attacks made up 56.9% of records rather than 46.5%.

Re-thresholding confirms this interpretation (Table 6). Lowering the XGBoost threshold from 0.5 to 0.005 (the lowest value examined) raised its detection rate from 0.652 to 0.812 and its F1 to 0.881, while FPR rose from 0.028 to 0.042. Because this threshold was chosen on the test data it is an *oracle* upper bound, not a deployable result; its purpose is to show that much of the F1 gap between detectors under shift reflects calibration of the decision threshold rather than an inability to rank attacks. A deployable remedy would require labelled data from the new environment or an unsupervised prior-shift correction, which we did not evaluate.

Table 6. Operating-point analysis for XGBoost on KDDTest+ (oracle sensitivity analysis; thresholds were varied on the test set).

Threshold on P(attack)	Detection rate	FPR	Precision	F1	MCC
0.5	0.652	0.028	0.968	0.780	0.636
0.3	0.664	0.029	0.968	0.787	0.644
0.2	0.676	0.029	0.968	0.796	0.655
0.1	0.697	0.030	0.968	0.810	0.671
0.05	0.715	0.032	0.967	0.823	0.686
0.02	0.747	0.034	0.966	0.843	0.711
0.01	0.779	0.036	0.966	0.862	0.738
0.005	0.812	0.042	0.963	0.881	0.764

Note: The 0.50 row reproduces Table 5. Results at other thresholds are descriptive and optimistic because the test labels were used to choose them.

3.1.1 Attack categories and novel attacks

Performance differed sharply by attack family (Figure 3). DoS attacks, which are numerous in training, were detected at rates of 0.78–0.92. Remote-to-local (R2L) attacks were almost invisible to most detectors: detection rates were 0.005–0.212, with only DT (0.212) and MLP (0.183) exceeding 0.10. The 17 attack types that never appear in KDDTrain+ account for 3,750 test records. Across detectors, the detection rate on these novel attacks (0.24–0.54) was on average 37.9 pp lower than on known attack types (0.71–0.81). Random forest, the most "accurate" model in cross-validation, detected only 23.5% of novel attacks, whereas the MLP and DT detected 54.1% and 53.0%.

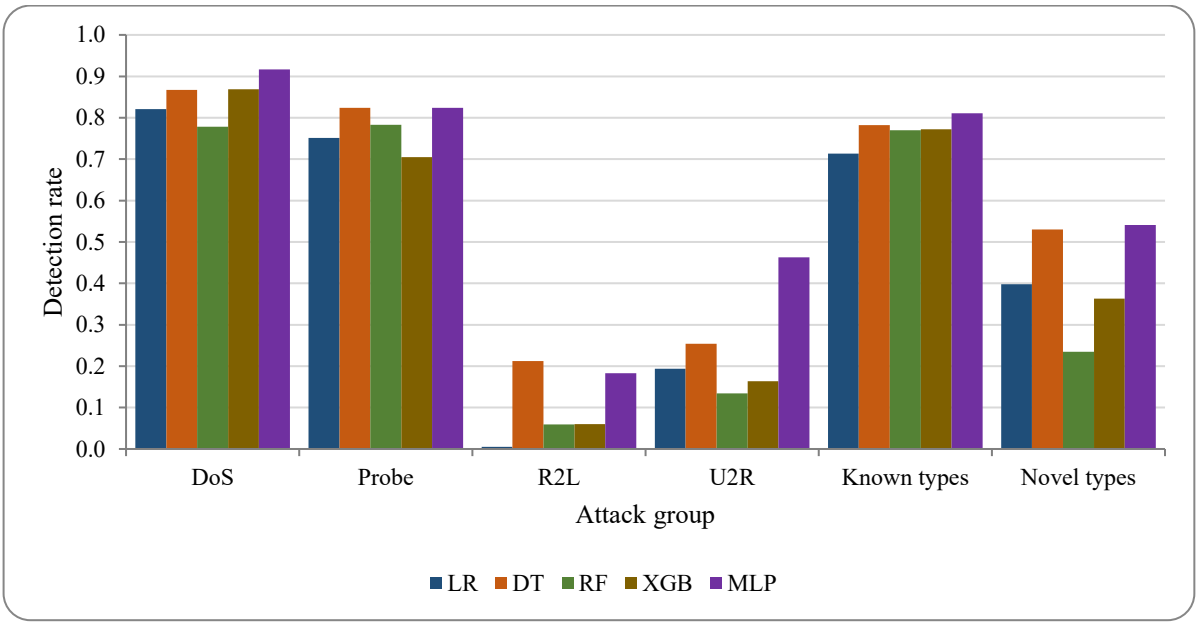


Figure 3. Detection rate by attack family and for attack types that are known (seen in training) or novel (absent from training), on KDDTest+ at threshold 0.5.

In the five-class task (Table 7) the ensembles reached accuracies of 0.749–0.779 but macro-F1 of only 0.513–0.575, because R2L and U2R recall was close to zero. The highest macro-F1 was obtained by XGB. Normal-traffic recall was ≥ 0.97 for every model, which again shows a decision boundary biased towards the majority behaviour seen in training.

Table 7. Five-class intrusion detection on KDDTest+: per-class recall / F1 for three tree ensembles.

Class	Test *n*	RF	XGBoost	LightGBM
Normal	9,711	0.974 / 0.780	0.972 / 0.803	0.970 / 0.777
DoS	7,458	0.794 / 0.870	0.843 / 0.899	0.766 / 0.852
Probe	2,421	0.603 / 0.709	0.655 / 0.723	0.606 / 0.702
R2L	2,887	0.052 / 0.099	0.085 / 0.157	0.093 / 0.170
U2R	67	0.060 / 0.110	0.179 / 0.293	0.209 / 0.289
Macro-F1		0.513	0.575	0.558
Weighted-F1		0.713	0.742	0.715
Accuracy		0.754	0.779	0.749

Note: The five-class LightGBM model uses $\eta = 0.05$ (see Section 2.4); with $\eta = 0.1$ its training accuracy was only 0.553, indicating divergence.

The binary XGBoost model concentrated its split gain on a few inputs: service (0.607), src bytes (0.163), wrong fragment (0.046), hot (0.038), dst host same srv rate (0.021) (gain shares, with one-hot levels of categorical fields summed). The network service type alone accounted for most of the gain, followed by the volume of bytes sent by the source. Gain shares summed over many one-hot levels favour high-cardinality fields, so this ranking is indicative. The reliance on service and byte-volume signatures is a demonstrated property of the fitted model; the claim that this reliance *causes* the poor transfer to novel attacks, which often use services and payload sizes absent from training, is a hypothesis that would require feature-ablation or counterfactual experiments to confirm.

3.1.2 Edge deployment cost

Table 8. Edge-deployment profile of the fitted pipelines (two-core virtual CPU, no accelerator).

Model	F1	AUC	Single-record latency, median (ms)	Latency, p95 (ms)	Batch time per record (µs)	Serialised size	Training time (s)
LR	0.740	0.778	5.9	8.2	3.7	7 kB	6.3
kNN	0.763	0.818	22.3	28.0	587.1	117.3 MB	0.3
DT	0.801	0.855	7.1	9.7	2.0	34 kB	2.8
RF	0.751	0.961	65.5	79.2	16.2	18.0 MB	14.2
ET	0.777	0.960	64.2	79.3	18.1	38.8 MB	26.5
XGB	0.780	0.964	7.6	10.5	4.4	638 kB	5.5
LGBM	0.769	0.961	7.5	11.4	11.0	1.0 MB	3.0
MLP	0.834	0.905	6.6	10.4	2.4	246 kB	22.4

Note: Latency includes the pre-processing pipeline (encoding and scaling). kNN stores the full training set, hence its size and batch cost.

Table 8 and Figure 4 show that accuracy and deployability diverged. Median single-record latency (including pre-processing) was 5.9–7.6 ms for LR, DT, XGBoost, LightGBM and the MLP, but 66 ms for the 200-tree RF and 64 ms for ET, whose serialised sizes were 18 MB and 39 MB. kNN, which must store the training set, occupied 117 MB and needed 587 µs per record even in batch mode, two orders of magnitude more than the tree boosters. The MLP (246 kB) and XGBoost (638 kB) therefore occupied the efficient frontier for gateway deployment, combining the best shifted-test F1 or AUC with millisecond latency and sub-megabyte memory. Training cost told a similar story: the MLP was the slowest to train among the compact models (22 s), but training happens off-line in the cloud layer, whereas inference cost is paid on every flow at the edge.

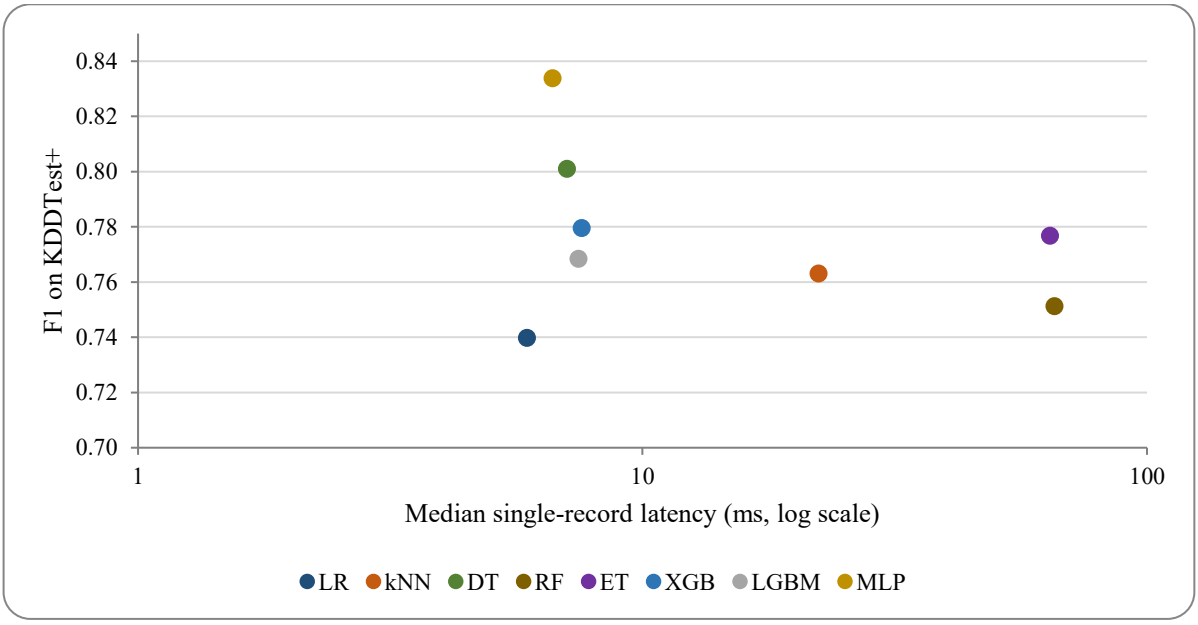


Figure 4. Accuracy–latency trade-off for edge deployment: shifted-test F1 against median single-record latency Points toward the upper left are preferable; the latency axis is logarithmic.

3.2 Layer 2: federated cloud–edge learning

The ten-client partitions differed markedly. Under IID partitioning every client held ≈ 47% attack traffic, whereas the Dirichlet(0.3) partition produced clients whose attack share ranged from 2% to 100% and whose sizes ranged from 3,105 to 36,136 records. Table 9 and Figure 5 summarise convergence over 40 rounds and five seeds.

Table 9. Federated versus centralised intrusion-detection training (MLP, 10 clients, 40 rounds, 5 seeds; evaluated on KDDTest+).

Training regime	Final-round F1 (mean ± SD)	Final-round AUC (mean ± SD)	Mean F1 at round 5 / 20	Best-round F1 (mean)	Total communication
Centralised (cloud) training	0.795 ± 0.013	0.912 ± 0.014	0.773 / 0.781	0.802	Raw data upload ≈ 61 MB
FedAvg, IID clients	0.765 ± 0.009	0.886 ± 0.007	0.741 / 0.743	0.767	25.2 MB
FedAvg, non-IID clients	0.760 ± 0.017	0.887 ± 0.018	0.728 / 0.744	0.764	25.2 MB
FedProx (μ = 0.01), non-IID	0.759 ± 0.016	0.886 ± 0.018	0.728 / 0.743	0.764	25.2 MB
FedAvg, non-IID, 50% participation	0.754 ± 0.019	0.886 ± 0.019	0.723 / 0.747	0.805	12.6 MB
Local-only (each client alone)	0.716 ± 0.122	0.869 ± 0.046	—	—	No exchange

Note: Communication = 7,873 parameters × 4 bytes × 2 directions × participating clients × 40 rounds. Local-only = mean over client models trained alone for up to 40 epochs (three seeds; clients holding a single class were excluded). Best-round F1 is the maximum over rounds, averaged over seeds, and is optimistic because it is selected on the test set.

Centralised training reached a final F1 of 0.795 ± 0.013 (AUC 0.912). FedAvg over IID clients reached 0.765 ± 0.009 (AUC 0.886), a shortfall of 3.1 pp in F1 (Welch $t = 4.39$, $p = 0.003$, five seeds per arm). Moving from IID to strongly label-skewed clients cost surprisingly little at round 40: F1 0.760 ± 0.017 (difference 0.5 pp, $p = 0.612$), with AUC essentially unchanged (0.887). Heterogeneity mainly increased variability between seeds (SD 0.017 vs. 0.009). The FedProx correction did not improve on FedAvg (F1 0.759 ± 0.016; paired difference −0.08 pp, $p = 0.249$). Halving client participation per round halved communication (12.6 MB instead of 25.2 MB) at a cost of 0.6 pp in final F1 and a noticeably noisier trajectory (Figure 5).

Natural Resources for Human Health
eISSN: 2583-1194

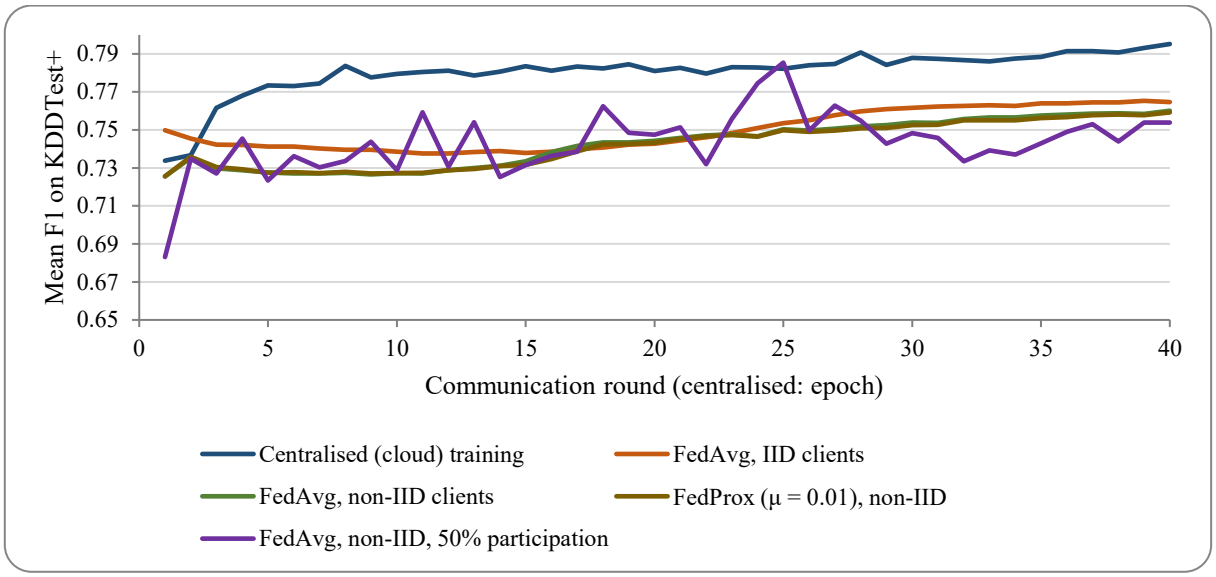


Figure 5. Convergence of the global model on KDDTest+ (mean F1 over five seeds) for centralised training and four federated regimes.

The decisive comparison is with isolation. A model trained by a single client on its own data achieved a mean F1 of 0.716 ± 0.122 (range 0.21–0.88 across 28 client models), well below every federated regime, and the worst clients, those holding almost exclusively one traffic class, produced detectors that were close to useless. Federation therefore delivered most of the benefit of pooling data while moving only model parameters: 25.2 MB over 40 rounds for ten clients, compared with roughly 61 MB to upload the encoded training data once. All federated and centralised models remained well below the shifted-test AUC of the tree ensembles in Section 3.1, because the same distribution shift applies and the MLP here is deliberately small.

The trajectories also differed in shape. Centralised training improved steadily, from a mean F1 of 0.773 after five epochs to 0.781 after twenty. FedAvg on IID clients plateaued early (0.741 at round 5, 0.743 at round 20) and only resumed improving late, whereas the non-IID regimes rose slowly (0.728 to 0.744). With half of the clients sampled per round, the global model oscillated from round to round because successive aggregates were dominated by different label mixtures; its mean best-round F1 (0.805) was therefore well above its final-round value, a gap that a deployed system could exploit only with a held-out validation set at the server. Because every curve is measured on the shifted test set, part of the round-to-round movement reflects how each aggregate happens to align with the test distribution rather than progress on the training objective.

3.3 Layer 3a: credit-risk assessment

Table 10. Credit-risk performance over 30 cross-validation folds (mean ± SD) and calibration on out-of-fold predictions.

Model	AUC	Brier score	KS	Balanced accuracy	ECE	Expected cost	Mean AUC rank
German Credit							
LR	0.784 ± 0.042	0.166 ± 0.018	0.499	0.674	0.034	0.540 ± 0.098	3.97
DT	0.718 ± 0.050	0.189 ± 0.019	0.384	0.602	0.070	0.608 ± 0.088	6.33
MLP	0.688 ± 0.106	0.190 ± 0.019	0.359	0.540	0.039	0.641 ± 0.076	5.97
RF	0.795 ± 0.042	0.164 ± 0.011	0.499	0.646	0.046	0.542 ± 0.072	2.95
XGB	0.796 ± 0.046	0.161 ± 0.020	0.505	0.684	0.035	0.555 ± 0.086	3.05
LGBM	0.791 ± 0.046	0.168 ± 0.023	0.499	0.684	0.074	0.591 ± 0.105	3.30

Model	AUC	Brier score	KS	Balanced accuracy	ECE	Expected cost	Mean AUC rank
Stack	0.800 ± 0.043	0.159 ± 0.016	0.517	0.679	0.021	0.530 ± 0.095	2.43
Credit scoring							
LR	0.831 ± 0.024	0.142 ± 0.009	0.537	0.701	0.017	—	4.93
DT	0.775 ± 0.026	0.163 ± 0.010	0.451	0.672	0.029	—	7.00
MLP	0.835 ± 0.026	0.142 ± 0.010	0.544	0.703	0.012	—	4.23
RF	0.840 ± 0.023	0.141 ± 0.008	0.557	0.700	0.029	—	3.90
XGB	0.846 ± 0.022	0.138 ± 0.009	0.564	0.720	0.015	—	2.43
LGBM	0.840 ± 0.021	0.141 ± 0.010	0.558	0.723	0.026	—	3.93
Stack	0.848 ± 0.023	0.138 ± 0.010	0.569	0.718	0.035	—	1.57

Note: Lower is better for Brier score, ECE (expected calibration error), expected cost and rank. Expected cost per applicant uses the German Credit 5:1 cost matrix at the Bayes threshold P(bad) ≥ 1/6. Ranks are averaged over folds (1 = best AUC).

All seven classifiers were evaluated on identical folds (Table 10). The stacking ensemble had the highest mean AUC on both datasets (0.800 on German Credit and 0.848 on the credit-scoring data), followed by XGBoost (0.796 and 0.846); logistic regression reached 0.784 and 0.831. The Friedman test rejected equal performance on both datasets (German: $\chi^2(6) = 91.8$, $p < 0.001$, Kendall's $W = 0.51$; credit scoring: $\chi^2(6) = 117.7$, $p < 0.001$, $W = 0.65$), indicating moderate-to-strong agreement among folds on the ordering. With a Nemenyi critical difference of 1.64 ranks, the stacking ensemble ranked significantly better than LR on the credit-scoring data (mean ranks 1.57 vs. 4.93) but not on German Credit (2.43 vs. 3.97). Much of the omnibus effect came from the weak single tree and, on German Credit, from the unstable MLP (AUC SD 0.106, reflecting early stopping on very small validation splits).

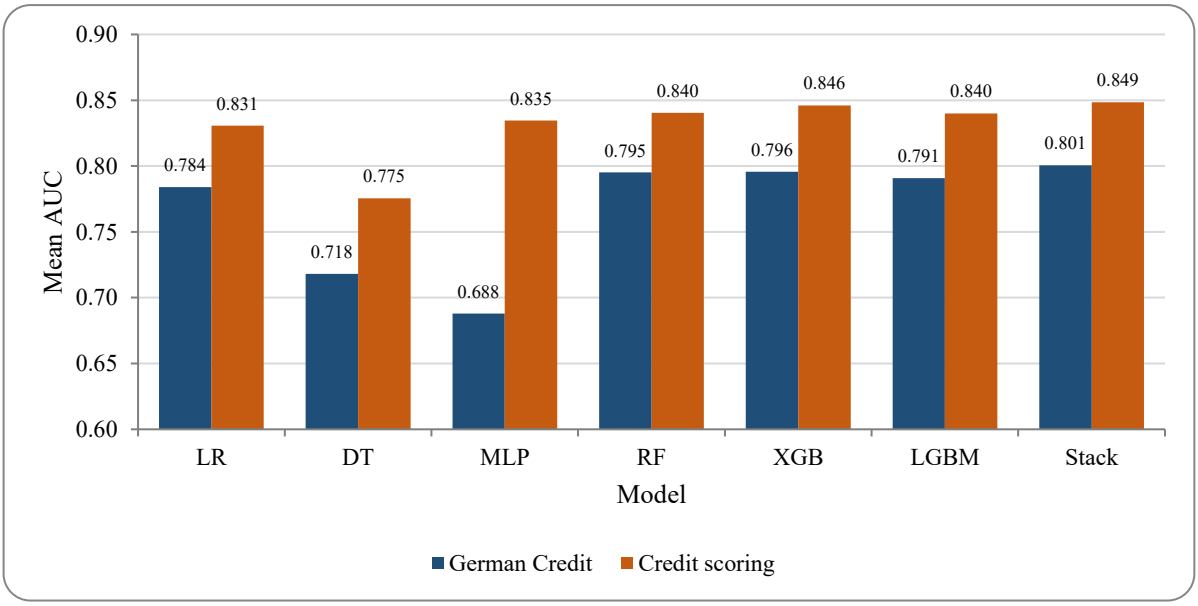


Figure 6. Mean cross-validated AUC (30 folds) of the seven credit-risk models on the two datasets. Fold-level SDs are reported in Table 10.

Table 11. Pairwise comparison with logistic regression: corrected resampled *t*-tests over 30 folds.

Model	ΔAUC	Corrected 95% CI	*p*	d_z	ΔBrier (*p*)	ΔExpected cost [95% CI]; *p*
German Credit						
DT	−0.066	[−0.100, −0.032]	< 0.001	−1.51	+0.0226 (0.001)	+0.068 [−0.019, 0.155]; 0.120
MLP	−0.096	[−0.179, −0.013]	0.025	−0.90	+0.0236 (0.005)	+0.101 [0.008, 0.193]; 0.034
RF	+0.011	[−0.009, 0.031]	0.268	0.43	−0.0023 (0.578)	+0.002 [−0.065, 0.069]; 0.952
XGB	+0.012	[−0.003, 0.027]	0.125	0.60	−0.0050 (0.180)	+0.014 [−0.051, 0.079]; 0.655
LGBM	+0.007	[−0.019, 0.032]	0.593	0.21	+0.0017 (0.783)	+0.051 [−0.024, 0.126]; 0.174
Stack	+0.017	[0.005, 0.028]	0.005	1.16	−0.0069 (0.003)	−0.010 [−0.070, 0.050]; 0.728
Credit scoring						
DT	−0.055	[−0.074, −0.037]	< 0.001	−2.35	+0.0205 (< 0.001)	—
MLP	+0.004	[−0.003, 0.010]	0.227	0.47	−0.0002 (0.857)	—
RF	+0.010	[−0.003, 0.022]	0.120	0.61	−0.0008 (0.694)	—
XGB	+0.015	[0.003, 0.027]	0.013	1.01	−0.0041 (0.076)	—
LGBM	+0.009	[−0.004, 0.022]	0.161	0.55	−0.0012 (0.657)	—
Stack	+0.018	[0.009, 0.027]	< 0.001	1.54	−0.0041 (0.011)	—

Note: Δ = model minus LR. Positive ΔAUC and negative ΔBrier/Δcost favour the model. CIs and *p*-values use the Nadeau–Bengio variance correction (1/30 + 1/9). d_z = mean paired difference / SD of paired differences (uncorrected). Expected cost is available for German Credit only.

The size of the gain over LR is the central credit-risk result (Table 11). After correcting for the dependence among resampled folds, only the stacking ensemble improved AUC significantly on both datasets: by 0.017 (95% CI [0.005, 0.028], $p = 0.005$, $d_z = 1.16$) on German Credit and 0.018 ([0.009, 0.027], $p < 0.001$, $d_z = 1.54$) on the credit-scoring data. XGBoost was significant only on the larger dataset (ΔAUC = 0.015, $p = 0.013$); RF and LightGBM gains of 0.007–0.011 had CIs spanning zero. The paired effect sizes are large because fold-to-fold variation is shared by all models, but the absolute improvements are 1–2 AUC points. In decision terms, no model reduced the German Credit expected misclassification cost relative to LR: the stacking ensemble changed cost by −0.010 per applicant (95% CI [−0.070, 0.050], $p = 0.728$), and the MLP increased it significantly (+0.101, $p = 0.034$).

Threshold-dependent metrics told the same story with more nuance. On German Credit, XGBoost and LightGBM reached the highest balanced accuracy (0.684 and 0.684 vs. 0.674 for LR), whereas RF, despite its competitive AUC, had a lower balanced accuracy (0.646), probably because averaging over many trees compresses its probabilities towards the base rate so that fewer bad applicants cross the 0.5 threshold. The KS statistic, widely used in retail credit, ranked stacking first on both datasets (0.517 and 0.569) but separated LR, RF and XGBoost by less than 0.01 on German Credit. Fold-to-fold SDs of AUC (0.042–0.050 on German Credit; 0.021–0.026 on the larger dataset, excluding the MLP) were several times larger than the

mean differences between the leading models, which is why uncorrected comparisons of single train–test splits can easily produce contradictory rankings.

3.3.1 Calibration, explainability and fairness

Calibration did not follow discrimination. On German Credit, stacking had the lowest ECE (0.021) and LightGBM the highest among the boosted models (0.074); on the credit-scoring data the ordering reversed for stacking, which had the highest ECE of all models (0.035) despite its best AUC, while the MLP (0.012), XGBoost (0.015) and LR (0.017) were best calibrated (Table 10; Figure 7). A model selected on AUC alone would therefore have produced the least reliable probabilities on that dataset, which matters wherever probabilities feed pricing or provisioning.

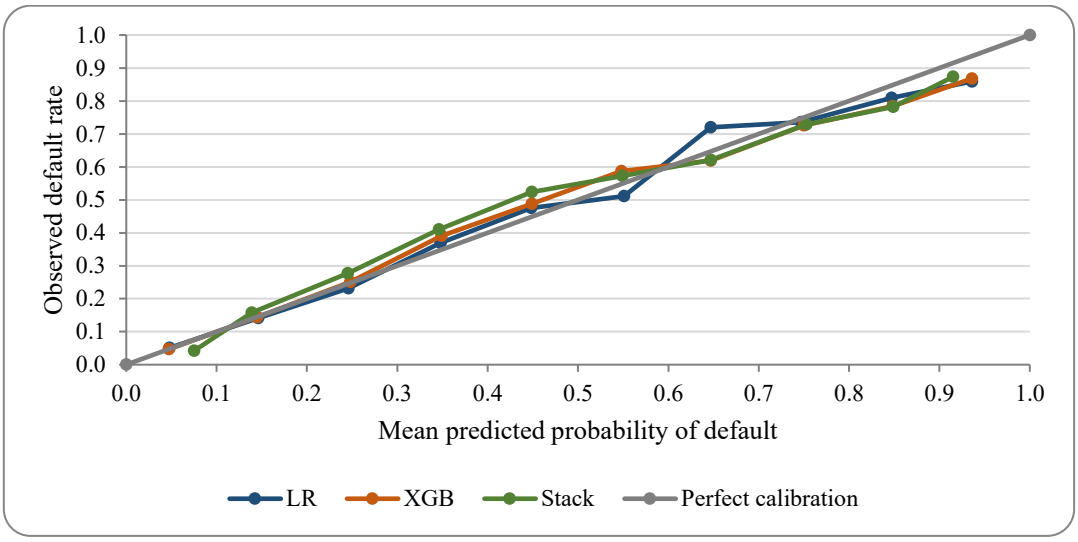


Figure 7. Reliability diagram for three models on the credit-scoring data (out-of-fold predictions, 10 equal-width bins; bins with fewer than 20 applicants omitted). Points on the diagonal indicate perfect calibration.

Global TreeSHAP attributions identified economically plausible drivers (Figure 8). For the credit-scoring data the most influential attributes were Seniority (0.535), Records (0.495), Assets (0.446), Finrat (0.406), Job (0.361), Savings (0.320) (mean |SHAP| on the log-odds scale). Directions were consistent with credit theory: longer employment seniority (Spearman ρ between value and attribution = −0.98), higher assets (−0.96), savings (−0.92) and income (−0.87) lowered predicted risk, whereas a higher financing ratio raised it (0.95). On German Credit, checking-account status dominated (mean |SHAP| 0.716), followed by loan duration (0.390; ρ = 0.92) and credit amount (0.339). Age ranked seventh (0.189) with a negative direction (ρ = −0.81): younger applicants were scored as riskier. These attributions describe how the fitted model uses the data; they are not causal effects of the attributes on default.

Natural Resources for Human Health
eISSN: 2583-1194

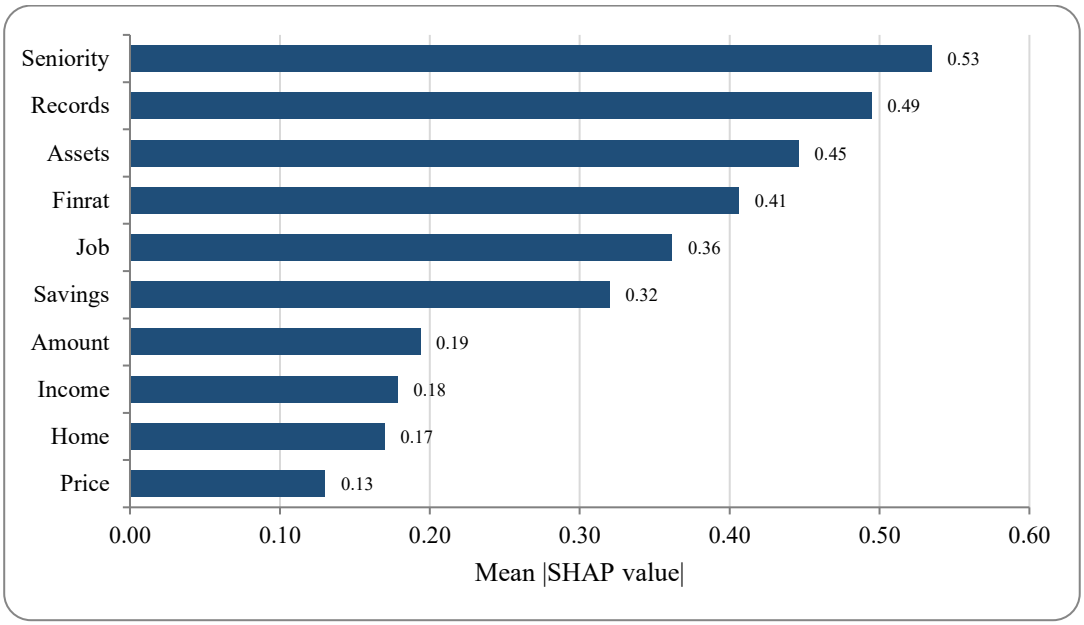


Figure 8. Global feature importance (mean absolute TreeSHAP value, log-odds scale) of the XGBoost credit model on the credit-scoring data.

Table 12. Group fairness by age (< 25 vs. ≥ 25 years) for out-of-fold predictions at threshold P(bad) < 0.5 (approve).

Model	Approval rate, young / older	Disparate-impact ratio	Equal-opportunity diff.	False-approval diff.	AUC young	AUC older
German Credit (young *n* = 149, observed bad rate 40.9% vs. 28.1%)						
LR	0.638 / 0.793	0.804	−0.141	−0.069	0.730	0.790
XGB	0.671 / 0.799	0.840	−0.089	−0.064	0.747	0.804
Stack	0.664 / 0.820	0.810	−0.116	−0.097	0.740	0.809
Credit scoring (young *n* = 558, observed bad rate 37.3% vs. 26.8%)						
LR	0.722 / 0.813	0.888	−0.070	−0.013	0.771	0.837
XGB	0.683 / 0.797	0.857	−0.095	−0.023	0.776	0.852
Stack	0.692 / 0.800	0.865	−0.076	−0.039	0.781	0.855

Note: Differences are young minus older. DIR < 0.80 is a common screening threshold for adverse impact (the "four-fifths" convention). Equal-opportunity difference compares approval rates among applicants who repaid.

All models approved younger applicants less often (Table 12). Disparate-impact ratios ranged from 0.80 to 0.89, just above the four-fifths screening value; LR on German Credit was closest to it (0.804). Equal-opportunity differences were negative throughout (−0.141 to −0.070): creditworthy young applicants were approved 7–14 pp less often than creditworthy older applicants. Part of the approval gap reflects genuinely higher observed default among the young (Table 12 header rows), but the equal-opportunity metric conditions on repayment, so the residual gap is not explained by base rates. Discrimination was also weaker for the young (AUC lower by 0.06–0.08). No model was uniformly fairest: XGBoost had the best DIR and EOD on German Credit but the worst on the credit-scoring data, which shows that fairness must be measured per deployment rather than inferred from model class.

Natural Resources for Human Health
eISSN: 2583-1194

3.4 Layer 3b: financial analytics

3.4.1 Volatility forecasting

Table 13. One-day-ahead VIX forecasts, walk-forward with annual re-estimation (2010-01-04 to 2026-09-21, n = 4,237 days).

Model	RMSE (log)	MAE (log)	RMSE (points) [95% CI]	MAE (points)	Directional accuracy	DM vs. HAR (*p*)	RMSE COVID-19 shock	RMSE 2017 (calm)
Random walk	0.0777	0.0545	1.852 [1.559, 2.165]	1.090	—	2.39 (0.017)	5.46	0.84
HAR (OLS)	0.0769	0.0539	1.827 [1.536, 2.156]	1.068	0.532	(benchmark)	5.54	0.82
Random forest	0.0786	0.0556	1.867 [1.562, 2.204]	1.098	0.516	3.94 (< 0.001)	5.75	0.89
LightGBM	0.0795	0.0559	1.939 [1.610, 2.288]	1.118	0.518	5.81 (< 0.001)	6.41	0.90
HAR + LightGBM average	0.0774	0.0544	1.848 [1.548, 2.164]	1.081	0.532	2.61 (0.009)	5.64	0.83

Note: DM = Diebold–Mariano statistic on squared log errors with the Harvey–Leybourne–Newbold correction; positive values mean larger losses than HAR. RMSE CIs from a moving-block bootstrap (block 20). COVID-19 shock = 15 Feb–30 Jun 2020. Directional accuracy = share of days on which the forecast correctly signs the change from today's value.

The HAR model was the most accurate VIX forecaster (Table 13). Its log RMSE (0.0769) was lower than that of the random walk (0.0777; DM = 2.39, $p = 0.017$), random forest (0.0786; DM = 3.94, $p < 0.001$) and LightGBM (0.0795; DM = 5.81, $p < 0.001$). Averaging HAR with LightGBM did not help (DM = 2.61, $p = 0.009$). The differences are statistically clear but economically small: in index points the RMSEs were 1.83 (HAR) and 1.94 (LightGBM), with overlapping bootstrap intervals. Directional accuracy was barely above chance for every model (0.516–0.532). The fitted HAR placed 0.89 of its weight on the daily component, consistent with the strong persistence of implied volatility.

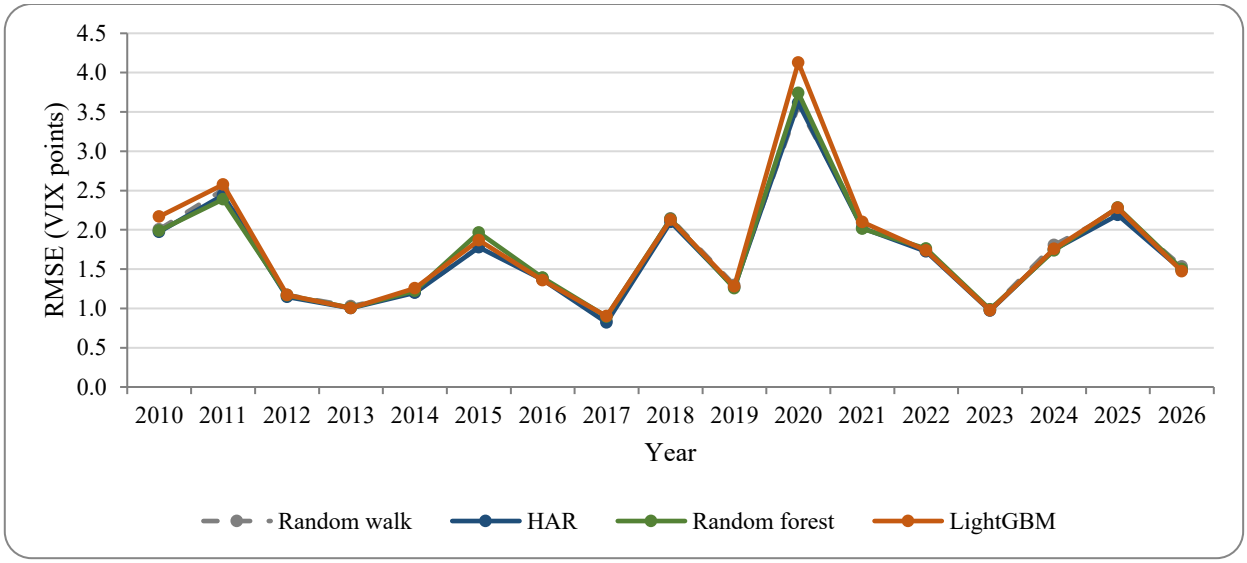


Figure 9. Annual out-of-sample RMSE (index points) of one-day-ahead VIX forecasts, 2010–2026. All models share the 2020 COVID-19 spike; LightGBM deteriorates most in that year.

Figure 9 shows where the ML models lost ground. In calm years their errors were indistinguishable from HAR, but in the COVID-19 shock LightGBM's RMSE (6.41 points) exceeded HAR's (5.54) by 16%. Regression trees cannot extrapolate beyond the range of the target seen during training, and on 16 March 2020 the VIX closed at 82.69, above the previous record close of 80.86 (20 November 2008) in the training data; this is a known property of tree models rather than a new finding, but it explains why their losses are concentrated in extreme regimes.

3.4.2 Market-risk measurement (value-at-risk)

Monthly S&P 500 log returns (n = 1,868) had mean 0.40% and SD 4.04%, negative skewness (−0.52) and heavy tails (excess kurtosis 11.2; Jarque–Bera $p < 0.001$). Absolute returns were strongly autocorrelated (Ljung–Box $Q(12)$ $p < 0.001$), confirming volatility clustering. Returns themselves were also autocorrelated ($p < 0.001$), which is partly an artefact of the series being a monthly average of daily prices and is acknowledged as a limitation.

Table 14. Value-at-risk backtests for monthly S&P 500 returns (1950-01 to 2026-09, n = 921 months).

Model	Violations (rate)	Kupiec UC LR (*p*)	Christoffersen IND LR (*p*)	CC LR (*p*)	Tick loss	Mean VaR (%)
5% VaR (expected violations 46.1)						
Historical simulation	54 (5.86%)	1.37 (0.241)	6.13 (0.013)	7.50 (0.023)	0.4755	−5.08
EWMA-normal	62 (6.73%)	5.27 (0.022)	7.07 (0.008)	12.34 (0.002)	0.4719	−5.11
GARCH(1,1)-*t*	56 (6.08%)	2.12 (0.145)	0.75 (0.387)	2.87 (0.238)	0.4580	−5.12
Quantile LightGBM	65 (7.06%)	7.32 (0.007)	0.04 (0.840)	7.36 (0.025)	0.4488	−4.43
1% VaR (expected violations 9.2)						
Historical simulation	18 (1.95%)	6.63 (0.010)	8.40 (0.004)	15.03 (< 0.001)	0.1794	−9.44
EWMA-normal	24 (2.61%)	16.63 (< 0.001)	5.16 (0.023)	21.79 (< 0.001)	0.1722	−7.46
GARCH(1,1)-*t*	17 (1.85%)	5.33 (0.021)	1.00 (0.317)	6.33 (0.042)	0.1516	−8.49
Quantile LightGBM	23 (2.50%)	14.73 (< 0.001)	1.18 (0.277)	15.91 (< 0.001)	0.1601	−7.22

Note: UC = unconditional coverage (χ^2_1); IND = independence of violations (χ^2_1); CC = conditional coverage (χ^2_2). Lower tick (quantile) loss is better.

Only GARCH(1,1)-*t* passed all three backtests at the 5% level (Table 14): 56 violations against 46 expected (UC $p = 0.145$, IND $p = 0.387$, CC $p = 0.238$). Historical simulation and EWMA produced clustered violations (IND $p = 0.013$ and 0.008), the classic failure of risk models that react too slowly to volatility regimes. The quantile LightGBM model is instructive: it attained the *lowest* 5% tick loss (0.4488 vs. 0.4580) and independent violations (IND $p = 0.840$), yet it breached its VaR 65 times, a 7.1% rate that failed the coverage test (UC $p = 0.007$). Its mean VaR (−4.43%) was less conservative than GARCH-*t* (−5.12%). At the 1% level every model under-covered the tail; GARCH-*t* came closest (17 vs. 9.2 expected, UC $p = 0.021$) and combined independent violations with the lowest tick loss (0.1516). At the 1% level, violations clustered in the 1970s and 2000s for every model (Figure 10).

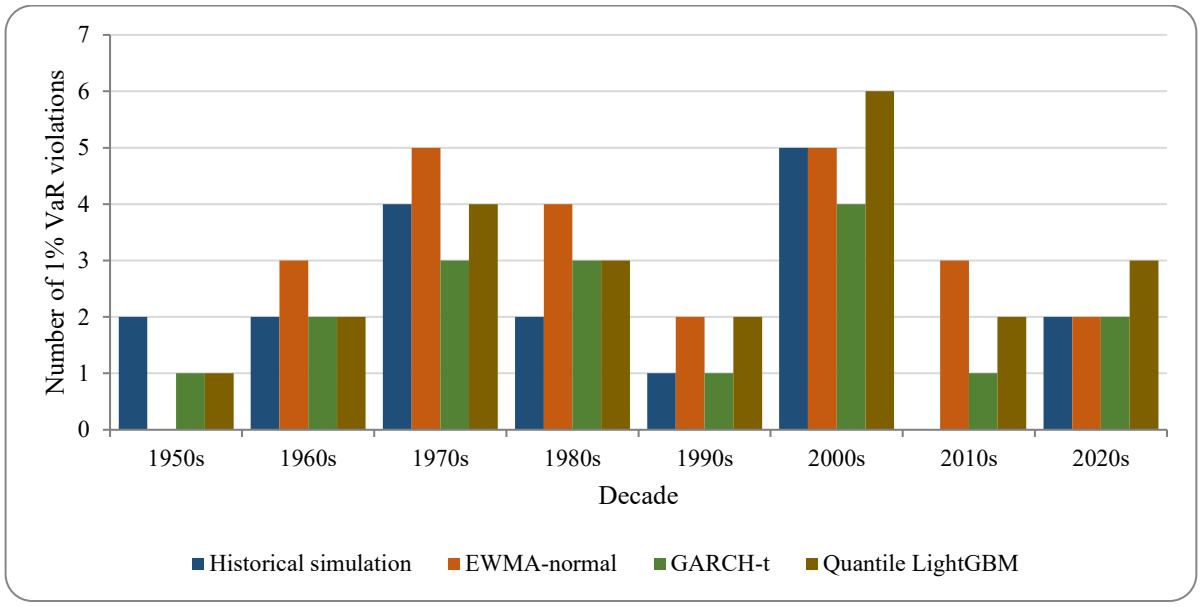


Figure 10. 1% VaR violations per decade for four risk models (monthly S&P 500, 1950–2026). With roughly 120 months per decade, about 1.2 violations per decade are expected under correct coverage.

3.5 Cross-domain synthesis

Table 15 brings the four layers together by comparing the best transparent baseline in each task with the best flexible AI model on the metric that matters for deployment. The "complexity premium" was large only where the data were high-dimensional, plentiful and drawn from the training distribution, and it shrank, vanished or reversed under distribution shift, small samples or time-series forecasting.

Table 15. Complexity premium by layer: best transparent baseline versus best flexible AI model, with the hypothesis verdict.

Layer / task	Deployment metric	Transparent baseline	Best flexible model	Premium (95% CI or test)	Hypothesis verdict
L1 IDS, in-distribution	F1 (5-fold CV)	LR 0.972	LGBM 0.9992	+2.7 pp (fold SD < 0.002)	H1 (part 1) supported
L1 IDS, shifted test	F1 (KDDTest+)	LR 0.740	MLP 0.834; best ensemble XGB 0.780	MLP +9.4 pp; ensembles lose to MLP (McNemar p < 0.001)	H1 (part 2) supported; ensemble superiority not supported
L1 IDS, ranking	AUC (KDDTest+)	LR 0.778	XGB 0.964	+0.186 (non-overlapping bootstrap CIs)	Ensembles rank best
L2 Federated IDS	F1 vs. centralised	Centralised 0.795	FedAvg-IID 0.765; non-IID 0.760	−3.1 pp (IID); −3.5 pp (non-IID)	H2 partly supported (skew effect small; FedProx no gain)
L3 Credit (German)	AUC; expected cost	LR 0.784	Stack 0.800	ΔAUC +0.017 [0.005, 0.028]; Δcost n.s.	H3 supported

Layer / task	Deployment metric	Transparent baseline	Best flexible model	Premium (95% CI or test)	Hypothesis verdict
L3 Credit (scoring)	AUC; ECE	LR 0.831	Stack 0.848	ΔAUC +0.018 [0.009, 0.027]; ECE worse (0.035 vs. 0.017)	H3 supported
L3 VIX forecasting	RMSE (log)	HAR 0.0769	RF 0.0786	Worse than HAR (DM $p < 0.001$)	H4 supported
L3 Value-at-risk (5%)	Coverage tests; tick loss	GARCH-t: passes UC/IND/CC	QGBM: tick 0.4488	Lower loss but fails UC ($p = 0.007$)	H4 supported

Read across rows, Table 15 suggests three regularities. First, the premium is largest for ranking quality in high-volume, high-dimensional classification (intrusion detection), where ensembles can exploit many weak interactions. Second, the premium on decisions (F1, cost, coverage) is consistently smaller than the premium on ranking or loss (AUC, tick loss), because decisions also depend on thresholds and calibration that flexible models do not automatically get right. Third, wherever the target is a persistent low-dimensional process (implied volatility, monthly market risk), the premium is zero or negative. These regularities are inductive generalisations from eight comparisons on five data sources and should be treated as propositions for further testing rather than laws.

4. DISCUSSION

This study evaluated AI across the security, computing and financial functions of a digitally transforming organisation under one statistical protocol. The unifying result is that the value of model complexity is conditional. Flexible learners dominated when training and deployment data matched, delivered small but statistically defensible gains in credit scoring, and offered no advantage, or a disadvantage, in persistent financial time series and under attack-distribution shift. We discuss each layer, compare the findings with recent literature, and then separate what the experiments demonstrate from what they only suggest.

4.1 Security analytics: benchmark accuracy is not field accuracy

H1 was only partly supported. Tree ensembles did outperform LR in cross-validation (F1 ≈ 0.999 vs. 0.972) and retained the best ranking quality on KDDTest+ (AUC 0.964 for XGBoost vs. 0.778 for LR), but at the operational threshold they were overtaken by an MLP and even by a single decision tree. The 16–25 pp F1 loss between cross-validation and the official test split is direct evidence of the evaluation pitfall described by Arp et al. [22]: random splits leak the test distribution into training. It is also consistent with the poor cross-dataset transfer reported by Sarhan et al. [25] and the drift-induced decay documented by Andresini et al. [24]. The NSL-KDD designers built KDDTest+ to be harder than the training data [52], and the same split continues to serve as the reference test for newer deep-learning detectors [57]. Our contribution is to decompose the loss: the ensembles' high AUC with low recall shows that a large part of the shifted-test deficit arose from a mis-placed decision threshold under a change in attack prevalence, while the remainder came from attack families (R2L, U2R and novel types) whose feature signatures were barely represented in training.

The practical implication for IoT and cloud security operations is that accuracy reported on a static benchmark should not be used to select a detector. A deployment process needs (i) a shift-aware validation set, ideally collected from the target network; (ii) threshold recalibration as part of routine model operations [38]; and (iii) monitoring of class prevalence and novel-signature rates. The superiority of the MLP on novel attacks is a demonstrated empirical fact in this experiment, but our explanation for it (smoother decision surfaces that extrapolate more gracefully than axis-aligned tree partitions) is a hypothesis; it should be tested with controlled feature-ablation studies and on modern IoT datasets such as TON_IoT and Edge-IIoTset [31,32]. Edge profiling adds a second selection criterion. Random forest and kNN, which looked competitive in accuracy, carried 10^1–10^2 MB footprints and tens of milliseconds of latency, whereas XGBoost and the MLP combined millisecond inference with sub-megabyte models, the profile required for edge intelligence [12].

4.2 Cloud–edge learning: isolation, not label skew, is the main cost

H2 was partly supported. FedAvg on IID clients approached, but did not reach, centralised performance (−3.1 pp F1). Contrary to the second part of H2, strong label skew reduced final F1 by only 0.5 pp, and FedProx gave no measurable benefit. This contrasts with reports that non-IID data severely degrade FedAvg [35,36] and with the motivation for FedProx [37]. A plausible explanation, which we did not test, is that with one local epoch per round and a small model, client drift is limited, so the proximal term has little to correct; FedProx is designed for many local steps and system heterogeneity. The more robust lesson is the contrast with isolated training: edge clients learning alone produced highly variable and sometimes useless detectors, whereas federation recovered most of the pooled-data performance while exchanging only model parameters. This supports the position of Rahman et al. and Campos et al. that FL is a realistic middle path between centralised and on-device learning for IoT intrusion detection [14,15]. Privacy is not guaranteed by federation alone; gradient leakage and poisoning attacks require secure aggregation and robust aggregation rules [35], which were outside the scope of this experiment.

4.3 Credit risk: small, real gains that are not free

H3 was supported. The stacking ensemble improved AUC over LR by 0.017 and 0.018 on the two datasets, magnitudes close to those reported in large benchmarks and in recent comparisons of ensembles, deep networks and augmented logistic regression [16,17,40]. Gradient boosting matched or exceeded deep architectures, as found more generally for tabular data [41,42]; our MLP was competitive only on the larger dataset and unstable on the 1,000-applicant German data. Whether a 1–2 point AUC gain is worth the loss of transparency depends on portfolio size and margins: Berg et al. show that modest discriminatory gains can be economically meaningful at scale [6], yet in our cost-sensitive evaluation of German Credit no model reduced expected misclassification cost significantly relative to LR.

Two governance findings qualify the AUC ranking. First, calibration was model- and dataset-specific; the best-discriminating model on the credit-scoring data was the worst calibrated. Probability calibration should therefore be checked, and if necessary corrected, separately from discrimination. Second, all models approved younger applicants at lower rates, and creditworthy young applicants were approved 7–14 pp less often than creditworthy older applicants. This is consistent with evidence that more flexible credit models can redistribute outcomes across groups [46] and with the conclusion of Kozodoi et al. that fairness must be assessed explicitly and traded off against profit [19]. Because the EU AI Act treats creditworthiness assessment of natural persons as high-risk [28], such audits, together with explanation tools of the kind applied here [18,43], would be part of the conformity evidence for deployment. Age was used as an illustrative attribute; we do not claim that the observed disparities are unlawful, and whether age may legitimately enter a credit model depends on jurisdiction.

4.4 Financial analytics: a simplicity frontier

H4 was supported. HAR beat both tree ensembles for one-day-ahead VIX forecasting, and GARCH-*t* was the only VaR model to pass all backtests at the 5% level. These results agree with recent large-scale evidence that a well-specified HAR is hard to beat [21], and specifically for the VIX [48]. They do not contradict findings that ML improves return prediction in high-dimensional cross-sections [20]; rather, they indicate that where the signal is a low-dimensional, highly persistent process, flexible learners add variance without adding information. The quantile LightGBM result is a useful caution for risk practice: the model had the lowest 5% quantile loss (0.4488) yet failed the coverage test with 65 breaches, so it would be penalised by regulatory backtesting despite being "best" on a statistical loss. Blom et al. report that LightGBM-type quantile models can match benchmarks for currency VaR when forward-looking implied volatilities are available [49]; our feature set was purely backward-looking, which is a plausible reason for the discrepancy and a hypothesis for future work.

4.5 Integration: what the framework demonstrates and what it hypothesises

The ICDT framework is intended as an organising and assurance device rather than a single integrated model, and the evidence should be read accordingly (Table 16). Demonstrated findings are those directly measured in this study under stated conditions. Hypothesised mechanisms are explanations that fit the data but were not isolated experimentally. Cross-layer propagation, for example the way an undetected intrusion might corrupt data used by a credit or trading model, is motivated by evidence on systemic cyber risk in finance [7,8] but was not simulated here, and we make no quantitative claim about it.

Table 16. Evidence status of the main claims.

Claim	Status	Basis in this study
Random-split CV overstates field performance of intrusion detectors by 16–25 pp F1	Demonstrated	Same detectors, CV vs. official shifted split (Table 5, Figure 2)
Under shift, much of the ensembles' F1 deficit is a threshold (operating-point) problem	Demonstrated (oracle)	High AUC with low recall; re-thresholding analysis (Table 6)
Novel attack types are detected far less often than known types	Demonstrated	Known vs. novel detection rates (Figure 3)
MLP extrapolates better to novel attacks because of smoother decision surfaces	Hypothesis	Consistent with results; no ablation performed
Federation recovers most of the pooled-data benefit relative to isolated clients	Demonstrated (simulation)	Table 9; local-only baseline
Small client drift explains why FedProx did not help	Hypothesis	Not isolated experimentally
Ensembles give small but significant AUC gains over LR in credit scoring	Demonstrated	Corrected *t*-tests (Table 11)
AUC-best credit model is not necessarily best calibrated or fairest	Demonstrated	ECE and fairness metrics (Tables 10, 12)
SHAP attributions reflect causal drivers of default	Not claimed	Attributions describe model behaviour only
HAR/GARCH-*t* outperform ML for persistent financial series	Demonstrated for these series	DM and VaR backtests (Tables 13, 14)
Tree models fail in extreme regimes because they cannot extrapolate	Partly demonstrated	2020 error spike; mechanism is a known model property
Cyber incidents propagate into financial-risk model failures	Hypothesis (not tested)	Motivated by literature [7,8]

For organisations pursuing digital transformation, the operational lesson is a portfolio approach to AI adoption [1,4]. Resources should flow to settings where the complexity premium is demonstrably large (high-volume, stationary classification), while parsimonious, auditable baselines should be retained, and treated as challengers that new models must beat out of sample, in persistent financial time series and in small or regulated decision domains. A shared assurance layer, implementing the same statistical, calibration, explanation and fairness checks across security and finance, operationalises the governance functions envisaged by the NIST AI RMF [29] and complements zero-trust security architecture [39].

Conceptually, the results support treating the complexity premium as a function of the data-generating regime rather than as a property of algorithms. Three regime characteristics appear to matter: the dimensionality and interaction richness of the inputs, the stationarity of the relationship between inputs and target, and the number of independent training observations relative to model capacity. High dimensionality with stationarity favours flexible learners (in-distribution intrusion detection); limited samples and moderate dimensionality narrow the premium (credit scoring); persistence in a low-dimensional state variable removes it (implied volatility and market risk); and non-stationarity can reverse it at the decision level (shifted intrusion detection). Framed this way, digital-transformation programmes can anticipate where AI investment is likely to pay off before committing to it, and the proposition is falsifiable on new domains.

4.6 Implications for practice and policy

For *security operations and IoT engineering* teams, the results argue for selecting detectors on a validation set that mimics the deployment environment, for scheduling threshold recalibration whenever traffic composition changes, and for treating

latency and memory as first-class selection criteria at the gateway. A compact boosted-tree or neural detector at the edge, with heavier models and retraining in the cloud, is consistent with both the accuracy and the cost profiles we observed [12,13]. For *cloud architects*, federated training is a credible default where raw data cannot be pooled; the dominant risk is not moderate label skew but clients that are too small or too homogeneous to learn alone, and the design question is therefore how to recruit enough diverse participants per round. For *credit-risk officers*, a 1–2 point AUC gain should be weighed against calibration, expected cost and fairness, each measured on the institution's own portfolio; an augmented logistic model of the kind proposed by Dumitrescu et al. [17] may capture much of the gain at lower governance cost. For *market-risk functions*, ML quantile models should be benchmarked against GARCH-type models on coverage tests, not only on statistical loss. For *regulators and auditors*, the study illustrates why high-risk AI obligations [28] are best met with evidence that combines out-of-distribution performance, uncertainty intervals, calibration and group metrics, rather than a single headline accuracy figure.

4.7 Limitations

Several limitations bound the conclusions. (1) *Data age and representativeness.* NSL-KDD is derived from late-1990s traffic and is not IoT traffic; the German Credit data are small and historical; the credit-scoring dataset's provenance is less documented than that of regulatory data; and the S&P 500 series is a monthly average of daily prices, which smooths returns and induces autocorrelation. The results should be replicated on contemporary IoT testbeds and loan-level portfolio data. (2) *Fixed hyper-parameters.* We deliberately avoided per-dataset tuning to prevent test-set leakage; nested tuning could raise the ceiling for every model, although recent evidence suggests it rarely overturns HAR-type benchmarks [21]. (3) *Simulated federation.* Clients were simulated on one machine with synchronous rounds and no dropouts, adversarial clients, secure aggregation or differential privacy; communication was counted, not measured on a network. (4) *Hardware-specific profiling.* Latency was measured on a general-purpose virtual CPU, not on embedded IoT hardware. (5) *Oracle threshold analysis.* The re-thresholding results in Table 6 used test labels and are an upper bound. (6) *Multiple testing.* Many hypothesis tests are reported; we did not apply a family-wise correction across layers, so isolated p-values near 0.05 should be interpreted cautiously. (7) *Fairness scope.* Fairness was assessed for one binary age split, at one threshold, without intersectional or causal analysis. (8) *No cross-layer experiment.* The integration claim is architectural and comparative; it does not include an end-to-end simulation of cyber incidents propagating into financial decisions.

5. CONCLUSION

This study integrated cybersecurity, IoT, cloud–edge computing, credit-risk assessment and financial analytics in a single, reproducible, statistically controlled evaluation of AI. Four conclusions follow. First, intrusion detectors that appear nearly perfect under random cross-validation lose 16–25 pp of F1 when attack distributions shift, and novel attacks are the main blind spot; detector selection must therefore rest on shift-aware validation, threshold management and edge-cost profiling, not benchmark accuracy. Second, federated learning is a practical cloud–edge compromise: it recovered most of the benefit of pooled data, was robust to label skew in this setting, and far outperformed isolated clients. Third, in credit risk, ensembles yield small but statistically defensible discrimination gains that do not automatically translate into lower expected cost, better calibration or fairer outcomes, so model governance must evaluate all of these. Fourth, for persistent financial time series, parsimonious econometric models (HAR and GARCH-t) remained superior to flexible ML. The common thread is that the value of AI in digital transformation is conditional on data regime and evaluation rigour, and the ICDT framework's cross-cutting assurance layer is the mechanism that makes this value visible and auditable.

Future work should (i) replicate the protocol on contemporary IoT and cloud traffic and on loan-level credit data; (ii) evaluate prior-shift correction and continual learning for intrusion detection; (iii) add secure aggregation, differential privacy and adversarial-client defences to the federated layer; (iv) test forward-looking and alternative-data features for ML risk models; and (v) build an explicit cross-layer simulation linking security incidents to financial risk outcomes.

REFERENCES

[1] P.C. Verhoef, T. Broekhuizen, Y. Bart, A. Bhattacharya, J.Q. Dong, N. Fabian, et al., "Digital transformation: A multidisciplinary reflection and research agenda," J. Bus. Res., vol. 122, pp. 889–901, 2021, doi: 10.1016/j.jbusres.2019.09.022.

[2] L. Cao, "AI in finance: Challenges, techniques, and opportunities," ACM Comput. Surv., vol. 55, no. 3, Art. no. 64, pp. 1–38, 2023, doi: 10.1145/3502289.

[3] J.W. Goodell, S. Kumar, W.M. Lim, and D. Pattnaik, "Artificial intelligence and machine learning in finance: Identifying foundations, themes, and research clusters from bibliometric analysis," J. Behav. Exp. Finance, vol. 32, Art. no. 100577, 2021, doi: 10.1016/j.jbef.2021.100577.

[4] P. Mikalef and M. Gupta, "Artificial intelligence capability: Conceptualization, measurement calibration, and empirical study on its impact on organizational creativity and firm performance," Inf. Manag., vol. 58, no. 3, Art. no. 103434, 2021, doi: 10.1016/j.im.2021.103434.

[5] D. Mhlanga, "Industry 4.0 in finance: The impact of artificial intelligence (AI) on digital financial inclusion," Int. J. Financ. Stud., vol. 8, no. 3, Art. no. 45, 2020, doi: 10.3390/ijfs8030045.

[6] T. Berg, V. Burg, A. Gombović, and M. Puri, "On the rise of FinTechs: Credit scoring using digital footprints," Rev. Financ. Stud., vol. 33, no. 7, pp. 2845–2897, 2020, doi: 10.1093/rfs/hhz099.

[7] T.M. Eisenbach, A. Kovner, and M.J. Lee, "Cyber risk and the U.S. financial system: A pre-mortem analysis," J. Financ. Econ., vol. 145, no. 3, pp. 802–826, 2022, doi: 10.1016/j.jfineco.2021.10.007.

[8] I. Aldasoro, L. Gambacorta, P. Giudici, and T. Leach, "The drivers of cyber risk," J. Financ. Stab., vol. 60, Art. no. 100989, 2022, doi: 10.1016/j.jfs.2022.100989.

[9] Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah, and F. Ahmad, "Network intrusion detection system: A systematic study of machine learning and deep learning approaches," Trans. Emerg. Telecommun. Technol., vol. 32, no. 1, Art. no. e4150, 2021, doi: 10.1002/ett.4150.

[10] A. Thakkar and R. Lohiya, "A survey on intrusion detection system: Feature selection, model, performance measures, application perspective, challenges, and future research directions," Artif. Intell. Rev., vol. 55, no. 1, pp. 453–563, 2022, doi: 10.1007/s10462-021-10037-9.

[11] A. Khraisat and A. Alazab, "A critical review of intrusion detection systems in the internet of things: Techniques, deployment strategy, validation strategy, attacks, public datasets and challenges," Cybersecurity, vol. 4, Art. no. 18, 2021, doi: 10.1186/s42400-021-00077-7.

[12] S. Deng, H. Zhao, W. Fang, J. Yin, S. Dustdar, and A.Y. Zomaya, "Edge intelligence: The confluence of edge computing and artificial intelligence," IEEE Internet Things J., vol. 7, no. 8, pp. 7457–7469, 2020, doi: 10.1109/JIOT.2020.2984887.

[13] S.S. Gill, M. Xu, C. Ottaviani, P. Patros, R. Bahsoon, et al., "AI for next generation computing: Emerging trends and future directions," Internet Things, vol. 19, Art. no. 100514, 2022, doi: 10.1016/j.iot.2022.100514.

[14] S.A. Rahman, H. Tout, C. Talhi, and A. Mourad, "Internet of Things intrusion detection: Centralized, on-device, or federated learning?," IEEE Netw., vol. 34, no. 6, pp. 310–317, 2020, doi: 10.1109/MNET.011.2000286.

[15] E.M. Campos, P.F. Saura, A. González-Vidal, J.L. Hernández-Ramos, J.B. Bernabé, G. Baldini, et al., "Evaluating federated learning for intrusion detection in Internet of Things: Review and challenges," Comput. Netw., vol. 203, Art. no. 108661, 2022, doi: 10.1016/j.comnet.2021.108661.

[16] B.R. Gunnarsson, S. vanden Broucke, B. Baesens, M. Óskarsdóttir, and W. Lemahieu, "Deep learning for credit scoring: Do or don't?," Eur. J. Oper. Res., vol. 295, no. 1, pp. 292–305, 2021, doi: 10.1016/j.ejor.2021.03.006.

[17] E. Dumitrescu, S. Hué, C. Hurlin, and S. Tokpavi, "Machine learning for credit scoring: Improving logistic regression with non-linear decision-tree effects," Eur. J. Oper. Res., vol. 297, no. 3, pp. 1178–1192, 2022, doi: 10.1016/j.ejor.2021.06.053.

[18] M. Bücker, G. Szepannek, A. Gosiewska, and P. Biecek, "Transparency, auditability, and explainability of machine learning models in credit scoring," J. Oper. Res. Soc., vol. 73, no. 1, pp. 70–90, 2022, doi: 10.1080/01605682.2021.1922098.

[19] N. Kozodoi, J. Jacob, and S. Lessmann, "Fairness in credit scoring: Assessment, implementation and profit implications," Eur. J. Oper. Res., vol. 297, no. 3, pp. 1083–1094, 2022, doi: 10.1016/j.ejor.2021.06.023.

[20] S. Gu, B. Kelly, and D. Xiu, "Empirical asset pricing via machine learning," Rev. Financ. Stud., vol. 33, no. 5, pp. 2223–2273, 2020, doi: 10.1093/rfs/hhaa009.

[21] J. Chassot and F. Audrino, "HARd to beat: The overlooked impact of rolling windows in the era of machine learning," Int. J. Forecast., vol. 42, no. 2, pp. 330–343, 2026, doi: 10.1016/j.ijforecast.2025.06.003.

[22] D. Arp, E. Quiring, F. Pendlebury, A. Warnecke, F. Pierazzi, C. Wressnegger, et al., "Dos and don'ts of machine learning in computer security," in Proc. 31st USENIX Security Symp., Boston, MA, USA, 2022, pp. 3971–3988.

[23] G. Apruzzese, P. Laskov, E. Montes de Oca, W. Mallouli, L. Brdalo Rapa, A.V. Grammatopoulos, et al., "The role of machine learning in cybersecurity," Digit. Threats Res. Pract., vol. 4, no. 1, Art. no. 8, pp. 1–38, 2023, doi: 10.1145/3545574.

[24] G. Andresini, F. Pendlebury, F. Pierazzi, C. Loglisci, A. Appice, and L. Cavallaro, "INSOMNIA: Towards concept-drift robustness in network intrusion detection," in Proc. 14th ACM Workshop Artif. Intell. Secur. (AISec '21), 2021, pp. 111–122, doi: 10.1145/3474369.3486864.

[25] M. Sarhan, S. Layeghy, and M. Portmann, "Towards a standard feature set for network intrusion detection system datasets," Mobile Netw. Appl., vol. 27, no. 1, pp. 357–370, 2022, doi: 10.1007/s11036-021-01843-0.

[26] C. Nadeau and Y. Bengio, "Inference for the generalization error," Mach. Learn., vol. 52, no. 3, pp. 239–281, 2003, doi: 10.1023/A:1024068626366.

[27] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," J. Mach. Learn. Res., vol. 7, pp. 1–30, 2006.

[28] European Parliament and Council of the European Union, "Regulation (EU) 2024/1689 of 13 June 2024 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act)," Off. J. Eur. Union, L series, 12 July 2024.

[29] National Institute of Standards and Technology, "Artificial intelligence risk management framework (AI RMF 1.0)," NIST AI 100-1, Gaithersburg, MD, USA, 2023, doi: 10.6028/NIST.AI.100-1.

[30] I.H. Sarker, A.S.M. Kayes, S. Badsha, H. Alqahtani, P. Watters, and A. Ng, "Cybersecurity data science: An overview from machine learning perspective," J. Big Data, vol. 7, Art. no. 41, 2020, doi: 10.1186/s40537-020-00318-5.

[31] N. Moustafa, "A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets," Sustain. Cities Soc., vol. 72, Art. no. 102994, 2021, doi: 10.1016/j.scs.2021.102994.

[32] M.A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, and H. Janicke, "Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning," IEEE Access, vol. 10, pp. 40281–40306, 2022, doi: 10.1109/ACCESS.2022.3165809.

[33] N. Capuano, G. Fenza, V. Loia, and C. Stanzione, "Explainable artificial intelligence in cybersecurity: A survey," IEEE Access, vol. 10, pp. 93575–93600, 2022, doi: 10.1109/ACCESS.2022.3204171.

[34] H.B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. Agüera y Arcas, "Communication-efficient learning of deep networks from decentralized data," in Proc. 20th Int. Conf. Artif. Intell. Stat. (AISTATS), PMLR vol. 54, 2017, pp. 1273–1282.

[35] P. Kairouz, H.B. McMahan, B. Avent, A. Bellet, M. Bennis, A.N. Bhagoji, et al., "Advances and open problems in federated learning," Found. Trends Mach. Learn., vol. 14, no. 1–2, pp. 1–210, 2021, doi: 10.1561/2200000083.

[36] H. Zhu, J. Xu, S. Liu, and Y. Jin, "Federated learning on non-IID data: A survey," Neurocomputing, vol. 465, pp. 371–390, 2021, doi: 10.1016/j.neucom.2021.07.098.

[37] T. Li, A.K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in Proc. Mach. Learn. Syst. (MLSys), vol. 2, 2020, pp. 429–450.

[38] D. Kreuzberger, N. Kühl, and S. Hirschl, "Machine learning operations (MLOps): Overview, definition, and architecture," IEEE Access, vol. 11, pp. 31866–31879, 2023, doi: 10.1109/ACCESS.2023.3262138.

[39] S. Rose, O. Borchert, S. Mitchell, and S. Connelly, "Zero trust architecture," NIST Special Publication 800-207, National Institute of Standards and Technology, Gaithersburg, MD, USA, 2020, doi: 10.6028/NIST.SP.800-207.

[40] S. Lessmann, B. Baesens, H.-V. Seow, and L.C. Thomas, "Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research," Eur. J. Oper. Res., vol. 247, no. 1, pp. 124–136, 2015, doi: 10.1016/j.ejor.2015.05.030.

[41] R. Shwartz-Ziv and A. Armon, "Tabular data: Deep learning is not all you need," Inf. Fusion, vol. 81, pp. 84–90, 2022, doi: 10.1016/j.inffus.2021.11.011.

[42] L. Grinsztajn, E. Oyallon, and G. Varoquaux, "Why do tree-based models still outperform deep learning on typical tabular data?," in Adv. Neural Inf. Process. Syst. 35 (NeurIPS 2022), Datasets and Benchmarks Track, 2022.

[43] N. Bussmann, P. Giudici, D. Marinelli, and J. Papenbrock, "Explainable machine learning in credit risk management," Comput. Econ., vol. 57, no. 1, pp. 203–216, 2021, doi: 10.1007/s10614-020-10042-0.

[44] S.M. Lundberg, G. Erion, H. Chen, A. DeGrave, J.M. Prutkin, B. Nair, et al., "From local explanations to global understanding with explainable AI for trees," Nat. Mach. Intell., vol. 2, no. 1, pp. 56–67, 2020, doi: 10.1038/s42256-019-0138-9.

[45] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan, "A survey on bias and fairness in machine learning," ACM Comput. Surv., vol. 54, no. 6, Art. no. 115, pp. 1–35, 2021, doi: 10.1145/3457607.

[46] A. Fuster, P. Goldsmith-Pinkham, T. Ramadorai, and A. Walther, "Predictably unequal? The effects of machine learning on credit markets," J. Finance, vol. 77, no. 1, pp. 5–47, 2022, doi: 10.1111/jofi.13090.

[47] F. Corsi, "A simple approximate long-memory model of realized volatility," J. Financ. Econom., vol. 7, no. 2, pp. 174–196, 2009, doi: 10.1093/jjfinec/nbp001.

[48] M. Guidolin and G.F. Panzeri, "Forecasting the CBOE VIX and SKEW indices using heterogeneous autoregressive models," Forecasting, vol. 6, no. 3, pp. 782–814, 2024, doi: 10.3390/forecast6030040.

[49] H.M. Blom, P.E. de Lange, and M. Risstad, "Estimating value-at-risk in the EURUSD currency cross from implied volatilities using machine learning methods and quantile regression," J. Risk Financ. Manag., vol. 16, no. 7, Art. no. 312, 2023, doi: 10.3390/jrfm16070312.

[50] P.H. Kupiec, "Techniques for verifying the accuracy of risk measurement models," J. Deriv., vol. 3, no. 2, pp. 73–84, 1995, doi: 10.3905/jod.1995.407942.

[51] P.F. Christoffersen, "Evaluating interval forecasts," Int. Econ. Rev., vol. 39, no. 4, pp. 841–862, 1998, doi: 10.2307/2527341.

[52] M. Tavallaee, E. Bagheri, W. Lu, and A.A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in Proc. IEEE Symp. Comput. Intell. Secur. Def. Appl. (CISDA), 2009, pp. 1–6, doi: 10.1109/CISDA.2009.5356528.

[53] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., 2016, pp. 785–794, doi: 10.1145/2939672.2939785.

[54] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, et al., "LightGBM: A highly efficient gradient boosting decision tree," in Adv. Neural Inf. Process. Syst. 30 (NIPS 2017), 2017, pp. 3146–3154.

[55] F.X. Diebold and R.S. Mariano, "Comparing predictive accuracy," J. Bus. Econ. Stat., vol. 13, no. 3, pp. 253–263, 1995, doi: 10.1080/07350015.1995.10524599.

[56] D. Harvey, S. Leybourne, and P. Newbold, "Testing the equality of prediction mean squared errors," Int. J. Forecast., vol. 13, no. 2, pp. 281–291, 1997, doi: 10.1016/S0169-2070(96)00719-4.

[57] T. Su, H. Sun, J. Zhu, S. Wang, and Y. Li, "BAT: Deep learning methods on network intrusion detection using NSL-KDD dataset," IEEE Access, vol. 8, pp. 29575–29585, 2020, doi: 10.1109/ACCESS.2020.2972627.