



Explainable Federated Learning for Privacy-Preserving Cyberattack Detection and Resilient Energy Management in Distributed Smart Grids

Abdur Rashid¹;

[1]. Junior Project Coordinator, Virtual American Companies, LLC, Texas, USA;
M.S. Engineering Management, Westcliff University, USA
Email: rabdurrashid091@gmail.com

Doi: [10.63125/cvnp5r22](https://doi.org/10.63125/cvnp5r22)

Received: 12 May 2026; **Revised:** 11 June 2026; **Accepted:** 14 July 2026; **Published:** 02 August 2026;

Abstract

Distributed smart grids (feeders, microgrids and substations operated by different parties) generate the telemetry that could train cyberattack detectors, but the parties cannot pool it: it reveals customers and operations, and regulation forbids its transfer. Federated learning (FL) trains a shared detector without moving the data, differential privacy bounds what the shared model reveals, and explainable AI (XAI) tells an operator why an alarm was raised, which an energy-management system must know before it acts. This article designs a data-driven analytics framework that joins the three and evaluates it on a transparent simulation testbed: twelve heterogeneous grid clients; five attack classes (false-data injection, replay, denial of service, scaling and command injection) injected into 15-minute telemetry windows; and six training schemes (local, centralised, FedAvg, FedProx, FedAvg with record-level differential privacy, and robust aggregation under model-poisoning clients). On each client's own attack mix the schemes were within 0.03 AUROC of one another and local models were as good as any; on attacks a client had not seen, the federated and central detectors held their quality (AUROC 0.87–0.88) while local models fell to 0.74, with FedProx the better federated scheme under heterogeneity. Record-level differential privacy at single-digit ϵ erased all but the strongest attack signature with 700 windows per client; one poisoning client in twelve destroyed plain averaging while the coordinate-wise median withstood three. At cost-tuned thresholds federation saved 3.3 thousand per client-year over local detection on unseen attacks (positive in 71% of economic draws) and lost 6.6 thousand on seen ones; the attack rate and damage drive the value, poisoners and privacy noise reverse it. Shapley explanations of the federated and centralised detectors agree (rank correlation 0.72–0.79) and a four-rule surrogate reproduces 96% of the federated detector's decisions. All inputs are stylised with stated ranges; the testbed and results file accompany the article.

Keywords

Federated Learning, Differential Privacy, Explainable AI, Cyberattack Detection, False-Data Injection, Smart Grid, Resilience, Energy Management, Robust Aggregation, Decision Analytics

Introduction

The electricity system is becoming a federation of digital systems. Distribution feeders, microgrids, aggregators, charging hubs and prosumers each run their own telemetry, control and energy management, exchange a limited set of signals with the system operator, and are exposed (through SCADA, phasor measurement units, smart meters and the communication networks that connect them) to attacks that the engineering of the grid never anticipated (Mo et al., 2012; Sridhar et al., 2012; Yan et al., 2012; Wang & Lu, 2013). The threat is not hypothetical: the Ukrainian outages of 2015 and 2016 were caused by intrusions into distribution-utility networks and by malware written for grid protocols (Lee et al., 2016; ICS-CERT, 2016; Whitehead et al., 2017; Cherepanov, 2017; Slowik, 2019), and the literature on false-data injection has shown since Liu, Ning and Reiter that an attacker who knows the network model can corrupt state estimation without triggering the bad-data detectors that operators rely on (Liu et al., 2011; Liang et al., 2017; Deng et al., 2017; Musleh et al., 2020). Machine learning can detect such attacks from telemetry (Özay et al., 2016; Esmalifalak et al., 2017; He et al., 2017; Ashrafuzzaman et al., 2020), but the detectors that perform best are trained on large, diverse datasets, and the diverse data live with different parties who cannot share them: the telemetry reveals customers' consumption and operating practice (McDaniel & McLaughlin, 2009; Molina-Markham et al., 2010), data-protection and critical-infrastructure regulation restrict its transfer (European Parliament and Council, 2016; European Parliament and Council, 2022; NERC, n.d.), and no single feeder sees enough attacks of every kind to learn them all.

Federated learning is the architecture that answers this constraint. Clients train a shared model on their own data and exchange only model updates, which a server averages (McMahan et al., 2017; Kairouz et al., 2021; Li et al., 2020b; Yang et al., 2019); heterogeneity across clients is handled by proximal and control-variate methods (Li et al., 2020c; Karimireddy et al., 2020; Zhao et al., 2018); and what the updates themselves reveal can be bounded by differential privacy (Abadi et al., 2016; Dwork & Roth, 2014; Geyer et al., 2017; McMahan et al., 2018) or hidden by secure aggregation (Bonawitz et al., 2017). Applications to intrusion detection in the Internet of Things and to load forecasting and attack detection in smart grids have multiplied (Nguyen et al., 2019; Preuveneers et al., 2018; Rahman et al., 2020; Mothukuri et al., 2022; Agrawal et al., 2022; Campos et al., 2022; Taïk & Cherkaoui, 2020; Fekri et al., 2022; Li et al., 2022; Jithish et al., 2023; Husnoo et al., 2023). Two gaps remain between these studies and a deployable system. The first is trust in the alarm: an energy-management system that switches to a safe mode, curtails a resource or isolates a feeder on a detector's say-so needs to know what the detector saw, and the explainable-AI literature has shown both that post-hoc explanations can be produced for any model (Ribeiro et al., 2016; Lundberg & Lee, 2017; Štrumbelj & Kononenko, 2014) and that they can be fragile or misleading (Ghorbani et al., 2019; Slack et al., 2020; Rudin, 2019); whether explanations of a federated detector agree with those of a centralised one, and across clients, has rarely been measured. The second gap is the decision: the literature reports detection metrics (AUROC, F1) but an operator deciding whether to join a federation, how much privacy noise to accept, and how to react to alarms needs the cost of missed attacks, false alarms and safe-mode operation under uncertainty, the drivers of that cost, and the value of the data that federation makes available (Axelsson, 2000; Sommer & Paxson, 2010; Arp et al., 2022).

This article designs a data-driven decision-analytics framework for explainable federated learning in distributed smart grids and evaluates it on a transparent simulation testbed. The framework, in the tradition of design-science research (Hevner et al., 2004; Peffers et al., 2007; Gregor & Hevner, 2013), has four layers: a physical and telemetry layer (feeders, microgrids, measurements, attacks), a federated learning and privacy layer (local training, aggregation, differential privacy, robust aggregation), an explanation layer (Shapley attributions, surrogate rules, agreement checks) and a decision and resilience layer (energy-management rules, costs of errors, Monte Carlo economics, governance). The testbed has twelve clients with heterogeneous loads, photovoltaic shares, network sensitivities, noise and attack mixes; 15-minute telemetry windows with 48 features derived from bus voltages, line flows, net load, photovoltaic output, frequency, packet rates, latency and command counts; five attack classes injected into 8% of windows; and six training schemes evaluated with three seeds on held-out windows: local models, a centralised model on pooled data (the privacy-violating reference), FedAvg, FedProx, FedAvg with record-level differential privacy at three noise levels, and FedAvg with coordinate-wise

median and trimmed-mean aggregation under one to three model-poisoning clients. The testbed uses no measured data; its purpose is to show how the framework's outputs behave (which scheme detects what, at what privacy cost, with what explanation stability, and what that is worth to an energy-management decision) so that an operator can replace the ranges with its own data. The testbed and the complete results file accompany the article.

Four research questions structure the study. RQ1: what do the smart-grid security, federated learning, differential privacy, explainable AI and resilience literatures require of a decision framework for federated attack detection, and what framework meets them? RQ2: how do local, centralised, federated, differentially private and robust training schemes compare on detection performance by attack class, false-alarm rate, calibration and communication cost, and how does the comparison change with data heterogeneity and poisoning? RQ3: do explanations of the federated detector agree with those of the centralised detector and across clients, and can a readable surrogate reproduce its decisions? RQ4: what are the costs of missed attacks, false alarms and safe-mode operation under uncertainty, which inputs drive them, and what do the results imply for federation, privacy budgets, standards and governance? The results support a conditional answer. On the testbed, a client's own detector is as good as any on the attacks in its own history, and federation pays only on the attacks the client has not seen (where local models lose a third of their recall and federated ones keep theirs) so that the value of federation rises with a party's attack exposure and is reversed by poisoning clients and by privacy noise; record-level differential privacy at the budgets the literature calls meaningful leaves only the strongest attack signature when local datasets are small; one dishonest client in twelve destroys plain averaging while a median aggregation withstands three; the federated detector's explanations name the same physical signatures as a centralised one's and reduce to four readable rules; and at the default alarm threshold every detector costs more in false alarms than it saves, so that threshold tuning against the operator's cost model is part of the detector. The limitation to state at the outset is that every input is stylised (the attack signatures, the telemetry model and the costs of errors are representations of what the literature describes, not measurements of any grid) and the article reports the sensitivity of each conclusion to the ranges assumed.

Framing the problem as decision analytics rather than as detection alone changes what the analysis has to produce. The analytics literature distinguishes prediction from decision and shows that connecting models to the decisions they serve matters more than marginal accuracy (Bertsimas & Kallus, 2020; Elmachoub & Grigas, 2022); the intrusion-detection literature has long warned that base rates make false alarms the binding constraint (Axelsson, 2000) and that machine-learning detectors fail in deployment for reasons that offline metrics hide (Sommer & Paxson, 2010; Arp et al., 2022); the behavioural literature warns that single-run results anchor judgement (Tversky & Kahneman, 1974; Kahneman, 2011); and the resilience literature insists that the question is not whether an attack is detected but how the system's performance degrades and recovers (Linkov et al., 2013; Ganin et al., 2016; Panteli & Mancarella, 2015; Bie et al., 2017). A framework for federated attack detection therefore has to state its data and their provenance, evaluate with seeds and held-out windows against local and centralised references, bound what the shared model reveals, explain what it detects and check that the explanation is stable, propagate error costs into probabilities and ranges (Spiegelhalter, 2017), and record the decisions and their basis (ISO, 2018; National Institute of Standards and Technology, 2024). The remainder of the article reviews the literature and derives the requirements (RQ1), describes the testbed, the training schemes, the explanation methods and the evaluation (Methods), reports detection, privacy, robustness, explanation and economic results (RQ2–RQ4), and discusses implications, analytics practice, limitations and further research. Twelve figures and fifteen tables carry the detail.

Background and Related Work

Cyber-physical security of smart grids

The security problem of the grid is cyber-physical: attacks on measurements and commands have physical consequences, and physical models can detect attacks that network monitoring cannot (Cárdenas et al., 2011; Pasqualetti et al., 2013; Giraldo et al., 2018; Urbina et al., 2016; Humayed et al., 2017). The canonical attack is false-data injection against state estimation (Schweppe & Wildes, 1970; Abur & Gómez Expósito, 2004): an attacker who knows the measurement matrix can add a vector that

passes the residual test (Liu et al., 2011), with variants that need only partial knowledge, that target markets or that are coordinated across meters (Kosut et al., 2011; Kim & Poor, 2011; Giani et al., 2013; Cui et al., 2012); reviews catalogue attacks, impacts and defences (Liang et al., 2017; Deng et al., 2017; Musleh et al., 2020). Beyond state estimation, the threat landscape of industrial control systems and smart-grid communications includes replay, denial of service, command injection and protocol abuse (McLaughlin et al., 2016; Yan et al., 2012; Wang & Lu, 2013; Galloway & Hancke, 2013; Mackiewicz, 2006; Kumar et al., 2015), documented in the Stuxnet, Ukraine and Industroyer cases (Langner, 2011; Falliere et al., 2011; Lee et al., 2016; Whitehead et al., 2017; Cherepanov, 2017; Slowik, 2019) and codified in standards and guidance, NIST SP 800-82 for operational-technology security, NISTIR 7628 for smart-grid cybersecurity, IEC 62443 for industrial automation, the NERC CIP standards for bulk-system assets, and the NIST Cybersecurity Framework and the EU NIS 2 Directive at the governance level (Stouffer et al., 2023; Smart Grid Interoperability Panel, 2014; International Electrotechnical Commission, 2023; NERC, n.d.; National Institute of Standards and Technology, 2024; European Parliament and Council, 2022). Privacy is the other half of the problem: smart-meter data reveal household activity (McDaniel & McLaughlin, 2009; Molina-Markham et al., 2010), and the General Data Protection Regulation and sector rules restrict their transfer (European Parliament and Council, 2016).

Machine learning for attack detection

Machine learning has been applied to attack detection in grids and control systems from several angles: supervised classifiers for false-data injection (Özay et al., 2016; Esmalifalak et al., 2017; He et al., 2017; Ashrafuzzaman et al., 2020; Zhang et al., 2021; Wang et al., 2018), data mining on the Mississippi State power-system disturbance dataset, which established the task of discriminating disturbances from cyber-attacks (Pan et al., 2015; Borges Hink et al., 2014), anomaly detection on physical process data (Goh et al., 2017; Kravchik & Shabtai, 2018; Inoue et al., 2017), and the broad intrusion-detection literature with its benchmark datasets and surveys (Buczak & Guven, 2016; Ferrag et al., 2020; Ahmed et al., 2016; Chandola et al., 2009; Sharafaldin et al., 2018; Moustafa & Slay, 2015; Koroniotis et al., 2019; Meidan et al., 2018; Alsaedi et al., 2020). Three cautions from that literature shape this study. The base-rate fallacy: when attacks are rare, even a small false-positive rate produces alarms that dwarf the true detections (Axelsson, 2000). The closed-world problem: detectors trained and tested in the laboratory fail on deployment because of distribution shift, data snooping and unrealistic evaluation (Sommer & Paxson, 2010; Arp et al., 2022; Kaufman et al., 2012). And imbalance and calibration: rare classes need weighting or resampling, and detector scores must be calibrated if they are to be acted on as probabilities (He & Garcia, 2009; Chawla et al., 2002; Guo et al., 2017; Niculescu-Mizil & Caruana, 2005; Saito & Rehmsmeier, 2015; Davis & Goadrich, 2006). The AI-based threat landscape also runs the other way: attackers use machine learning too (Kaloudi & Li, 2020; Mirsky et al., 2023).

Model Generalization and Real-World Deployment

Despite significant advances in machine learning-based attack detection, ensuring reliable performance across diverse operational environments remains a major challenge (Aminul Islam & Md, 2023; Jahanara, 2025; Syed Feroz, 2024). Many existing detection models demonstrate high accuracy when evaluated using benchmark datasets but experience reduced effectiveness when deployed in real-world power grids, industrial control systems, and other critical infrastructure environments (Jannatul, 2024; Mansura Akter & Sumaiya, 2022; Sumaiya, 2026). This performance gap is often attributed to differences in network configurations, evolving attack patterns, variations in operational conditions, and changes in data distributions over time (Ahmed, 2026; Md Ashfaq & Abdullah Mohammad, 2022; Nishat, 2023). Furthermore, models trained using controlled laboratory datasets may struggle to identify previously unseen attacks or distinguish malicious activities from legitimate operational disturbances (Abdullah Mohammad, 2023; Sumaiya, 2025; Tahmedur & Rifat, 2023). The complexity of modern interconnected infrastructures introduces additional challenges because communication patterns, sensor measurements, and system behavior can change dynamically in response to equipment failures, maintenance activities, renewable energy fluctuations, and external disturbances (Jahanara & Mst Kaniz, 2023; Md Asif Ali Sheak & Tahmedur, 2023; Md Mahamud Pervaz, 2025). Such variations may increase false-positive rates and reduce the effectiveness of conventional classification approaches, particularly when detection models rely heavily on predefined attack signatures or historical training patterns (Asheke, 2024; Ashik & Shabbir, 2022; Md Anisur Rahman, 2023).

Addressing these limitations requires the development of machine learning frameworks capable of maintaining consistent detection performance under varying operational conditions (Mashfiqu, 2026; Md Arif Uz et al., 2025; Stabak & Md Shahid, 2022). Adaptive learning, transfer learning, domain adaptation, and continuous model evaluation offer potential approaches for improving generalization and reducing dependence on static training datasets (Gazi Yeasin, 2023; Md Kaium, 2023; Sumaiya, 2024). In addition, incorporating diverse data sources, including network traffic, physical sensor measurements, and operational control signals, may provide a more comprehensive representation of system behavior and improve the identification of coordinated cyber-physical attacks (Ashrafuzzaman, 2024; Jahanara, 2026; Maishar Mostari, 2025). However, integrating heterogeneous information introduces challenges related to data synchronization, computational complexity, feature selection, and system scalability (Mashiat, 2025; Md Mahamud Pervaz, 2023; Sadia & Maishar Mostari, 2022). Detection frameworks must also balance accuracy with response time, particularly in environments where delayed identification of malicious activities can affect operational safety and service continuity (Aminul Islam, 2024; Mst Kaniz & Jahanara, 2022; Syed Feroz, 2025). Therefore, future research should emphasize realistic testing environments, cross-dataset validation, and robust performance evaluation under distribution shifts, imbalanced attack scenarios, and evolving adversarial strategies. These considerations are essential for translating laboratory-based machine learning achievements into practical, scalable, and dependable cybersecurity solutions.

Adaptive Machine Learning for Cybersecurity

Another important direction in machine learning-based attack detection involves developing explainable and adaptive detection frameworks capable of responding effectively to increasingly sophisticated cyber threats (Ahmed, 2024; Kynatun & Saba, 2024; Mashiat & Md Anisur Rahman, 2022). Although advanced machine learning and deep learning algorithms can identify complex relationships within high-dimensional sensor measurements, communication records, and operational data, their limited interpretability may reduce the confidence of system operators in automated security decisions. This limitation is particularly significant in smart grids and industrial control systems, where security alerts may trigger operational interventions that influence equipment performance, system stability, and the continuity of essential services (Asheke, 2023; Md & Gazi Yeasin, 2022; Mohammad Badrul & Md. Mominul, 2023). Traditional black-box models often provide limited information about the specific features, measurements, or behavioral deviations responsible for their predictions, making it difficult for operators to distinguish genuine malicious activities from abnormal but legitimate operating conditions (Mashfiqu & Md, 2022; Md Imran & Jannatul, 2022; Rony & Md, 2026). Explainable artificial intelligence techniques can address this concern by identifying influential input variables, illustrating relationships between operational anomalies and attack indicators, and generating interpretable explanations for classification outcomes (Adnan & Iftekhar, 2023; Mashfiqu, 2025; Nuzhat Sadia, 2023). Such capabilities may strengthen human oversight, improve incident investigation, and support more transparent cybersecurity decision-making across interconnected infrastructure systems (Ahmed & Nuzhat Sadia, 2023; Jannatul, 2026; Syed Feroz, 2023).

Furthermore, the growing sophistication of cyberattacks highlights the importance of integrating adaptability, adversarial robustness, and uncertainty estimation into machine learning-based detection systems (Kazi Rakib Hasan & Adnan, 2022; Md. Morshedul & Aminul Islam, 2022; Stabak & Md Siam, 2023). Attackers may deliberately manipulate input measurements, exploit weaknesses in classification boundaries, or develop previously unseen attack strategies to evade conventional detection mechanisms. Static machine learning models may therefore become less effective as threat characteristics and operational environments evolve (Jannatul, 2025; Mashiat, 2024; Syed Feroz & Md Mahamud Pervaz, 2022). Adaptive frameworks incorporating online learning, incremental model updates, and feedback-driven optimization may provide opportunities to maintain detection effectiveness without requiring complete model retraining after every operational change (Abdullah Mohammad, 2024; Md Imran & Jannatul, 2021; Shaiful et al., 2022).

Federated learning: optimisation, heterogeneity and systems

Federated averaging trains a shared model by iterating local stochastic-gradient steps on each client and weighted averaging of the resulting models (McMahan et al., 2017); the field's open problems (heterogeneity, communication, privacy, robustness, fairness) are catalogued in a comprehensive

survey (Kairouz et al., 2021), with shorter treatments of the challenges and of the horizontal, vertical and transfer settings (Li et al., 2020b; Yang et al., 2019). Statistical heterogeneity (non-identically distributed client data) degrades FedAvg (Zhao et al., 2018; Hsu et al., 2019; Sattler et al., 2020), and is addressed by a proximal term that keeps local models near the global one (FedProx) (Li et al., 2020c), by control variates (SCAFFOLD) (Karimireddy et al., 2020) and by adaptive server optimisers (Reddi et al., 2021). Systems work has shown federated learning at scale and in edge networks (Bonawitz et al., 2019; Lim et al., 2020; Wang et al., 2019; Beutel et al., 2020), and surveys map federated learning onto the Internet of Things and smart-grid settings (Nguyen et al., 2021; Cheng et al., 2022).

Privacy, attacks on federated learning and robust aggregation

Federated learning does not by itself protect privacy: gradients leak training data (Zhu et al., 2019; Geiping et al., 2020), membership and property inference succeed against both centralised and federated models (Shokri et al., 2017; Nasr et al., 2019; Melis et al., 2019), and surveys of security and privacy in federated learning catalogue the threats (Mothukuri et al., 2021; Lyu et al., 2024). Differential privacy is the formal remedy (Dwork et al., 2006; Dwork & Roth, 2014): DP-SGD clips per-example gradients and adds Gaussian noise, with a moments accountant to track the privacy loss (Abadi et al., 2016); client-level variants protect whole clients' contributions (Geyer et al., 2017; McMahan et al., 2018); the trade-off between privacy and utility has been analysed for federated learning (Wei et al., 2020); and hybrid designs combine differential privacy with secure multi-party computation (Truex et al., 2019; Bonawitz et al., 2017). Robustness is the complementary concern: a malicious client can poison the global model or insert a backdoor (Bagdasaryan et al., 2020; Fang et al., 2020), and Byzantine-robust aggregation rules (Krum, coordinate-wise median and trimmed mean) limit the damage at a cost in statistical efficiency (Blanchard et al., 2017; Yin et al., 2018).

Federated learning for intrusion detection and smart grids

Federated intrusion detection began with device-type-specific anomaly models for the Internet of Things (Nguyen et al., 2019) and chained anomaly detectors (Preuveneers et al., 2018), and has been compared with centralised and on-device learning (Rahman et al., 2020) and surveyed for its concepts, challenges and evaluation practice (Agrawal et al., 2022; Campos et al., 2022; Mothukuri et al., 2022). In smart grids, federated learning has been applied to load forecasting from smart-meter data (Taïk & Cherkaoui, 2020; Fekri et al., 2022; Badr et al., 2023), consumer-characteristic identification (Wang et al., 2021), secure and efficient edge-cloud training (Su et al., 2022), false-data-injection detection (Li et al., 2022), distributed anomaly detection (Jithish et al., 2023) and disturbance-versus-attack discrimination on the power-system dataset (Husnoo et al., 2023). The common findings are that federated detectors approach centralised performance when data are reasonably homogeneous, that heterogeneity and small local datasets open a gap that proximal methods partly close, that differential privacy costs accuracy, and that robustness to poisoning requires explicit aggregation rules. What these studies do not provide is the explanation and decision layer: whether the federated detector's reasons agree with a centralised one's, and what its operating point is worth.

Explainable AI for security and energy

Post-hoc explanation methods attribute a prediction to features: LIME fits a local linear surrogate (Ribeiro et al., 2016), Shapley values from cooperative game theory give an axiomatic attribution computed by sampling or by the SHAP family (Shapley, 1953; Štrumbelj & Kononenko, 2014; Lundberg & Lee, 2017; Lundberg et al., 2020), integrated gradients attribute along a path (Sundararajan et al., 2017), counterfactual explanations say what would have to change (Wachter et al., 2018), and permutation and accumulated-local-effect methods summarise global behaviour (Fisher et al., 2019; Apley & Zhu, 2020; Molnar, 2025). Surveys organise the field and its evaluation (Guidotti et al., 2018; Adadi & Berrada, 2018; Barredo Arrieta et al., 2020; Doshi-Velez & Kim, 2017; Lipton, 2018; Miller, 2019), and the case for inherently interpretable models in high-stakes decisions is made by Rudin (Rudin, 2019). Two warnings apply directly to security: explanations of neural networks are fragile to small input perturbations (Ghorbani et al., 2019), and post-hoc explainers can be fooled by an adversarial model (Slack et al., 2020). Explainable intrusion detection has its own literature of decision-tree explanations for trust management, explainable frameworks for intrusion-detection systems, surveys of explainable AI in cybersecurity and a systematisation of explainable machine learning for security (Mahbooba et al., 2021; Wang et al., 2020; Capuano et al., 2022; Neupane et al., 2022; Nadeem

et al., 2023), which emphasises that the audience is an analyst who must act, and that explanation stability matters as much as fidelity.

Resilience and energy management under attack

Power-system resilience (the ability to anticipate, absorb, adapt to and recover from disruption) has conceptual frameworks and metrics (Panteli & Mancarella, 2015; Bie et al., 2017; Arghandeh et al., 2016; Wang et al., 2016; Linkov et al., 2013; Ganin et al., 2016), with microgrids and distribution-system restoration as the operational levers (Chen et al., 2017; Olivares et al., 2014; Parhizi et al., 2015; Hatziargyriou et al., 2007) and home and building energy-management systems as the distributed controllers whose decisions an attack can corrupt (Beaudin & Zareipour, 2015; Zhou et al., 2016). Attack detection matters to resilience through the energy-management response: a detected attack can trigger a safe mode (falling back to last-good state, local measurements and conservative setpoints) whose cost is the price of resilience, while a missed attack lets corrupted data drive dispatch, and a false alarm imposes the safe-mode cost needlessly (Cárdenas et al., 2011; Pasqualetti et al., 2013; Zhang et al., 2023). The incident-response and risk-management standards frame these as decisions with recorded rationale (Cichonski et al., 2012; Joint Task Force Transformation Initiative, 2012; International Organization for Standardization, 2022; ISO, 2018; Kaplan & Garrick, 1981; Hubbard & Seiersen, 2016).

Decision analytics and interpretable learning

The management-science literature supplies the framework's remaining requirements: prescriptive analytics connects models to decisions (Bertsimas & Kallus, 2020; Elmachtoub & Grigas, 2022; Ban & Rudin, 2019); aleatoric uncertainty (which attacks will come) is distinguished from epistemic uncertainty (what they cost, how good the detector is) because only the second can be bought down (Hüllermeier & Waegeman, 2021; Kendall & Gal, 2017); risk is communicated as probabilities and ranges (Spiegelhalter, 2017); flexible models are made readable with importance, partial dependence and shallow trees (Friedman, 2001; Ke et al., 2017; Lou et al., 2012; Caruana et al., 2015); human judgement is supported rather than replaced (Amershi et al., 2019; Shneiderman, 2020; Fildes et al., 2009); variance-based sensitivity analysis and value of information connect uncertainty to decisions (Sobol', 2001; Saltelli et al., 2010; Herman & Usher, 2017; Howard, 1966); simulation studies follow conventions of verification and common random numbers (Law, 2015; Sargent, 2013); evaluation of learned models respects seeds, held-out data and the dependence between folds (Demšar, 2006; Nadeau & Bengio, 2003; Bergmeir & Benítez, 2012; Henderson et al., 2018); and analytics capability pays through organisational routines and data quality (Mikalef et al., 2019; Brynjolfsson et al., 2021; Hazen et al., 2014; Haug et al., 2011).

Requirements and the framework

Six requirements follow. R1, data that stay where they are: the detector must be trained without transferring telemetry, because the telemetry reveals customers and operations and its transfer is restricted. R2, heterogeneity and rarity as the normal case: clients differ in loads, networks and attack exposure, and most see few examples of most attacks, so the scheme must be evaluated under non-identical data and label skew. R3, bounded disclosure and bounded damage: what the shared model reveals must be quantified by a privacy budget, and what a malicious client can do to it must be limited by the aggregation rule. R4, explanations that an operator can check: attributions per attack class and per client, their agreement across schemes and clients, and a readable surrogate. R5, decision outputs: costs of missed attacks, false alarms and safe-mode operation under uncertainty, their drivers, and the value of federation over local and centralised alternatives. R6, feedback and governance: measured alarms, incidents and explanation drift must return to retraining, and the data provenance, privacy parameters, aggregation rules and decisions must be recorded. From these, the framework (Figure 1) is designed as four layers (the physical and telemetry layer, the federated learning and privacy layer, the explanation layer and the decision and resilience layer) with feedback from governance to training and from economics to the privacy budget. Figure 2 shows how the federated schemes, the privacy mechanism, the aggregation rules and the explanation methods fit together; the Methods give the models. Table 1 organises the literature into nine strands and the gap each leaves.

Table 1: Literature strands, what they establish, representative sources and the gap addressed

Strand	What the literature establishes	Representative sources	Gap addressed here
Cyber-physical security of smart grids	Attacks on measurements and commands have physical consequences; false-data injection can pass bad-data detection; documented incidents (Stuxnet, Ukraine, Industroyer); standards for OT and smart-grid security	Mo et al. (2012); Sridhar et al. (2012); Liu et al. (2011); Liang et al. (2017); Deng et al. (2017); Musleh et al. (2020); Lee et al. (2016); Cherepanov (2017); Stouffer et al. (2023); Smart Grid Interoperability Panel (2014)	Five attack classes injected into heterogeneous telemetry with known ground truth
Machine learning for attack detection	Supervised and anomaly detectors for FDIA, disturbances and process attacks; benchmark datasets; the base-rate fallacy, the closed-world problem and calibration	Özay et al. (2016); He et al. (2017); Pan et al. (2015); Borges Hink et al. (2014); Goh et al. (2017); Axelsson (2000); Sommer & Paxson (2010); Arp et al. (2022); Guo et al. (2017)	Detection evaluated by class, false-alarm rate and calibration, with seeds and held-out windows
Federated learning: optimisation and heterogeneity	FedAvg; open problems; non-IID degradation and remedies (FedProx, SCAFFOLD, adaptive optimisers); systems at scale and at the edge	McMahan et al. (2017); Kairouz et al. (2021); Li et al. (2020b, 2020c); Zhao et al. (2018); Karimireddy et al. (2020); Reddi et al. (2021); Bonawitz et al. (2019); Lim et al. (2020)	Local, central, FedAvg and FedProx compared under a heterogeneity sweep and label skew
Privacy, attacks on FL and robust aggregation	Gradient leakage and inference attacks; differential privacy and DP-SGD with a moments accountant; client-level DP; secure aggregation; poisoning and backdoors; Byzantine-robust rules	Zhu et al. (2019); Geiping et al. (2020); Shokri et al. (2017); Nasr et al. (2019); Dwork & Roth (2014); Abadi et al. (2016); Geyer et al. (2017); Bonawitz et al. (2017); Bagdasaryan et al. (2020); Fang et al. (2020); Blanchard et al. (2017); Yin et al. (2018)	A privacy-utility curve with accounted ϵ and a poisoning experiment with median and trimmed-mean aggregation
Federated learning for intrusion detection and grids	Federated IoT anomaly detection; comparisons with central and on-device learning; federated load forecasting, FDIA detection, anomaly detection and disturbance discrimination in grids	Nguyen et al. (2019); Preuveneers et al. (2018); Rahman et al. (2020); Agrawal et al. (2022); Campos et al. (2022); Taïk & Cherkaoui (2020); Fekri et al. (2022); Li et al. (2022); Jithish et al. (2023); Husnoo et al. (2023)	Explanation and decision layers added to the federated detector; value of federation measured
Explainable AI for security	LIME, Shapley attributions, integrated gradients, counterfactuals, surrogates; fragility of explanations; explainable intrusion detection and its analyst audience	Ribeiro et al. (2016); Lundberg & Lee (2017); Štrumbelj & Kononenko (2014); Sundararajan et al. (2017); Wachter et al. (2018); Rudin (2019); Ghorbani et al. (2019); Slack et al. (2020); Capuano et al. (2022); Nadeem et al. (2023)	Agreement of explanations across schemes and clients, and a surrogate tree with fidelity
Resilience and energy management under attack	Resilience frameworks and metrics; microgrids and restoration; energy-management systems as the controllers an attack corrupts; safe-mode responses	Panteli & Mancarella (2015); Bie et al. (2017); Arghandeh et al. (2016); Linkov et al. (2013); Ganin et al. (2016); Olivares et al. (2014); Beaudin & Zareipour (2015); Zhou et al. (2016); Cárdenas et al. (2011); Pasqualetti et al. (2013)	A safe-mode rule whose costs of missed attacks, false alarms and safe-mode operation are priced
Standards, regulation and governance	OT security guidance, smart-grid cybersecurity guidelines, IEC 62443, NERC CIP, NIST CSF 2.0, NIS 2, GDPR; incident response and risk management	Stouffer et al. (2023); Smart Grid Interoperability Panel (2014); IEC (2023); NERC (n.d.); NIST (2024); European Parliament and Council (2016, 2022); Cichonski et al. (2012); ISO (2018)	Governance record of data provenance, privacy budget, aggregation rule and decisions
Decision analytics and interpretable learning	Prescriptive analytics; aleatoric vs epistemic uncertainty; risk communication; interpretable models; Sobol sensitivity; simulation and evaluation conventions; analytics capability	Bertsimas & Kallus (2020); Elmachoutb & Grigas (2022); Hüllermeier & Waegeman (2021); Spiegelhalter (2017); Rudin (2019); Sobol' (2001); Saltelli et al. (2010); Law (2015); Sargent (2013); Demšar (2006); Mikalef et al. (2019)	Costs under uncertainty, Sobol indices and a readable metamodel as the framework's outputs

Figure 1: Explainable federated learning decision-analytics architecture for distributed smart grids: physical and telemetry layer, federated learning and privacy layer, explanation layer, and decision and resilience layer, with feedback loops

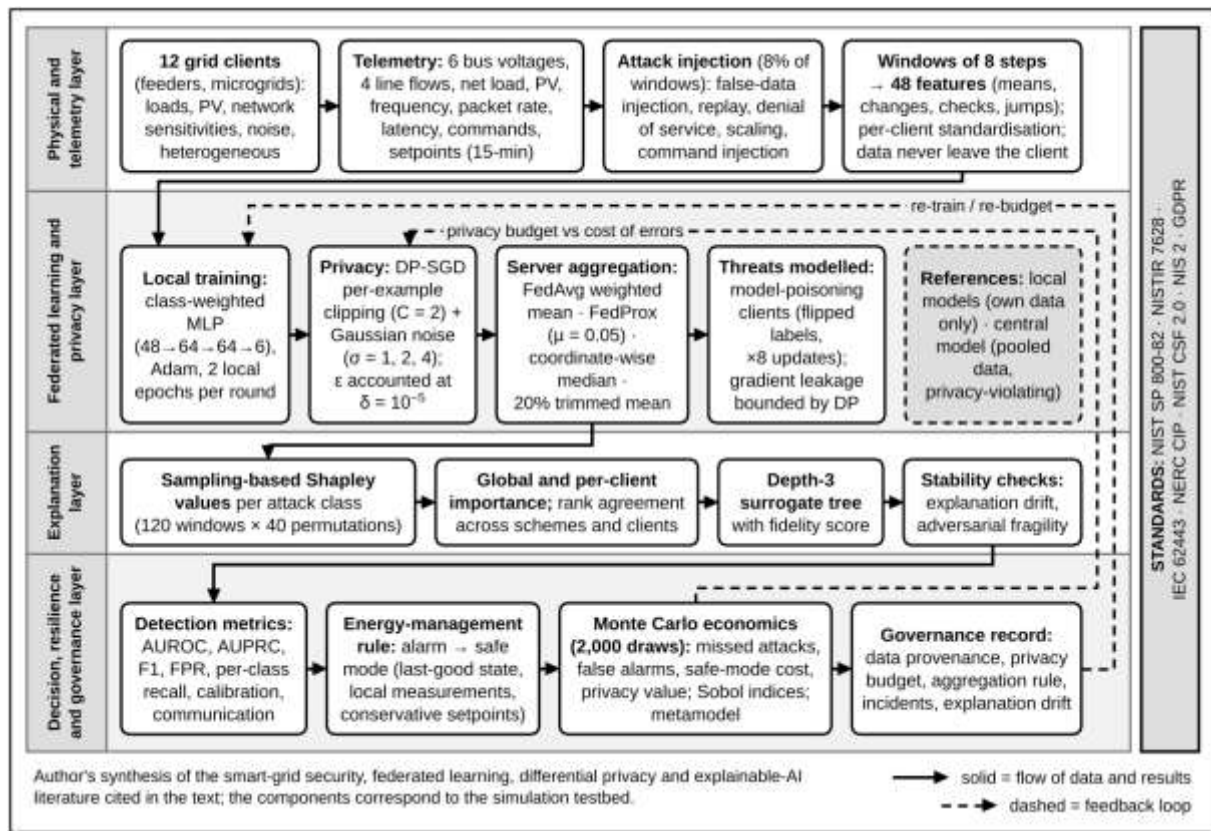
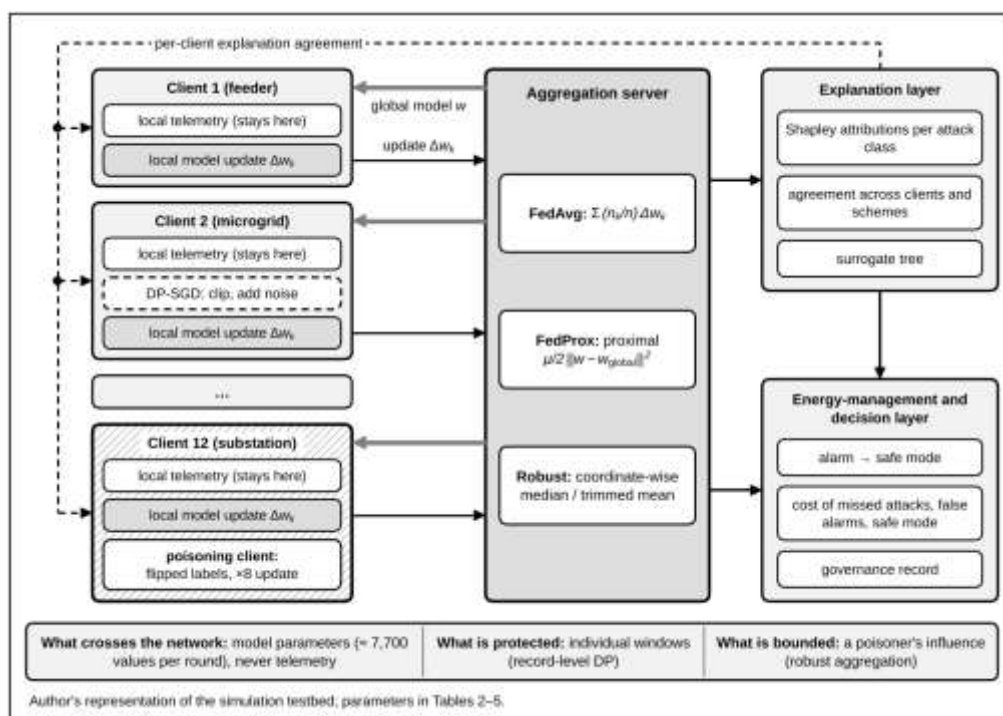


Figure 2: Federated training schematic: clients with local telemetry and local training, the server's aggregation rules (weighted mean, median, trimmed mean), the record-level differential-privacy mechanism, poisoning clients, and the explanation and energy-management layers

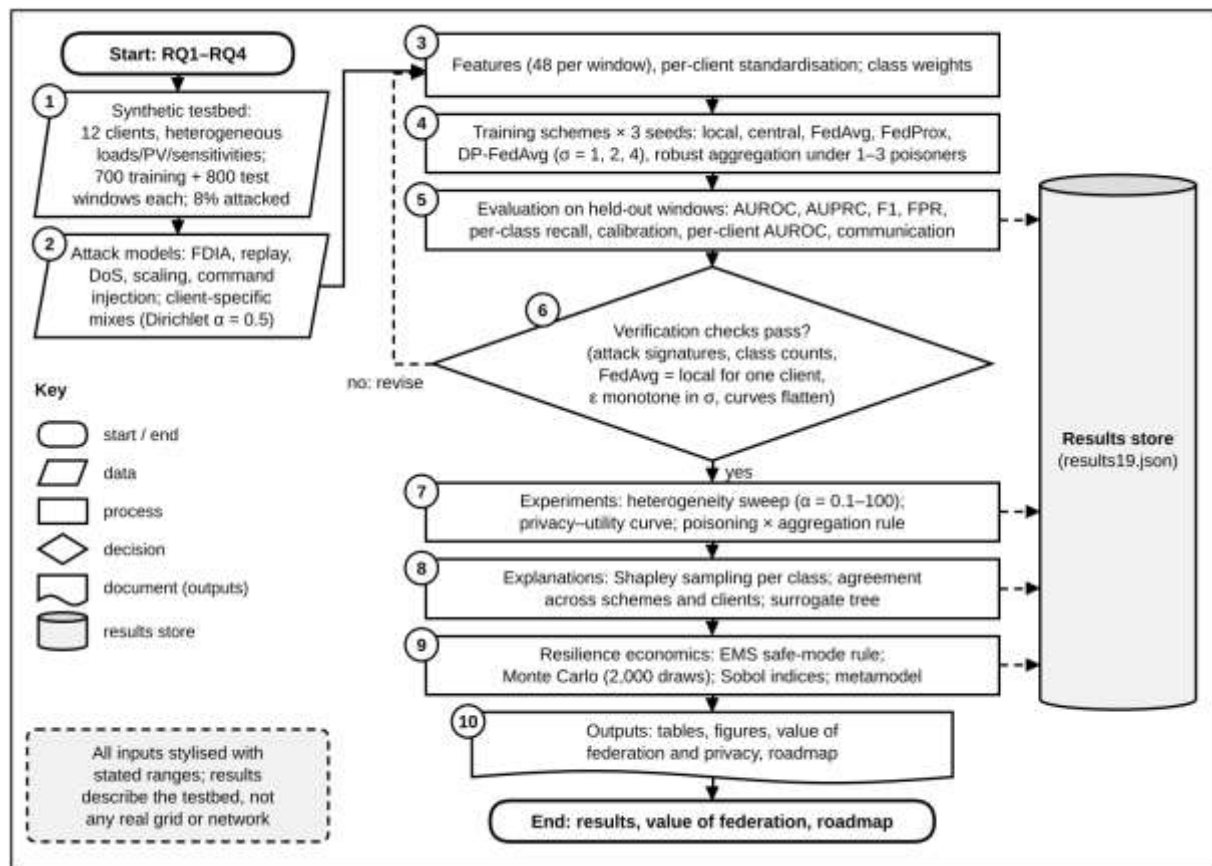


Methods

Study design

The study follows the design-science process of problem and requirements, design of the artefact, evaluation and communication (Peffers et al., 2007; Hevner et al., 2004), with an artificial evaluation on a simulation testbed, chosen because the framework's outputs (detection by attack class under known ground truth, privacy budgets, explanation agreement, costs of errors) can only be checked where the truth is known (Venable et al., 2016; Sonnenberg & vom Brocke, 2012). Figure 3 shows the design as a flowchart: a synthetic testbed of twelve heterogeneous grid clients with injected attacks, six training schemes with three seeds, evaluation on held-out windows, verification, experiments on heterogeneity, privacy and poisoning, explanation and agreement analysis, resilience economics with Monte Carlo, Sobol indices and a metamodel, and outputs. The simulation follows the conventions of simulation studies (explicit assumptions, verification of components, common random numbers (the same held-out windows for every scheme within a seed) (Law, 2015; Sargent, 2013)) and the evaluation conventions of machine learning in security: held-out data, no leakage of test statistics into training, attention to base rates, seeds and interval estimates (Sommer & Paxson, 2010; Arp et al., 2022; Kaufman et al., 2012; Henderson et al., 2018). It is implemented in Python with NumPy (the detector is a small NumPy multilayer perceptron with hand-written backpropagation, so that per-example gradient clipping for differential privacy is explicit), with SALib, LightGBM and scikit-learn for the analysis (Harris et al., 2020; Herman & Usher, 2017; Ke et al., 2017; Pedregosa et al., 2011), runs in under half an hour on two cores, and writes a results file from which every table and figure is produced. No measured data from any grid, utility or network were used.

Figure 3: Study design as a flowchart: testbed and attack injection, training schemes, evaluation on held-out windows, verification, experiments, explanation analysis, resilience economics and metamodel, outputs



Testbed: clients, telemetry and attacks

Table 2 lists the testbed and Figure 4 shows an example client. Twelve clients (feeders or microgrids

operated by different parties) each produce a 15-minute telemetry stream of seventeen channels: six bus voltages, four line flows, net load, photovoltaic output, frequency, SCADA packet rate, latency, command count and setpoint deviation. Each client has fixed physical characteristics drawn once and shared by its training and test streams: a load scale of 0.6–1.6, a photovoltaic share of 0–60%, measurement noise of 0.4–2%, and voltage and flow sensitivities to net load (-0.02 to -0.08 pu per unit load and 1.0 to 0.25 per unit load, each perturbed by $\pm 20\%$) in the manner of a linearised distribution model (Farivar & Low, 2013; Baran & Wu, 1989); load follows evening and morning peaks with noise, photovoltaics a clear-sky shape scaled by a daily cloudiness factor (Pfenninger & Staffell, 2016). The stream is cut into windows of eight steps (two hours); each client has 700 training windows, 300 validation windows (used only to tune the alarm threshold) and 800 test windows drawn from its own attack mix, plus a second test stream of 800 windows with a uniform mix over the five classes (the attacks the client may never have seen) and 8% of windows carry an attack. The attack class is drawn from a client-specific mix sampled from a Dirichlet distribution with concentration 0.5 over the five classes, so that clients differ in which attacks they see and some see almost none of a class, the label skew that federated learning is meant to overcome (Zhao et al., 2018; Hsu et al., 2019); the concentration is varied (0.1, 2, 100) in the heterogeneity experiment. Table 3 gives the attack models, each a stylised representation of a documented class: false-data injection shifts voltages and flows coherently, as if net load had moved by 15–50% of the client's load scale, from the window's midpoint: a shift that passes a voltage–flow consistency check but not a comparison with the measured load, in the spirit of the stealthy attacks of Liu et al. (Liu et al., 2011; Liang et al., 2017); replay overwrites the last four to eight steps of the measurement channels with a segment recorded ten to fifty hours earlier (Mo et al., 2012; McLaughlin et al., 2016); denial of service cuts the packet rate by 40–95%, raises latency by 30–110 ms and freezes the measurements at their last value (Yan et al., 2012; Wang & Lu, 2013); scaling multiplies the load and photovoltaic channels by 0.85–1.25 (Deng et al., 2017); and command injection adds three to twelve commands and a setpoint deviation of ± 0.05 –0.2 in the second half of the window (Cherepanov, 2017; Slowik, 2019). From each window 48 features are computed: the means of the seventeen channels; seven derived checks (the residual of a crude local state check, voltage spread, flow balance, load change, photovoltaic change, packet-rate change and the hour of day); the first-to-last change of each channel; and seven variability and jump features (standard deviation of net load, frequency, command count and latency, and the largest step change in voltages, flows and packet rate). Features are standardised per client on its own training data, so no cross-client statistic is exchanged.

Table 2: Testbed: clients, telemetry channels, physical characteristics, windows, attack share and heterogeneity

Element	Model	Values (stylised)	Basis
Clients	12 feeders / microgrids operated by different parties; fixed physical characteristics drawn once per client	Load scale 0.6–1.6; PV share 0–60%; noise 0.4–2%	Author's stylisation; Farivar & Low (2013)
Telemetry channels (17)	Six bus voltages and four line flows from net load through linear sensitivities; net load; PV; frequency; packet rate; latency; command count; setpoint deviation	Voltage sensitivities -0.02 to -0.08 pu per unit load, flows 1.0–0.25 per unit load, each $\pm 20\%$ per client; load with evening and morning peaks; PV clear-sky $\times$ cloudiness 0.3–1.0	Baran & Wu (1989); Pfenninger & Staffell (2016); De La Ree et al. (2010)
Windows and features	Windows of 8 steps (2 h); 48 features: 17 channel means, 7 derived checks, 17 first-to-last changes, 7 variability and jump features; per-client z-scores on training data	700 training and 800 test windows per client	Chandola et al. (2009); Kaufman et al. (2012)
Attacks	Five classes injected into a share of windows; class drawn from a client-specific Dirichlet mix	Attack share 8%; Dirichlet concentration 0.5 (0.1, 2, 100 in the heterogeneity sweep)	Zhao et al. (2018); Hsu et al. (2019)
Heterogeneity	Loads, PV, sensitivities and noise differ by client; attack mixes differ (label skew); some clients see almost none of a class	See Figure 4(b) and 4(d)	Kairouz et al. (2021); Li et al. (2020b)
Seeds	The seed fixes client characteristics, data, initialisation and sampling; the same held-out windows serve every scheme within a seed	3 seeds (2 for the poisoning and heterogeneity experiments)	Law (2015); Henderson et al. (2018)

Figure 4: Testbed: (a) an example client's telemetry over a day with injected attacks, (b) attack-class mixes across the twelve clients, (c) feature-space separation of attack classes, (d) class counts per client

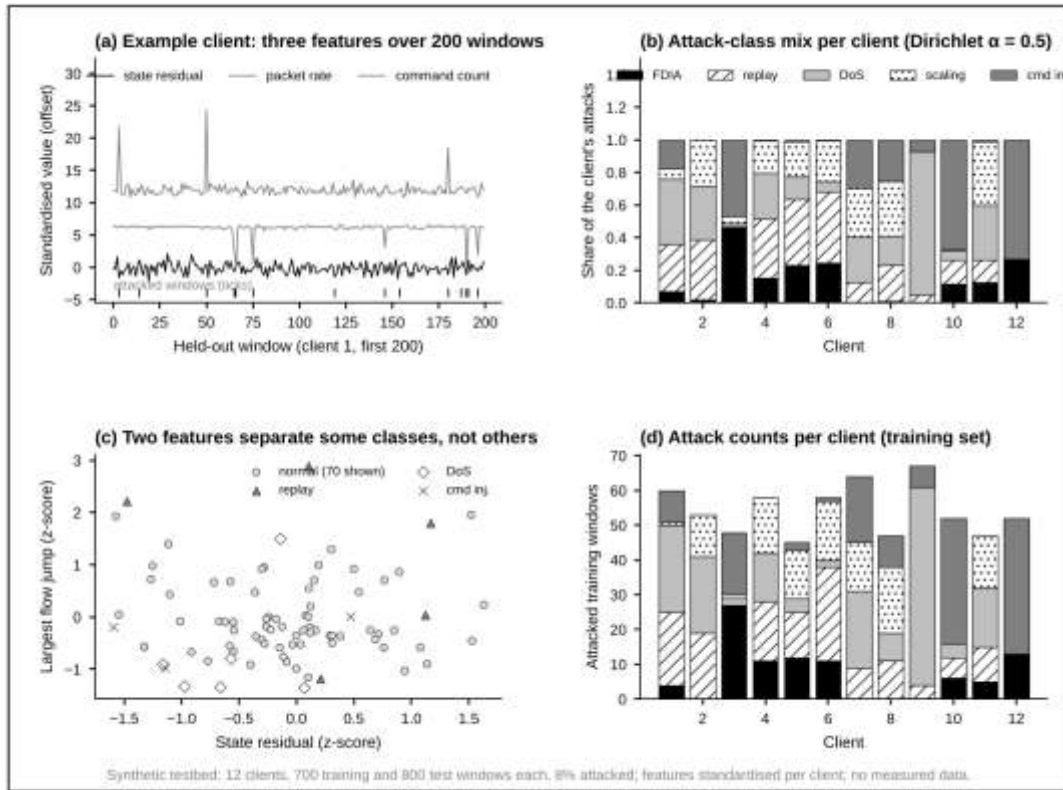


Table 3: Attack models (stylised): mechanism, affected channels, parameter ranges and the documented class each represents

Class	Mechanism in the testbed	Channels affected	Parameters	Documented class it represents
False-data injection	Coherent shift of voltages and flows as if net load had moved, from the window midpoint; passes a voltage-flow consistency check but not a comparison with the measured load	6 voltages, 4 flows	Shift $0.15\text{--}0.5 \times \text{load}$ scale, either sign	Liu et al. (2011); Liang et al. (2017); Deng et al. (2017)
Replay	Last steps of the measurement channels replaced by a segment recorded 10–50 h earlier	Voltages, flows, load, PV, frequency	Replayed length 4–8 steps	Mo et al. (2012); McLaughlin et al. (2016)
Denial of service	Packet rate collapses, latency rises and measurements freeze at their last value from the window midpoint	Packet rate, latency, voltages, flows (frozen)	Packet-rate drop 40–95%; latency +30–110 ms	Yan et al. (2012); Wang & Lu (2013)
Scaling	Load and PV measurement channels multiplied by a constant for the whole window	Net load, PV	Factor 0.85–1.25	Deng et al. (2017); Giani et al. (2013)
Command injection	Burst of extra commands and a setpoint deviation in the second half of the window	Command count, setpoint deviation	3–12 commands; deviation $\pm 0.05\text{--}0.2$	Cherepanov (2017); Slowik (2019); Lee et al. (2016)

Detector, training schemes, privacy and robustness

Table 4 gives the detector and the schemes. The detector is a multilayer perceptron (48 inputs, two hidden layers of 64 rectified units, six-way softmax over normal and the five attack classes) trained with class-weighted cross-entropy (weights proportional to the square root of inverse class frequency, clipped to 0.5–4, against imbalance (He & Garcia, 2009)), Adam at 3×10^{-3} (Kingma & Ba, 2015), minibatches of 64 and per-client standardised features. Six schemes train it. Local: each client trains its own model for 30 epochs on its own windows and is evaluated on its own test windows; a transfer score evaluates one client's model on the others. Central: one model trained for 60 epochs on the pooled training windows of all clients: the privacy-violating reference that federated learning tries to match (Rahman et al., 2020). FedAvg: 60 rounds in which every client trains two local epochs from the global model and the server averages the updates weighted by client size (McMahan et al., 2017). FedProx: the same with a proximal term ($\mu = 0.05$) that keeps local models near the global one under heterogeneity (Li et al., 2020c). Differentially private FedAvg: each client trains with DP-SGD, per-example gradients clipped to L2 norm 2 and Gaussian noise with multiplier $\sigma \in \{1, 2, 4\}$ added to the batch sum, so that the model protects individual windows (records) of each client's data (Abadi et al., 2016; Dwork & Roth, 2014); the privacy loss (ϵ at $\delta = 10^{-5}$) is accounted with the leading term of the moments-accountant bound of Abadi et al. composed over all local steps (sampling rate 64/700 per step, 1,320 steps), which is an approximation whose omitted term is small for $\sigma \geq 1$; an exact Rényi accountant would give similar or slightly smaller values (Abadi et al., 2016; Dwork & Roth, 2014). Robust aggregation: FedAvg with coordinate-wise median or 20% trimmed-mean aggregation (Yin et al., 2018; Blanchard et al., 2017) under one, two or three model-poisoning clients that train on flipped labels (attacks relabelled as normal) and scale their updates eightfold (Bagdasaryan et al., 2020; Fang et al., 2020), compared with plain averaging under the same poisoners. Each scheme is run with three seeds (two for the poisoning and heterogeneity experiments), with the seed fixing the clients' characteristics, data, initialisation and sampling. Communication cost is counted as the bytes of model parameters exchanged (two directions, 32-bit) per client per round; the central scheme is charged the bytes of the raw training data it moves.

Table 4: Detector, training schemes, differential privacy, robust aggregation and poisoning model: architecture, hyperparameters, protocol and accounting

Element	Specification
Detector	Multilayer perceptron 48→64→64→6 (softmax over normal and five attack classes); class-weighted cross-entropy (square-root inverse frequency, clipped 0.5–4); Adam at 0.003; minibatches of 64; NumPy implementation with explicit per-example gradients
Local	Each client trains its own model for 30 epochs on its own windows; evaluated on its own test windows (pooled); transfer score = client 1's model on the other clients
Central	One model trained for 60 epochs on the pooled training windows of all clients (privacy-violating reference); charged the bytes of the raw data moved
FedAvg	60 rounds; every client trains 2 local epochs from the global model; server averages updates weighted by client size (McMahan et al., 2017)
FedProx	As FedAvg with a proximal term $\mu = 0.05$ toward the global model (Li et al., 2020c)
DP-FedAvg	Local DP-SGD: per-example gradient clipping to L2 norm 2, Gaussian noise with multiplier $\sigma \in \{1.0, 2.0, 4.0\}$ on the batch sum (Abadi et al., 2016); record-level guarantee; ϵ at $\delta = 1e-05$ from the leading term of the moments-accountant bound composed over 1320 steps at sampling rate 0.091 (approximate)
Robust aggregation	Coordinate-wise median or 20% trimmed mean of the client updates (Yin et al., 2018; Blanchard et al., 2017), compared with plain averaging
Poisoning model	1–3 clients train on flipped labels (attacks relabelled as normal) and scale their update by 8 (Bagdasaryan et al., 2020; Fang et al., 2020)
Communication	Bytes of parameters exchanged per client per round (two directions, 32-bit); 7,686 parameters

Explanations

Table 5 gives the explanation methods. For the central, FedAvg, FedProx, differentially private ($\sigma = 1$) and local (client 1) detectors, sampling-based Shapley values (Štrumbelj & Kononenko, 2014; Shapley, 1953; Lundberg & Lee, 2017) of the predicted probability of each attack class are computed for 120 held-out windows of that class with 40 random feature permutations against a background of 60 training windows per client; the mean absolute value per feature gives the importance of the feature for that class, and the mean over classes gives a global importance vector. Agreement between schemes is the Spearman rank correlation between their global importance vectors; agreement across clients is the rank correlation between the importance vector computed on each client's attacked test windows and the global vector, which measures whether the federated detector's reasons are the same everywhere (Ghorbani et al., 2019; Nadeem et al., 2023). A depth-3 decision tree fitted to the FedAvg detector's predicted class on the pooled test windows gives a readable surrogate with a fidelity score (Rudin, 2019; Guidotti et al., 2018; Molnar, 2025).

Table 5: Explanation methods: Shapley sampling design, importance aggregation, agreement measures, surrogate tree and the questions each answers

Method	Design	Question answered
Shapley sampling per class	Štrumbelj & Kononenko (2014) permutation sampling of the predicted probability of each attack class: 120 held-out windows of the class, 40 permutations, background of 60 training windows per client	Which features drive each class's detection (Lundberg & Lee, 2017; Shapley, 1953)
Global importance	Mean absolute attribution per feature, averaged over the five classes	What the detector relies on overall
Agreement across schemes	Spearman rank correlation between global importance vectors of central, FedAvg, FedProx, DP ($\sigma = 1$) and local (client 1) detectors	Does federation (or privacy noise) change the detector's reasons? (Ghorbani et al., 2019)
Agreement across clients	Rank correlation between each client's importance vector (its attacked test windows) and the global vector	Are the federated detector's reasons the same everywhere? (Nadeem et al., 2023)
Surrogate tree	Depth-3 decision tree fitted to the FedAvg detector's predicted class on pooled test windows; fidelity = agreement with the detector	Can an operator read the decision rule? (Rudin, 2019; Guidotti et al., 2018)

Evaluation, resilience economics and experiments

Table 6 lists the metrics and Table 7 the experiments. Detection is evaluated on the held-out test windows of all clients: AUROC and area under the precision–recall curve for attack versus normal, F1, precision, recall and false-positive rate at a threshold of 0.5 on the attack probability, recall per attack class and macro-averaged over classes, accuracy, expected calibration error of the attack probability in ten bins (Fawcett, 2006; Davis & Goadrich, 2006; Saito & Rehmsmeier, 2015; Guo et al., 2017; Naeini et al., 2015), and per-client AUROC, on both test streams, the in-mix stream (each client's own attack mix, the usual protocol) and the uniform-mix stream (every class equally, the deployment question). Because attacks are rare, the 0.5 threshold is not the operating point an operator would use (Axelsson, 2000): for each scheme a cost-minimising alarm threshold is chosen on the clients' validation windows (each client evaluates the annual-cost curve of thresholds from 0.5 to 0.999 locally and only the curves are pooled), and recall per class and false-positive rate at that threshold are reported for both streams.

Table 6: Evaluation metrics: detection, calibration, per-client and transfer performance, communication and the resilience cost model

Metric or cost	Definition	Baseline value / source
AUROC, AUPRC	Attack-versus-normal ranking of the attack probability ($1 - P(\text{normal})$); area under ROC and precision-recall curves	Fawcett (2006); Davis & Goadrich (2006); Saito & Rehmsmeier (2015)
F1, precision, recall, FPR	At a threshold of 0.5 on the attack probability	Axelsson (2000)
Per-class and macro recall	Share of each attack class predicted as that class; mean over the five classes	He & Garcia (2009)
Calibration (ECE)	Expected calibration error of the attack probability in ten bins	Guo et al. (2017); Naeini et al. (2015)
Per-client AUROC; transfer	AUROC on each client's test windows; local model of client 1 applied to the other clients	Rahman et al. (2020)
Communication	Parameter bytes exchanged (federated) or raw data bytes moved (central)	Bonawitz et al. (2019)
Safe-mode rule	Alarm $\rightarrow$ safe mode (last-good state, local measurements, conservative setpoints) for 8 steps	35 per step; Cárdenas et al. (2011)
Missed-attack damage	Per missed event, by class	FDIA 1,800; other classes 700; attacks 18 per client-year
False-alarm cost	Per false alarm plus two safe-mode steps; applied to $\text{FPR} \times \text{normal windows per year}$	90 per alarm
Annual cost of a scheme	Missed-attack damage (class shares $\times$ (1 – recall) $\times$ damage $\times$ rate) + safe-mode cost of detected attacks + false-alarm cost + communication	Author's stylisation

Table 7: Experiments: scheme comparison, heterogeneity sweep, privacy-utility curve, poisoning and robust aggregation, explanation agreement, Monte Carlo economics, Sobol and metamodel

Experiment	Design	Outputs
Scheme comparison	Local, central, FedAvg, FedProx, DP-FedAvg ($\sigma = 1, 2, 4$) $\times$ 3 seeds on held-out windows	Detection, calibration, per-class recall, per-client AUROC, communication
Heterogeneity sweep	Dirichlet concentration 0.1, 0.5, 2, 100 $\times$ local, FedAvg, FedProx (2 seeds)	Gap between local, federated and the pooled reference as label skew varies
Privacy-utility curve	$\sigma = 1, 2, 4$ with accounted ϵ	Detection lost per unit of privacy gained; which classes vanish first
Poisoning and robust aggregation	1, 2, 3 poisoners $\times$ mean, median, trimmed mean (2 seeds)	Damage from poisoners and the cost of robustness
Explanation agreement	Shapley importance for central, FedAvg, FedProx, DP, local; agreement across schemes and across clients; surrogate tree	Whether reasons survive federation and privacy; a readable rule
Monte Carlo economics	2,000 draws of damage, false-alarm, safe-mode, attack-rate, communication and privacy-value inputs on measured operating points	Annual cost by scheme; saving of federation over local; value over central including privacy
Sobol and metamodel	Sobol indices of the federation-over-local saving; LightGBM on the experiment grid $\times$ sampled economics with a depth-3 tree	Drivers of the saving; conditions that make federation worth most

The resilience economics translate the operating point into money: an energy-management rule switches a client into a safe mode (last-good state, local measurements, conservative setpoints) for eight steps when the detector raises an alarm, at a cost per step; a missed attack causes damage (1,800 per false-data-injection event, 700 per event of the other classes, in the testbed's units); a false alarm costs 90 plus two safe-mode steps; attacks arrive at 18 per client-year; and the annual cost of a scheme is the sum of missed-attack damage (weighted by one minus recall per class and the class shares), safe-mode cost of detected attacks and false-alarm cost (false-positive rate times the normal windows in a year), plus communication (Cárdenas et al., 2011; Pasqualetti et al., 2013; Axelsson, 2000). The Monte Carlo analysis draws 2,000 sets of these inputs (damage per false-data-injection event (triangular 500–1,800–6,000), damage per other event (150–700–2,500), false-alarm cost (20–250), safe-mode cost per step (10–80), attack rate (2–18–60 per year), communication cost (0.002–0.05 per MB) and the value a party places on not pooling its data (0–20,000 per year)) and recomputes the annual cost of each scheme from its cost-tuned operating point on each test stream (in-mix and uniform-mix), reporting distributions, the saving of federation over local detection and the value of federation over centralised training including the privacy value (Morgan & Henrion, 1990; Spiegelhalter, 2017); Sobol indices attribute the variance of the federation-over-local saving to its inputs (Sobol', 2001; Saltelli et al., 2010; Herman & Usher, 2017; Iwanaga et al., 2022). A gradient-boosting metamodel on the experiment grid (heterogeneity concentration, privacy noise, number of poisoners) crossed with sampled economics, with a depth-3 tree as a readable summary, says which conditions make federation worth most (Ke et al., 2017; Friedman, 2001; Rudin, 2019). The experiments are: the scheme comparison (six schemes, three seeds); the heterogeneity sweep (Dirichlet concentration 0.1, 0.5, 2, 100 for local, FedAvg and FedProx); the privacy-utility curve ($\sigma = 1, 2, 4$ with their ϵ); the poisoning experiment (one to three poisoners $\times$ mean, median, trimmed mean); the explanation agreement analysis; and the economics.

Verification

Verification checks were that attacked windows carry the intended signatures (false-data injection passes the voltage-flow consistency check while raising the state residual; denial of service collapses the packet rate), that the class counts per client match the Dirichlet mixes, that per-client standardisation uses training data only and that alarm thresholds are tuned on validation windows never used for training or testing, that FedAvg with a single client reduces to local training, that the privacy loss ϵ decreases monotonically with the noise multiplier, that median and trimmed-mean aggregation without poisoners track plain averaging, that learning curves (global AUROC every five rounds) flatten within the round budget, and that every table and figure is produced from the results file (Law, 2015; Sargent, 2013). The code is organised so that the telemetry model, the attack models, the detector, the aggregation rules and the economics can be replaced independently.

Results

Learning and detection by scheme

Figure 5 shows the learning curves and the spread of detection quality across clients; Table 8 and Figure 6 give the scheme comparison on the held-out windows. The federated detectors reached most of their final quality within ten rounds (global AUROC 0.878 for FedAvg and 0.880 for FedProx at round 10, against 0.864 and 0.884 at round 60) and the central model within five epochs; the differentially private detector with $\sigma = 1$ learned more slowly, from 0.23 to 0.52 at round 10 and 0.867 at round 60, as the noise slowed the early rounds. On the in-mix test (each client's own attack mix, the protocol most studies use) the schemes are close: central 0.894 ± 0.007 AUROC over three seeds, local 0.885 ± 0.002 , FedProx 0.884 ± 0.006 , FedAvg 0.864 ± 0.008 ; the ranking by area under the precision-recall curve is the same (0.764, 0.754, 0.762, 0.718), and false-positive rates at the default threshold of 0.5 range from 0.99% (local) to 2.82% (central). On the uniform-mix test (every client exposed to all five classes equally) the picture changes: the central and federated detectors hold their quality (0.882, 0.880 and 0.870) while the local models fall to 0.743 ± 0.018 and their macro recall over the five classes from 0.613 to 0.396, because a client whose history held few or no examples of a class cannot detect it; a local model transferred to the other clients scores 0.749. The per-client spread (Figure 5b) is wide for every scheme (0.70–1.00 for local, 0.72–0.99 for the federated detectors) and reflects the clients' attack mixes and noise levels more than the scheme. By attack class (Table 9, Figure 6b), denial of service is detected almost perfectly by every scheme except the noisiest private one (0.94–1.00), command injection and false-data injection

well (0.76–0.94 and 0.70–0.87 in-mix), replay poorly (0.27–0.50) and scaling worst (0.15–0.26), the last two because their signatures (a splice in the measurement history and a proportional error in two channels) are small relative to the day-to-day variation of the features; the central model detects replay best (0.50) and the local models detect scaling best (0.24), the one class where client-specific sensitivities help. Calibration of the attack probability is good for all non-private schemes (expected calibration error 0.023–0.043).

Figure 5: Learning and spread: (a) global AUROC on held-out windows by round for central, FedAvg, FedProx and differentially private FedAvg (mean and range over seeds), (b) per-client AUROC by scheme

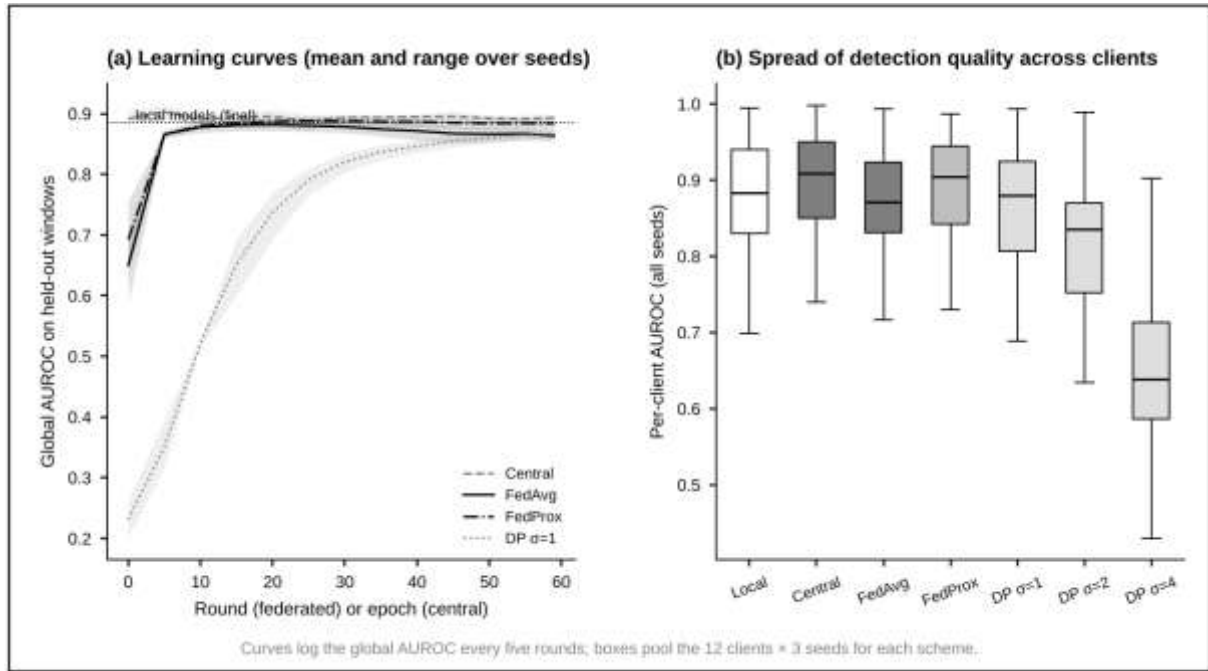


Table 8: Scheme comparison on held-out windows: AUROC, AUPRC, F1, false-positive rate, macro recall and calibration on the in-mix test; per-client AUROC range; AUROC and macro recall on the uniform-mix test; communication

Scheme	AUROC, in-mix test (mean $\pm$ sd over seeds)	AUPRC	F1 (0.5)	FPR (0.5), %	Macro recall	ECE	Per-client AUROC range	AUROC, uniform-mix test	Macro recall, uniform-mix	Communication (MB)
Local (own data)	0.885 $\pm$ 0.002	0.754	0.738	0.99	0.613	0.026	0.70–1.00	0.743 $\pm$ 0.018	0.396	0.0
Central (pooled)	0.894 $\pm$ 0.007	0.764	0.721	2.82	0.712	0.043	0.74–1.00	0.882 $\pm$ 0.012	0.716	1.6
FedAvg	0.864 $\pm$ 0.008	0.718	0.687	1.83	0.584	0.037	0.72–0.99	0.870 $\pm$ 0.010	0.636	44.3
FedProx	0.884 $\pm$ 0.006	0.762	0.744	1.07	0.621	0.023	0.73–0.99	0.880 $\pm$ 0.001	0.651	44.3
DP-FedAvg, $\sigma = 1$	0.867 $\pm$ 0.007	0.707	0.651	0.20	0.427	0.026	0.69–0.99	0.841 $\pm$ 0.004	0.488	44.3
DP-FedAvg, $\sigma = 2$	0.822 $\pm$ 0.014	0.590	0.483	0.08	0.279	0.035	0.63–0.99	0.795 $\pm$ 0.009	0.342	44.3
DP-FedAvg, $\sigma = 4$	0.651 $\pm$ 0.016	0.302	0.222	0.00	0.115	0.047	0.43–0.90	0.602 $\pm$ 0.041	0.130	44.3
Local model transferred to other clients	0.749	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a

Figure 6: Detection by scheme: (a) AUROC on the in-mix and uniform-mix tests, (b) recall by attack class, (c) false-positive rate at the default and cost-tuned thresholds with recall at the tuned threshold, (d) macro recall over classes on both tests

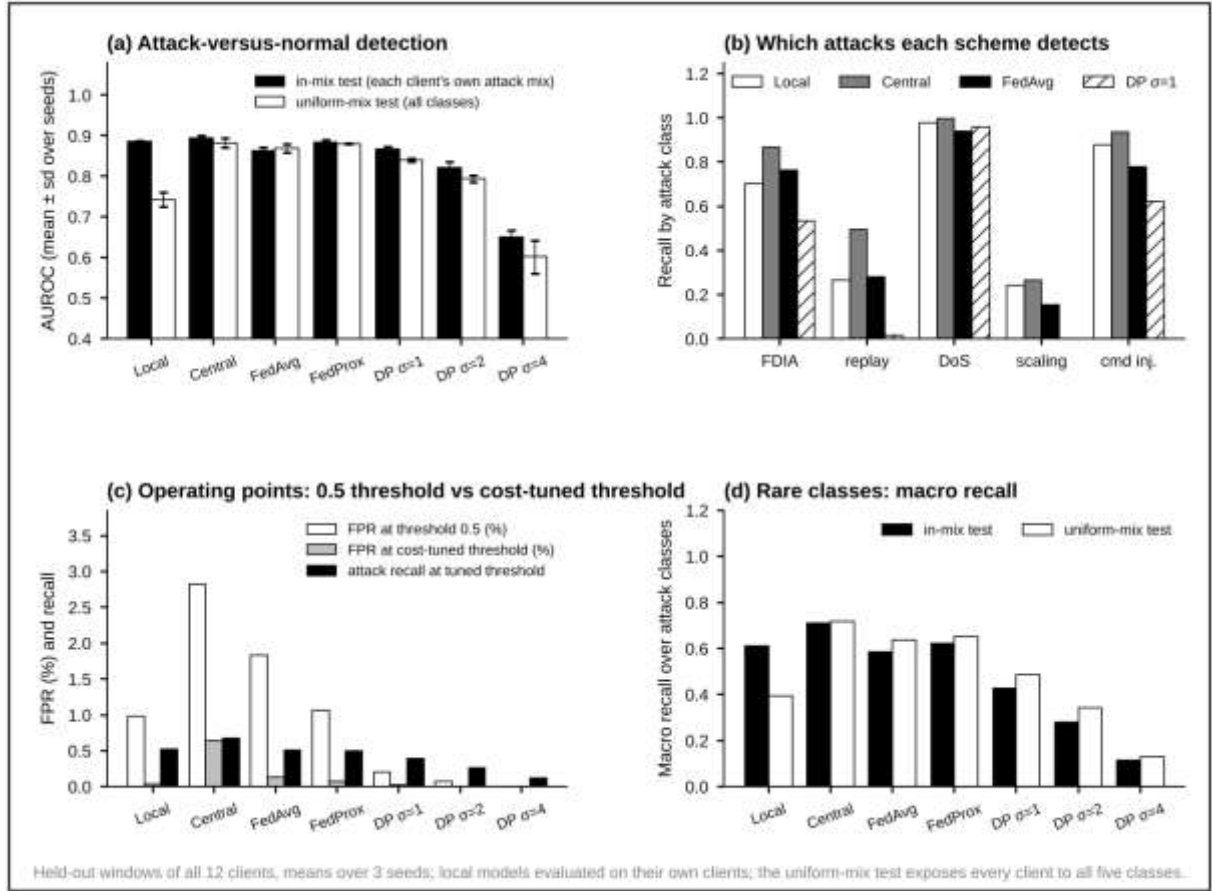


Table 9: Recall by attack class on the in-mix and uniform-mix tests, cost-tuned alarm thresholds, and attack recall and false-positive rate at the tuned threshold, by scheme

Scheme	False-data injection (in-mix / uniform)	Replay (in-mix / uniform)	Denial of service (in-mix / uniform)	Scaling (in-mix / uniform)	Command injection (in-mix / uniform)	Cost-tuned threshold	Attack recall at tuned threshold (in-mix / uniform)	FPR at tuned threshold, % (in-mix / uniform)
Local (own data)	0.70 / 0.43	0.27 / 0.12	0.98 / 0.81	0.24 / 0.10	0.88 / 0.52	0.985	0.52 / 0.33	0.03 / 0.03
Central (pooled)	0.87 / 0.90	0.50 / 0.49	1.00 / 1.00	0.26 / 0.21	0.94 / 0.98	0.999	0.68 / 0.67	0.65 / 0.30
FedAvg	0.76 / 0.79	0.28 / 0.36	0.94 / 0.98	0.15 / 0.13	0.78 / 0.92	0.996	0.51 / 0.55	0.13 / 0.08
FedProx	0.79 / 0.82	0.33 / 0.38	0.97 / 1.00	0.17 / 0.12	0.84 / 0.93	0.993	0.50 / 0.55	0.07 / 0.03
DP-FedAvg, $\sigma = 1$	0.53 / 0.65	0.02 / 0.02	0.96 / 0.99	0.00 / 0.00	0.62 / 0.78	0.860	0.39 / 0.44	0.03 / 0.02
DP-FedAvg, $\sigma = 2$	0.12 / 0.21	0.00 / 0.00	0.95 / 0.98	0.00 / 0.00	0.33 / 0.52	0.733	0.26 / 0.30	0.01 / 0.01
DP-FedAvg, $\sigma = 4$	0.00 / 0.00	0.00 / 0.00	0.57 / 0.65	0.00 / 0.00	0.00 / 0.00	0.567	0.12 / 0.12	0.00 / 0.01

Communication and compute are small at this scale but instructive. Each federated round exchanges the 7,686 parameters of the detector in both directions, 44 MB per client over 60 rounds, whereas moving a client's 700 training windows of 48 features to a central server would take 0.13 MB; the raw data are smaller than the model traffic by two orders of magnitude, so federation here buys confidentiality, not bandwidth, and the communication cost contributed nothing to the Sobol indices below. Training took 9–14 seconds per scheme and seed on two cores; the Shapley explanations took longer than the training. The time-to-quality favours the federated schemes in one respect: because every client trains in parallel, ten rounds (a few seconds of wall-clock in the testbed and a few minutes in a deployment with 15-minute windows) are enough to reach 0.88, and a new client joining a trained federation inherits the global model at once, which a local model cannot do.

The operating point matters more than the ranking. At the default threshold, false-positive rates of 1–3% mean 300–900 false alarms per client-year at 35,040 windows per year, which the cost model prices above the damage of every attack the detector catches; the cost-tuned thresholds chosen on the validation windows lie at 0.985–0.999 for the non-private schemes and at 0.57–0.86 for the private ones (whose attack probabilities are less concentrated), and at those thresholds the false-positive rates fall to 0.03–0.13% for local, FedAvg and FedProx while attack recall falls to 0.50–0.52 in-mix and 0.33 (local) to 0.55 (federated) on the uniform mix (Figure 6c). The central model is the exception: its attack probabilities on normal windows are more concentrated near one than the federated models' (it overfits the pooled data; its calibration error is the highest of the non-private schemes), the threshold grid's upper value of 0.999 was not high enough, and its tuned false-positive rate stays at 0.65% in-mix and 0.30% uniform, a calibration problem that a finer grid or a calibration step would address, and that the economics below penalise.

Heterogeneity

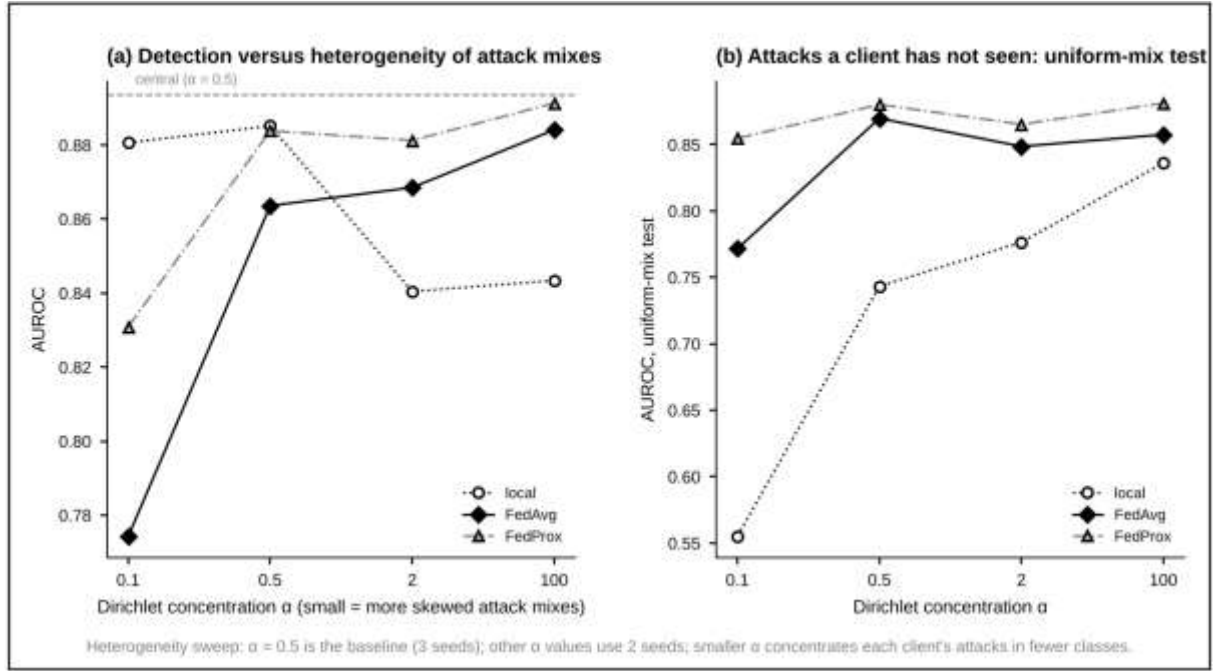
Table 10 and Figure 7 give the heterogeneity sweep. As the Dirichlet concentration falls from 100 (every client sees every class) through 2 and 0.5 to 0.1 (each client's attacks concentrated in one or two classes), the local models' in-mix AUROC rises from 0.843 to 0.881 (a client specialised in one class learns it well) while their uniform-mix AUROC falls from 0.836 to 0.555 and their uniform-mix macro recall from 0.526 to 0.219: the local model is only as good as the attacks the client has already suffered.

Table 10: Heterogeneity sweep: AUROC, macro recall and false-positive rate on the in-mix test, AUROC and macro recall on the uniform-mix test, and annual EMS cost at the tuned threshold for local, FedAvg and FedProx at Dirichlet concentrations of 0.1, 0.5, 2 and 100

Dirichlet α	Scheme	AUROC, in-mix (mean $\pm$ sd)	Macro recall, in-mix	FPR (0.5), %	AUROC, uniform-mix	Macro recall, uniform-mix	Annual EMS cost per client, uniform-mix, tuned (thousand)	Seeds
0.1	Local (own data)	0.881 $\pm$ 0.036	0.641	1.33	0.555	0.219	19.5	2
	FedAvg	0.774 $\pm$ 0.016	0.390	1.98	0.772	0.440	13.9	2
	FedProx	0.831 $\pm$ 0.023	0.484	0.36	0.855	0.577	11.0	2
0.5	Local (own data)	0.885 $\pm$ 0.002	0.613	0.99	0.743	0.396	13.7	3
	FedAvg	0.864 $\pm$ 0.008	0.584	1.83	0.870	0.636	13.6	3
	FedProx	0.884 $\pm$ 0.006	0.621	1.07	0.880	0.651	11.1	3
2	Local (own data)	0.840 $\pm$ 0.006	0.520	0.79	0.776	0.437	18.9	2
	FedAvg	0.869 $\pm$ 0.007	0.639	1.48	0.849	0.643	11.3	2
	FedProx	0.881 $\pm$ 0.007	0.646	1.15	0.865	0.651	16.6	2
100	Local (own data)	0.843 $\pm$ 0.019	0.533	0.96	0.836	0.526	12.2	2
	FedAvg	0.884 $\pm$ 0.018	0.679	1.91	0.857	0.676	11.5	2
	FedProx	0.891 $\pm$ 0.017	0.679	1.82	0.881	0.690	10.2	2

FedAvg degrades with heterogeneity on both tests (uniform-mix AUROC 0.857 at $\alpha = 100$, 0.772 at $\alpha = 0.1$), the client-drift effect that the non-IID literature describes; FedProx degrades less (0.881 to 0.855) and is the best federated scheme at every level of heterogeneity, with uniform-mix macro recall of 0.577 at $\alpha = 0.1$ against 0.440 for FedAvg and 0.219 for local. At the homogeneous end the federated schemes beat the local models on both tests (0.884–0.891 against 0.843 in-mix), because pooling knowledge of attacks that every client sees adds data; at the heterogeneous end they lose in-mix and win on the uniform mix, because pooling adds classes. Which test an operator should weight is the deployment question: the uniform-mix test asks whether a client is protected against attacks it has not yet seen.

Figure 7: Heterogeneity sweep: (a) in-mix AUROC of local, FedAvg and FedProx by Dirichlet concentration, (b) uniform-mix AUROC (detection of attacks a client has not seen)



Privacy and robustness

Table 11 and Figure 8 give the privacy–utility curve. Record-level DP-SGD inside FedAvg with noise multipliers of 1, 2 and 4 gives privacy losses of $\epsilon \approx 36, 15$ and 6.7 at $\delta = 10^{-5}$ under the approximate accounting of the Methods, and AUROC of 0.867, 0.822 and 0.651 against 0.864 without noise; at $\sigma = 1$ the ranking quality is intact but the classes with weak signatures vanish first (replay and scaling recall fall to 0.02 and 0.00, false-data injection to 0.53, command injection to 0.62) while denial of service stays at 0.96; at $\sigma = 2$ false-data injection falls to 0.12 and at $\sigma = 4$ only denial of service is still detected (0.57). The private detectors are also conservative: their false-positive rate at the default threshold is 0.00–0.20%, which, as the economics show, is not a disadvantage. The privacy budgets are large by the standards of the differential-privacy literature (single-digit ϵ is the usual target) and the testbed's conclusion is that record-level DP-SGD at those budgets costs most of the detector's value when each client has 700 windows: the privacy that small local datasets can afford is limited, and client-level mechanisms with more clients, or larger local datasets, are the routes the literature offers. Table 12 and Figure 9 give the poisoning experiment. One poisoning client among twelve, training on flipped labels and scaling its update eightfold, collapses plain averaging from 0.864 to 0.719 AUROC and attack recall from 0.64 to 0.21; coordinate-wise median aggregation holds at 0.860 and 0.60 recall and the 20% trimmed mean at 0.844 and 0.57. With two poisoners the median holds at 0.821 and the trimmed mean at 0.819 against 0.677 for averaging; with three (a quarter of the clients) averaging detects nothing (recall 0.003), the trimmed mean, which removes only two values per coordinate, fails (0.716, recall 0.14), and the median still detects 0.45 of attacks at 0.794 AUROC. Robustness has a price without poisoners: the median's false-positive rate is higher (2.29% against 1.83% at one poisoner) because it discards the

information in the averaged update's tails.

Table 11: Privacy-utility curve: noise multiplier, privacy loss ϵ , AUROC, F1, false-positive rate, macro recall, recall by class and annual EMS cost for FedAvg without noise and with DP-SGD at $\sigma = 1, 2$ and 4 , with the no-detector cost

Scheme	Noise multiplier σ	ϵ ($\delta = 10^{-5}$, approximate)	AUROC (mean $\pm$ sd)	F1 (0.5)	FPR (0.5), %	Macro recall	Recall by class: FDIA / replay / DoS / scaling / cmd injection	Annual EMS cost per client, uniform-mix, tuned (thousand)
FedAvg, no DP	n/a	∞	0.864 ± 0.008	0.687	1.83	0.584	0.76 / 0.28 / 0.94 / 0.15 / 0.78	13.6
DP-FedAvg	1	35.8	0.867 ± 0.007	0.651	0.20	0.427	0.53 / 0.02 / 0.96 / 0.00 / 0.62	12.3
DP-FedAvg	2	14.9	0.822 ± 0.014	0.483	0.08	0.279	0.12 / 0.00 / 0.95 / 0.00 / 0.33	14.3
DP-FedAvg	4	6.7	0.651 ± 0.016	0.222	0.00	0.115	0.00 / 0.00 / 0.57 / 0.00 / 0.00	16.0
No detector	n/a	n/a	n/a	n/a	n/a	n/a	n/a	16.5

Figure 8: Privacy and utility: (a) AUROC against the accounted privacy loss ϵ , (b) recall by attack class as noise rises, (c) F1 and false-positive rate at the default threshold, (d) annual EMS cost at the tuned threshold on the uniform-mix test

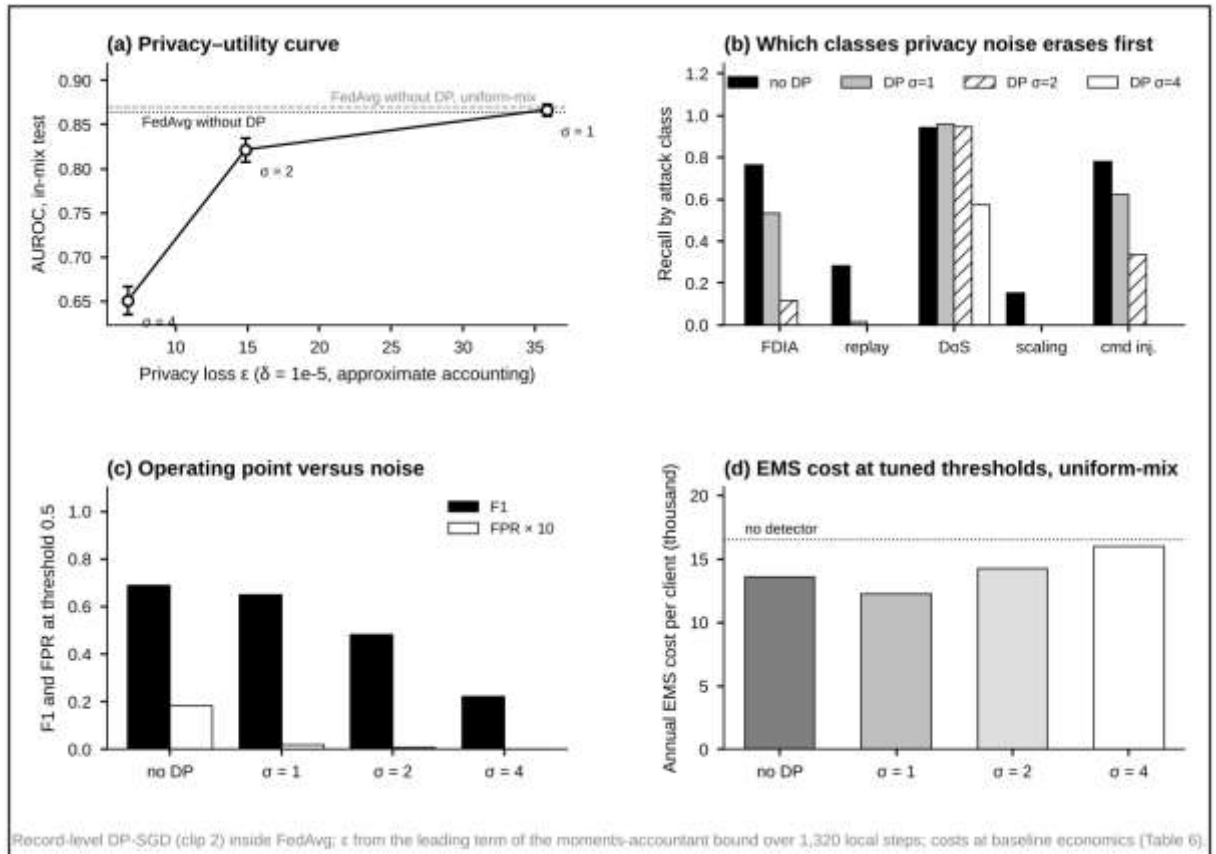
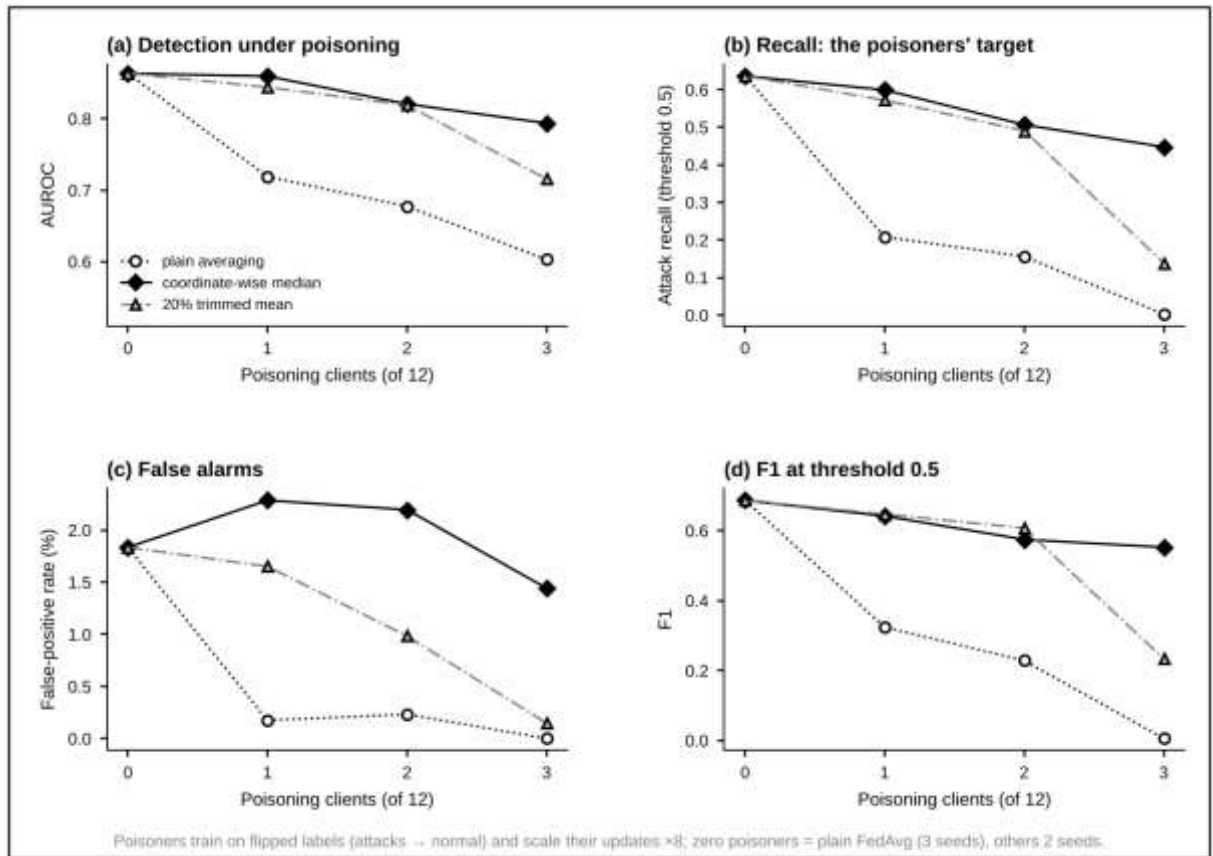


Table 12: Poisoning and robust aggregation: AUROC, F1, false-positive rate, attack recall, macro recall, recall of false-data injection and command injection, and annual EMS cost for plain averaging, coordinate-wise median and 20% trimmed mean under one to three poisoning clients

Poisoning clients	Aggregation	AUROC (mean $\pm$ sd)	F1 (0.5)	FPR (0.5), %	Attack recall (0.5)	Macro recall	Recall: FDIA / command injection	Annual EMS cost per client, uniform-mix, tuned (thousand)
0	Plain averaging (no poisoners)	0.864 $\pm$ 0.008	0.687	1.83	0.636	0.584	0.76 / 0.78	13.6
1	Plain averaging	0.719 $\pm$ 0.022	0.324	0.17	0.208	0.204	0.29 / 0.24	14.4
	Coordinate-wise median	0.860 $\pm$ 0.000	0.642	2.29	0.599	0.555	0.69 / 0.72	14.2
	20% trimmed mean	0.844 $\pm$ 0.008	0.648	1.65	0.572	0.539	0.67 / 0.73	14.1
2	Plain averaging	0.677 $\pm$ 0.106	0.229	0.23	0.156	0.138	0.19 / 0.25	17.2
	Coordinate-wise median	0.821 $\pm$ 0.005	0.575	2.20	0.506	0.482	0.65 / 0.66	13.3
	20% trimmed mean	0.819 $\pm$ 0.021	0.607	0.98	0.490	0.466	0.60 / 0.60	13.2
3	Plain averaging	0.603 $\pm$ 0.088	0.005	0.00	0.003	0.003	0.00 / 0.00	16.5
	Coordinate-wise median	0.794 $\pm$ 0.023	0.553	1.44	0.447	0.422	0.49 / 0.39	14.1
	20% trimmed mean	0.716 $\pm$ 0.020	0.233	0.15	0.138	0.112	0.07 / 0.00	15.3

Figure 9: Poisoning: (a) AUROC, (b) attack recall, (c) false-positive rate and (d) F1 of plain averaging, coordinate-wise median and trimmed mean as the number of poisoning clients rises from zero to three



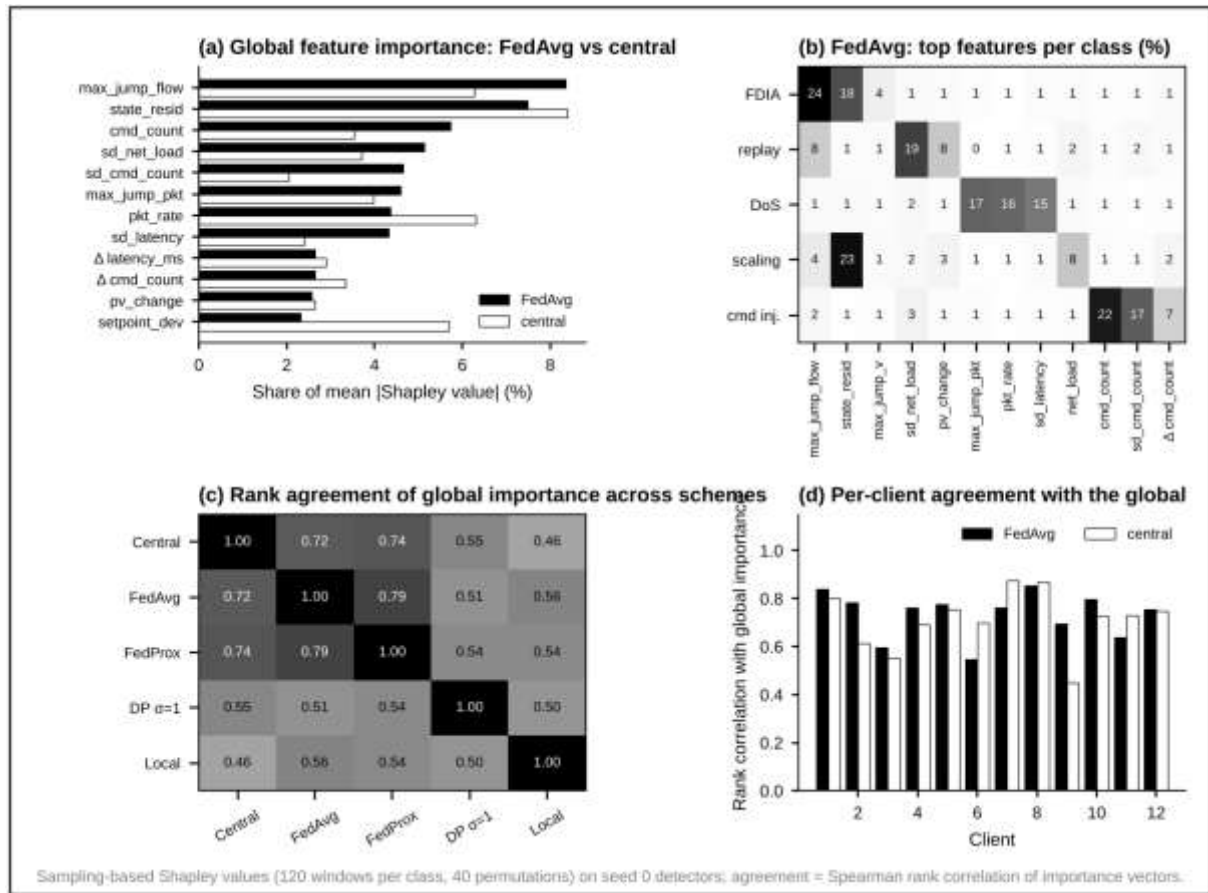
Explanations

Table 13 and Figure 10 give the explanations. The Shapley attributions of the FedAvg detector are physically readable: false-data injection is attributed to the largest flow jump (24% of the mean absolute attribution) and the state residual (18%) (the splice and the inconsistency with the measured load that the attack leaves); replay to the variability of net load (19%) and the photovoltaic change (8%); denial of service to the packet-rate jump, the packet rate and latency variability (17%, 16%, 15%); scaling to the state residual (23%) and net load (8%); and command injection to the command count and its variability (22%, 17%). The central detector's attributions name the same features in a slightly different order (false-data injection: flow jump 20%, state residual 20%; command injection: setpoint deviation 21%, command count 13%), and the rank agreement of the global importance vectors is 0.72 between central and FedAvg, 0.74 between central and FedProx and 0.79 between FedAvg and FedProx; the private detector agrees less (0.51–0.55) because noise flattens its reliance on the weak-signature features, and the local model of client 1 least (0.46–0.56) because it has learned its own client's mix. Across clients, the FedAvg detector's explanation on each client's attacked windows agrees with the global vector at a mean rank correlation of 0.73 (range 0.55–0.85), the central detector's at 0.71 (0.45–0.88), FedProx at 0.77 and the private detector at 0.79; the federated detectors' reasons are at least as uniform across clients as the pooled model's. The depth-3 surrogate tree reproduces 96.4% of the FedAvg detector's predictions on the pooled test windows with four rules that an operator can read: a packet-rate z-score below -2.7 is denial of service; otherwise a command count above 3.2 is command injection; otherwise a flow jump above 3.4 is false-data injection; otherwise the window is normal; the tree ignores replay and scaling, which the detector itself detects poorly, and its fidelity is the fidelity of a detector that mostly sees three classes.

Table 13: Explanations: top Shapley features per attack class for the FedAvg and central detectors, rank agreement of global importance across schemes, per-client agreement, and the surrogate tree with its fidelity

Item	FedAvg detector (seed 0)	Central detector (seed 0)
False-data injection	max_jump_flow 24%; state_resid 18%; max_jump_v 4%; flow_l2 2%	max_jump_flow 20%; state_resid 20%; Δ flow_l1 5%; Δ flow_l2 3%
Replay	sd_net_load 19%; pv_change 8%; max_jump_flow 8%; Δ pv_out 5%	sd_net_load 13%; pv_change 10%; Δ pv_out 9%; max_jump_flow 7%
Denial of service	max_jump_pkt 17%; pkt_rate 16%; sd_latency 15%; Δ latency_ms 8%	pkt_rate 24%; max_jump_pkt 14%; latency_ms 11%; Δ latency_ms 10%
Scaling	state_resid 23%; net_load 8%; max_jump_flow 4%; v_bus5 3%	state_resid 25%; net_load 9%; flow_l1 3%; hour_sin 2%
Command injection	cmd_count 22%; sd_cmd_count 17%; Δ cmd_count 7%; setpoint_dev 7%	setpoint_dev 21%; cmd_count 13%; Δ cmd_count 11%; Δ setpoint_dev 8%
Rank agreement of global importance (Spearman)	central-FedAvg 0.72; central-FedProx 0.74; central-DP $\sigma=1$ 0.55; central-local 0.46; FedAvg-FedProx 0.79; FedAvg-DP $\sigma=1$ 0.51; FedAvg-local 0.56; FedProx-DP $\sigma=1$ 0.54; FedProx-local 0.54; DP $\sigma=1$ -local 0.50	
Per-client agreement with the global vector	FedAvg: mean 0.73, range 0.55–0.85	Central: mean 0.71, range 0.45–0.88
Surrogate tree (depth 3) on the FedAvg detector; fidelity 96.4%	$\text{pkt_rate} \leq -2.73 \ \& \ \text{pkt_change} \leq -3.93 \rightarrow \text{dos};$ $\text{pkt_rate} \leq -2.73 \ \& \ \text{pkt_change} > -3.93 \rightarrow \text{dos};$ $\text{pkt_rate} > -2.73 \ \& \ \text{cmd_count} \leq 3.17 \ \& \ \text{max_jump_flow} \leq 3.38 \rightarrow \text{normal};$ $\text{pkt_rate} > -2.73 \ \& \ \text{cmd_count} \leq 3.17 \ \& \ \text{max_jump_flow} > 3.38 \rightarrow \text{fdia};$ $\text{pkt_rate} > -2.73 \ \& \ \text{cmd_count} > 3.17 \ \& \ \text{cmd_count} \leq 5.12 \rightarrow \text{cmd_injection};$ $\text{pkt_rate} > -2.73 \ \& \ \text{cmd_count} > 3.17 \ \& \ \text{cmd_count} > 5.12 \rightarrow \text{cmd_injection}$	Thresholds are z-scores of per-client standardised features

Figure 10: Explanations: (a) global feature importance of the FedAvg and central detectors, (b) top features per attack class for FedAvg, (c) rank agreement of global importance across schemes, (d) agreement of each client's explanation with the global one



Economics: the cost of errors and the value of federation

Table 14 and Figure 11 give the resilience economics. At baseline economics and tuned thresholds, on the uniform-mix test, a client with no detector bears 16.5 thousand per year of attack damage; a local detector reduces the total of missed attacks, safe-mode operation and false alarms to 13.7 thousand, FedAvg to 13.6, FedProx to 11.1, the private detector at $\sigma = 1$ to 12.3, at $\sigma = 2$ to 14.3 and at $\sigma = 4$ to 16.0, and the central model raises it to 23.4 because of its false alarms. On the in-mix test the local detector does best (12.1 thousand), FedProx next (13.9) and FedAvg no better than no detector (16.7): where every client's future attacks resemble its past ones, its own model, tuned to its own base rates, is hard to beat. Under 2,000 draws of economic inputs the annual saving of FedAvg over local detection on the uniform mix is 3.3 thousand per client (median 2.3, 5th–95th percentile –3.1 to 13.0), positive in 71% of draws; on the in-mix test it is –6.6 thousand (–10.5 to –2.6), negative in all but a handful of draws; and the value of FedAvg over the centralised reference, counting the value a party places on not pooling its data (0–20 thousand per year) and the central model's false alarms, is 22.2 thousand (8.1–36.4), positive in all draws. Sobol indices attribute 60% of the variance of the federation-over-local saving to the attack rate and 18–24% each to the damage per false-data-injection and per other event, with the false-alarm and safe-mode costs contributing 2–6% and communication nothing: the value of federation rises with the frequency and severity of attacks, which is where pooled knowledge of unseen attacks pays, and the uncertainty an operator should buy down first is its own attack exposure. The metamodel over the experiment grid (Table 14) attributes the variance of the saving to the number of poisoning clients (36%), the privacy noise (26%), heterogeneity (12%) and the attack rate (11%); its tree reads: without poisoners and with $\sigma \leq 1$ the saving is 4.8 thousand at attack rates below 28 per year and 9.0 thousand above; with $\sigma > 1$ it is negative; with any poisoner it is negative unless robust aggregation is used; the conditions, in order, under which a party should join a federation.

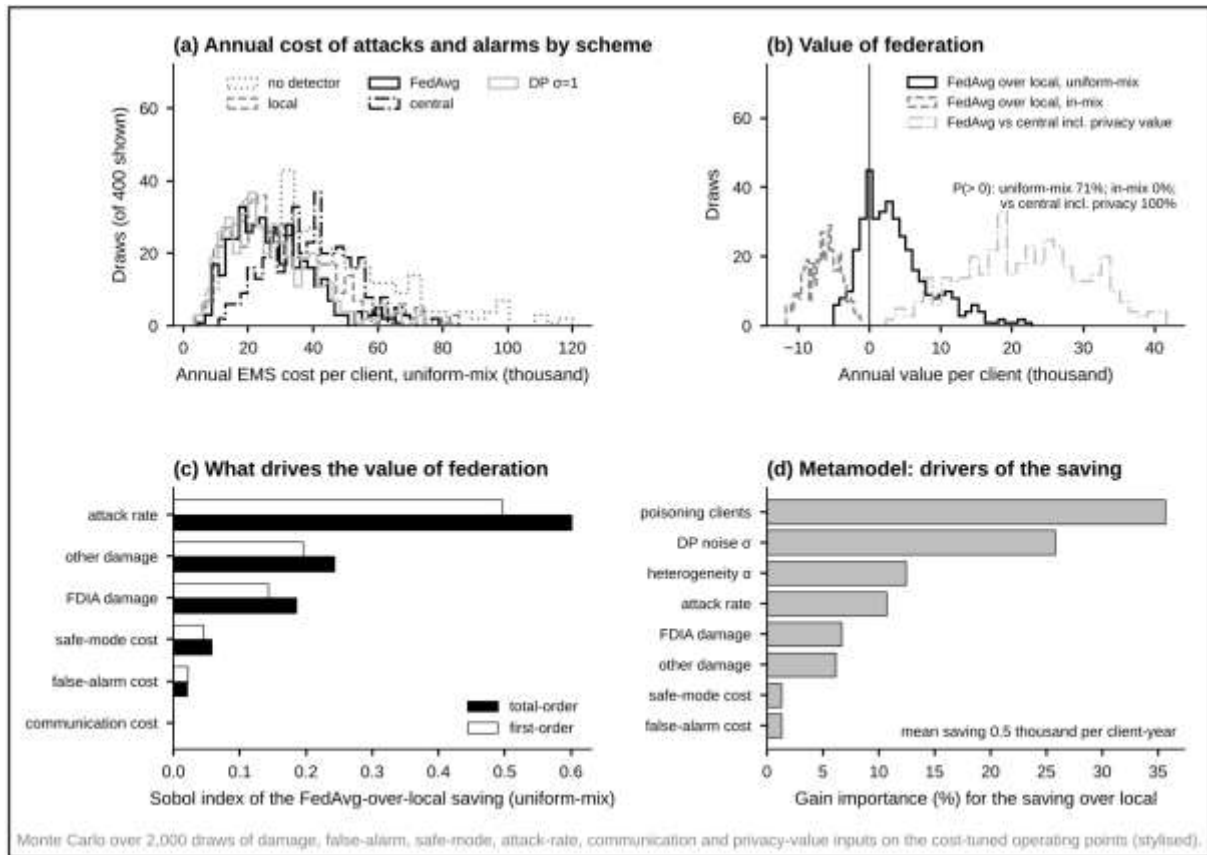
Table 14: Economics at tuned thresholds: annual cost components by scheme on the uniform-mix and in-mix tests, Monte Carlo distributions of the value of federation over local and central detection, and Sobol indices

Scheme / quantity	Tuned threshold / mean	Missed attacks / median	Safe mode / 5th–95th	False alarms / $P(> 0)$	Total, uniform-mix	Total, in-mix	Saving vs no detector
Local (own data)	0.985	10.7	1.6	1.4	13.7	12.1	2.8
Central (pooled)	0.999	4.6	3.4	15.3	23.4	41.4	-6.9
FedAvg	0.996	6.7	2.8	4.1	13.6	16.7	2.9
FedProx	0.993	6.8	2.8	1.6	11.1	13.9	5.4
DP-FedAvg, $\sigma = 1$	0.860	8.9	2.2	1.2	12.3	13.6	4.2
DP-FedAvg, $\sigma = 2$	0.733	12.2	1.5	0.6	14.3	14.7	2.2
DP-FedAvg, $\sigma = 4$	0.567	15.0	0.6	0.4	16.0	15.9	0.5
No detector	n/a	16.5	0	0	16.5	16.7	0
Monte Carlo: FedAvg saving over local, uniform-mix (thousand per client-year)	3.3	2.3	-3.1 to 13.0	71			
Monte Carlo: FedAvg saving over local, in-mix	-6.6	-6.6	-10.5 to -2.6	0			
Monte Carlo: FedAvg value over central incl. privacy value, uniform-mix	22.2	22.2	8.1–36.4	100			
Monte Carlo: cost with no detector, uniform-mix	38.4	33.4	10.5–83.2	100			
Monte Carlo: cost with FedAvg, uniform-mix	26.5	24.3	10.8–50.6	100			
Sobol indices of the FedAvg-over-local saving (S_1 / ST)	attack rate: 0.50 / 0.60; other damage: 0.19 / 0.24; FDIA damage: 0.14 / 0.18; safe-mode cost: 0.04 / 0.06; false-alarm cost: 0.02 / 0.02; communication cost: 0.00 / 0.00			4,096 evaluations	76		

Verification

Attacked windows carried the intended signatures: false-data injection raised the state residual and the flow jump while leaving the voltage–flow consistency check within noise, denial of service collapsed the packet rate by the drawn fraction; class counts per client matched the Dirichlet mixes (7.75% of training windows attacked, 89–177 windows per class across the twelve clients, with four to five clients holding three or fewer examples of a given class); standardisation used training windows only and thresholds validation windows only; FedAvg with a single client reproduced local training; ϵ fell monotonically with σ (35.8, 14.9, 6.7); median and trimmed-mean aggregation without poisoners tracked averaging within seed variation; the learning curves flattened within the round budget; and every table and figure was produced from the results file. The run took eight minutes on two cores.

Figure 11: Economics: (a) annual EMS cost per client by scheme under economic uncertainty, (b) value of federation over local and central detection, (c) Sobol indices of the saving, (d) metamodel drivers



Discussion

Answers to the research questions

RQ1 asked what a decision framework for federated attack detection requires. The literature yields six requirements (data that stay where they are, heterogeneity and rarity as the normal case, bounded disclosure and bounded damage, explanations an operator can check, decision outputs in money under uncertainty, and feedback with governance) and the four-layer framework of Figure 1 meets them; the testbed shows the outputs each produces. RQ2 asked how the schemes compare. On each client's own attack mix the schemes are within 0.03 AUROC of each other and the local models are as good as any; on the attacks a client has not seen the federated detectors hold their quality (0.87–0.88) and the local models fall to 0.74; FedProx is the better federated scheme at every level of heterogeneity; record-level differential privacy at single-digit ϵ costs most of the detector's value with 700 windows per client; one poisoning client in twelve destroys plain averaging and the coordinate-wise median withstands three. RQ3 asked whether the explanations agree. The federated and central detectors attribute each class to the same physically meaningful features with rank agreement of 0.72–0.79, agree with each client's explanation at 0.71–0.77 on average, and a four-rule surrogate reproduces 96% of the federated detector's decisions; privacy noise and local training lower the agreement to about 0.5. RQ4 asked what the operating point is worth. At the default threshold every detector costs more than it saves because of false alarms; at cost-tuned thresholds the federated detectors save 3–5 thousand per client-year against no detector and FedAvg saves 3.3 thousand over local detection on unseen attacks (positive in 71% of draws) but loses 6.6 thousand on seen ones; the attack rate and damage drive the value, and poisoners and privacy noise are the conditions that reverse it.

What federation is for, and what it is not

The two tests give opposite verdicts on federation, and both are right about something. A local detector trained on a client's own history is the best detector of that history: it knows the client's sensitivities, its noise and its base rates, and no averaging with eleven other clients improves that. A federated detector is a detector of what the other clients have seen, and its value is realised only when an attack arrives that the client has not seen, which, for a security system, is the case that matters, because the attacker chooses. The uniform-mix test is a stylised version of that case, and on it the local models lose a third of their macro recall while the federated models keep theirs. The economics follow the same logic: the saving of federation over local detection is positive on unseen attacks and scales with the attack rate and damage, which is why the Sobol indices point to the operator's own exposure as the uncertainty to resolve. Two corollaries follow for the design of a federation. First, the heterogeneity that makes federation valuable (clients see different attacks) is the heterogeneity that makes FedAvg drift, so the proximal term of FedProx (or a personalised layer on a shared backbone, which the literature has since developed) should be the default (Li et al., 2020c; Kairouz et al., 2021). Second, the base rate, not the model, sets the operating point: with attacks in 8% of windows, a detector that raises alarms on 1–3% of normal windows is a liability in any cost model with a non-trivial false-alarm cost, and threshold tuning on validation data at the operator's costs is as much part of the detector as its weights (Axelsson, 2000; Sommer & Paxson, 2010).

Privacy, robustness and the size of the federation

The privacy result is sobering and should be read with its conditions. Record-level DP-SGD protects each window of each client's data against inference from the shared model, and with 700 windows per client and 1,320 noisy steps, the budgets at which the detector still works ($\epsilon \approx 36$ at $\sigma = 1$) are far above the single-digit values the literature treats as meaningful, while budgets below 10 leave only the denial-of-service signature (Abadi et al., 2016; Dwork & Roth, 2014; Wei et al., 2020). The arithmetic is the well-known one: the privacy cost of a step scales with the sampling rate, which is large when local datasets are small, and the noise that masks one record also masks the rare classes whose signatures are weakest. Three routes exist. Larger local datasets lower the sampling rate; client-level differential privacy, where the unit of protection is a client's whole contribution, suits federations of organisations better than record-level protection and becomes affordable with many clients (Geyer et al., 2017; McMahan et al., 2018); and secure aggregation hides the individual updates from the server without noise, at the cost of trusting the aggregate (Bonawitz et al., 2017; Truex et al., 2019). The testbed's twelve clients are the smallest federation that shows the effects; the privacy and robustness arithmetic both improve with scale, which is an argument for sector-wide federations under a system operator rather than bilateral ones. The robustness result is the complementary warning: with twelve clients, one dishonest participant is 8% of the federation, enough to destroy plain averaging; the median withstood three, the trimmed mean did not, and both cost a little accuracy without poisoners. A federation needs an aggregation rule chosen for the number of participants it can afford to lose, attestation of the participants, and drills (Blanchard et al., 2017; Yin et al., 2018; Fang et al., 2020).

Explanations as a control, not an ornament

The explanation layer earned its place in two ways that a detection metric cannot show. First, the attributions are physically interpretable and stable across schemes: the federated detector found the same signatures (the flow jump and state residual of false-data injection, the packet-rate collapse of denial of service, the command burst of command injection) as the centralised one, with rank agreement of 0.72–0.79, and the four-rule surrogate reproduces 96% of its decisions. An analyst receiving an alarm with "largest flow jump 3.5 standard deviations, state residual 2.9" and the rule "flow jump > 3.4 → false-data injection" can check the telemetry; one receiving only a probability cannot. Second, the agreement measures are a monitor: a fall in per-client agreement signals a client whose data or model has drifted, and a fall in cross-scheme agreement under privacy noise or poisoning signals that the detector's reasons have changed even when its AUROC has not: the private detector at $\sigma = 1$ kept its AUROC (0.867 against 0.864) but lost agreement (0.51), because it had stopped using the weak-signature features. That explanations can themselves be attacked, and that a surrogate's fidelity is the fidelity of the detector's blind spots as much as its strengths (the tree has no rule for replay or scaling), are the two cautions the literature adds and the testbed confirms (Ghorbani et al., 2019; Slack et al.,

2020; Rudin, 2019; Nadeem et al., 2023).

Implications for energy management and resilience

The energy-management layer turns the detector into a resilience mechanism, and the economics show what that mechanism is worth and what it costs. A safe mode that holds the last good state, trusts local measurements and applies conservative setpoints for two hours after an alarm absorbs an attack at a price (in the testbed, 35 per step, 280 per alarm) and that price is paid for every true alarm and every false one. The cost model therefore has three terms that pull against each other: damage avoided (which rises with recall), safe-mode cost (which rises with recall and with false alarms) and false-alarm cost (which rises with the false-positive rate), and the tuned thresholds sit where their sum is least. In the resilience vocabulary, the detector shortens the time an attack goes unabsorbed and the safe mode bounds the performance loss during it, while false alarms impose a performance loss of their own (Panteli & Mancarella, 2015; Linkov et al., 2013; Ganin et al., 2016); a detector that is tuned without regard to the safe-mode cost makes the grid less resilient, not more, because it spends the system's tolerance for degraded operation on windows that were never attacked. Two design consequences follow. The safe-mode rule itself should be graded (a short, cheap response to a low-confidence alarm and a longer one to a confirmed attack) which turns the single threshold of this study into a policy, and makes the explanation layer part of the response: an alarm whose attribution names the packet-rate collapse calls for a communications fallback, one that names the flow jump and state residual calls for a state-estimation check before dispatch acts. And the measured outcomes of the safe mode (how often it was entered, for how long, at what cost, and whether an attack was in progress) are the data that the governance record needs and that the framework's economics would be recalibrated from; without them the cost model remains the stylisation it is here (Cárdenas et al., 2011; Pasqualetti et al., 2013; Chen et al., 2017).

The centralised reference

The centralised model is the reference that federation is usually measured against, and on the testbed it was the best detector by every ranking metric and the worst by cost. Both facts are informative. Pooling the data gave it the most examples of every class and the highest recall on replay and scaling, which is the ceiling that the federated schemes approach (within 0.01–0.02 AUROC on the uniform mix). But a single model fitted to twelve clients with different sensitivities, noise and base rates overfits the pooled normal windows in a way that the per-client standardisation does not undo: its attack probabilities on normal windows are the most concentrated near one, its calibration error the highest of the non-private schemes, and its false-positive rate at any threshold the highest (at the tuned threshold still 0.3–0.65%), which the cost model prices at 15–33 thousand per client-year. The lesson is not that centralisation is bad but that a pooled model needs per-client calibration and per-client thresholds, which is exactly what the federated schemes get for free from local training and local threshold curves (Guo et al., 2017; Niculescu-Mizil & Caruana, 2005; Rahman et al., 2020). It is also a reminder that privacy is not the only reason to keep data and decisions local: the base rates are local too.

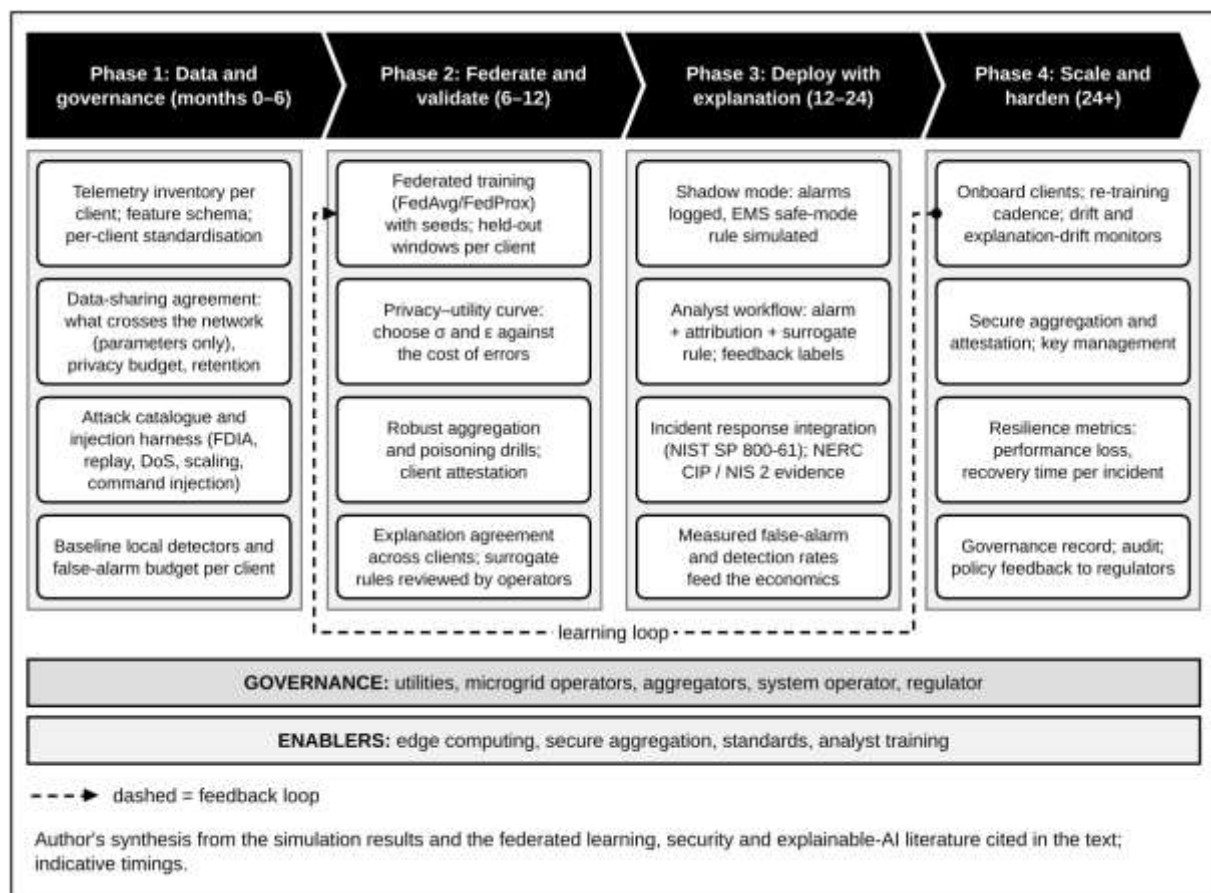
Implications for operators and federations

Three implications follow for practice. First, join a federation for the attacks you have not seen, and evaluate it on them: an operator comparing a local detector with a federated one on its own history will conclude, correctly for that history, that the local one is as good; the comparison that bears on risk is on attack classes the client has not experienced, which the uniform-mix test stylises and which a red-team catalogue would make concrete. Second, set the operating point from a cost model, and measure the two costs that the Sobol indices and the tuned thresholds make decisive (the cost of a false alarm and of safe-mode operation, and the client's attack rate) before trusting any annual value; in the testbed the choice of threshold changed the sign of the detector's value. Third, treat privacy budget and aggregation rule as design parameters with a price: the testbed's privacy-utility curve and poisoning table are the form in which that price can be shown to a data-protection officer and a security officer respectively, and the federation's size is the lever that lowers it (Kairouz et al., 2021; Mothukuri et al., 2021; Lyu et al., 2024).

Implications for standards, regulation and governance

The results connect to the standards and regulatory threads in four ways. NIST SP 800-82 and NISTIR 7628 ask operators to monitor operational-technology networks and to protect the confidentiality of grid data; federated training satisfies both at once only if the model updates are themselves protected, which is what the differential-privacy and secure-aggregation results speak to (Stouffer et al., 2023; Smart Grid Interoperability Panel, 2014; Bonawitz et al., 2017). IEC 62443 and the NERC CIP standards define security zones and the evidence an operator must keep; a federated detector crosses zones by design, and its governance record (data provenance, privacy budget, aggregation rule, incidents and explanation drift) is the evidence that the detector's behaviour can be audited (International Electrotechnical Commission, 2023; NERC, n.d.; National Institute of Standards and Technology, 2024). The EU NIS 2 Directive extends incident-reporting duties to energy operators, and the General Data Protection Regulation restricts the transfer of personal data such as consumption profiles; federation keeps the telemetry where it is, but the regulation also asks what a shared model reveals, which only a privacy budget answers (European Parliament and Council, 2022; European Parliament and Council, 2016; Wachter et al., 2018). And the incident-response and risk-management standards frame the alarm as a decision with recorded rationale (Cichonski et al., 2012; Joint Task Force Transformation Initiative, 2012; International Organization for Standardization, 2022; ISO, 2018): the surrogate rules and attributions of this study are the form in which that rationale can be recorded and reviewed. Figure 12 sets out the implementation as four phases (data and governance, federate and validate, deploy with explanation, scale and harden) with the governance bodies and enablers each needs.

Figure 12: Implementation roadmap for explainable federated attack detection in distributed smart grids: four phases from data and governance through federation and validation and deployment with explanation to scaling and hardening, with the governance bodies and enablers



Implications for analytics practice

The study illustrates five analytics practices for federated detectors. Evaluate on the mix the deployment will face, not only on each client's history: the in-mix and uniform-mix tests give opposite answers about federation, and the uniform-mix test is the one that asks whether a client is protected against attacks it has not yet seen (Sommer & Paxson, 2010; Arp et al., 2022). Tune the operating point to the cost model and report the base-rate arithmetic: at an 8% attack share, a false-positive rate of 1–3% at the default threshold produces hundreds of false alarms a year whose cost dwarfs the damage avoided, and the cost-tuned thresholds near 0.95–0.99 are what make any detector worth having (Axelsson, 2000). Account for privacy with a stated mechanism, a stated accountant and its approximations, and report what each unit of privacy costs in detection (Abadi et al., 2016; Dwork & Roth, 2014). Check explanations for agreement across schemes and clients before trusting them, and keep a surrogate rule that an analyst can read (Ghorbani et al., 2019; Rudin, 2019; Nadeem et al., 2023). And propagate the operating point into money under uncertainty with Sobol indices, so that the operator knows which cost to measure first: here, the false-alarm and safe-mode costs (Sobol', 2001; Saltelli et al., 2010; Spiegelhalter, 2017). These practices are the analytics counterpart of the federated-learning content; they are what turns a detection metric into a decision about federation, privacy and response.

Limitations

Table 15 lists the assumptions and their likely effects. The testbed is stylised throughout: the telemetry model is linear with per-client sensitivities, the attack models are representations of documented classes, and the features were designed with those classes in mind, so absolute detection rates describe the testbed: replay and scaling are hard here because their signatures are small relative to the features' day-to-day variation, and real attacks may be easier or harder. Twelve clients with 700 training windows each is a small federation with small local datasets, which favours federation on rare classes and makes record-level privacy expensive; the privacy accounting is an approximation (the leading term of the moments-accountant bound) and its ϵ values are upper bounds whose exact counterparts would be somewhat smaller. The detector is a small multilayer perceptron trained for 60 rounds with three seeds (two in the sweeps), and differences of one to two points of AUROC between schemes are within seed variation. The poisoners are strong but simple; stealthier attackers would behave differently against the median. The explanations rest on sampling-based Shapley values of seed-0 detectors and on rank agreement, which is sensitive to the many low-importance features; the surrogate's fidelity is measured on the detector's predictions, not on the truth. The cost model is stylised (fixed safe-mode duration, per-event damages, a false-alarm cost) and the thresholds tuned on validation windows at baseline costs are optimistic for a central model whose probabilities needed a finer grid; the economics omit platform capital, incident tails and discounting; and the uniform-mix test assumes every client faces every class equally, which is a stylisation of "attacks you have not seen" rather than a measurement of any threat landscape.

Further research

Five directions follow. First, calibration to real telemetry: SCADA and phasor-measurement streams from feeders and microgrids, with red-team or replayed attack traces, would replace the generators, and the public power-system datasets would anchor the attack classes (Pan et al., 2015; Borges Hink et al., 2014; Goh et al., 2017). Second, stronger and more varied detectors under the same protocol (sequence models on raw windows, physics-informed residuals from a full state estimator, one-class and anomaly detectors for unseen attacks) to raise recall on replay and scaling and to test whether federation's advantage on unseen classes persists (Özay et al., 2016; Pasqualetti et al., 2013; Liu et al., 2012; Schölkopf et al., 2001). Third, privacy and robustness together: client-level differential privacy with secure aggregation, exact accountants, personalised federated models for heterogeneous clients, and adaptive poisoners against Krum, norm clipping and attestation (McMahan et al., 2018; Truex et al., 2019; Blanchard et al., 2017; Fang et al., 2020). Fourth, explanation under adversaries: whether attributions can be manipulated by a poisoning client, how explanation drift signals model compromise, and how analysts use attributions and surrogate rules in incident response (Slack et al., 2020; Capuano et al., 2022; Nadeem et al., 2023). Fifth, resilience beyond a safe-mode rule: closed-loop energy management under detected and undetected attacks, restoration and islanding, and the

resilience metrics (performance loss and recovery time) that the response actually changes (Panteli & Mancarella, 2015; Chen et al., 2017; Zhang et al., 2023). Each would use the same framework; only the layer would change.

Conclusion

This article asked what a decision-analytics framework for explainable federated learning in distributed smart grids should contain, and what it says on a transparent testbed. The framework has four layers (physical and telemetry, federated learning and privacy, explanation, and decision, resilience and governance) and meets six requirements drawn from the smart-grid security, federated learning, differential privacy, explainable AI and resilience literatures: data that stay where they are, heterogeneity and rarity as the normal case, bounded disclosure and bounded damage, explanations an operator can check, decision outputs in money under uncertainty, and feedback with a governance record. On a stylised federation of twelve grid clients with five injected attack classes, local, central and federated detectors were within 0.03 AUROC of each other on each client's own attack mix, but only the federated and central detectors held their quality on attacks a client had not seen (0.87–0.88 against 0.74 for local models), with FedProx the better federated scheme under heterogeneity; record-level differential privacy at single-digit ϵ erased all but the strongest attack signature with 700 windows per client; one poisoning client in twelve destroyed plain averaging while the coordinate-wise median withstood three.

REFERENCES

- [1]. Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., & Zhang, L. (2016). Deep learning with differential privacy. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS '16) (pp. 308–318). ACM. <https://doi.org/10.1145/2976749.2978318>
- [2]. Abdullah Mohammad, S. (2023). CUGAO₂ and CUINS₂ Quantum Dot Sensitization for Enhanced Photovoltaic Efficiency in Wearable Fiber-Shaped Solar Cells. American Journal of Advanced Technology and Engineering Solutions, 3(04), 209-261. <https://doi.org/10.63125/wysdvy71>
- [3]. Abdullah Mohammad, S. (2024). Triboelectric And Piezoelectric Nanogenerators Using Binary and Ternary Oxide Nanostructures: Optimization of Energy Conversion Efficiency for Wearable Harvesting. Review of Applied Science and Technology, 3(03), 31-72. <https://doi.org/10.63125/w7q9cs55>
- [4]. Abur, A., & Gómez Expósito, A. (2004). Power system state estimation: Theory and implementation. CRC Press. <https://doi.org/10.1201/9780203913673>
- [5]. Adadi, A., & Berrada, M. (2018). Peeking inside the black-box: A survey on explainable artificial intelligence (XAI). IEEE Access, 6, 52138–52160. <https://doi.org/10.1109/ACCESS.2018.2870052>
- [6]. Adnan, S., & Iftikhar, A. (2023). Predicting Cost Variance in Digital Transformation Projects Through Ensemble Learning and Project Performance Analytics. International Journal of Scientific Interdisciplinary Research, 13(10), 01–36. <https://doi.org/10.63125/j18hxs51>
- [7]. Agrawal, S., Sarkar, S., Aouedi, O., Yenduri, G., Piamrat, K., Alazab, M., Bhattacharya, S., Maddikunta, P. K. R., & Gadekallu, T. R. (2022). Federated learning for intrusion detection system: Concepts, challenges and future directions. Computer Communications, 195, 346–361. <https://doi.org/10.1016/j.comcom.2022.09.012>
- [8]. Ahmed, H. C. H. R. (2024). Machine Learning Enhanced Demand Forecasting and Portfolio Analytics for Scalable Growth Management in Digital Logistics Operations. American Journal of Advanced Technology and Engineering Solutions, 4(02), 33-81. <https://doi.org/10.63125/5xq3nd53>
- [9]. Ahmed, H. C. H. R. (2026). An Artificial Intelligence Integrated IT Management Framework for Analytics-Driven Operational Decision-Making in U.S. Supply Chain and Logistics Systems. American Journal of Interdisciplinary Studies, 16(06), 01-37. <https://doi.org/10.63125/b5hgg259>
- [10]. Ahmed, H. C. H. R., & Nuzhat Sadia, P. (2023). A Quantitative Assessment of CRM Analytics and KPI Dashboard Design on Enterprise Revenue Conversion in Emerging Market Logistics Platforms. American Journal of Data Science and Analytics, 4(09), 50-95. <https://doi.org/10.63125/315bge15>
- [11]. Ahmed, M., Mahmood, A. N., & Hu, J. (2016). A survey of network anomaly detection techniques. Journal of Network and Computer Applications, 60, 19–31. <https://doi.org/10.1016/j.jnca.2015.11.016>
- [12]. Alsaedi, A., Moustafa, N., Tari, Z., Mahmood, A., & Anwar, A. (2020). TON_IoT telemetry dataset: A new generation dataset of IoT and IIoT for data-driven intrusion detection systems. IEEE Access, 8, 165130–165150. <https://doi.org/10.1109/ACCESS.2020.3022862>
- [13]. Amershi, S., Weld, D., Vorvoreanu, M., Fournery, A., Nushi, B., Collisson, P., Suh, J., Iqbal, S., Bennett, P. N., Inkpen, K., Teevan, J., Kikin-Gil, R., & Horvitz, E. (2019). Guidelines for human-AI interaction. In Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19) (pp. 1–13). ACM. <https://doi.org/10.1145/3290605.3300233>
- [14]. Aminul Islam, M. (2024). Interpreting Cyber-Threat Indicators in Enterprise Memory Forensics: A Mixed-Methods Study of Machine Learning Performance and Explainability. Review of Applied Science and Technology, 14(12), 01–13. <https://doi.org/10.63125/2p8hte92>
- [15]. Aminul Islam, M., & Md, A. (2023). Adaptive Class-Imbalance Learning for High-Sensitivity Fraud Detection in

- Large-Scale Digital Transaction Networks. *International Journal of Business and Economics Insights*, 13(09), 01-33. <https://doi.org/10.63125/bmxwv741>
- [16]. Apley, D. W., & Zhu, J. (2020). Visualizing the effects of predictor variables in black box supervised learning models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 82(4), 1059–1086. <https://doi.org/10.1111/rssb.12377>
- [17]. Arghandeh, R., von Meier, A., Mehrmanesh, L., & Mili, L. (2016). On the definition of cyber-physical resilience in power systems. *Renewable and Sustainable Energy Reviews*, 58, 1060–1069. <https://doi.org/10.1016/j.rser.2015.12.193>
- [18]. Arp, D., Quiring, E., Pendlebury, F., Warnecke, A., Pierazzi, F., Wressnegger, C., Cavallaro, L., & Rieck, K. (2022). Dos and don'ts of machine learning in computer security. In *Proceedings of the 31st USENIX Security Symposium* (pp. 3971–3988). USENIX Association. <https://www.usenix.org/conference/usenixsecurity22/presentation/arp>
- [19]. Asheke, M. (2023). Machine Learning-Based Predictive Quality Modeling for Defect Detection and Process Optimization in Advanced Manufacturing Systems. *Review of Applied Science and Technology*, 13(12), 01–49. <https://doi.org/10.63125/thy73b16>
- [20]. Asheke, M. (2024). Digital Twin and Industrial Internet of Things-Enabled Predictive Maintenance for Reliability Enhancement in Smart Manufacturing Systems. *International Journal of Scientific Interdisciplinary Research*, 5(2), 792–838. <https://doi.org/10.63125/jgrdq62>
- [21]. Ashik, M., & Shabbir, A. (2022). Digital Twin-Enabled Predictive Maintenance for Rotating and Static Equipment: Bridging FEA-Based Failure Analysis with Real-Time Asset Health Monitoring in Energy and Power Facilities. *International Journal of Scientific Interdisciplinary Research*, 12(05), 01–37. <https://doi.org/10.63125/0qaysz49>
- [22]. Ashrafuzzaman, M. (2024). The impact of cloud-based management information systems on hrm efficiency: an analysis of small and medium-sized enterprises (SMES). *Academic Journal on Artificial Intelligence, Machine Learning, Data Science and Management Information Systems*, 1(01), 40-56. <https://doi.org/10.69593/ajaimldsmis.v1i01.124>
- [23]. Ashrafuzzaman, M., Das, S., Chakhchoukh, Y., Shiva, S. G., & Sheldon, F. T. (2020). Detecting stealthy false data injection attacks in the smart grid using ensemble-based machine learning. *Computers & Security*, 97, Article 101994. <https://doi.org/10.1016/j.cose.2020.101994>
- [24]. Axelsson, S. (2000). The base-rate fallacy and the difficulty of intrusion detection. *ACM Transactions on Information and System Security*, 3(3), 186–205. <https://doi.org/10.1145/357830.357849>
- [25]. Badr, M. M., Mahmoud, M. M. E. A., Fang, Y., Abdulaal, M. J., Aljohani, A. J., Alasmay, W., & Ibrahim, M. I. (2023). Privacy-preserving and communication-efficient energy prediction scheme based on federated learning for smart grids. *IEEE Internet of Things Journal*, 10(9), 7719–7736. <https://doi.org/10.1109/JIOT.2022.3230586>
- [26]. Bagdasaryan, E., Veit, A., Hua, Y., Estrin, D., & Shmatikov, V. (2020). How to backdoor federated learning. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS)*, PMLR 108 (pp. 2938–2948). PMLR. <https://proceedings.mlr.press/v108/bagdasaryan20a.html>
- [27]. Ban, G.-Y., & Rudin, C. (2019). The big data newsvendor: Practical insights from machine learning. *Operations Research*, 67(1), 90–108. <https://doi.org/10.1287/opre.2018.1757>
- [28]. Baran, M. E., & Wu, F. F. (1989). Network reconfiguration in distribution systems for loss reduction and load balancing. *IEEE Transactions on Power Delivery*, 4(2), 1401–1407. <https://doi.org/10.1109/61.25627>
- [29]. Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., & Herrera, F. (2020). Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, 58, 82–115. <https://doi.org/10.1016/j.inffus.2019.12.012>
- [30]. Beaudin, M., & Zareipour, H. (2015). Home energy management systems: A review of modelling and complexity. *Renewable and Sustainable Energy Reviews*, 45, 318–335. <https://doi.org/10.1016/j.rser.2015.01.046>
- [31]. Bergmeir, C., & Benítez, J. M. (2012). On the use of cross-validation for time series predictor evaluation. *Information Sciences*, 191, 192–213. <https://doi.org/10.1016/j.ins.2011.12.028>
- [32]. Bertsimas, D., & Kallus, N. (2020). From predictive to prescriptive analytics. *Management Science*, 66(3), 1025–1044. <https://doi.org/10.1287/mnsc.2018.3253>
- [33]. Beutel, D. J., Topal, T., Mathur, A., Qiu, X., Fernandez-Marques, J., Gao, Y., Sani, L., Li, K. H., Parcollet, T., de Gusmão, P. P. B., & Lane, N. D. (2020). Flower: A friendly federated learning research framework (2007.14390). arXiv. <https://arxiv.org/abs/2007.14390>
- [34]. Bie, Z., Lin, Y., Li, G., & Li, F. (2017). Battling the extreme: A study on the power system resilience. *Proceedings of the IEEE*, 105(7), 1253–1266. <https://doi.org/10.1109/JPROC.2017.2679040>
- [35]. Blanchard, P., El Mhamdi, E. M., Guerraoui, R., & Stainer, J. (2017). Machine learning with adversaries: Byzantine tolerant gradient descent. In *Advances in Neural Information Processing Systems 30 (NIPS 2017)*. Curran Associates. <https://papers.nips.cc/paper/2017/hash/f4b9ec30ad9f68f89b29639786cb62ef-Abstract.html>
- [36]. Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., Ivanov, V., Kiddon, C., Konečný, J., Mazzocchi, S., McMahan, H. B., Van Overveldt, T., Petrou, D., Ramage, D., & Roselander, J. (2019). Towards federated learning at scale: System design. In *Proceedings of Machine Learning and Systems 1 (MLSys 2019)* (pp. 374–388). mlsys.org. https://proceedings.mlsys.org/paper_files/paper/2019/hash/7b770da633baf74895be22a8807f1a8f-Abstract.html
- [37]. Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H. B., Patel, S., Ramage, D., Segal, A., & Seth, K. (2017). Practical secure aggregation for privacy-preserving machine learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS '17)* (pp. 1175–1191). ACM.

- <https://doi.org/10.1145/3133956.3133982>
- [38]. Borges Hink, R., Beaver, J. M., Buckner, M. A., Morris, T., Adhikari, U., & Pan, S. (2014). Machine learning for power system disturbance and cyber-attack discrimination. In 2014 7th International Symposium on Resilient Control Systems (ISRCs) (pp. 1–8). IEEE. <https://doi.org/10.1109/ISRCs.2014.6900095>
- [39]. Brynjolfsson, E., Rock, D., & Syverson, C. (2021). The productivity J-curve: How intangibles complement general purpose technologies. *American Economic Journal: Macroeconomics*, 13(1), 333–372. <https://doi.org/10.1257/mac.20180386>
- [40]. Buczak, A. L., & Guven, E. (2016). A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications Surveys & Tutorials*, 18(2), 1153–1176. <https://doi.org/10.1109/COMST.2015.2494502>
- [41]. Campos, E. M., Saura, P. F., González-Vidal, A., Hernández-Ramos, J. L., Bernabé, J. B., Baldini, G., & Skarmeta, A. (2022). Evaluating federated learning for intrusion detection in Internet of Things: Review and challenges. *Computer Networks*, 203, Article 108661. <https://doi.org/10.1016/j.comnet.2021.108661>
- [42]. Capuano, N., Fenza, G., Loia, V., & Stanzione, C. (2022). Explainable artificial intelligence in cybersecurity: A survey. *IEEE Access*, 10, 93575–93600. <https://doi.org/10.1109/ACCESS.2022.3204171>
- [43]. Cárdenas, A. A., Amin, S., Lin, Z.-S., Huang, Y.-L., Huang, C.-Y., & Sastry, S. (2011). Attacks against process control systems: Risk assessment, detection, and response. In *Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security* (pp. 355–366). ACM. <https://doi.org/10.1145/1966913.1966959>
- [44]. Caruana, R., Lou, Y., Gehrke, J., Koch, P., Sturm, M., & Elhadad, N. (2015). Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1721–1730). ACM. <https://doi.org/10.1145/2783258.2788613>
- [45]. Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM Computing Surveys*, 41(3), Article 15. <https://doi.org/10.1145/1541880.1541882>
- [46]. Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16, 321–357. <https://doi.org/10.1613/jair.953>
- [47]. Chen, C., Wang, J., & Ton, D. (2017). Modernizing distribution system restoration to achieve grid resiliency against extreme weather events: An integrated solution. *Proceedings of the IEEE*, 105(7), 1267–1288. <https://doi.org/10.1109/JPROC.2017.2684780>
- [48]. Cheng, X., Li, C., & Liu, X. (2022). A review of federated learning in energy systems. In *2022 IEEE/IAS Industrial and Commercial Power System Asia (I&CPS Asia)* (pp. 2089–2095). IEEE. <https://doi.org/10.1109/ICPSAsia55496.2022.9949863>
- [49]. Cherepanov, A. (2017). WIN32/INDUSTROYER: A new threat for industrial control systems. ESET. https://web-assets.esetstatic.com/wls/2017/06/Win32_Industroyer.pdf
- [50]. Cichonski, P., Millar, T., Grance, T., & Scarfone, K. (2012). Computer security incident handling guide (NIST Special Publication 800-61 Rev. 2). National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.SP.800-61r2>
- [51]. Cui, S., Han, Z., Kar, S., Kim, T. T., Poor, H. V., & Tager, A. (2012). Coordinated data-injection attack and detection in the smart grid: A detailed look at enriching detection solutions. *IEEE Signal Processing Magazine*, 29(5), 106–115. <https://doi.org/10.1109/MSP.2012.2185911>
- [52]. Davis, J., & Goadrich, M. (2006). The relationship between precision-recall and ROC curves. In *Proceedings of the 23rd International Conference on Machine Learning* (pp. 233–240). ACM. <https://doi.org/10.1145/1143844.1143874>
- [53]. Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7, 1–30. <https://jmlr.org/papers/v7/demsar06a.html>
- [54]. Deng, R., Xiao, G., Lu, R., Liang, H., & Vasilakos, A. V. (2017). False data injection on state estimation in power systems, attacks, impacts, and defense: A survey. *IEEE Transactions on Industrial Informatics*, 13(2), 411–423. <https://doi.org/10.1109/TII.2016.2614396>
- [55]. Doshi-Velez, F., & Kim, B. (2017). Towards a rigorous science of interpretable machine learning (arXiv:1702.08608). arXiv. <https://arxiv.org/abs/1702.08608>
- [56]. Dwork, C., & Roth, A. (2014). The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3–4), 211–407. <https://doi.org/10.1561/04000000042>
- [57]. Dwork, C., McSherry, F., Nissim, K., & Smith, A. (2006). Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography (TCC 2006)*, *Lecture Notes in Computer Science* 3876 (pp. 265–284). Springer. https://doi.org/10.1007/11681878_14
- [58]. Elmachtoub, A. N., & Grigas, P. (2022). Smart "predict, then optimize". *Management Science*, 68(1), 9–26. <https://doi.org/10.1287/mnsc.2020.3922>
- [59]. Esmalifalak, M., Liu, L., Nguyen, N., Zheng, R., & Han, Z. (2017). Detecting stealthy false data injection using machine learning in smart grid. *IEEE Systems Journal*, 11(3), 1644–1652. <https://doi.org/10.1109/JSYST.2014.2341597>
- [60]. European Parliament and Council. (2016). Regulation (EU) 2016/679 on the protection of natural persons with regard to the processing of personal data (General Data Protection Regulation). *Official Journal of the European Union*, L 119, 1–88. <https://eur-lex.europa.eu/eli/reg/2016/679/oj>
- [61]. European Parliament and Council. (2022). Directive (EU) 2022/2555 on measures for a high common level of cybersecurity across the Union (NIS 2 Directive). *Official Journal of the European Union*, L 333, 80–152. <https://eur->

- lex.europa.eu/eli/dir/2022/2555/oj
- [62]. Falliere, N., Murchu, L. O., & Chien, E. (2011). W32.Stuxnet dossier (Version 1.4). Symantec Corporation. <https://docs.broadcom.com/doc/security-response-w32-stuxnet-dossier-11-en>
- [63]. Fang, M., Cao, X., Jia, J., & Gong, N. (2020). Local model poisoning attacks to Byzantine-robust federated learning. In Proceedings of the 29th USENIX Security Symposium (USENIX Security 20) (pp. 1605–1622). USENIX Association. <https://www.usenix.org/conference/usenixsecurity20/presentation/fang>
- [64]. Farivar, M., & Low, S. H. (2013). Branch flow model: Relaxations and convexification, Part I. IEEE Transactions on Power Systems, 28(3), 2554–2564. <https://doi.org/10.1109/TPWRS.2013.2255317>
- [65]. Fawcett, T. (2006). An introduction to ROC analysis. Pattern Recognition Letters, 27(8), 861–874. <https://doi.org/10.1016/j.patrec.2005.10.010>
- [66]. Fekri, M. N., Grolinger, K., & Mir, S. (2022). Distributed load forecasting using smart meter data: Federated learning with recurrent neural networks. International Journal of Electrical Power & Energy Systems, 137, Article 107669. <https://doi.org/10.1016/j.ijepes.2021.107669>
- [67]. Ferrag, M. A., Maglaras, L., Moschogiannis, S., & Janicke, H. (2020). Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study. Journal of Information Security and Applications, 50, 102419. <https://doi.org/10.1016/j.jisa.2019.102419>
- [68]. Fildes, R., Goodwin, P., Lawrence, M., & Nikolopoulos, K. (2009). Effective forecasting and judgmental adjustments: An empirical evaluation and strategies for improvement in supply-chain planning. International Journal of Forecasting, 25(1), 3–23. <https://doi.org/10.1016/j.ijforecast.2008.11.010>
- [69]. Fisher, A., Rudin, C., & Dominici, F. (2019). All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. Journal of Machine Learning Research, 20(177), 1–81. <https://jmlr.org/papers/v20/18-760.html>
- [70]. Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. Annals of Statistics, 29(5), 1189–1232. <https://doi.org/10.1214/aos/1013203451>
- [71]. Galloway, B., & Hancke, G. P. (2013). Introduction to industrial control networks. IEEE Communications Surveys & Tutorials, 15(2), 860–880. <https://doi.org/10.1109/SURV.2012.071812.00124>
- [72]. Ganin, A. A., Massaro, E., Gutfraind, A., Steen, N., Keisler, J. M., Kott, A., Mangoubi, R., & Linkov, I. (2016). Operational resilience: Concepts, design and analysis. Scientific Reports, 6, 19540. <https://doi.org/10.1038/srep19540>
- [73]. Gazi Yeasin, A. (2023). Impact of Accessibility-First Design Systems on WCAG 2.1 Compliance and Inclusive User Experience in Enterprise Software Platforms: A Mixed-Methods Evaluation. Review of Applied Science and Technology, 13(07), 01–49. <https://doi.org/10.63125/nxh4gp19>
- [74]. Geiping, J., Bauermeister, H., Dröge, H., & Moeller, M. (2020). Inverting gradients: How easy is it to break privacy in federated learning? In Advances in Neural Information Processing Systems 33 (NeurIPS 2020) (pp. 16937–16947). Curran Associates. <https://proceedings.neurips.cc/paper/2020/hash/c4ede56bbd98819ae6112b20ac6bf145-Abstract.html>
- [75]. Geyer, R. C., Klein, T., & Nabi, M. (2017). Differentially private federated learning: A client level perspective (1712.07557). arXiv. <https://arxiv.org/abs/1712.07557>
- [76]. Ghorbani, A., Abid, A., & Zou, J. (2019). Interpretation of neural networks is fragile. In Proceedings of the AAAI Conference on Artificial Intelligence, 33(1) (pp. 3681–3688). AAAI Press. <https://doi.org/10.1609/aaai.v33i01.33013681>
- [77]. Giani, A., Bitar, E., Garcia, M., McQueen, M., Khargonekar, P., & Poolla, K. (2013). Smart grid data integrity attacks. IEEE Transactions on Smart Grid, 4(3), 1244–1253. <https://doi.org/10.1109/TSG.2013.2245155>
- [78]. Giraldo, J., Urbina, D., Cardenas, A., Valente, J., Faisal, M., Ruths, J., Tippenhauer, N. O., Sandberg, H., & Candell, R. (2018). A survey of physics-based attack detection in cyber-physical systems. ACM Computing Surveys, 51(4), Article 76. <https://doi.org/10.1145/3203245>
- [79]. Goh, J., Adepu, S., Tan, M., & Lee, Z. S. (2017). Anomaly detection in cyber physical systems using recurrent neural networks. In 2017 IEEE 18th International Symposium on High Assurance Systems Engineering (HASE) (pp. 140–145). IEEE. <https://doi.org/10.1109/HASE.2017.36>
- [80]. Gregor, S., & Hevner, A. R. (2013). Positioning and presenting design science research for maximum impact. MIS Quarterly, 37(2), 337–355. <https://doi.org/10.25300/MISQ/2013/37.2.01>
- [81]. Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., & Pedreschi, D. (2018). A survey of methods for explaining black box models. ACM Computing Surveys, 51(5), Article 93. <https://doi.org/10.1145/3236009>
- [82]. Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017). On calibration of modern neural networks. In Proceedings of the 34th International Conference on Machine Learning (PMLR 70, pp. 1321–1330). <https://proceedings.mlr.press/v70/guo17a.html>
- [83]. Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., Fernández del Río, J., Wiebe, M., Peterson, P., . . . Oliphant, T. E. (2020). Array programming with NumPy. Nature, 585(7825), 357–362. <https://doi.org/10.1038/s41586-020-2649-2>
- [84]. Hatziargyriou, N. D., Asano, H., Irvani, R., & Marnay, C. (2007). Microgrids. IEEE Power and Energy Magazine, 5(4), 78–94. <https://doi.org/10.1109/MPAE.2007.376583>
- [85]. Haug, A., Zachariassen, F., & van Liempd, D. (2011). The costs of poor data quality. Journal of Industrial Engineering

- and Management, 4(2), 168–193. <https://doi.org/10.3926/jiem.2011.v4n2.p168-193>
- [86]. Hazen, B. T., Boone, C. A., Ezell, J. D., & Jones-Farmer, L. A. (2014). Data quality for data science, predictive analytics, and big data in supply chain management: An introduction to the problem and suggestions for research and applications. *International Journal of Production Economics*, 154, 72–80. <https://doi.org/10.1016/j.ijpe.2014.04.018>
- [87]. He, H., & Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9), 1263–1284. <https://doi.org/10.1109/TKDE.2008.239>
- [88]. He, Y., Mendis, G. J., & Wei, J. (2017). Real-time detection of false data injection attacks in smart grid: A deep learning-based intelligent mechanism. *IEEE Transactions on Smart Grid*, 8(5), 2505–2516. <https://doi.org/10.1109/TSG.2017.2703842>
- [89]. Henderson, P., Islam, R., Bachman, P., Pineau, J., Precup, D., & Meger, D. (2018). Deep reinforcement learning that matters. In *Proceedings of the AAAI Conference on Artificial Intelligence* (pp. 3207–3214). AAAI Press. <https://doi.org/10.1609/aaai.v32i1.11694>
- [90]. Herman, J., & Usher, W. (2017). SALib: An open-source Python library for sensitivity analysis. *Journal of Open Source Software*, 2(9), Article 97. <https://doi.org/10.21105/joss.00097>
- [91]. Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly*, 28(1), 75–105. <https://doi.org/10.2307/25148625>
- [92]. Howard, R. A. (1966). Information value theory. *IEEE Transactions on Systems Science and Cybernetics*, 2(1), 22–26. <https://doi.org/10.1109/TSSC.1966.300074>
- [93]. Hsu, T.-M. H., Qi, H., & Brown, M. (2019). Measuring the effects of non-identical data distribution for federated visual classification (1909.06335). *arXiv*. <https://arxiv.org/abs/1909.06335>
- [94]. Hubbard, D. W., & Seiersen, R. (2016). *How to measure anything in cybersecurity risk*. John Wiley & Sons. <https://doi.org/10.1002/9781119162315>
- [95]. Hüllermeier, E., & Waegeman, W. (2021). Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine Learning*, 110(3), 457–506. <https://doi.org/10.1007/s10994-021-05946-3>
- [96]. Humayed, A., Lin, J., Li, F., & Luo, B. (2017). Cyber-physical systems security: A survey. *IEEE Internet of Things Journal*, 4(6), 1802–1831. <https://doi.org/10.1109/JIOT.2017.2703172>
- [97]. Husnoo, M. A., Anwar, A., Reda, H. T., Hosseinzadeh, N., Islam, S. N., Mahmood, A. N., & Doss, R. (2023). FedDiSC: A computation-efficient federated learning framework for power systems disturbance and cyber attack discrimination. *Energy and AI*, 14, Article 100271. <https://doi.org/10.1016/j.egyai.2023.100271>
- [98]. Industrial Control Systems Cyber Emergency Response Team. (2016). *Cyber-attack against Ukrainian critical infrastructure (Alert IR-ALERT-H-16-056-01)*. Cybersecurity and Infrastructure Security Agency. <https://www.cisa.gov/uscert/ics/alerts/IR-ALERT-H-16-056-01>
- [99]. Inoue, J., Yamagata, Y., Chen, Y., Poskitt, C. M., & Sun, J. (2017). Anomaly detection for a water treatment system using unsupervised machine learning. In *2017 IEEE International Conference on Data Mining Workshops (ICDMW)* (pp. 1058–1065). IEEE. <https://doi.org/10.1109/ICDMW.2017.149>
- [100]. International Electrotechnical Commission. (2023). Understanding IEC 62443. International Electrotechnical Commission. <https://www.iec.ch/blog/understanding-iec-62443>
- [101]. International Organization for Standardization. (2018). *ISO 31000:2018 Risk management: Guidelines*. ISO. <https://www.iso.org/standard/65694.html>
- [102]. International Organization for Standardization. (2022). *Information security, cybersecurity and privacy protection: Guidance on managing information security risks (ISO/IEC 27005:2022)*. International Organization for Standardization. <https://www.iso.org/standard/80585.html>
- [103]. Iwanaga, T., Usher, W., & Herman, J. (2022). Toward SALib 2.0: Advancing the accessibility and interpretability of global sensitivity analyses. *Socio-Environmental Systems Modelling*, 4, Article 18155. <https://doi.org/10.18174/sesmo.18155>
- [104]. Jahanara, A. (2025). AI-Enabled Workforce Management: Its Impact on Employee Burnout, Staffing Efficiency, and Organizational Performance in Healthcare. *Review of Applied Science and Technology*, 4(04), 247–295. <https://doi.org/10.63125/5617ff74>
- [105]. Jahanara, A. (2026). Impact of Artificial Intelligence on Operational Efficiency and Cost Reduction in U.S. Healthcare Organizations. *American Journal of Interdisciplinary Studies*, 7(02), 225–276. <https://doi.org/10.63125/3p24ma43>
- [106]. Jahanara, A., & Mst Kaniz, F. (2023). Demand Variability and Capacity Planning in Home and Community-Based Services: A Decision-Analytic Framework for Health Services Managers. *American Journal of Data Science and Analytics*, 4(09), 01–49. <https://doi.org/10.63125/3y62vr16>
- [107]. Jannatul, F. (2024). Alzheimer's Disease and Mobility Multimodal Prediction of Gait Decline, Fall Risk, and Loss of Functional Independence in Mild Cognitive Impairment and Early Alzheimer's Disease. *Review of Applied Science and Technology*, 3(01), 420–483. <https://doi.org/10.63125/q2d90m63>
- [108]. Jannatul, F. (2025). Spinal Cord Injury and Neurorehabilitation Deep Learning-Based Prediction of Functional Recovery Trajectories After Traumatic Spinal Cord Injury. *ASRC Procedia: Global Perspectives in Science and Scholarship*, 1(01), 2865–2922. <https://doi.org/10.63125/m738bh82>
- [109]. Jannatul, F. (2026). Predicting Falls and Mobility Decline Among Older Adults Using Longitudinal Functional Data and Ensemble Machine Learning. *International Journal of Scientific Interdisciplinary Research*, 7(1), 581–631. <https://doi.org/10.63125/56hans40>
- [110]. Jithish, J., Alangot, B., Mahalingam, N., & Yeo, K. S. (2023). Distributed anomaly detection in smart grids: A federated

- learning-based approach. *IEEE Access*, 11, 7157–7179. <https://doi.org/10.1109/ACCESS.2023.3237554>
- [111]. Joint Task Force Transformation Initiative. (2012). Guide for conducting risk assessments (NIST Special Publication 800-30 Rev. 1). National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.SP.800-30r1>
- [112]. Kahneman, D. (2011). Thinking, fast and slow. Farrar, Straus and Giroux. <https://openlibrary.org/isbn/9780374275631>
- [113]. Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., D'Oliveira, R. G. L., Eichner, H., El Rouayheb, S., Evans, D., Gardner, J., Garrett, Z., Gascón, A., Ghazi, B., Gibbons, P. B., ... Zhao, S. (2021). Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 14(1-2), 1–210. <https://doi.org/10.1561/22000000083>
- [114]. Kaloudi, N., & Li, J. (2020). The AI-based cyber threat landscape: A survey. *ACM Computing Surveys*, 53(1), Article 20. <https://doi.org/10.1145/3372823>
- [115]. Kaplan, S., & Garrick, B. J. (1981). On the quantitative definition of risk. *Risk Analysis*, 1(1), 11–27. <https://doi.org/10.1111/j.1539-6924.1981.tb01350.x>
- [116]. Karimireddy, S. P., Kale, S., Mohri, M., Reddi, S., Stich, S., & Suresh, A. T. (2020). SCAFFOLD: Stochastic controlled averaging for federated learning. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, PMLR 119 (pp. 5132–5143). PMLR. <https://proceedings.mlr.press/v119/karimireddy20a.html>
- [117]. Kaufman, S., Rosset, S., Perlich, C., & Stitelman, O. (2012). Leakage in data mining: Formulation, detection, and avoidance. *ACM Transactions on Knowledge Discovery from Data*, 6(4), Article 15. <https://doi.org/10.1145/2382577.2382579>
- [118]. Kazi Rakib Hasan, S., & Adnan, S. (2022). Quantitative Assessment of Machine Learning-Based Schedule Risk Prediction for Reducing Delays in Complex Technology Projects. *Review of Applied Science and Technology*, 1(04), 464–501. <https://doi.org/10.63125/dn8nkm85>
- [119]. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. *Advances in Neural Information Processing Systems*, 30, 3146–3154. <https://papers.nips.cc/paper/6907-lightgbm-a-highly-efficient-gradient-boosting-decision-tree>
- [120]. Kendall, A., & Gal, Y. (2017). What uncertainties do we need in Bayesian deep learning for computer vision? In *Advances in Neural Information Processing Systems 30 (NeurIPS 2017)*. Curran Associates. https://proceedings.neurips.cc/paper_files/paper/2017/hash/2650d6089a6d640c5e85b2b88265dc2b-Abstract.html
- [121]. Kim, T. T., & Poor, H. V. (2011). Strategic protection against data injection attacks on power grids. *IEEE Transactions on Smart Grid*, 2(2), 326–333. <https://doi.org/10.1109/TSG.2011.2119336>
- [122]. Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations (ICLR 2015)*. arXiv. <https://arxiv.org/abs/1412.6980>
- [123]. Koroniotis, N., Moustafa, N., Sitnikova, E., & Turnbull, B. (2019). Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset. *Future Generation Computer Systems*, 100, 779–796. <https://doi.org/10.1016/j.future.2019.05.041>
- [124]. Kosut, O., Jia, L., Thomas, R. J., & Tong, L. (2011). Malicious data attacks on the smart grid. *IEEE Transactions on Smart Grid*, 2(4), 645–658. <https://doi.org/10.1109/TSG.2011.2163807>
- [125]. Kravchik, M., & Shabtai, A. (2018). Detecting cyber attacks in industrial control systems using convolutional neural networks. In *Proceedings of the 2018 Workshop on Cyber-Physical Systems Security and PrivaCy* (pp. 72–83). ACM. <https://doi.org/10.1145/3264888.3264896>
- [126]. Kumar, S., Soni, M. K., & Jain, D. K. (2015). Cyber security threats in synchrophasor system in WAMS. *International Journal of Computer Applications*, 115(8), 17–22. <https://doi.org/10.5120/20172-2355>
- [127]. Kynatun, M., & Saba, A. (2024). Machine Learning Enhanced Intrusion Detection and Anomaly Classification for MIS-Driven Enterprise Network Security. *International Journal of Scientific Interdisciplinary Research*, 14(09), 01–41. <https://doi.org/10.63125/xk6kpc72>
- [128]. Langner, R. (2011). Stuxnet: Dissecting a cyberwarfare weapon. *IEEE Security & Privacy*, 9(3), 49–51. <https://doi.org/10.1109/MSP.2011.67>
- [129]. Law, A. M. (2015). Simulation modeling and analysis (5th ed.). McGraw-Hill Education. <https://www.mheducation.com/>
- [130]. Lee, R. M., Assante, M. J., & Conway, T. (2016). Analysis of the cyber attack on the Ukrainian power grid: Defense use case. SANS Institute and Electricity Information Sharing and Analysis Center (E-ISAC). https://media.kasperskycontenthub.com/wp-content/uploads/sites/43/2016/05/20081514/E-ISAC_SANS_Ukraine_DUC_5.pdf
- [131]. Li, T., Sahu, A. K., Talwalkar, A., & Smith, V. (2020b). Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3), 50–60. <https://doi.org/10.1109/MSP.2020.2975749>
- [132]. Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., & Smith, V. (2020c). Federated optimization in heterogeneous networks. In *Proceedings of Machine Learning and Systems 2 (MLSys 2020)* (pp. 429–450). mlsys.org. https://proceedings.mlsys.org/paper_files/paper/2020/hash/1f5fe83998a09396ebe6477d9475ba0c-Abstract.html
- [133]. Li, Y., Wei, X., Li, Y., Dong, Z., & Shahidehpour, M. (2022). Detection of false data injection attacks in smart grid: A secure federated deep learning approach. *IEEE Transactions on Smart Grid*, 13(6), 4862–4872. <https://doi.org/10.1109/TSG.2022.3204796>
- [134]. Liang, G., Zhao, J., Luo, F., Weller, S. R., & Dong, Z. Y. (2017). A review of false data injection attacks against modern

- power systems. *IEEE Transactions on Smart Grid*, 8(4), 1630–1638. <https://doi.org/10.1109/TSG.2015.2495133>
- [135]. Lim, W. Y. B., Luong, N. C., Hoang, D. T., Jiao, Y., Liang, Y.-C., Yang, Q., Niyato, D., & Miao, C. (2020). Federated learning in mobile edge networks: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 22(3), 2031–2063. <https://doi.org/10.1109/COMST.2020.2986024>
- [136]. Linkov, I., Eisenberg, D. A., Plourde, K., Seager, T. P., Allen, J., & Kott, A. (2013). Resilience metrics for cyber systems. *Environment Systems and Decisions*, 33(4), 471–476. <https://doi.org/10.1007/s10669-013-9485-y>
- [137]. Lipton, Z. C. (2018). The mythos of model interpretability. *Queue*, 16(3), 31–57. <https://doi.org/10.1145/3236386.3241340>
- [138]. Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2012). Isolation-based anomaly detection. *ACM Transactions on Knowledge Discovery from Data*, 6(1), Article 3. <https://doi.org/10.1145/2133360.2133363>
- [139]. Liu, Y., Ning, P., & Reiter, M. K. (2011). False data injection attacks against state estimation in electric power grids. *ACM Transactions on Information and System Security*, 14(1), Article 13. <https://doi.org/10.1145/1952982.1952995>
- [140]. Lou, Y., Caruana, R., & Gehrke, J. (2012). Intelligible models for classification and regression. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 150–158). ACM. <https://doi.org/10.1145/2339530.2339556>
- [141]. Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30, 4765–4774. <https://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions>
- [142]. Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., Katz, R., Himmelfarb, J., Bansal, N., & Lee, S.-I. (2020). From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, 2(1), 56–67. <https://doi.org/10.1038/s42256-019-0138-9>
- [143]. Lyu, L., Yu, H., Ma, X., Chen, C., Sun, L., Zhao, J., Yang, Q., & Yu, P. S. (2024). Privacy and robustness in federated learning: Attacks and defenses. *IEEE Transactions on Neural Networks and Learning Systems*, 35(7), 8726–8746. <https://doi.org/10.1109/TNNLS.2022.3216981>
- [144]. Mackiewicz, R. E. (2006). Overview of IEC 61850 and benefits. In *2006 IEEE PES Power Systems Conference and Exposition* (pp. 623–630). IEEE. <https://doi.org/10.1109/PSCE.2006.296392>
- [145]. Mahbooba, B., Timilsina, M., Sahal, R., & Serrano, M. (2021). Explainable artificial intelligence (XAI) to enhance trust management in intrusion detection systems using decision tree model. *Complexity*, 2021, Article 6634811. <https://doi.org/10.1155/2021/6634811>
- [146]. Maishar Mostari, Z. (2025). Impact of Deep Learning on Real-Time Network Intrusion Detection: A Quantitative Evaluation Using Benchmark Datasets. *Review of Applied Science and Technology*, 4(04), 296–338. <https://doi.org/10.63125/taksd40>
- [147]. Mansura Akter, E., & Sumaiya, M. (2022). Antimicrobial Resistance Profiles of *Acinetobacter Baumannii* and *Vibrio Cholerae* in Hospital Wastewater and Adjacent Household Water Sources. *American Journal of Data Science and Analytics*, 3(09), 01-55. <https://doi.org/10.63125/gbervt14>
- [148]. Mashfiqu, R. (2025). Multivariate Time-Series Forecasting of Working Capital Requirements for Improving Cash-Flow Stability in Export-Oriented Manufacturing. *American Journal of Data Science and Analytics*, 6(12), 217-249. <https://doi.org/10.63125/ty3dtt46>
- [149]. Mashfiqu, R. (2026). Explainable Artificial Intelligence for Supplier Cost-Risk Assessment and Margin Optimization in Global Manufacturing Supply Chains. *American Journal of Interdisciplinary Studies*, 16(08), 01-37. <https://doi.org/10.63125/np6bw920>
- [150]. Mashfiqu, R., & Md, A. (2022). Monte Carlo Simulation and Stochastic Cash-Flow Modeling for Reducing Liquidity Risk in Project-Based Enterprises. *International Journal of Scientific Interdisciplinary Research*, 1(01), 382–431. <https://doi.org/10.63125/zf3fwv21>
- [151]. Mashiat, N. (2024). Reducing Telecommunications Service Interruptions Through Predictive Root-Cause Analytics of Multivariate System Telemetry. *International Journal of Scientific Interdisciplinary Research*, 14(11), 01-37. <https://doi.org/10.63125/nqjh0c29>
- [152]. Mashiat, N. (2025). Improving Enterprise Data Integrity Through Automated Validation and Schema-Drift Detection in Large-Scale Data Integration Systems. *Review of Applied Science and Technology*, 4(02), 889–934. <https://doi.org/10.63125/21zzkk04>
- [153]. Mashiat, N., & Md Anisur Rahman, M. (2022). Machine Learning Approaches to Credit Risk Scoring and Default Prediction in Small-Business Lending: Empirical Evidence from Lending Portfolio Data. *American Journal of Data Science and Analytics*, 3(06), 137-182. <https://doi.org/10.63125/j256rs96>
- [154]. McDaniel, P., & McLaughlin, S. (2009). Security and privacy challenges in the smart grid. *IEEE Security & Privacy*, 7(3), 75–77. <https://doi.org/10.1109/MSP.2009.76>
- [155]. McLaughlin, S., Konstantinou, C., Wang, X., Davi, L., Sadeghi, A.-R., Maniatakos, M., & Karri, R. (2016). The cybersecurity landscape in industrial control systems. *Proceedings of the IEEE*, 104(5), 1039–1057. <https://doi.org/10.1109/JPROC.2015.2512235>
- [156]. McMahan, B., Moore, E., Ramage, D., Hampson, S., & Agüera y Arcas, B. (2017). Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, PMLR 54 (pp. 1273–1282). PMLR. <https://proceedings.mlr.press/v54/mcmahan17a.html>
- [157]. McMahan, H. B., Ramage, D., Talwar, K., & Zhang, L. (2018). Learning differentially private recurrent language models. In *International Conference on Learning Representations (ICLR 2018)*. OpenReview.

- <https://openreview.net/forum?id=BJ0hF1Z0b>
- [158]. Md Anisur Rahman, M. (2023). Development of AI-Driven Business Intelligence Frameworks for Financial Risk and Liquidity Monitoring in Real Time. *International Journal of Scientific Interdisciplinary Research*, 4(2), 152–203. <https://doi.org/10.63125/jm55ca64>
 - [159]. Md Arif Uz, Z., Sharmin, S., & Md Abdur, R. (2025). Factors Influencing Behavioural Intention to Use the Innovative Open And Distance Learning Systems With The Role Of Continuance Intention. *American Journal of Scholarly Research and Innovation*, 4(01), 202–220. <https://doi.org/10.63125/pbjxp014>
 - [160]. Md Ashfaq, S., & Abdullah Mohammad, S. (2022). CNT/PANI Composite Electrodes for Flexible Fiber-Shaped Dye-Sensitized Solar Cells: Synthesis, Characterization, and Photovoltaic Performance. *American Journal of Interdisciplinary Studies*, 3(02), 221–274. <https://doi.org/10.63125/m334mf41>
 - [161]. Md Asif Ali Sheak, A., & Tahmedur, R. (2023). A Data-Driven Framework for Engineering Project Risk Assessment Under Schedule and Cost Uncertainty. *International Journal of Scientific Interdisciplinary Research*, 4(2), 204–251. <https://doi.org/10.63125/kq4zzh16>
 - [162]. Md Imran, H., & Jannatul, F. (2021). Translational Stroke and Neurorehabilitation Research Integrating Neuroimaging, Circulating Biomarkers, and Functional Assessments to Predict Recovery After Stroke. *American Journal of Data Science and Analytics*, 2(12), 156–198. <https://doi.org/10.63125/c28rzv23>
 - [163]. Md Imran, H., & Jannatul, F. (2022). Molecular and Precision Rehabilitation Molecular Mechanisms Linking Mitochondrial Dysfunction, Inflammation, and Muscle Degeneration with Functional Decline and Rehabilitation Response in Neuromuscular Disease. *American Journal of Interdisciplinary Studies*, 12(08), 01–44. <https://doi.org/10.63125/0ym5bh79>
 - [164]. Md Kaium, H. (2023). A GIS-Based Multi-Criteria Decision-Support and Land-Suitability Optimization Framework for Climate-Resilient Urban and Regional Planning. *American Journal of Data Science and Analytics*, 4(06), 43–103. <https://doi.org/10.63125/wvq0sx87>
 - [165]. Md Mahamud Pervaz, P. (2023). Integration of MES, SAP, And PI Historian Platforms for Real-Time Industrial Asset Performance Management. *Review of Applied Science and Technology*, 2(04), 393–445. <https://doi.org/10.63125/9ebtas97>
 - [166]. Md Mahamud Pervaz, P. (2025). Development of a CMMS-Integrated Predictive Maintenance Framework for Industrial PLC and Drive Systems. *American Journal of Scholarly Research and Innovation*, 4(01), 894–942. <https://doi.org/10.63125/j33gtf31>
 - [167]. Md, A., & Gazi Yeasin, A. (2022). Machine Learning Enhanced Spatial-Data Workflow Redesign: A Quantitative Evaluation of Task Success and Operational Cost Reduction in Enterprise LiDAR Platforms. *American Journal of Data Science and Analytics*, 3(06), 80–136. <https://doi.org/10.63125/0den7c02>
 - [168]. Md. Morshedul, I., & Aminul Islam, M. (2022). A Quantitative Evaluation of Explainable Ensemble Learning for Probabilistic Risk Estimation and Decision Reliability in Enterprise Analytics. *American Journal of Data Science and Analytics*, 3(02), 01–38. <https://doi.org/10.63125/k22shk74>
 - [169]. Meidan, Y., Bohadana, M., Mathov, Y., Mirsky, Y., Shabtai, A., Breitenbacher, D., & Elovici, Y. (2018). N-BaIoT: Network-based detection of IoT botnet attacks using deep autoencoders. *IEEE Pervasive Computing*, 17(3), 12–22. <https://doi.org/10.1109/MPRV.2018.03367731>
 - [170]. Melis, L., Song, C., De Cristofaro, E., & Shmatikov, V. (2019). Exploiting unintended feature leakage in collaborative learning. In *2019 IEEE Symposium on Security and Privacy (SP)* (pp. 691–706). IEEE. <https://doi.org/10.1109/SP.2019.00029>
 - [171]. Mikalef, P., Boura, M., Lekakos, G., & Krogstie, J. (2019). Big data analytics and firm performance: Findings from a mixed-method approach. *Journal of Business Research*, 98, 261–276. <https://doi.org/10.1016/j.jbusres.2019.01.044>
 - [172]. Miller, T. (2019). Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267, 1–38. <https://doi.org/10.1016/j.artint.2018.07.007>
 - [173]. Mirsky, Y., Demontis, A., Kotak, J., Shankar, R., Gelei, D., Yang, L., Zhang, X., Pintor, M., Lee, W., Elovici, Y., & Biggio, B. (2023). The threat of offensive AI to organizations. *Computers & Security*, 124, 103006. <https://doi.org/10.1016/j.cose.2022.103006>
 - [174]. Mo, Y., Kim, T. H.-J., Brancik, K., Dickinson, D., Lee, H., Perrig, A., & Sinopoli, B. (2012). Cyber-physical security of a smart grid infrastructure. *Proceedings of the IEEE*, 100(1), 195–209. <https://doi.org/10.1109/JPROC.2011.2161428>
 - [175]. Mohammad Badrul, A., & Md. Mominul, H. (2023). Machine Learning-Based Assessment of Environmental and Public Health Risks in Sewerage Sludge Management Systems. *American Journal of Advanced Technology and Engineering Solutions*, 3(02), 33–77. <https://doi.org/10.63125/y3yfxa92>
 - [176]. Molina-Markham, A., Shenoy, P., Fu, K., Cecchet, E., & Irwin, D. (2010). Private memoirs of a smart meter. In *Proceedings of the 2nd ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Building (BuildSys '10)* (pp. 61–66). ACM. <https://doi.org/10.1145/1878431.1878446>
 - [177]. Molnar, C. (2025). *Interpretable machine learning: A guide for making black box models explainable* (3rd ed.). Independently published. <https://christophm.github.io/interpretable-ml-book/>
 - [178]. Morgan, M. G., & Henrion, M. (1990). *Uncertainty: A guide to dealing with uncertainty in quantitative risk and policy analysis*. Cambridge University Press. <https://doi.org/10.1017/CBO9780511840609>
 - [179]. Mothukuri, V., Khare, P., Parizi, R. M., Pouriyeh, S., Dehghantanha, A., & Srivastava, G. (2022). Federated-learning-based anomaly detection for IoT security attacks. *IEEE Internet of Things Journal*, 9(4), 2545–2554. <https://doi.org/10.1109/JIOT.2021.3077803>
 - [180]. Mothukuri, V., Parizi, R. M., Pouriyeh, S., Huang, Y., Dehghantanha, A., & Srivastava, G. (2021). A survey on security

- and privacy of federated learning. *Future Generation Computer Systems*, 115, 619–640. <https://doi.org/10.1016/j.future.2020.10.007>
- [181]. Moustafa, N., & Slay, J. (2015). UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In *2015 Military Communications and Information Systems Conference (MilCIS)* (pp. 1–6). IEEE. <https://doi.org/10.1109/MilCIS.2015.7348942>
- [182]. Mst Kaniz, F., & Jahanara, A. (2022). Lean Six Sigma Implementation in Home Health Administrative Operations: A DMAIC Case Study of Managerial Workflow Redesign and Error Reduction. *International Journal of Scientific Interdisciplinary Research*, 12(04), 01–60. <https://doi.org/10.63125/xchca204>
- [183]. Musleh, A. S., Chen, G., & Dong, Z. Y. (2020). A survey on the detection algorithms for false data injection attacks in smart grids. *IEEE Transactions on Smart Grid*, 11(3), 2218–2234. <https://doi.org/10.1109/TSG.2019.2949998>
- [184]. Nadeau, C., & Bengio, Y. (2003). Inference for the generalization error. *Machine Learning*, 52(3), 239–281. <https://doi.org/10.1023/A:1024068626366>
- [185]. Nadeem, A., Vos, D., Cao, C., Pajola, L., Dieck, S., Baumgartner, R., & Verwer, S. (2023). SoK: Explainable machine learning for computer security applications. In *2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P)* (pp. 221–240). IEEE. <https://doi.org/10.1109/EuroSP57164.2023.00022>
- [186]. Naeini, M. P., Cooper, G. F., & Hauskrecht, M. (2015). Obtaining well calibrated probabilities using Bayesian binning. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence* (pp. 2901–2907). AAAI Press. <https://doi.org/10.1609/aaai.v29i1.9602>
- [187]. Nasr, M., Shokri, R., & Houmansadr, A. (2019). Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *2019 IEEE Symposium on Security and Privacy (SP)* (pp. 739–753). IEEE. <https://doi.org/10.1109/SP.2019.00065>
- [188]. National Institute of Standards and Technology. (2024). The NIST Cybersecurity Framework (CSF) 2.0 (NIST CSWP 29). National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.CSWP.29>
- [189]. Neupane, S., Ables, J., Anderson, W., Mittal, S., Rahimi, S., Banicescu, I., & Seale, M. (2022). Explainable intrusion detection systems (X-IDS): A survey of current methods, challenges, and opportunities (2207.06236). *arXiv*. <https://arxiv.org/abs/2207.06236>
- [190]. Nguyen, D. C., Ding, M., Pathirana, P. N., Seneviratne, A., Li, J., & Poor, H. V. (2021). Federated learning for Internet of Things: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 23(3), 1622–1658. <https://doi.org/10.1109/COMST.2021.3075439>
- [191]. Nguyen, T. D., Marchal, S., Miettinen, M., Fereidooni, H., Asokan, N., & Sadeghi, A.-R. (2019). DfIoT: A federated self-learning anomaly detection system for IoT. In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)* (pp. 756–767). IEEE. <https://doi.org/10.1109/ICDCS.2019.00080>
- [192]. Niculescu-Mizil, A., & Caruana, R. (2005). Predicting good probabilities with supervised learning. In *Proceedings of the 22nd International Conference on Machine Learning* (pp. 625–632). ACM. <https://doi.org/10.1145/1102351.1102430>
- [193]. Nishat, J. (2023). Communicable Disease Prevention in a Post-Pandemic Era: A Global Review of Public Health Education, Surveillance, and Community-Level Interventions. *International Journal of Scientific Interdisciplinary Research*, 4(4), 630–678. <https://doi.org/10.63125/mnqef36>
- [194]. North American Electric Reliability Corporation. (n.d.). CIP (Critical Infrastructure Protection) reliability standards. NERC. <https://www.nerc.com/standards/reliability-standards/cip>
- [195]. Nuzhat Sadia, P. (2023). A Multi-Criteria Decision-Support Framework For ERP-Based Business Process Optimization in Mid-Sized Manufacturing Enterprises. *American Journal of Scholarly Research and Innovation*, 2(02), 519–573. <https://doi.org/10.63125/j58k5d55>
- [196]. Olivares, D. E., Mehrizi-Sani, A., Etemadi, A. H., Cañizares, C. A., Iravani, R., Kazerani, M., Hajimiragha, A. H., Gomis-Bellmunt, O., Saeedifard, M., Palma-Behnke, R., Jiménez-Estévez, G. A., & Hatzigiargyriou, N. D. (2014). Trends in microgrid control. *IEEE Transactions on Smart Grid*, 5(4), 1905–1919. <https://doi.org/10.1109/TSG.2013.2295514>
- [197]. Özay, M., Esnaola, I., Yarman Vural, F. T., Kulkarni, S. R., & Poor, H. V. (2016). Machine learning methods for attack detection in the smart grid. *IEEE Transactions on Neural Networks and Learning Systems*, 27(8), 1773–1786. <https://doi.org/10.1109/TNNLS.2015.2404803>
- [198]. Pan, S., Morris, T., & Adhikari, U. (2015). Developing a hybrid intrusion detection system using data mining for power systems. *IEEE Transactions on Smart Grid*, 6(6), 3104–3113. <https://doi.org/10.1109/TSG.2015.2409775>
- [199]. Panteli, M., & Mancarella, P. (2015). The grid: Stronger, bigger, smarter? Presenting a conceptual framework of power system resilience. *IEEE Power and Energy Magazine*, 13(3), 58–66. <https://doi.org/10.1109/MPE.2015.2397334>
- [200]. Parhizi, S., Lotfi, H., Khodaei, A., & Bahramirad, S. (2015). State of the art in research on microgrids: A review. *IEEE Access*, 3, 890–925. <https://doi.org/10.1109/ACCESS.2015.2443119>
- [201]. Pasqualetti, F., Dörfler, F., & Bullo, F. (2013). Attack detection and identification in cyber-physical systems. *IEEE Transactions on Automatic Control*, 58(11), 2715–2729. <https://doi.org/10.1109/TAC.2013.2266831>
- [202]. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830. <https://jmlr.org/papers/v12/pedregosa11a.html>
- [203]. Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3), 45–77.

- <https://doi.org/10.2753/MIS0742-1222240302>
- [204]. Pfenninger, S., & Staffell, I. (2016). Long-term patterns of European PV output using 30 years of validated hourly reanalysis and satellite data. *Energy*, 114, 1251–1265. <https://doi.org/10.1016/j.energy.2016.08.060>
- [205]. Preuveneers, D., Rimmer, V., Tsingenopoulos, I., Spooren, J., Joosen, W., & Ilie-Zudor, E. (2018). Chained anomaly detection models for federated learning: An intrusion detection case study. *Applied Sciences*, 8(12), Article 2663. <https://doi.org/10.3390/app8122663>
- [206]. Rahman, S. A., Tout, H., Talhi, C., & Mourad, A. (2020). Internet of Things intrusion detection: Centralized, on-device, or federated learning? *IEEE Network*, 34(6), 310–317. <https://doi.org/10.1109/MNET.011.2000286>
- [207]. Reddi, S. J., Charles, Z., Zaheer, M., Garrett, Z., Rush, K., Konečný, J., Kumar, S., & McMahan, H. B. (2021). Adaptive federated optimization. In *International Conference on Learning Representations (ICLR 2021)*. OpenReview. <https://openreview.net/forum?id=LkFG3IB13U5>
- [208]. Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). “Why should I trust you?”: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1135–1144). ACM. <https://doi.org/10.1145/2939672.2939778>
- [209]. Rony, S., & Md, A. (2026). Multi-Agency Coordination in Sustained Healthcare Supply Shortages: Lessons from Covid-19, Drug Shortages, and Critical Medical Supply Disruptions. *American Journal of Data Science and Analytics*, 7(01), 181–221. <https://doi.org/10.63125/5v6t8515>
- [210]. Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5), 206–215. <https://doi.org/10.1038/s42256-019-0048-x>
- [211]. Sadia, Z., & Maishar Mostari, Z. (2022). The Impact of Interactive Dashboard Visualization on Data-Driven Decision Making in Financial Services: A Qualitative Evaluation. *International Journal of Scientific Interdisciplinary Research*, 12(03), 01–35. <https://doi.org/10.63125/3jstsv75>
- [212]. Saito, T., & Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLoS ONE*, 10(3), e0118432. <https://doi.org/10.1371/journal.pone.0118432>
- [213]. Saltelli, A., Annoni, P., Azzini, I., Campolongo, F., Ratto, M., & Tarantola, S. (2010). Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index. *Computer Physics Communications*, 181(2), 259–270. <https://doi.org/10.1016/j.cpc.2009.09.018>
- [214]. Sargent, R. G. (2013). Verification and validation of simulation models. *Journal of Simulation*, 7(1), 12–24. <https://doi.org/10.1057/jos.2012.20>
- [215]. Sattler, F., Wiedemann, S., Müller, K.-R., & Samek, W. (2020). Robust and communication-efficient federated learning from non-i.i.d. data. *IEEE Transactions on Neural Networks and Learning Systems*, 31(9), 3400–3413. <https://doi.org/10.1109/TNNLS.2019.2944481>
- [216]. Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J., & Williamson, R. C. (2001). Estimating the support of a high-dimensional distribution. *Neural Computation*, 13(7), 1443–1471. <https://doi.org/10.1162/089976601750264965>
- [217]. Schweppe, F. C., & Wildes, J. (1970). Power system static-state estimation, Part I: Exact model. *IEEE Transactions on Power Apparatus and Systems*, PAS-89(1), 120–125. <https://doi.org/10.1109/TPAS.1970.292678>
- [218]. Shaiful, M., Anisur, R., & Md, A. (2022). A Systematic Literature Review on the Role of Digital Health Twins in Preventive Healthcare For Personal and Corporate Wellbeing. *American Journal of Interdisciplinary Studies*, 3(04), 1–31. <https://doi.org/10.63125/negjw373>
- [219]. Shapley, L. S. (1953). A value for n-person games. In *Contributions to the Theory of Games (AM-28), Volume II* (pp. 307–318). Princeton University Press. <https://doi.org/10.1515/9781400881970-018>
- [220]. Sharafaldin, I., Habibi Lashkari, A., & Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. In *Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP 2018)* (pp. 108–116). SciTePress. <https://doi.org/10.5220/0006639801080116>
- [221]. Shneiderman, B. (2020). Human-centered artificial intelligence: Reliable, safe & trustworthy. *International Journal of Human-Computer Interaction*, 36(6), 495–504. <https://doi.org/10.1080/10447318.2020.1741118>
- [222]. Shokri, R., Stronati, M., Song, C., & Shmatikov, V. (2017). Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)* (pp. 3–18). IEEE. <https://doi.org/10.1109/SP.2017.41>
- [223]. Slack, D., Hilgard, S., Jia, E., Singh, S., & Lakkaraju, H. (2020). Fooling LIME and SHAP: Adversarial attacks on post hoc explanation methods. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society (AIES '20)* (pp. 180–186). ACM. <https://doi.org/10.1145/3375627.3375830>
- [224]. Slowik, J. (2019). Evolution of ICS attacks and the prospects for future disruptive events. In *Dragos, Inc..* <https://www.dragos.com/wp-content/uploads/Evolution-of-ICS-Attacks-and-the-Prospects-for-Future-Disruptive-Events-Joseph-Slowik-1.pdf>
- [225]. Smart Grid Interoperability Panel, Smart Grid Cybersecurity Committee. (2014). Guidelines for smart grid cybersecurity (NISTIR 7628 Rev. 1). National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.IR.7628r1>
- [226]. Sobol', I. M. (2001). Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. *Mathematics and Computers in Simulation*, 55(1–3), 271–280. [https://doi.org/10.1016/S0378-4754\(00\)00270-6](https://doi.org/10.1016/S0378-4754(00)00270-6)
- [227]. Sommer, R., & Paxson, V. (2010). Outside the closed world: On using machine learning for network intrusion detection. In *2010 IEEE Symposium on Security and Privacy* (pp. 305–316). IEEE. <https://doi.org/10.1109/SP.2010.25>
- [228]. Sonnenberg, C., & vom Brocke, J. (2012). Evaluations in the science of the artificial – Reconsidering the build-evaluate pattern in design science research. In *Design Science Research in Information Systems. Advances in Theory and Practice (DESRIST 2012), Lecture Notes in Computer Science, Vol. 7286* (pp. 381–397). Springer.

- https://doi.org/10.1007/978-3-642-29863-9_28
- [229]. Spiegelhalter, D. (2017). Risk and uncertainty communication. *Annual Review of Statistics and Its Application*, 4, 31–60. <https://doi.org/10.1146/annurev-statistics-010814-020148>
- [230]. Sridhar, S., Hahn, A., & Govindarasu, M. (2012). Cyber-physical system security for the electric power grid. *Proceedings of the IEEE*, 100(1), 210–224. <https://doi.org/10.1109/JPROC.2011.2165269>
- [231]. Stabak, D., & Md Shahid, N. (2022). Empirical Assessment of Uncertainty-Aware Real-Time Decision-Making for Reliable and Resilient Cyber-Physical Systems. *International Journal of Scientific Interdisciplinary Research*, 1(01), 432–477. <https://doi.org/10.63125/q3r4kk16>
- [232]. Stabak, D., & Md Siam, T. (2023). Real-Time Edge Computing for Cyber-Physical Systems: A Meta-Analysis of Energy Efficiency and Fault Tolerance. *American Journal of Scholarly Research and Innovation*, 13(06), 01–44. <https://doi.org/10.63125/g9c9cp41>
- [233]. Stouffer, K., Pease, M., Tang, C., Zimmerman, T., Pillitteri, V., Lightman, S., Hahn, A., Saravia, S., Sherule, A., & Thompson, M. (2023). Guide to operational technology (OT) security (NIST Special Publication 800-82 Rev. 3). National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.SP.800-82r3>
- [234]. Štrumbelj, E., & Kononenko, I. (2014). Explaining prediction models and individual predictions with feature contributions. *Knowledge and Information Systems*, 41(3), 647–665. <https://doi.org/10.1007/s10115-013-0679-x>
- [235]. Su, Z., Wang, Y., Luan, T. H., Zhang, N., Li, F., Chen, T., & Cao, H. (2022). Secure and efficient federated learning for smart grid with edge-cloud collaboration. *IEEE Transactions on Industrial Informatics*, 18(2), 1333–1344. <https://doi.org/10.1109/TII.2021.3095506>
- [236]. Sumaiya, M. (2024). Hospital Wastewater as A Reservoir of Antimicrobial Resistance Genes: Implications for Public Health and Genomic Surveillance in The United States. *Review of Applied Science and Technology*, 3(04), 448–508. <https://doi.org/10.63125/ahvas976>
- [237]. Sumaiya, M. (2025). CRISPR-Cas Systems in Precision Oncology: Current Applications and Future Challenges. *American Journal of Interdisciplinary Studies*, 6(1), 513–564. <https://doi.org/10.63125/rqpqr418>
- [238]. Sumaiya, M. (2026). Artificial Intelligence Driven Multi-Omics Approaches for Precision Medicine: From Molecular Discovery to Clinical Translation. *American Journal of Scholarly Research and Innovation*, 5(01), 429–477. <https://doi.org/10.63125/yfme0q12>
- [239]. Sundararajan, M., Taly, A., & Yan, Q. (2017). Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, PMLR 70 (pp. 3319–3328). PMLR. <https://proceedings.mlr.press/v70/sundararajan17a.html>
- [240]. Syed Feroz, A. (2023). Quality-Management Systems for Resilient Energy Infrastructure Fabrication. *International Journal of Scientific Interdisciplinary Research*, 13(10), 37–91. <https://doi.org/10.63125/mbhhrm88>
- [241]. Syed Feroz, A. (2024). AI-Assisted NDT Analytics for Early Crack Detection in Energy-Infrastructure Pressure Vessels Using Advanced Manufacturing Data. *Review of Applied Science and Technology*, 14(07), 01–52. <https://doi.org/10.63125/8bdqs538>
- [242]. Syed Feroz, A. (2025). Intelligent Process Monitoring for Enhancing Weld Integrity and Reliability in Advanced Manufacturing of LPG Energy Infrastructure. *American Journal of Interdisciplinary Studies*, 6(3), 394–441. <https://doi.org/10.63125/p3b5mb44>
- [243]. Syed Feroz, A., & Md Mahamud Pervaz, P. (2022). Root-Cause Analysis and FMEA-Driven Reliability Improvement in Gas Production Facilities: A Multi-Year Empirical Study. *International Journal of Scientific Interdisciplinary Research*, 12(10), 01–53. <https://doi.org/10.63125/70msp878>
- [244]. Tahmedur, R., & Rifat, C. (2023). Critical Path Analysis and Resource-Leveling Optimization for Multi-Project Engineering Portfolios. *American Journal of Interdisciplinary Studies*, 13(12), 01–47. <https://doi.org/10.63125/8eqnea46>
- [245]. Taïk, A., & Cherkaoui, S. (2020). Electrical load forecasting using edge computing and federated learning. In *ICC 2020 – 2020 IEEE International Conference on Communications (ICC)* (pp. 1–6). IEEE. <https://doi.org/10.1109/ICC40277.2020.9148937>
- [246]. Truex, S., Baracaldo, N., Anwar, A., Steinke, T., Ludwig, H., Zhang, R., & Zhou, Y. (2019). A hybrid approach to privacy-preserving federated learning. In *Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security (AISec '19)* (pp. 1–11). ACM. <https://doi.org/10.1145/3338501.3357370>
- [247]. Tversky, A., & Kahneman, D. (1974). Judgment under uncertainty: Heuristics and biases. *Science*, 185(4157), 1124–1131. <https://doi.org/10.1126/science.185.4157.1124>
- [248]. Urbina, D. I., Giraldo, J. A., Cardenas, A. A., Tippenhauer, N. O., Valente, J., Faisal, M., Ruths, J., Candell, R., & Sandberg, H. (2016). Limiting the impact of stealthy attacks on industrial control systems. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security* (pp. 1092–1105). ACM. <https://doi.org/10.1145/2976749.2978388>
- [249]. Venable, J., Pries-Heje, J., & Baskerville, R. (2016). FEDS: A framework for evaluation in design science research. *European Journal of Information Systems*, 25(1), 77–89. <https://doi.org/10.1057/ejis.2014.36>
- [250]. Wachter, S., Mittelstadt, B., & Russell, C. (2018). Counterfactual explanations without opening the black box: Automated decisions and the GDPR. *Harvard Journal of Law & Technology*, 31(2), 841–887. <https://jolt.law.harvard.edu/assets/articlePDFs/v31/Counterfactual-Explanations-without-Opening-the-Black-Box-Sandra-Wachter-et-al.pdf>

- [251]. Wang, H., Ruan, J., Wang, G., Zhou, B., Liu, Y., Fu, X., & Peng, J. (2018). Deep learning-based interval state estimation of AC smart grids against sparse cyber attacks. *IEEE Transactions on Industrial Informatics*, 14(11), 4766–4778. <https://doi.org/10.1109/TII.2018.2804669>
- [252]. Wang, M., Zheng, K., Yang, Y., & Wang, X. (2020). An explainable machine learning framework for intrusion detection systems. *IEEE Access*, 8, 73127–73141. <https://doi.org/10.1109/ACCESS.2020.2988359>
- [253]. Wang, S., Tuor, T., Salonidis, T., Leung, K. K., Makaya, C., He, T., & Chan, K. (2019). Adaptive federated learning in resource constrained edge computing systems. *IEEE Journal on Selected Areas in Communications*, 37(6), 1205–1221. <https://doi.org/10.1109/JSAC.2019.2904348>
- [254]. Wang, W., & Lu, Z. (2013). Cyber security in the smart grid: Survey and challenges. *Computer Networks*, 57(5), 1344–1371. <https://doi.org/10.1016/j.comnet.2012.12.017>
- [255]. Wang, Y., Bennani, I. L., Liu, X., Sun, M., & Zhou, Y. (2021). Electricity consumer characteristics identification: A federated learning approach. *IEEE Transactions on Smart Grid*, 12(4), 3637–3647. <https://doi.org/10.1109/TSG.2021.3066577>
- [256]. Wang, Y., Chen, C., Wang, J., & Baldick, R. (2016). Research on resilience of power systems under natural disasters: A review. *IEEE Transactions on Power Systems*, 31(2), 1604–1613. <https://doi.org/10.1109/TPWRS.2015.2429656>
- [257]. Wei, K., Li, J., Ding, M., Ma, C., Yang, H. H., Farokhi, F., Jin, S., Quek, T. Q. S., & Poor, H. V. (2020). Federated learning with differential privacy: Algorithms and performance analysis. *IEEE Transactions on Information Forensics and Security*, 15, 3454–3469. <https://doi.org/10.1109/TIFS.2020.2988575>
- [258]. Whitehead, D. E., Owens, K., Gammel, D., & Smith, J. (2017). Ukraine cyber-induced power outage: Analysis and practical mitigation strategies. In *2017 70th Annual Conference for Protective Relay Engineers (CPRE)* (pp. 1–8). IEEE. <https://doi.org/10.1109/CPRE.2017.8090056>
- [259]. Yan, Y., Qian, Y., Sharif, H., & Tipper, D. (2012). A survey on cyber security for smart grid communications. *IEEE Communications Surveys & Tutorials*, 14(4), 998–1010. <https://doi.org/10.1109/SURV.2012.010912.00035>
- [260]. Yang, Q., Liu, Y., Chen, T., & Tong, Y. (2019). Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology*, 10(2), Article 12. <https://doi.org/10.1145/3298981>
- [261]. Yin, D., Chen, Y., Kannan, R., & Bartlett, P. (2018). Byzantine-robust distributed learning: Towards optimal statistical rates. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, PMLR 80 (pp. 5650–5659). PMLR. <https://proceedings.mlr.press/v80/yin18a.html>
- [262]. Zhang, H., Yue, D., Dou, C., Xie, X., Li, K., & Hancke, G. P. (2023). Resilient optimal defensive strategy of TSK fuzzy-model-based microgrids' system via a novel reinforcement learning approach. *IEEE Transactions on Neural Networks and Learning Systems*, 34(4), 1921–1931. <https://doi.org/10.1109/TNNLS.2021.3105668>
- [263]. Zhang, Y., Wang, J., & Chen, B. (2021). Detecting false data injection attacks in smart grids: A semi-supervised deep learning approach. *IEEE Transactions on Smart Grid*, 12(1), 623–634. <https://doi.org/10.1109/TSG.2020.3010510>
- [264]. Zhao, Y., Li, M., Lai, L., Suda, N., Civin, D., & Chandra, V. (2018). Federated learning with non-IID data (1806.00582). *arXiv*. <https://arxiv.org/abs/1806.00582>
- [265]. Zhou, B., Li, W., Chan, K. W., Cao, Y., Kuang, Y., Liu, X., & Wang, X. (2016). Smart home energy management systems: Concept, configurations, and scheduling strategies. *Renewable and Sustainable Energy Reviews*, 61, 30–40. <https://doi.org/10.1016/j.rser.2016.03.047>
- [266]. Zhu, L., Liu, Z., & Han, S. (2019). Deep leakage from gradients. In *Advances in Neural Information Processing Systems 32 (NeurIPS 2019)*. Curran Associates. <https://papers.nips.cc/paper/2019/hash/60a6c4002cc7b29142def8871531281a-Abstract.html>