

On the Interaction of Aggregation Strategies, Privacy Budgets, and Data Heterogeneity in Differentially Private Federated Learning

Ivana Dasovic
TU Wien, Vienna, Austria

Anastasia Pustozero
SBA Research, Vienna, Austria
apustozero@sba-research.org

Rudolf Mayer
SBA Research, Vienna, Austria
rmayer@sba-research.org

Abstract—Federated Learning (FL) enables collaborative model training across decentralized mobile, edge, and institutional devices without exchanging raw data, which makes it a key enabler of privacy-preserving artificial intelligence for wireless and mobile networks. However, FL does not provide formal privacy guarantees, as shared model updates can leak information about local training data; this is a concern for distributed learning over untrusted networks. Differential Privacy (DP) addresses this by bounding the influence of any single record, but it introduces a privacy–utility trade-off whose behavior depends on how local updates are aggregated at the server and on how heterogeneously data is distributed across clients. Most prior work studies these factors in isolation. In this paper, we present a controlled empirical study of their *joint* interaction in differentially private federated learning. Using example-level DP-SGD applied locally on each client, we evaluate three aggregation strategies—FedAvg, FedAdam, and FedYogi—across a wide range of privacy budgets and non-IID data distributions on two real-world tabular benchmarks: the well-known Adult Income dataset, and a more challenging, heterogeneous healthcare dataset (MIMIC-III). Our results reveal a sharp privacy-budget threshold below which training collapses, and show that adaptive aggregation (FedAdam, FedYogi) is substantially more robust than FedAvg to both DP noise and data heterogeneity—training more smoothly, with smaller global weight changes and higher F1 score at equal privacy. MIMIC-III is consistently more fragile under the same privacy constraints than the Adult Income dataset, and non-IID partitioning amplifies the degradation. We distill these findings into practical guidelines for deploying privacy-preserving FL on heterogeneous, distributed data.

Index Terms—Federated learning, differential privacy, aggregation strategies, non-IID, privacy–utility trade-off

I. INTRODUCTION

Machine learning is increasingly embedded in mobile, healthcare, and financial services, where data is sensitive and often legally protected. Federated Learning (FL) offers a way to train models on data that remains on edge devices or within institutional silos, sharing only model updates with a

coordinating server [1], [2]. Because raw data is never centralized, the impact of any individual breach is contained, the communication overhead of moving large datasets is avoided, and computation is pushed to clients such as smartphones, IoT devices, or hospitals. In the common FL setting, clients train on their local data and send model parameters or gradients to a server, which aggregates them into a global model that is redistributed for further rounds until convergence. As intelligence migrates to wireless and mobile networks, FL has become a cornerstone for privacy-preserving, on-device AI and a recurring security concern for distributed systems operating over partially trusted infrastructure [3].

Keeping data local, however, does not by itself guarantee full privacy. Prior work has shown that shared gradients or models can be exploited to reconstruct or infer properties of the underlying training records [4]–[6]. Differential Privacy (DP) provides a formal, quantifiable defense by bounding how much any single record can influence the released model, controlled by a privacy budget ϵ [7], [8]. Differentially Private Stochastic Gradient Descent (DP-SGD) realizes this for model training by clipping per-example gradients and adding calibrated Gaussian noise [9]. Smaller ϵ means stronger privacy and more noise; the inevitable cost is reduced model utility.

Two further factors influence the outcome. First, the *aggregation strategy* determines how client updates are combined at the server: simple averaging (FedAvg) [1] and adaptive server optimizers such as FedAdam and FedYogi [10] react very differently to noise and heterogeneity. Second, real deployments are *non-IID*: clients hold heterogeneous local distributions, which is known to harm convergence and final quality [11], [12].

Research gap. While each of these factors has been studied individually, their combined effect on the privacy–utility trade-off remains poorly understood. A recent systematic review of DP-FL confirms that the interaction between the privacy mechanism, the aggregation rule, and data heterogeneity is comparatively under-explored [13]. For example, earlier DP-FL studies on the real-world dataset MIMIC-III compare only a small set of strategies (e.g., FedAvg vs. FedProx) and do not cover adaptive server optimizers nor a joint analysis of privacy budget and heterogeneity [14]. The literature therefore lacks a controlled study that varies privacy budget, aggregation

This work was partly funded by the European Union’s Horizon Europe Research and Innovation Programme (GA No 101136305). Partner Semmelweis University received funding from the Hungarian National Research, Development, and Innovation Fund. This publication reflects only the authors’ view, and the European Commission is not responsible for any use that may be made of the information it contains. The work was also partly funded by the COMET – Competence Centers for Excellent Technologies Programme and funded by BMIMI, BMWET, and the federal state of Vienna, managed by FFG.

strategy, and heterogeneity *simultaneously* on realistic data. This gap is a practical obstacle to deploying privacy-preserving FL in edge and institutional environments where data is unevenly distributed, noisy, and imbalanced.

Research questions. Rather than treating the three factors separately, we study them as an interacting system, and structure the investigation around three questions that progressively introduce and then combine them:

RQ1: How does the privacy budget ϵ shape the effectiveness–stability trade-off in DP-FL, and how does this trade-off depend on the dataset?

RRQ2 At a given privacy budget, how does the choice of aggregation strategy (FedAvg, FedAdam, FedYogi) change effectiveness, stability, and convergence—that is, how do the privacy mechanism and the aggregation rule *interact*?

RQ3: When data is additionally non-IID, how do privacy budget, aggregation strategy, and heterogeneity act *jointly* on the trade-off, and which strategies remain robust under their combined pressure?

The questions are: RQ1 establishes the privacy axis, RQ2 adds the aggregation axis on top of it, and RQ3 adds heterogeneity, so that the overall study characterizes the joint behavior rather than three effects in isolation.

Addressing these questions, our contributions are:

- We build a local DP-FL system that enforces example-level DP via DP-SGD on each client under federated orchestration, and use it to study the *joint* interaction of privacy budget, aggregation strategy, and data heterogeneity, rather than any single factor in isolation.
- We characterize how the privacy budget governs the effectiveness–stability trade-off and show that this trade-off is dataset-dependent, being notably stricter for the MIMIC-III than for the Adult Income dataset (RQ1).
- We show that, at equal privacy, the aggregation rule substantially changes the outcome: adaptive aggregation (FedAdam, FedYogi) is markedly more robust to DP noise than FedAvg—smoother training, smaller global weight changes, higher F1—and that this privacy–aggregation interaction is strongest at tight budgets and fades as ϵ grows (RQ2).
- We show that introducing non-IID data compounds the degradation and that privacy, aggregation, and heterogeneity act together rather than additively, with adaptive optimizers remaining the more stable choice under their combined pressure; we distill the combined findings into deployment guidelines (RQ3).

The remainder of the paper is organized as follows. Section II reviews background and related work. Section III describes the system, datasets, and experimental setup. Section IV presents and discusses the results. Section V concludes and outlines future work.

II. BACKGROUND AND RELATED WORK

This section introduces the FL setting and aggregation strategies we study, the differential-privacy mechanism we use,

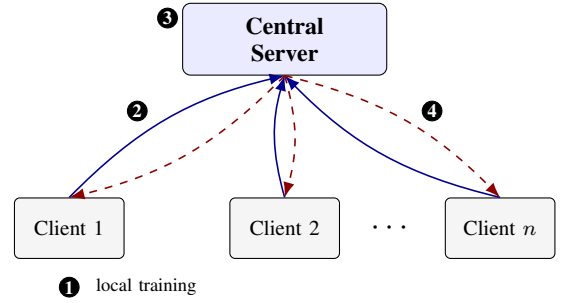


Fig. 1: One federated learning round. ❶ clients train a local model on their own data; ❷ each client sends its local update to the server (solid); ❸ the server aggregates the updates into a new global model; and ❹ the global model is redistributed to the clients (dashed).

and the data-heterogeneity model, and then position our work against the most closely related prior studies.

A. Federated Learning and Aggregation

We consider server-orchestrated *horizontal* FL in a *cross-silo* setting: a small, stable set of well-resourced participants (e.g., hospitals or banks) that share the same feature space but hold disjoint records. Figure 1 summarizes one communication round: clients train locally, transmit updates, the server aggregates them into a new global model, and redistributes it.

The aggregation rule is the component that combines client contributions and is therefore central to FL behavior, especially when updates are noisy or heterogeneous. **FedAvg** [1] forms a sample-weighted average of the client models,

$$\mathbf{w}_{t+1} = \sum_k \frac{n_k}{n} \mathbf{w}_{t+1}^{(k)}, \quad (1)$$

where n_k is the number of local samples of client k and $n = \sum_k n_k$. FedAvg is communication-efficient and the de facto baseline, but it weights every (possibly noisy) update equally and is known to suffer from *client drift* under non-IID data, where divergent local optima bias the global model [12].

Adaptive server optimizers generalize this step. Following Reddi et al. [10], the server treats the negative aggregated client update as a *pseudo-gradient* $\Delta_t = \mathbf{w}_t - \sum_k \frac{n_k}{n} \mathbf{w}_{t+1}^{(k)}$ and applies an adaptive optimizer with server learning rate η , decay rates β_1, β_2 , and a small constant τ :

$$\mathbf{m}_t = \beta_1 \mathbf{m}_{t-1} + (1 - \beta_1) \Delta_t, \quad (2)$$

$$\textbf{FedAdam: } \mathbf{v}_t = \beta_2 \mathbf{v}_{t-1} + (1 - \beta_2) \Delta_t^2, \quad (3)$$

$$\textbf{FedYogi: } \mathbf{v}_t = \mathbf{v}_{t-1} - (1 - \beta_2) \Delta_t^2 \text{sign}(\mathbf{v}_{t-1} - \Delta_t^2), \quad (4)$$

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \mathbf{m}_t / (\sqrt{\mathbf{v}_t} + \tau). \quad (5)$$

Here, Eq. (2) is the first-moment (momentum) estimate of the pseudo-gradient; Eqs. (3) and (4) are the two alternative second-moment estimates that distinguish FedAdam from FedYogi; and Eq. (5) applies the resulting preconditioned step to the global model, with plain FedAvg recovered as the

TABLE I: The three server-side aggregation strategies compared in this study

Strategy	Server-side state	Update rule / property
FedAvg	none (stateless)	sample-weighted average, Eq. (1); canonical baseline
FedAdam	1st & 2nd moments ($\mathbf{m}_t, \mathbf{v}_t$)	Adam-style EMA of squared pseudo-gradient, Eq. (3)
FedYogi	1st & 2nd moments ($\mathbf{m}_t, \mathbf{v}_t$)	sign-controlled second moment, Eq. (4); more stable $\mathbf{v}_t$

special case in which no adaptive preconditioning is used. The key difference is the second-moment update: FedAdam uses an exponential moving average (Adam-style [15]), whereas FedYogi uses a sign-based additive update (Yogi-style [16]) that prevents the rapid growth of $\mathbf{v}_t$ and yields more stable effective per-coordinate step sizes. Because the per-coordinate denominator $\sqrt{\mathbf{v}_t}$ in Eq. (5) rescales updates, noisy or inconsistent coordinates—exactly those produced by DP noise and heterogeneity—are implicitly damped, which motivates studying these optimizers under privacy constraints. Other families address heterogeneity differently: FedProx adds a proximal term to the client objective [17], SCAFFOLD uses control variates to correct client drift [12], and robust rules such as Krum [18] and Trimmed Mean [19] target Byzantine threats to the model integrity or availability.

We selected FedAvg, FedAdam, and FedYogi for achieving a comparison of rather different methods. FedAvg is the canonical, ubiquitous baseline against which all FL methods are measured. FedAdam and FedYogi are the two most widely adopted adaptive server optimizers, are natively available as drop-in server-side strategies, and were specifically designed to improve convergence under client heterogeneity [10]. Crucially, all three modify only the server-side aggregation while leaving local client training identical, so observed differences can be attributed to the aggregation rule alone. FedAdam and FedYogi differ from each other only in their second-moment update (Eqs. (3)–(4)), which makes them an ideal controlled pair for probing sensitivity to DP noise. By contrast, methods that (i) change the client objective (FedProx), (ii) require additional client state and communication (SCAFFOLD), or (iii) assume an adversarial threat model (Krum, Trimmed Mean) would confound a clean privacy-versus-aggregation comparison, and are therefore out of scope for this work.

B. Differential Privacy in Federated Learning

A randomized mechanism is (ϵ, δ) -differentially private if, for any two datasets differing in a single record and any output set, the output probabilities differ by at most a factor e^ϵ up to an additive δ [7]. In FL, DP is typically applied either *centrally* (the server perturbs the aggregate, assuming it is trusted) or *locally* (each client perturbs its own updates, tolerating an untrusted server). We adopt local, example-level DP via DP-SGD [9]: each client clips per-sample gradients

to a fixed norm and adds Gaussian noise before updating, bounding the contribution of any individual record even if the server is compromised. Composing the privacy loss across many rounds with basic composition is loose; we therefore use Rényi Differential Privacy (RDP), which gives substantially tighter cumulative bounds for the sampled Gaussian mechanism [20]. DP for FL has been combined with secure aggregation and related primitives [21]–[23], and adaptive-clipping or dynamic-noise schemes aim to soften the utility cost [24], [25]. Healthcare has been a prominent application domain, given strict data-protection regimes and siloed clinical records [14], [26]. Nonetheless, the *interaction* between the DP mechanism, the aggregation rule, and data heterogeneity remains the focus of comparatively little work [13].

C. Data Heterogeneity

Under non-IID data, client distributions diverge in label proportions (label skew) and in the number of samples (quantity skew), which slows convergence and lowers final quality [11]. A standard way to simulate controllable heterogeneity is Dirichlet partitioning with concentration parameter α : small α yields strongly skewed (non-IID) partitions, while large α approaches IID [27]. Combining DP noise with non-IID skew compounds the difficulty of stable optimization, motivating our joint study.

D. Positioning

Our study differs from prior work in combining three axes at once. Unlike DP-FL optimization analyses that fix the aggregation method, and unlike FL benchmarks that compare aggregation strategies without privacy, we vary privacy budget, aggregation strategy, and heterogeneity together. Unlike MIMIC DP-FL studies limited to FedAvg and FedProx [14], we evaluate adaptive server optimizers, and we do so on two benchmark datasets from different domains to test whether conclusions transfer across domains.

III. METHODOLOGY AND EXPERIMENTAL SETUP

This section describes the DP-FL system we implemented, the datasets and model, the data-partitioning regimes, how each aggregation strategy and the differential-privacy accounting are realized, and the metrics used for evaluation.

A. System and Tools

We implement server-orchestrated horizontal FL with the Flower framework¹, simulating all clients on a single machine for controlled, reproducible cross-silo experiments. The shared model and local optimization use PyTorch. Example-level DP is enforced with Opacus²: each client’s optimizer is wrapped with a privacy engine that performs per-sample gradient clipping, Gaussian noise addition, and Rényi Differential Privacy (RDP) accounting. This separation of concerns lets Flower handle aggregation while Opacus enforces DP locally on every client. To isolate effects, we compare three model types:

¹<https://flower.ai/>

²<https://opacus.ai/>

a centralized model (an upper bound without federation or privacy), a non-private federated model (capturing decentralization only), and a differentially private federated model.

B. Aggregation Strategies

The three strategies are realized as server-side aggregation rules and are the only component varied across runs. FedAvg performs the weighted average of Eq. (1). FedAdam and FedYogi maintain server-side first- and second-moment state ($\mathbf{m}_t, \mathbf{v}_t$) across communication rounds, treat the averaged client update as the pseudo-gradient Δ_t , and apply the updates in Eqs. (3)–(4) with tuned server learning rate η and decay rates β_1, β_2 . Client-side local training (model, loss, local optimizer, number of local steps, and—in private runs—the DP mechanism) is held identical regardless of the server strategy, so that any differences in stability or accuracy stem from the aggregation rule.

C. Differential Privacy and Accounting

On each client, DP-SGD [9] clips every per-sample gradient g_i to a fixed ℓ_2 norm C and adds Gaussian noise to the batch sum,

$$\tilde{g} = \frac{1}{B} \left(\sum_{i=1}^B g_i / \max(1, \|g_i\|_2 / C) + \mathcal{N}(0, \sigma^2 C^2 \mathbf{I}) \right), \quad (6)$$

where σ is the noise multiplier and B the batch size. The noise multiplier is calibrated by Opacus to reach a target per-epoch budget ϵ given δ and the sampling rate, and the cumulative budget over the 50 communication rounds is tracked with the RDP accountant [20], which converts back to an (ϵ, δ) guarantee. We set $\delta = 1/N$ with N the number of training examples. Because ϵ is specified *per epoch*, the overall guarantee of a run accumulates over rounds; we report the per-epoch ϵ throughout. The clipping norm C trades off clipping bias against added noise: a small C limits gradient magnitude (less noise but more bias), a large C does the opposite. We fix C per configuration to the 90th percentile of per-sample gradient norms observed during 10 non-private warm-up epochs [28]; the resulting C ranges from about 1.96–2.66 on Adult and 3.26–3.79 on MIMIC-III across configurations.

D. Datasets

We use two widely adopted tabular benchmarks, both posed as binary classification, chosen to span two distinct domains.

The *MIMIC-III* dataset [29] is a large, publicly available critical-care database of de-identified electronic health records from over 40,000 ICU patients. It is a real-world healthcare benchmark and is particularly relevant and challenging for FL: clinical data is naturally siloed across institutions and care units, is strongly heterogeneous across patients, mixes continuous, categorical, and time-stamped variables, and is severely class-imbalanced for outcomes such as mortality. These properties make it an ideal test for privacy-preserving, heterogeneity-aware FL, and they mirror the regulatory and distributional constraints of deploying AI in real clinical networks [14], [26]. Following the preprocessing of Purushotham et al. [30] we use Feature Set A (17 clinically relevant features

over the first 24 hours of admission) for in-hospital mortality prediction, yielding 35,992 records.

The *Adult Income* dataset [31] predicts whether an individual earns more than \$50,000 from demographic and employment attributes. It is one of the most established and reproducible benchmarks for binary classification in the ML community, is openly available, and is frequently used in prior FL and DP studies; this makes it valuable for cross-study comparability and as a clean, well-understood complement to MIMIC-III. It also exhibits real class imbalance and feature skew, so it is far from a trivial task. After cleaning, one-hot encoding, and removal of redundant/missing entries it reduces to 30,139 records. Continuous features are standardized per client.

E. Model

To keep the focus on privacy and aggregation rather than architecture, we use a logistic regression classifier: a single linear layer followed by a sigmoid, trained with binary cross-entropy loss and SGD (DP-SGD in private settings). A low-complexity, interpretable model reduces confounding from architectural choices, converges quickly, and is reproducible—appropriate for an in-depth controlled study. Still, the model provides sufficient complexity to effectively learn the relevant aspects of the data and achieve results comparable to state-of-the-art models. Because both datasets are imbalanced, the decision threshold is tuned post-hoc on a validation set by sweeping values from 0 to 1 in steps of 0.01 and selecting the threshold that maximizes F1 [32].

F. Data Partitioning

We study two regimes. *IID* partitioning randomly distributes records so each client mirrors the global distribution. *Non-IID* partitioning uses a Dirichlet partitioner with concentration $\alpha \in \{0.2, 0.3, 0.5, 1, 5, 15\}$, inducing both label and quantity skew; smaller α is more heterogeneous.

G. Hyperparameters

Hyperparameters are tuned separately for each configuration (aggregation method $\times$ client count $\times$ dataset), resulting in 18 different searches across the search space as listed in Table II. The decision threshold is then optimized in a second search. For FedAdam and FedYogi we additionally tune β_1, β_2 , and the server learning rate η , as these directly govern the server update. Federated training runs for 50 rounds with all clients participating each round; client counts of 5, 10, and 15 are evaluated.

H. Evaluation Metrics

We report five complementary metrics. The *F1 score* is our primary metric given class imbalance, aggregated globally from per-client true/false positive/negative counts. *Accuracy* and *AUROC* (the latter averaged across clients in IID settings) measure predictive quality, AUROC being threshold-independent. The *global model weight change* between consecutive rounds measures training stability—a desirable run

TABLE II: Hyperparameter search space (grid search)

Hyperparameter	Tested values
Learning rate	0.001, 0.01, 0.1
Batch size	64, 128
Weight decay	0, 10^{-2} , 10^{-4}
Momentum	0, 0.9
L_1 norm	0, 10^{-4} , 10^{-5}
β_1 (FedAdam/FedYogi)	0, 0.1, 0.3, 0.4, 0.5, 0.9
β_2 (FedAdam/FedYogi)	0.5, 0.9, 0.99
η (server LR)	0.01, 0.03, 0.1, 0.3

drives this toward zero. We also track *loss* to monitor optimization. All metrics are tracked across the 50 rounds to expose convergence dynamics, not only final values.

I. Environment

Experiments run on a CPU-only workstation (Intel i5-1235U, 32 GB RAM) using Python 3.10, PyTorch 2.7.0, Opacus 1.5.4, and Flower (flwr) 1.18.0.

IV. RESULTS AND DISCUSSION

We first establish non-private baselines, then study the three factors in turn: the privacy budget (RQ1), the aggregation strategy (RQ2), and non-IID data (RQ3), before distilling practical guidelines.

A. Non-Private Federated Baseline

We first establish strong non-private baselines so that later degradation can be attributed to DP and heterogeneity rather than to poor tuning. Table III reports the tuned federated results (5 clients) alongside centralized logistic-regression benchmarks from the literature [30], [33]. The federated models come close to the centralized references on both datasets. With optimal thresholds, F1 is consistently higher than at the default 0.5 threshold, confirming the value of threshold tuning under imbalance. Without privacy and under IID data, all three aggregation strategies converge to nearly identical final quality; they differ mainly in convergence speed and smoothness, with FedAvg typically attaining the strongest mid-training scores on the easier Adult task and the adaptive optimizers catching up by round 50. Scaling from 5 to 15 clients does not meaningfully change final effectiveness (10- and 15-client results, omitted for space, follow the same pattern). This establishes that any later differences stem from DP constraints or data heterogeneity.

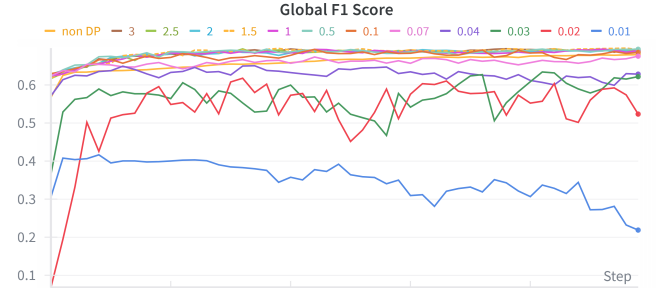
B. Impact of the Privacy Budget (RQ1)

We next vary the per-epoch privacy budget over twelve values (0.01, 0.02, 0.03, 0.04, 0.07, 0.1, 0.5, 1, 1.5, 2, 2.5, 3) with FedAvg and IID data, comparing against the non-private baseline. Figure 2 shows global F1 over rounds for both datasets, and Fig. 3 the corresponding weight changes on Adult.

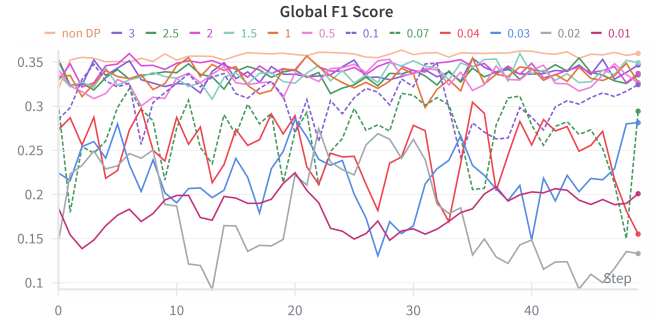
A clear, monotone trade-off emerges across all metrics: as ϵ decreases, more noise is injected and effectiveness drops while

TABLE III: Non-private federated baseline vs. centralized literature references (5 clients; best threshold)

Dataset	Method	F1	F1@0.5	AUROC
Adult	Centralized [33]	0.657	—	—
	FedAvg	0.640	0.517	0.895
	FedYogi	0.650	0.548	0.891
	FedAdam	0.647	0.563	0.882
MIMIC-III	Centralized [30]	—	—	0.824
	FedAvg	0.383	0.192	0.794
	FedYogi	0.383	0.371	0.795
	FedAdam	0.369	0.164	0.795



(a) Adult Income.



(b) MIMIC-III (healthcare).

Fig. 2: Global F1 over 50 rounds for FedAvg under varying per-epoch privacy budgets ϵ (5 clients, IID). Larger ϵ (weaker privacy) tracks the non-private baseline; small ϵ degrades and destabilizes F1, with MIMIC-III more sensitive than Adult.

instability rises. For sufficiently large budgets ($\epsilon \gtrsim 0.5$), F1, AUROC, and accuracy stay close to the non-private baseline with smooth trajectories and small, decreasing weight changes. A threshold region exists below which quality deteriorates sharply: on Adult, models remain usable down to about $\epsilon = 0.04$ (with mild delays and fluctuations), degrade markedly below that, and *fail entirely* at $\epsilon = 0.01$, where the model does not learn and F1 even drifts downward. The loss curves confirm this picture: the smallest budget yields the highest, most unstable loss, never approaching the levels reached at larger budgets, while AUROC exhibits a comparable threshold below which discriminative power collapses. MIMIC-III is

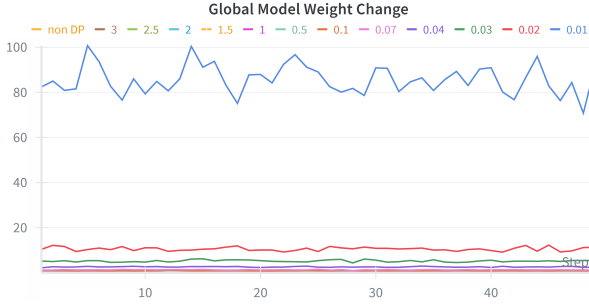


Fig. 3: Global model weight change for FedAvg on Adult under varying ϵ (5 clients, IID). Stronger privacy (smaller ϵ) produces larger, less stable weight updates; the non-private run converges smoothly toward zero.

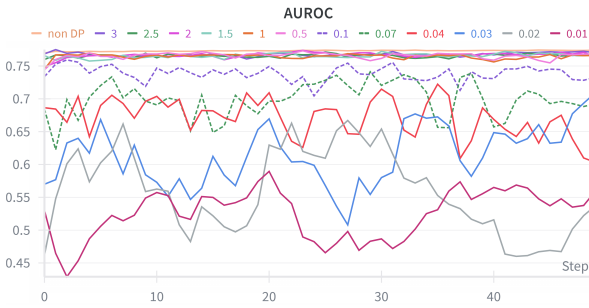


Fig. 4: AUROC for FedAvg on MIMIC-III under varying ϵ (5 clients, IID). Budgets $\epsilon \gtrsim 0.5$ cluster near the non-private baseline; budgets $\epsilon \leq 0.04$ collapse toward chance level with strong oscillation, illustrating the threshold behavior of the privacy–utility trade-off.

consistently more fragile: under identical budgets it shows larger degradation and stronger oscillations, retaining acceptable F1 only for $\epsilon \gtrsim 0.5$.

The AUROC trajectories in Fig. 4 make the threshold behavior explicit: budgets at or above 0.5 remain tightly clustered around the non-private level (AUROC ≈ 0.78), $\epsilon = 0.1$ and 0.07 sit moderately below, and budgets of 0.04 or smaller fall toward chance level with large swings between rounds—the model intermittently loses and regains discriminative power rather than converging.

Two cross-cutting observations hold on both datasets. First, F1 is the most privacy-sensitive metric, declining earliest and most. Accuracy, by contrast, is the least sensitive and can remain deceptively high even when the model fails on the minority class—so we rely primarily on F1 and AUROC throughout. Second, the number of clients (5/10/15) has little effect on these trends: the same degradation and fluctuation patterns recur as budgets tighten, indicating that the privacy budget, not the client population size, drives the behavior in this range. Because budgets $\epsilon \gtrsim 0.5$ essentially reproduce non-private behavior, we restrict the subsequent experiments to the more informative low- ϵ regime.

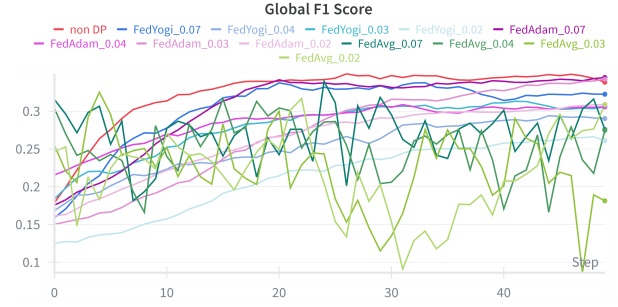


Fig. 5: Global F1 on MIMIC-III (5 clients, IID) for FedAvg, FedAdam, and FedYogi under low privacy budgets. FedAvg (green shades) oscillates strongly; the adaptive optimizers (blue/purple) learn smoothly and reach higher F1.

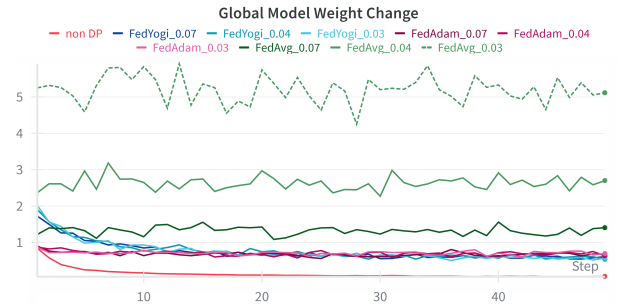


Fig. 6: Global model weight change on Adult (5 clients, IID) under low privacy budgets. FedAvg (green) exhibits much larger, more erratic updates than FedAdam and FedYogi, which remain low and stable.

C. Impact of Aggregation Strategy (RQ2)

We now fix IID data and compare FedAvg, FedAdam, and FedYogi under matched privacy budgets in the informative low- ϵ range. Figure 5 shows F1 on MIMIC-III and Fig. 6 the weight changes on Adult.

Under equal privacy guarantees, the adaptive optimizers FedAdam and FedYogi consistently outperform FedAvg in AUROC and F1, converge faster, and train more stably. FedAvg pays a higher “optimization cost” for privacy: it shows pronounced F1 oscillations—at the strictest budgets it sometimes makes essentially no learning progress—and the largest, most irregular global weight changes (Fig. 6), indicating unstable updates. This is consistent with the per-coordinate rescaling in Eqs. (3)–(4), which damps the noisy directions that FedAvg propagates directly. FedYogi tends to be the most stable overall, with FedAdam close behind; the two converge to similar weight magnitudes after the first few rounds. As before, accuracy understates these gaps: FedAvg’s accuracy can look comparable even when its F1 and AUROC are clearly worse. The advantage of adaptivity is largest under tight privacy and shrinks as ϵ increases, where all methods approach the non-private baseline and differences become

minor. The effect is more pronounced on MIMIC-III than on Adult, in line with the greater fragility of the healthcare task.

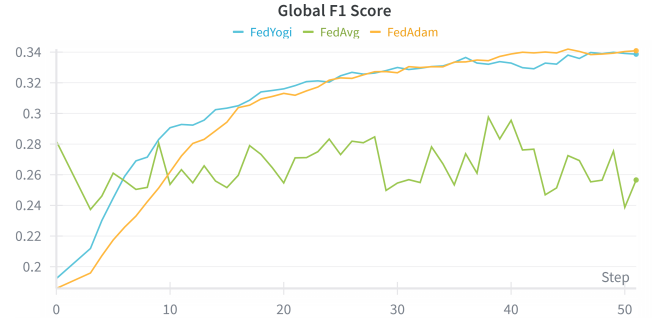
Beyond final quality, the aggregation rule also governs *convergence speed*. Measuring the number of communication rounds needed to first reach a target F1 threshold, the adaptive optimizers consistently require fewer rounds than FedAvg under DP noise: FedAdam and FedYogi climb steadily toward their plateau within the first 10–20 rounds, whereas FedAvg’s noisy, oscillating trajectory repeatedly crosses and falls back below the threshold, so it either reaches it later or fails to hold it. Faster, more reliable convergence is operationally valuable in federated deployments, where each additional round consumes communication bandwidth and—under DP—additional privacy budget; methods that converge in fewer rounds therefore reach a given utility at a lower cumulative privacy cost.

D. Effects of Non-IID Data (RQ3)

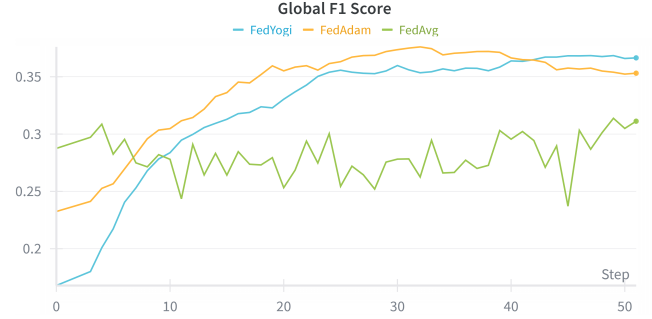
Finally, we combine DP with non-IID partitioning, varying the Dirichlet partitioners *concentration* α and ϵ for all three strategies (5 clients). Figure 7 contrasts a more heterogeneous setting ($\alpha = 5$) with a more homogeneous one ($\alpha = 10$) on MIMIC-III at $\epsilon = 0.5$.

As a reference, we first ran the same non-IID partitions *without* any privacy mechanism. There, heterogeneity alone already lowers and slows convergence: the smallest α (0.2) yields markedly lower F1 and slower, noisier learning, while larger α recovers stable convergence for all methods. The aggregation strategies behave similarly to the IID baseline at matched α , with FedAvg converging more slowly and showing higher loss; interestingly, on Adult the adaptive optimizers exhibit somewhat larger early weight oscillations than FedAvg in the non-private non-IID case. This isolates the heterogeneity effect and provides the baseline against which the differentially private runs below are interpreted.

Both stronger heterogeneity (smaller α) and tighter privacy (smaller ϵ) reduce F1 and increase fluctuations, and the two effects compound: the worst behavior appears when both are extreme. Increasing α toward IID improves convergence speed, final F1, and smoothness across all methods, and this benefit is most pronounced under strict privacy. Aggregation choice interacts with heterogeneity in a dataset-dependent way. On MIMIC-III, FedAvg performs worst—never producing a smooth learning curve and making weak progress under joint high heterogeneity and strict privacy—while FedAdam and FedYogi train stably and reach higher F1; their gap narrows as α and ϵ grow, and they overlap at high α with looser privacy. On Adult, the differences are smaller, and FedYogi can converge somewhat slower than FedAdam/FedAvg at the lowest α and ϵ . Crucially, the accuracy/F1 divergence persists: FedAvg often attains the *highest* accuracy while having a *lower* F1, because it predicts the majority class well but the minority class poorly—reinforcing that stability and F1 should be prioritized over raw accuracy in imbalanced, privacy-constrained FL. These observations are consistent with non-private clinical FL benchmarks that find simple averaging



(a) $\epsilon = 0.5$, $\alpha = 5$ (more heterogeneous).



(b) $\epsilon = 0.5$, $\alpha = 10$ (less heterogeneous).

Fig. 7: Global F1 on MIMIC-III (5 clients) under DP and non-IID data. FedAvg (green) is unstable and lower; FedAdam and FedYogi converge smoothly. Higher α (less skew) improves stability and final F1 for all methods.

brittle under heterogeneity and imbalance, and they extend that picture to the differentially private regime.

E. Practical Guidelines

Our findings suggest several recommendations for deploying DP-FL. (i) Calibrate the privacy budget per dataset: a sufficiently large ϵ preserves most utility, but the usable threshold is dataset-specific and notably stricter for sensitive healthcare data than for the Adult Income dataset. (ii) Prefer adaptive server optimizers (FedAdam, FedYogi) over plain FedAvg when operating under tight privacy or strong heterogeneity, as they deliver smoother training, smaller weight changes, and higher F1 at equal privacy. (iii) Choosing fitting metrics such as F1 and AUROC is important, since accuracy can mask failure on minority classes. (iv) Reducing heterogeneity (or designing for it) pays off most precisely when privacy is strongest.

F. Limitations

Several design choices bound the scope of our conclusions. We deliberately use a single, low-complexity model family (logistic regression) so that observed effects are attributable to privacy and aggregation rather than to architecture; richer models may interact differently with DP noise, and confirming our findings on deep architectures is left to future work.

However, the model is sufficiently powerful for the two chosen datasets. Experiments are run as a simulation of cross-silo FL with full client participation in every round, which controls for stragglers and sampling noise but does not capture the partial participation and communication constraints typical of large-scale mobile FL. We study two tabular benchmarks from two domains; while they were chosen for realism and comparability, broader dataset coverage would strengthen external validity. Finally, we enforce record-level DP with a per-epoch budget and RDP accounting; user-level guarantees and alternative accountants could shift the precise thresholds we report, even though we expect the qualitative trends—the budget threshold and the relative robustness of adaptive aggregation—to persist.

V. CONCLUSION AND FUTURE WORK

We presented a controlled empirical study of how privacy budgets, aggregation strategies, and non-IID data jointly affect differentially private federated learning on two benchmark datasets. Across both datasets we observed the expected privacy–utility trade-off with a sharp budget threshold below which training collapses, a consistent robustness advantage of adaptive aggregation (FedAdam, FedYogi) over FedAvg under DP noise and heterogeneity. While no single adaptive aggregation strategy dominates in every setting, and the best choice depends on the dataset, the degree of heterogeneity, and the privacy requirement, adaptive methods are favored over the FedAvg baseline. Future work includes evaluating additional and personalized aggregation rules, studying partial client participation per round (common in mobile FL), and extending the analysis from record-level to user-level differential privacy, which is required by many real applications.

REFERENCES

- [1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas, “Communication-Efficient Learning of Deep Networks from Decentralized Data,” in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2017.
- [2] Q. Yang, Y. Liu, T. Chen, and Y. Tong, “Federated Machine Learning: Concept and Applications,” *ACM Trans. Intell. Syst. Technol.*, vol. 10, no. 2, 2019.
- [3] E. T. Martínez Beltrán, M. Quiles Pérez, P. M. Sánchez Sánchez, S. López Bernal, G. Bovet, M. Gil Pérez, G. Martínez Pérez, and A. Huertas Celdrán, “Decentralized Federated Learning: Fundamentals, State of the Art, Frameworks, Trends, and Challenges,” *IEEE Communications Surveys & Tutorials*, vol. 25, no. 4, 2023.
- [4] L. Zhu, Z. Liu, and S. Han, “Deep Leakage from Gradients,” in *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2019.
- [5] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, “Membership Inference Attacks Against Machine Learning Models,” in *2017 IEEE Symposium on Security and Privacy (SP)*, 2017.
- [6] M. Fredrikson, S. Jha, and T. Ristenpart, “Model Inversion Attacks that Exploit Confidence Information and Basic Countermeasures,” in *ACM SIGSAC Conference on Computer and Communications Security*. New York, NY, USA: Association for Computing Machinery, 2015.
- [7] C. Dwork, F. McSherry, K. Nissim, and A. Smith, “Calibrating Noise to Sensitivity in Private Data Analysis,” in *Lecture Notes in Computer Science*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006.
- [8] C. Dwork and A. Roth, *The Algorithmic Foundations of Differential Privacy*, 2014.
- [9] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, “Deep Learning with Differential Privacy,” in *ACM SIGSAC Conference on Computer and Communications Security*. New York, NY, USA: Association for Computing Machinery, 2016.
- [10] S. J. Reddi, Z. Charles, M. Zaheer, Z. Garrett, K. Rush, J. Konečný, S. Kumar, and H. B. McMahan, “Adaptive Federated Optimization,” 2020.
- [11] H. Chen and H. Vikalo, “Federated Learning in Non-IID Settings Aided by Differentially Private Synthetic Data,” in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2023.
- [12] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, “SCAFFOLD: Stochastic Controlled Averaging for Federated Learning,” in *International Conference on Machine Learning*. PMLR, 2020.
- [13] J. Fu, Y. Hong, X. Ling, L. Wang, X. Ran, Z. Sun, W. H. Wang, Z. Chen, and Y. Cao, “Differentially private federated learning: A systematic review,” *arXiv preprint arXiv:2405.08299*, 2024.
- [14] A. N. Horvath, M. Berchier, F. Nooralahzadeh, A. Allam, and M. Krauthammer, “Exploratory Analysis of Federated Learning Methods with Differential Privacy on MIMIC-III,” 2023, arXiv:2302.04208.
- [15] D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” 2017, arXiv:1412.6980 [cs].
- [16] M. Zaheer, S. Reddi, D. Sachan, S. Kale, and S. Kumar, “Adaptive Methods for Nonconvex Optimization,” in *Advances in Neural Information Processing Systems*, vol. 31. Curran Associates, Inc., 2018.
- [17] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, “Federated Optimization in Heterogeneous Networks,” *Proceedings of Machine Learning and Systems*, vol. 2, 2020.
- [18] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, “Machine Learning with Adversaries: Byzantine Tolerant Gradient Descent,” in *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017.
- [19] D. Yin, Y. Chen, R. Kannan, and P. Bartlett, “Byzantine-Robust Distributed Learning: Towards Optimal Statistical Rates,” in *International Conference on Machine Learning*. PMLR, 2018.
- [20] I. Mironov, “Rényi Differential Privacy.” IEEE Computer Society, 2017.
- [21] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth, “Practical Secure Aggregation for Privacy-Preserving Machine Learning,” in *ACM SIGSAC Conference on Computer and Communications Security*. New York, NY, USA: Association for Computing Machinery, 2017.
- [22] H. B. McMahan, D. Ramage, K. Talwar, and L. Zhang, “Learning Differentially Private Recurrent Language Models,” 2018, arXiv:1710.06963 [cs].
- [23] X. Ren, S. Yang, C. Zhao, J. McCann, and Z. Xu, “Belt and Braces: When Federated Learning Meets Differential Privacy,” *Commun. ACM*, vol. 67, no. 12, 2024.
- [24] C. Wei, W. Li, C. Gong, and W. Chen, “DC-SGD: Differentially Private SGD With Dynamic Clipping Through Gradient Norm Distribution Estimation,” *IEEE Transactions on Information Forensics and Security*, vol. 20, 2025.
- [25] M. Noble, A. Bellet, and A. Dieuleveut, “Differentially private federated learning on heterogeneous data,” in *International Conference on Artificial Intelligence and Statistics (AISTATS)*, ser. PMLR, 2022.
- [26] G. A. Kaissis, M. R. Makowski, D. Rückert, and R. F. Braren, “Secure, privacy-preserving and federated machine learning in medical imaging,” *Nature Machine Intelligence*, vol. 2, no. 6, 2020.
- [27] M. Morafah, S. Vahidian, W. Wang, and B. Lin, “FLIS: Clustered Federated Learning Via Inference Similarity for Non-IID Data Distribution,” *IEEE Open Journal of the Computer Society*, vol. 4, 2023.
- [28] “Get Started | Responsible AI Toolkit,” 2022.
- [29] A. Johnson, T. Pollard, and R. Mark, “MIMIC-III Clinical Database,” 2016.
- [30] S. Purushotham, C. Meng, Z. Che, and Y. Liu, “Benchmarking deep learning models on large healthcare datasets,” *Journal of Biomedical Informatics*, vol. 83, 2018.
- [31] R. K. Barry Becker, “Adult,” 1996.
- [32] H. He, *Imbalanced Learning: Foundations, Algorithms, and Applications*, 1st ed. Somerset: John Wiley & Sons, Incorporated, 2013.
- [33] L.-P. Chen, “Supervised Learning for Binary Classification on US Adult Income,” *Journal of Modeling and Optimization*, vol. 13, no. 2, 2021.