DP-FedSOFIM: Differentially Private Federated Stochastic Optimization using Regularized Fisher Information Matrix

Anonymous authors
Paper under double-blind review

Abstract

Differentially private federated learning (DP-FL) often suffers from slow convergence under tight privacy budgets because the noise required for privacy preservation degrades gradient quality. Although second-order optimization can accelerate training, existing approaches for DP-FL face significant scalability limitations: Newton-type methods require clients to compute Hessians, while feature covariance methods scale poorly with model dimension. We propose **DP-FedSOFIM**, a simple and scalable Hessian approximation based second-order optimization method for DP-FL. The method constructs a regularized proxy for the Fisher information matrix at the server using only privatized aggregated gradients, capturing useful curvature information without requiring full Hessian computations or feature covariance estimation. Efficient rank-one updates based on the Sherman-Morrison formula enable communication costs proportional to the model size and require only $O(d)$ client side memory. Because all curvature and preconditioning operations are performed at the server on already privatized gradients, **DP-FedSOFIM** introduces no additional privacy cost beyond the underlying privatized gradient release mechanism. Experiments on CIFAR-10 and PathMNIST demonstrate that DP-FedSOFIM converges faster and consistently achieves higher accuracy than several competitive differentially private federated learning baselines across a wide range of privacy budgets.

1 Introduction

Federated learning (FL) has emerged as a widely adopted paradigm for training machine learning models across geographically distributed data sources without centralizing raw data (McMahan et al., 2017a; Kairouz & McMahan, 2021; Li et al., 2020a). Its appeal is especially pronounced in privacy sensitive domains such as healthcare and finance, where regulatory constraints and ethical obligations make direct data sharing infeasible (Rieke et al., 2020; Sheller et al., 2020). Yet the mere absence of raw data transfer does not constitute a formal privacy guarantee. A growing body of work has demonstrated that shared model gradients can leak sensitive training information through gradient inversion attacks (Zhu et al., 2019; Geiping et al., 2020), motivating the integration of rigorous privacy mechanisms into the federated training pipeline.

Differential privacy (DP) provides the gold standard for bounding information leakage in iterative learning algorithms (Dwork et al., 2006; 2014). The seminal DP-SGD framework of Abadi et al. (2016) operationalized DP for deep learning via per example gradient clipping and calibrated Gaussian noise addition, and introduced the moments accountant for tight privacy composition across training steps. Extending this approach to federated settings, McMahan et al. (2017b) proposed DP-FedAvg, which enforces user level differential privacy by clipping and noising client updates before aggregation. Subsequent work has sharpened the privacy utility tradeoff through more expressive accounting frameworks such as Rényi differential privacy (RDP) (Mironov, 2017), Gaussian DP (Dong et al., 2022), and privacy amplification by subsampling (Balle & Wang, 2018; Wang et al., 2019).

Despite these advances, differentially private federated learning (DP-FL) faces a persistent convergence–privacy tradeoff. To satisfy an (ϵ, δ)-DP guarantee, gradient updates must be clipped and perturbed with Gaussian noise scaled to the clipping threshold (Abadi et al., 2016). In the federated setting where communication costs

tightly limit the number of rounds and full client participation amplifies per round noise this perturbation can dominate the true gradient signal entirely, causing convergence to stall or degrade, particularly under stringent privacy budgets ($\epsilon < 2$) (Andrew et al., 2021). The problem is further compounded by client drift in heterogeneous data regimes (Karimireddy et al., 2020; Li et al., 2020b), where local gradients diverge from the global objective, and by the cumulative effect of noise composition over many rounds. These challenges together create a regime tight privacy, many clients, limited rounds where even well tuned first-order methods such as DP-FedAvg and DP-FedGD suffer severe utility loss.

A natural remedy is to exploit second-order curvature information, which can accelerate convergence by rescaling gradients along directions of high curvature precisely the directions where noise corrupted first-order steps are most wasteful. Adaptive methods such as Adam (Kingma & Ba, 2017) and AdaGrad (Duchi et al., 2011) pursue this idea via diagonal curvature estimates, and server side adaptive methods such as FedAdam and FedYogi (Reddi et al., 2020) carry this to federated settings without requiring second-order computation on clients. However, these methods do not explicitly leverage geometric curvature information and provide limited benefit under the anisotropic noise introduced by DP clipping. Exact second-order methods, including Newton-type approaches such as FedNL (Safaryan et al., 2022) and GIANT (Wang et al., 2018), as well as the Kronecker-factored curvature approximation K-FAC (Martens & Grosse, 2015), require clients to compute or transmit $O(d^2)$ Hessian or feature covariance information per round an $O(d)$-fold overhead over first-order methods that is prohibitive for high dimensional models and resource constrained edge devices.

The most closely related prior work, DP-FedNew (Krouka et al., 2025), introduces a second-order preconditioner for DP-FL based on local feature covariance matrices. While DP-FedNew demonstrates significant convergence speedups over DP-FedGD, its requirement that each client maintain and communicate an $O(d^2)$ feature covariance matrix limits its applicability to low dimensional generalized linear models and makes it impractical at the parameter scales of modern transfer learning pipelines. Moreover, computing second-order information at the client level introduces additional sensitivity that must be accounted for in the privacy analysis, potentially requiring tighter clipping or larger noise multipliers.

We introduce **DP-FedSOFIM** (Differentially Private Federated Stochastic Optimization using Regularized Fisher Information Matrix), a framework that resolves the scalability bottleneck of prior second-order DP-FL methods by relocating all curvature estimation and preconditioning entirely to the server. DP-FedSOFIM builds an online, rank one approximation to the Fisher Information Matrix (FIM) using only the privatized, aggregated gradients already available at the server, maintaining the inverse efficiently via the Sherman-Morrison formula (Sherman & Morrison, 1950) at $O(d)$ cost per round. Because all curvature computation is applied exclusively to already privatized quantities, the post-processing theorem (Dwork et al., 2014) guarantees that DP-FedSOFIM preserves the same (ϵ, δ)-DP guarantee as the underlying DP-FedGD baseline at no additional privacy cost.

While second-order optimization has long been known to accelerate convergence in centralized learning, extending such ideas to differentially private federated learning introduces several technical difficulties. In federated systems, curvature information cannot be computed from centralized data and must instead be inferred from aggregated client updates that are themselves corrupted by privacy noise. This creates a fundamental tension: reliable curvature estimation typically requires rich second-order statistics, yet differential privacy and communication constraints restrict what information can be transmitted from clients. Classical second-order approaches rely on dense Hessian or Fisher matrices with $O(d^2)$ memory and computation, rendering them impractical in large-scale federated environments. From a statistical perspective, the problem is further complicated by the fact that the curvature signal must be recovered from noisy gradient observations whose variance scales with the privacy mechanism.

The DP-FedSOFIM framework addresses this challenge by exploiting the structure of the aggregated gradient dynamics. Rather than attempting to estimate a full curvature matrix, we construct a rank-one Fisher proxy from the momentum buffer, which aggregates gradient information across rounds while attenuating differential privacy noise through exponential averaging. This structure enables an exact Sherman–Morrison inversion, yielding a curvature-adaptive preconditioner that can be applied in $O(d)$ time and memory. The resulting algorithm therefore integrates curvature-aware optimization with differential privacy in a manner

that preserves the communication efficiency required in cross-device federated learning, while still capturing the dominant curvature directions that govern convergence in ill-conditioned optimization landscapes.

Our key contributions are:

- **Server-Side Second-Order Preconditioning:** We design a curvature-aware natural-gradient preconditioner constructed entirely from privatized aggregated gradients, eliminating the need for client-side second-order computation.
- **Efficient $O(d)$ Implementation:** By leveraging the Sherman-Morrison formula for low-rank matrix updates (Sherman & Morrison, 1950), we achieve $O(d)$ computational complexity per round, making our approach scalable to high-dimensional models.
- **Privacy Preservation via Post-Processing:** We prove that server-side preconditioning preserves (ϵ, δ)-DP guarantees through the post-processing theorem (Dwork et al., 2014), as the FIM is computed on already-privatized gradient aggregates.
- **Empirical Validation:** Experiments on CIFAR-10 and PathMNIST with ResNet-20 frozen features and $n = 20$ clients demonstrate that DP-FedSOFIM consistently outperforms all baselines across all four privacy budgets ($\varepsilon \in \{0.5, 1, 5, 10\}$). It attains the best round-10 accuracy in seven of eight dataset/privacy regimes, with a $\sim 5\times$ reduction in rounds needed to reach 95% of DP-FedGD's final accuracy. On CIFAR-10, final-round gains over DP-FedGD reach up to +4.55% ($\varepsilon = 10$); on PathMNIST, where curvature-aware preconditioning is most effective on the ill-conditioned medical imaging task, gains reach up to +5.16% ($\varepsilon = 10$). DP-FedSOFIM achieves the best final-round accuracy in six of eight regimes and degrades gracefully under stringent privacy, retaining its convergence advantage down to $\varepsilon = 0.5$.

The remainder of this paper is organized as follows. Section 2 reviews related work on differentially private federated learning, second-order optimization in federated settings, and efficient matrix inversion techniques. Section 3 presents the DP-FedSOFIM algorithm and its efficient implementation. Section 4 provides theoretical convergence guarantees and privacy analysis. Section 5 reports experimental results, and Section 6 concludes with discussion and future directions. We provide detailed proofs of all results in the Appendix, along with several other methodological plus theoretical details including computational complexity analysis and runtime analysis, and additional experimental results.

2 Related Work

2.1 Differentially Private Federated Learning

Differential privacy in federated learning has been extensively studied since the foundational contributions of Dwork et al. (2006) and Dwork et al. (2014), who established the formal framework and core mechanisms including the Gaussian and Laplace mechanisms. Abadi et al. (2016) operationalized DP for deep learning via per-example gradient clipping and the moments accountant, providing the first practical technique for tracking privacy loss under iterative training. Building on this, McMahan et al. (2017b) proposed DP-FedAvg, extending user-level differential privacy to the federated setting by clipping and noising client updates prior to aggregation.

Subsequent work has focused on tightening the privacy-utility tradeoff through advanced accounting techniques. Mironov (2017) introduced Rényi differential privacy (RDP), which provides tighter composition bounds than the moments accountant and has become the standard tool for multi-round privacy analysis in DP-FL. Dong et al. (2022) developed Gaussian DP (GDP) as an alternative analytic framework that yields tight guarantees for the Gaussian mechanism specifically, and is the foundation for the Hockey-Stick divergence accounting used in our experiments. Privacy amplification by subsampling (Balle & Wang, 2018; Wang et al., 2019) provides further improvement when clients or data are sampled randomly, reducing the effective privacy cost per round and allowing smaller noise multipliers for a given (ϵ, δ) target.

On the optimization side, Andrew et al. (2021) demonstrated that adaptive clipping of client updates, where the clipping threshold is tuned to track a target gradient norm quantile, can substantially improve convergence under DP constraints without additional privacy cost. The interaction between client heterogeneity and differential privacy has also received attention: Karimireddy et al. (2020) introduced SCAFFOLD to correct for client drift via control variates, and our experiments show that this correction degrades under high privacy noise as the control variates become corrupted, an empirical finding we analyze in Section 5. Recent work has also studied personalized federated learning under differential privacy (Hu et al., 2020; Noble et al., 2022; Wei et al., 2023), heterogeneous privacy budgets across clients (Liu et al., 2022), and the interaction between local differential privacy and central DP (Naseri et al., 2022).

Second-order methods for DP-FL were first systematically explored by Krouka et al. (2025), who proposed DP-FedNew. DP-FedNew computes local feature covariance matrices at each client as an approximation to the FIM for generalized linear models, achieving substantial speedups over DP-FedGD. However, its requirement for local $O(d^2)$ memory per client limits scalability to high-dimensional models and resource-constrained devices. Furthermore, computing second-order statistics at the client introduces additional sensitivity that must be accounted for in the privacy analysis. Our work directly addresses these limitations by moving curvature estimation entirely to the server, preserving $O(d)$ client-side complexity and incurring no additional privacy cost by the post-processing theorem.

2.2 Adaptive and Second-Order Optimizers in Federated Learning

Adaptive optimization methods such as Adam (Kingma & Ba, 2017) and AdaGrad (Duchi et al., 2011) have become standard in centralized deep learning by maintaining per-parameter running estimates of gradient moments to adaptively rescale step sizes. Extending these methods to federated settings is non-trivial due to the need to aggregate adaptive statistics across heterogeneous clients (Reddi et al., 2020). FedAdam and FedYogi (Reddi et al., 2020) maintain server-side adaptive learning rates using pseudo-gradients formed by the aggregated client updates, but do not explicitly leverage second-order curvature information and therefore provide limited benefit in the highly noisy DP regime.

Beyond adaptive gradient methods, another line of work studies the Alternating Direction Method of Multipliers (ADMM) and related primal-dual optimization frameworks for distributed and federated learning. Several works have explored ADMM-based optimization for distributed and federated learning. FedADMM (Gong et al., 2022) introduces a primal-dual federated optimization framework that uses dual variables to address statistical and system heterogeneity while maintaining communication costs comparable to FedAvg. More recently, Möllenhoff et al. (2025) established a Bayesian interpretation of federated ADMM and derived Newton-like and Adam-like variants through variational Bayesian duality. ADMM based techniques have also been applied to communication efficient second-order federated optimization; for example, FedNPG-ADMM (Lan et al., 2023) approximates natural policy gradient directions while reducing communication complexity from $O(d^2)$ to $O(d)$. More broadly, ADMM has been studied as an alternative to gradient-based deep learning optimization and shown to possess favorable convergence properties in nonconvex neural network training (Zeng et al., 2021).

Natural gradient descent (Amari, 1998) uses the Fisher Information Matrix as a preconditioner, providing a principled, reparameterization-invariant approach to incorporating curvature. As established by Martens (2020), the empirical Fisher Information Matrix computed as the outer product of gradient vectors provides a tractable approximation that retains the key geometric properties of the true FIM and underpins many practical second-order methods. Computing and inverting the exact FIM requires $O(d^2)$ storage and $O(d^3)$ inversion, making it impractical for large-scale models. Approximation methods such as K-FAC (Martens & Grosse, 2015) and diagonal Fisher (Pascanu & Bengio, 2014) reduce these costs but introduce additional approximation errors and still require $O(d^2)$ factors per Kronecker block.

Several Newton-type methods have been proposed for distributed and federated settings. GIANT (Wang et al., 2018) communicates local Newton directions and performs a global approximate Newton step, achieving $O(d)$ communication per round but requiring $O(d^2)$ local Hessian computation. FedNL (Safaryan et al., 2022) uses compressed Hessian communication with contractive compressors, providing condition-number-independent local convergence, but requires clients to maintain and transmit Hessian information. SHED (Dal Fabbro et al.,

2024) incrementally shares eigenvalue-eigenvector pairs to reconstruct local Hessians at the server, reducing communication at the cost of stale curvature estimates. None of these methods address the differentially private setting, where client-side second-order computation introduces additional sensitivity that must be carefully managed.

DP-FedNew (Krouka et al., 2025) computes local feature covariance matrices at each client, which approximates the FIM for generalized linear models. While effective, this approach requires each client to store and communicate $O(d^2)$ parameters. In contrast, DP-FedSOFIM constructs a global FIM proxy on the server using aggregated noisy gradients, achieving $O(d)$ memory complexity while preserving the benefits of second-order preconditioning. Crucially, the server-side construction means that privacy and preconditioning are cleanly separated: any future improvement in privacy accounting or amplification applies equally to DP-FedSOFIM and DP-FedGD, and the accuracy gains from preconditioning are preserved across all privacy regimes.

2.3 Efficient Matrix Inversion Techniques

The Sherman-Morrison formula (Sherman & Morrison, 1950) provides an efficient method for updating matrix inverses when a matrix is modified by a rank-one perturbation, reducing an $O(d^3)$ inversion to an $O(d)$ update given the previous inverse. This technique has been applied in online convex optimization (Hazan et al., 2007), where it enables logarithmic regret algorithms that maintain curvature estimates in $O(d)$ space, and in recursive least squares (Haykin, 2002) for sequential parameter estimation. The broader family of rank-k updates is handled by the Woodbury matrix identity (Woodbury, 1950), of which Sherman-Morrison is the rank-one special case.

In the federated learning context, efficient low-rank matrix maintenance is essential for any server-side curvature method that must update its estimate every round. The SOFIM framework (Sen et al., 2024) demonstrated the viability of rank-one FIM approximations in the non-private federated setting, motivating our extension to the differentially private regime. We leverage the Sherman-Morrison formula to maintain the inverse of the regularized FIM in $O(d)$ per round, making DP-FedSOFIM computationally competitive with first-order baselines while delivering the convergence benefits of natural gradient descent.

3 Methodology

3.1 Federated Learning Formulation

We consider a synchronous federated learning system consisting of n clients indexed by $i \in [n] := \{1, \ldots, n\}$. Client i holds a private dataset $\mathcal{D}_i = \{(x_{i,j}, y_{i,j})\}_{j=1}^{|\mathcal{D}_i|}$. The global federated dataset is therefore $\mathcal{D} = \{\mathcal{D}_1, \ldots, \mathcal{D}_n\}$. The goal is to minimize the empirical risk

$$F(\theta) = \frac{1}{n} \sum_{i=1}^{n} F_i(\theta), \qquad F_i(\theta) = \frac{1}{|\mathcal{D}_i|} \sum_{(x,y) \in \mathcal{D}_i} \ell(\theta; x, y),$$

where $\theta \in \mathbb{R}^d$ denotes the model parameters and $\ell(\theta; x, y)$ is a differentiable loss function.

Optimization proceeds in communication rounds. At round t, clients compute local gradient information and transmit privatized updates to a central server. The server aggregates these updates and performs a global parameter update.

For clarity of exposition we assume full participation, i.e. all clients participate at every round. The mechanism itself does not rely on this assumption. Under partial participation, a subset $S_t \subset [n]$ of clients may be sampled at each round, in which case aggregation is performed over S_t rather than $[n]$. The remainder of the algorithm remains unchanged.

Remark 3.1 (Notation). Throughout the paper we use subscript time indices for all iterates and buffers. For example θ_t denotes the global parameter vector at round t, $G_t \in \mathbb{R}^d$ the aggregated gradient, and $M_t \in \mathbb{R}^d$ the server-side momentum buffer. The number of clients is denoted by n; the symbol K commonly used in federated learning literature is set equal to n here under full participation.

3.2 Privacy Model: Record-Level Differential Privacy

We adopt the record-level differential privacy framework introduced by Abadi et al. (2016). In this model the protected unit is an individual training example.

Definition 3.2 (Neighboring Datasets)**.** Two client datasets D_i and D_i' are said to be neighboring if they differ in exactly one data record while having the same size. Formally, $D_i' = (D_i \setminus \{z\}) \cup \{z'\}$ for some records z and z'.

This *replace-one* adjacency definition keeps the dataset size fixed, which simplifies sensitivity analysis for the normalized gradient releases used by the algorithm.

Definition 3.3 ((ε, δ)-Differential Privacy)**.** A randomized mechanism $\mathcal{M}$ satisfies (ε, δ)-DP if for all neighboring datasets $\mathcal{D}, \mathcal{D}'$ and all measurable sets S,

$$\mathbb{P}[\mathcal{M}(\mathcal{D}) \in S] \leq e^{\varepsilon} \mathbb{P}[\mathcal{M}(\mathcal{D}') \in S] + \delta.$$

Record-level DP provides protection for individual training examples even within a participating client's dataset. This notion of privacy is standard in privacy-preserving deep learning and federated optimization (Abadi et al., 2016).

3.3 Baseline Mechanism: DP-FedGD

We first recall the differentially private federated gradient descent algorithm (DP-FedGD), which forms the foundation of our method.

At round t, each client computes per-example gradients $g(x, y) = \nabla \ell(\theta_t; x, y)$.

Per-example gradient clipping. To control sensitivity, gradients are clipped to an ℓ_2 radius C_g:

$$\bar{g}(x, y) = g(x, y) \cdot \min\left(1, \frac{C_g}{\|g(x, y)\|_2}\right). \tag{1}$$

This guarantees $\|\bar{g}(x, y)\|_2 \leq C_g$, which yields a uniform ℓ_2 sensitivity bound for the summed gradients.

Client-side Gaussian perturbation. Let $S_{i,t} = \sum_{(x,y) \in \mathcal{D}_i} \bar{g}(x, y)$ denote the sum of clipped gradients on client i. Client i releases the noisy normalized update

$$g_{i,t} = \frac{1}{|\mathcal{D}_i|} \left(S_{i,t} + E_{i,t}\right), \qquad E_{i,t} \sim \mathcal{N}\left(0, \frac{(C_g \sigma_g)^2}{n} I_d\right). \tag{2}$$

The noise multiplier σ_g controls the privacy–utility tradeoff.

Server aggregation. The server aggregates client updates as $G_t = n^{-1} \sum_{i=1}^{n} g_{i,t}$. The global model is updated via $\theta_{t+1} = \theta_t - \eta_t G_t$.

Gradient clipping bounds the ℓ_2 sensitivity of the client update, while Gaussian noise is added according to the Gaussian mechanism. The resulting differential privacy guarantees depend on the clipping threshold, noise scale, dataset normalization, and the privacy accountant used across rounds. A formal derivation of the (ε, δ)-privacy guarantees for the full DP-FedSOFIM procedure is provided in Section 4.10. However, DP-FedGD is fundamentally a first-order optimization method. In ill-conditioned problems, anisotropic curvature and privacy-induced noise can significantly slow convergence. The concrete privacy parameters of the Gaussian release are derived in Appendix F, while Section 4.10 shows that the SOFIM preconditioning introduces no additional privacy loss beyond this release mechanism.

3.4 DP-FedSOFIM: Privacy Preserving Second Order Federated Optimization

To address this limitation we propose **DP-FedSOFIM**, which augments DP-FedGD with a curvature-aware server-side preconditioning step inspired by the SOFIM framework of Sen et al. (2024). The original SOFIM method constructs scalable structured Fisher approximations for non-private federated learning. Our contribution extends this idea to the differentially private regime while preserving record-level privacy guarantees. The design philosophy is deliberately conservative with respect to privacy. The client-side mechanism remains identical to DP-FedGD: clipping and Gaussian perturbation are unchanged. All curvature information is constructed solely from already-privatized aggregated gradients. Consequently, privacy guarantees follow directly from the post-processing property of differential privacy.

3.4.1 Natural Gradient Motivation

For a parametric model $p(y|x;\theta)$, the Fisher Information Matrix (FIM) is defined as

$$\mathcal{I}(\theta) = \mathbb{E}\left[\nabla \log p(y|x;\theta) \nabla \log p(y|x;\theta)^\top\right].$$

Natural gradient descent (Amari, 1998) updates parameters according to

$$\theta_{t+1} = \theta_t - \eta_t \mathcal{I}(\theta_t)^{-1} \nabla F(\theta_t),$$

which corresponds to steepest descent under the Riemannian metric induced by the Fisher information. Exact computation of the FIM is typically infeasible in federated settings. We therefore adopt a structured rank-one approximation inspired by the SOFIM framework.

3.4.2 Server-Side Curvature Proxy

The server maintains an exponential moving average of aggregated gradients

$$M_t = \beta M_{t-1} + (1-\beta) G_t, \qquad M_{-1} = \mathbf{0}_d, \quad \beta \in [0,1),$$

where G_t denotes the aggregated client update at communication round t. The momentum buffer stabilizes curvature estimation by smoothing stochastic fluctuations arising from client sampling, data heterogeneity, and privacy-induced noise.

Using the momentum estimate, we construct the curvature proxy

$$\widehat{\mathcal{I}}_t = M_t M_t^\top + \rho I_d,$$

where $\rho > 0$ is a regularization parameter that ensures positive definiteness and numerical stability.

The proposed approximation is intentionally low-rank and is not intended to recover the full Fisher Information Matrix (FIM), which is generally impractical to compute and store in large scale federated learning systems. Instead, the momentum estimate M_t serves as a stable and noise-reduced aggregation of historical gradient information, enabling the construction of a structured Fisher-inspired curvature surrogate. By accumulating information across communication rounds, the resulting proxy captures useful curvature characteristics while remaining computationally efficient.

A key advantage of the rank-one structure is that it enables efficient inverse preconditioning through the Sherman–Morrison formula, thereby avoiding the $O(d^2)$ memory and computational costs associated with full-matrix Fisher approximations. Consequently, the preconditioner $\widehat{\mathcal{I}}_t^{-1}$ introduces anisotropic rescaling of gradient updates, yielding curvature aware optimization at essentially first-order computational complexity. Importantly, the momentum buffer is used solely to construct the curvature proxy and does not directly determine the model update. Instead, the model update is obtained by preconditioning the aggregated gradient with the inverse Fisher-inspired curvature proxy $\widehat{\mathcal{I}}_t^{-1}$, resulting in a curvature-aware optimization step.

3.4.3 Efficient Inversion via Sherman-Morrison

Since $\widehat{\mathcal{I}}_t$ is a rank-one perturbation of ρI_d, its inverse admits the closed form

$$H_t = \widehat{\mathcal{I}}_t^{-1} = \frac{1}{\rho} I_d - \frac{M_t M_t^\top}{\rho^2 + \rho \|M_t\|_2^2},$$

which follows from the Sherman–Morrison identity (Sherman & Morrison, 1950).

Applying this inverse to the gradient yields

$$H_t G_t = \frac{1}{\rho} G_t - \frac{M_t (M_t^\top G_t)}{\rho^2 + \rho \|M_t\|_2^2}.$$

This expression requires only two inner products and vector operations, leading to an $O(d)$ computational cost per round. We discuss this in more detail in Appendix D.

Preconditioned server update. The global model is updated via

$$\theta_{t+1} = \theta_t - \eta_t H_t G_t.$$

Privacy preservation. The matrix H_t is a deterministic function of the privatized aggregated gradient G_t and the previous server state (θ_t, M_{t-1}). Since differential privacy is closed under post-processing, the mapping $G_t \mapsto H_t G_t$ does not incur any additional privacy loss. Therefore the DP-FedSOFIM update inherits the same (ε, δ)-DP guarantee as the underlying DP-FedGD mechanism.

Interpretation. DP-FedSOFIM can be viewed as a privacy-compatible, natural-gradient-motivated method in which curvature information is estimated entirely from privatized signals. The algorithm balances three competing factors: (i) clipping-induced bias, (ii) Gaussian noise required for privacy, and (iii) curvature-aware rescaling intended to mitigate ill-conditioning.

Warm-started preconditioning. Under very tight privacy budgets, the first few privatized aggregate gradients may be dominated by Gaussian noise, and the momentum buffer M_t may therefore be unstable during the early rounds. A simple stabilization is to introduce the preconditioner gradually. Define

$$q_t = (1 - \lambda_t)\rho^{-1} G_t + \lambda_t H_t G_t, \qquad 0 \le \lambda_t \le 1, \qquad \lambda_t \uparrow 1,$$

and update

$$\theta_{t+1} = \theta_t - \eta_t q_t.$$

For $\lambda_t = 0$, the method reduces to a scaled DP-FedGD step, while for $\lambda_t = 1$, it recovers DP-FedSOFIM. This warm-started variant can reduce early-round instability under stringent privacy budgets by allowing the momentum buffer to accumulate a more stable privatized gradient history before full anisotropic preconditioning is applied. Since q_t is still a deterministic function of the already privatized aggregate G_t and the previous server state, the warm-started update incurs no additional privacy cost.

3.5 Algorithmic Summary

Algorithm 1 summarizes the complete DP-FedSOFIM procedure. Client side operations coincide exactly with DP-FedGD; the only modification occurs on the server through the curvature-aware preconditioned update.

4 Convergence and Privacy Analysis

This section establishes optimization and privacy guarantees for DP-FedSOFIM. All algorithmic quantities are exactly those defined in Section 3; in particular, the server receives the privatized aggregated gradient

Algorithm 1 DP-FedSOFIM

Input: θ_0, learning rate η, clipping threshold C_g, noise multiplier σ_g, momentum β, regularization ρ, rounds T

Server initialization: $M_{-1} = \mathbf{0}_d$

for $t = 0, \ldots, T-1$ **do**

  // Client side (identical to DP-FedGD)

  for each client $i \in [n]$ **do**

    $S_{i,t} = \sum_{(x,y)\in\mathcal{D}_i} \frac{\nabla \ell(\theta_t; x, y)}{\max\big(1,\ \|\nabla \ell(\theta_t; x, y)\|_2 / C_g\big)}$

    Sample $E_{i,t} \sim \mathcal{N}\big(0,\ (C_g\sigma_g)^2 I_d / n\big)$

    $g_{i,t} = \big(S_{i,t} + E_{i,t}\big) \,/\, |\mathcal{D}_i|$

  end for

  // Server side (curvature-aware update)

  $G_t = \frac{1}{n}\sum_{i=1}^{n} g_{i,t}$

  $M_t = \beta M_{t-1} + (1-\beta)\, G_t$

  $H_t G_t = \frac{1}{\rho} G_t - \frac{M_t(M_t^\top G_t)}{\rho^2 + \rho\|M_t\|_2^2}$

  $\theta_{t+1} = \theta_t - \eta_t H_t G_t$

end for

$G_t = \frac{1}{n}\sum_{i=1}^{n} g_{i,t}$, updates the momentum buffer, forms the rank-one preconditioner, and performs the parameter update $\theta_{t+1} = \theta_t - \eta H_t G_t$.

The algorithm allows a general stepsize sequence $\{\eta_t\}_{t\geq 0}$. For the convergence analysis below, we specialize to the constant stepsize regime $\eta_t \equiv \eta$. A technical feature of DP-FedSOFIM is that the preconditioner is formed from the *current* privatized aggregate rather than from a delayed or auxiliary quantity. Consequently, H_t and G_t are coupled within the same round. The analysis below handles this same-step dependence directly.

To analyze the stochastic behavior of the update, we decompose the aggregated gradient into a clipped population component and a privacy noise term.

We define the aggregated clipped gradient

$$g_{\text{clip}}(\theta_t) := \frac{1}{n}\sum_{i=1}^{n} \frac{1}{|\mathcal{D}_i|} \sum_{(x,y)\in\mathcal{D}_i} \bar{g}(x, y),$$

where $\bar{g}(x, y)$ denotes the clipped per-example gradient defined in (1). The server aggregate can therefore be written as

$$G_t = g_{\text{clip}}(\theta_t) + \xi_t = \nabla F(\theta_t) + \zeta_t + \xi_t, \tag{3}$$

where $\zeta_t := g_{\text{clip}}(\theta_t) - \nabla F(\theta_t)$ is the clipping bias and ξ_t is the noise introduced by the privacy mechanism.

From the client update rule (2), the noise term admits the explicit representation

$$\xi_t = \frac{1}{n}\sum_{i=1}^{n} \frac{E_{i,t}}{|\mathcal{D}_i|}, \tag{4}$$

where the injected noises are independent Gaussian vectors

$$E_{i,t} \sim \mathcal{N}\left(0, \frac{(C_g\sigma_g)^2}{n} I_d\right).$$

Let $\mathcal{F}_t$ denote the *pre-round filtration*, i.e., the sigma-field generated by all algorithmic randomness up to the start of round t. In particular, θ_t, M_{t-1}, $\nabla F(\theta_t)$, $g_{\text{clip}}(\theta_t)$, and ζ_t are $\mathcal{F}_t$-measurable, whereas ξ_t is the fresh noise generated during round t. This gives, $\mathbb{E}[\xi_t \mid \mathcal{F}_t] = 0$.

Using independence across clients, the covariance of ξ_t is $\mathbb{E}[\xi_t \xi_t^\top \mid \mathcal{F}_t] = \nu_t^2 I_d$, where

$$\nu_t^2 = \frac{(C_g \sigma_g)^2}{n^3} \sum_{i=1}^{n} \frac{1}{|\mathcal{D}_i|^2}. \tag{5}$$

Lemma 4.1 (Conditional mean under preconditioning)**.** *Let $G_t = g_{\text{clip}}(\theta_t) + \xi_t$ and $H_t = (\rho I_d + M_t M_t^\top)^{-1}$. Then, conditional on the pre-round filtration $\mathcal{F}_t$,*

$$\mathbb{E}[G_t \mid \mathcal{F}_t] = g_{\text{clip}}(\theta_t).$$

Equivalently,

$$\mathbb{E}[G_t - g_{\text{clip}}(\theta_t) \mid \mathcal{F}_t] = 0.$$

Proof. By (3), $G_t = g_{\text{clip}}(\theta_t) + \xi_t$. Since $g_{\text{clip}}(\theta_t)$ is $\mathcal{F}_t$-measurable and $\mathbb{E}[\xi_t \mid \mathcal{F}_t] = 0$,

$$\mathbb{E}[G_t \mid \mathcal{F}_t] = g_{\text{clip}}(\theta_t) + \mathbb{E}[\xi_t \mid \mathcal{F}_t] = g_{\text{clip}}(\theta_t).$$

□

Thus the aggregated gradient estimator G_t is conditionally unbiased for the clipped population gradient and has isotropic Gaussian noise with variance parameter ν_t^2.

4.1 Assumptions

Assumption 4.2 (L-smoothness)**.** Each local objective F_i is L-smooth. Consequently, the global objective F is L-smooth:

$$\|\nabla F(\theta) - \nabla F(\theta')\|_2 \le L\|\theta - \theta'\|_2, \qquad \forall\, \theta, \theta' \in \mathbb{R}^d.$$

Equivalently, for all $\theta, \Delta \in \mathbb{R}^d$,

$$F(\theta + \Delta) \le F(\theta) + \langle \nabla F(\theta), \Delta \rangle + \frac{L}{2}\|\Delta\|_2^2. \tag{6}$$

Assumption 4.3 (μ-strong convexity)**.** The global objective F is μ-strongly convex for some $\mu > 0$, i.e.,

$$F(\theta') \ge F(\theta) + \langle \nabla F(\theta), \theta' - \theta \rangle + \frac{\mu}{2}\|\theta' - \theta\|_2^2, \qquad \forall\, \theta, \theta' \in \mathbb{R}^d.$$

In particular,

$$\|\nabla F(\theta)\|_2^2 \ge 2\mu\big(F(\theta) - F(\theta^\star)\big), \qquad \forall\, \theta \in \mathbb{R}^d, \tag{7}$$

where $\theta^\star$ denotes the unique minimizer of F.

Remark 4.4 (Strong Convexity in Frozen-Feature Models). When the feature extractor is frozen and only a linear prediction head is trained, strong convexity arises naturally in several common settings. For squared-loss linear regression, strong convexity holds whenever the feature covariance matrix is positive definite. For multiclass softmax models, strong convexity can be ensured by adding an explicit ℓ_2 regularization term or by imposing identifiability constraints on the parameterization. These conditions are standard in transfer learning scenarios where only the final prediction layer is optimized.

Assumption 4.5 (Bounded clipping bias)**.** There exists $\zeta_{\max} \ge 0$ such that $\|\zeta_t\|_2 \le \zeta_{\max}$, $\forall\, t \ge 0$.

Assumption 4.6 (Bounded gradient noise)**.** Let ξ_t denote the stochastic noise defined in (4). There exists a constant $\nu^2 < \infty$ such that $\mathbb{E}[\xi_t \mid \mathcal{F}_t] = 0$, $\mathbb{E}[\|\xi_t\|_2^2 \mid \mathcal{F}_t] \le d\nu^2$ for all iterations t.

Remark 4.7 (Noise magnitude in DP-FedSOFIM). For the mechanism defined in Section 3.3, the variance parameter in Assumption 4.6 is given explicitly by $\nu_t^2 = (C_g \sigma_g)^2 \sum_{i=1}^{n} |\mathcal{D}_i|^{-2}/n^3$. In the common case where all clients have equal dataset size $|\mathcal{D}_i| = m$, this simplifies to $\nu_t^2 = (C_g \sigma_g)^2/(n^2 m^2)$. Thus the variance of the aggregated noise decreases as $O(n^{-2})$ when client dataset sizes are equal, while the standard deviation decreases as $O(n^{-1})$.

Assumption 4.8 (Bounded gradient norm along the iterate sequence)**.** There exists $G_{\max} > 0$ such that $\|\nabla F(\theta_t)\|_2 \le G_{\max}$, $\forall\, t \ge 0$.

Assumption 4.8 is standard in analyses of adaptive or state-dependent preconditioners. In the present setting it is used only to control the same-step coupling between the current preconditioner H_t and the current privatized aggregate G_t. In the frozen-feature regime, it holds whenever features are bounded and the trainable head is confined to a bounded parameter region, or when explicit regularization prevents parameter growth.

Remark 4.9 (Role of bounded bias and bounded gradients). The bounded-bias and bounded-gradient assumptions play different roles in the analysis. The bounded-bias assumption controls the discrepancy between the privatized aggregate update and the target population gradient. In the one-step DP-FedGD setting this discrepancy is the clipping bias $\zeta_t = g_{\text{clip}}(\theta_t) - \nabla F(\theta_t)$, whereas in mini-batch or multi-local-step variants it may also contain stochastic approximation error and local-update drift. This assumption allows the descent bound to separate the optimization error from the perturbation introduced by clipping, privacy noise, and local computation.

The bounded-gradient assumption is used for a different reason. In DP-FedSOFIM, the preconditioner $H_t = (\rho I_d + M_t M_t^\top)^{-1}$ is formed using the same-round privatized aggregate G_t through the momentum recursion $M_t = \beta M_{t-1} + (1-\beta) G_t$. Thus H_t and G_t are statistically coupled within the same communication round. The bounded-gradient condition is used to control this same-step dependence, specifically the term arising from $\nabla F(\theta_t)^\top H_t \nabla F(\theta_t)$, where the lower bound contains a correction proportional to $G_{\max}^2 \|M_t\|_2^2/\rho^2$. The assumption is therefore not introduced to make the objective globally well-behaved, but to control the interaction between the current privatized gradient, the server-side momentum buffer, and the adaptive preconditioner. In the frozen-feature linear-head setting, such a condition is natural when features are bounded and the trainable parameter is regularized or kept in a bounded region.

The following lemma shows that the DP noise mechanism satisfies the bounded-noise assumption used in the optimization analysis. We provide the proof of the same in the Appendix.

Lemma 4.10 (Uniform bound on DP noise variance)**.** *Let ξ_t denote the noise defined in* (4)*. Assume that client dataset sizes satisfy $|\mathcal{D}_i| \ge m_{\min} > 0$ for all i. Then the variance parameter in* (5) *satisfies $\nu_t^2 \le (C_g \sigma_g)^2/(n^2 m_{\min}^2) =: \nu^2$. Consequently, $\mathbb{E}[\|\xi_t\|_2^2 \mid \mathcal{F}_t] \le d\nu^2$.*

4.2 Norm Control for the Clipped Aggregate and the Momentum Buffer

We begin with two elementary but fundamental estimates. The first is a deterministic norm bound coming directly from clipping. The second yields a uniform second-moment bound for the server momentum buffer. We provide proofs of both the lemmas in Appendix A.

Lemma 4.11 (Norm bound for the aggregated clipped gradient)**.** *For every $t \ge 0$,*

$$\|g_{\text{clip}}(\theta_t)\|_2 \le C_g. \tag{8}$$

Consequently,

$$\mathbb{E}\big[\|G_t\|_2^2 \mid \mathcal{F}_t\big] = \|g_{\text{clip}}(\theta_t)\|_2^2 + d\nu_t^2 \le C_g^2 + d\nu^2.$$

Lemma 4.12 (Conditional second-moment recursion for the momentum buffer)**.** *For every $t \ge 0$,*

$$\mathbb{E}\big[\|M_t\|_2^2 \mid \mathcal{F}_t\big] \le \beta \|M_{t-1}\|_2^2 + (1-\beta) C_g^2 + (1-\beta)^2 d\nu^2. \tag{9}$$

Define $u_t := \mathbb{E}\|M_t\|_2^2$. Then

$$u_t \le \beta u_{t-1} + (1-\beta) C_g^2 + (1-\beta)^2 d\nu^2, \qquad t \ge 0, \tag{10}$$

and, since $M_{-1} = 0_d$,

$$u_t \le \bar{M}^2 := C_g^2 + (1-\beta) d\nu^2, \qquad t \ge 0. \tag{11}$$

Remark 4.13 (Noise smoothing by the momentum buffer). Lemma 4.12 shows that the server-side buffer has uniformly bounded second moment despite the Gaussian privacy noise having unbounded support. This formalizes the stabilizing effect of the exponential moving average: the curvature proxy is built from a smoothed sequence of privatized aggregates rather than from a single noisy realization.

4.3 Properties of the Same-Step Preconditioner

We next establish deterministic structural bounds for the same-step preconditioner. These bounds hold pathwise for each realized M_t, and therefore remain valid even though M_t depends on the current privacy noise. Proof of this lemma is in Appendix A.

Lemma 4.14 (Sherman–Morrison representation and operator bounds)**.** *For every $t \geq 0$,*

$$H_t = \frac{1}{\rho} I_d - \frac{M_t M_t^\top}{\rho(\rho + \|M_t\|_2^2)}. \tag{12}$$

Consequently, for every $v \in \mathbb{R}^d$,

$$\|H_t v\|_2 \leq \frac{1}{\rho}\|v\|_2, \tag{13}$$

$$\|H_t v\|_2^2 \leq \frac{1}{\rho^2}\|v\|_2^2, \tag{14}$$

$$\frac{1}{\rho + \|M_t\|_2^2}\|v\|_2^2 \leq v^\top H_t v \leq \frac{1}{\rho}\|v\|_2^2. \tag{15}$$

In particular,

$$\nabla F(\theta_t)^\top H_t \nabla F(\theta_t) \geq \frac{1}{\rho + \|M_t\|_2^2}\|\nabla F(\theta_t)\|_2^2. \tag{16}$$

Corollary 4.15 (Non-vacuous lower bound on a bounded-momentum event)**.** *Fix $T \geq 1$ and $\delta_M \in (0,1)$. Let*

$$\bar{M}^2 = C_g^2 + (1-\beta)d\nu^2$$

be the uniform second-moment bound from Lemma 4.12, and define $M_\delta^2 := T\bar{M}^2/\delta_M$. Then

$$\mathbb{P}\left(\max_{0 \leq t \leq T-1} \|M_t\|_2^2 \leq M_\delta^2\right) \geq 1 - \delta_M.$$

On this event, for every $t = 0, \ldots, T-1$,

$$\nabla F(\theta_t)^\top H_t \nabla F(\theta_t) \geq \frac{1}{\rho + M_\delta^2}\|\nabla F(\theta_t)\|_2^2.$$

4.4 One-Step Descent for the Exact Same-Step Update

We now establish the central one-step descent inequality. Because the preconditioner H_t depends on the current privatized aggregate G_t, the proof must control the coupling among $\nabla F(\theta_t)$, H_t, and G_t within the same round. Before stating the one-step descent bound, we introduce two auxiliary Young parameters $\tau_1, \tau_2 > 0$. These constants are not algorithmic parameters. They are used only in the proof to control the cross-terms involving the clipping bias and the privacy noise. Specifically, Young's inequality is used to bound terms of the form

$$|\nabla F(\theta_t)^\top H_t \zeta_t| \quad \text{and} \quad |\nabla F(\theta_t)^\top H_t \xi_t|.$$

The parameter τ_1 controls the bias cross-term, while τ_2 controls the noise cross-term. Smaller choices of τ_1, τ_2 increase the descent coefficient but enlarge the additive bias and noise terms in the error floor.

Lemma 4.16 (Conditional one-step descent)**.** *Fix any $\tau_1, \tau_2 > 0$, and define*

$$c_\nabla := \frac{1}{\rho} - \frac{\tau_1 + \tau_2}{2}. \tag{17}$$

Under Assumptions 4.2, 4.5, 4.6, and 4.8, for every $t \geq 0$,

$$\begin{aligned}\mathbb{E}[F(\theta_{t+1}) \mid \mathcal{F}_t] \leq F(\theta_t) - \eta c_\nabla \|\nabla F(\theta_t)\|_2^2 + \frac{\eta G_{\max}^2}{\rho^2}\mathbb{E}\big[\|M_t\|_2^2 \mid \mathcal{F}_t\big] \\ + \frac{\eta\,\zeta_{\max}^2}{2\tau_1\rho^2} + \frac{\eta\, d\,\nu^2}{2\tau_2\rho^2} + \frac{L\eta^2}{2\rho^2}\big(C_g^2 + d\nu^2\big).\end{aligned} \tag{18}$$

The previous lemma still contains the conditional term $\mathbb{E}[\|M_t\|_2^2 \mid \mathcal{F}_t]$. The next lemma turns it into a deterministic constant.

Lemma 4.17 (Uniform unconditional one-step descent)**.** *Fix $\tau_1, \tau_2 > 0$ and define c_∇ as in* (17)*. Under Assumptions 4.2, 4.5, 4.6, and 4.8, for every $t \geq 0$,*

$$\mathbb{E}[F(\theta_{t+1})] \leq \mathbb{E}[F(\theta_t)] - \eta c_\nabla \, \mathbb{E}\|\nabla F(\theta_t)\|_2^2 + \Gamma, \tag{19}$$

where

$$\Gamma := \frac{\eta G_{\max}^2}{\rho^2} \bar{M}^2 + \frac{\eta \, \zeta_{\max}^2}{2\tau_1 \rho^2} + \frac{\eta \, d \, \nu^2}{2\tau_2 \rho^2} + \frac{L\eta^2}{2\rho^2}\big(C_g^2 + d\nu^2\big), \tag{20}$$

and $\bar{M}^2$ is given by (11)*.*

Proofs of both the above lemmas are in Appendix A.

Remark 4.18 (High-probability descent coefficient). The Young parameters τ_1, τ_2 in Lemma 4.16 control the bias and noise cross-terms. Combining the same Young bounds with Corollary 4.15 yields, on the bounded-momentum event, the pathwise coefficient

$$c_{\nabla,\delta} = \frac{1}{\rho + M_\delta^2} - \frac{\tau_1 + \tau_2}{2}.$$

This coefficient is non-vacuous whenever $\tau_1 + \tau_2 < 2/(\rho + M_\delta^2)$. We do not use $c_{\nabla,\delta}$ in the main expectation recursion; the main theorems use the conservative coefficient $c_\nabla = \rho^{-1} - (\tau_1 + \tau_2)/2$ and account for same-step coupling through the additive term in Γ.

4.5 Extension to General Privatized Aggregate Updates

The preceding analysis was stated for the one-step DP-FedGD-type update, where the server receives a clipped and privatized aggregate gradient. The same argument also applies to a broader class of privatized aggregate updates. This extension is useful for mini-batch implementations and multi-local-step DP-FedAvg-style variants.

Suppose the server receives a privatized aggregate update G_t satisfying

$$G_t = \nabla F(\theta_t) + b_t + \xi_t,$$

where b_t is an $\mathcal{F}_t$-measurable bias term and ξ_t is the fresh noise generated at round t. Assume

$$\|b_t\|_2 \leq B, \qquad \mathbb{E}[\xi_t \mid \mathcal{F}_t] = 0, \qquad \mathbb{E}[\|\xi_t\|_2^2 \mid \mathcal{F}_t] \leq d\nu^2.$$

Here b_t may contain clipping bias, stochastic mini-batch bias, and local-update drift. In the one-step DP-FedGD setting, b_t reduces to the clipping bias and $B = \zeta_{\max}$. In a multi-local-step DP-FedAvg-style implementation, one may write abstractly $B = B_{\text{clip}} + B_{\text{drift}}$, where B_{drift} captures the deviation between the aggregate local update and the global gradient at θ_t.

Proposition 4.19 (Descent under a biased privatized aggregate update)**.** *Suppose F is L-smooth and $\|\nabla F(\theta_t)\|_2 \leq G_{\max}$ along the iterate sequence. Let $M_t = \beta M_{t-1} + (1-\beta) G_t$, $H_t = (\rho I_d + M_t M_t^\top)^{-1}$, and $\theta_{t+1} = \theta_t - \eta H_t G_t$. Fix $\tau_1, \tau_2 > 0$, and define $c_\nabla = \rho^{-1} - (\tau_1 + \tau_2)/2$. If $\mathbb{E}[\|G_t\|_2^2 \mid \mathcal{F}_t] \leq G_0^2 + d\nu^2$ for some finite constant G_0, then*

$$\begin{aligned}\mathbb{E}[F(\theta_{t+1}) \mid \mathcal{F}_t] \leq F(\theta_t) - \eta c_\nabla \|\nabla F(\theta_t)\|_2^2 + \frac{\eta G_{\max}^2}{\rho^2} \mathbb{E}[\|M_t\|_2^2 \mid \mathcal{F}_t] \\ + \frac{\eta B^2}{2\tau_1 \rho^2} + \frac{\eta d \nu^2}{2\tau_2 \rho^2} + \frac{L\eta^2}{2\rho^2}(G_0^2 + d\nu^2).\end{aligned}$$

Consequently, if $\mathbb{E}\|M_t\|_2^2 \leq \overline{M}^2$ uniformly in t, then

$$\mathbb{E}[F(\theta_{t+1})] \leq \mathbb{E}[F(\theta_t)] - \eta c_\nabla \mathbb{E}\|\nabla F(\theta_t)\|_2^2 + \Gamma_B,$$

where

$$\Gamma_B = \frac{\eta G_{\max}^2}{\rho^2} \overline{M}^2 + \frac{\eta B^2}{2\tau_1 \rho^2} + \frac{\eta d \nu^2}{2\tau_2 \rho^2} + \frac{L\eta^2}{2\rho^2}(G_0^2 + d\nu^2).$$

Remark 4.20 (Relation to mini-batch and multi-local-step methods). Proposition 4.19 shows that the SOFIM preconditioner is not tied to a full-gradient, one-local-step implementation. The proof only requires a privatized aggregate update with controlled bias and second moment. Mini-batching changes the noise and bias constants, while multi-local-step local training contributes an additional drift term to B. Thus the same descent mechanism remains valid, with the price of local training appearing explicitly through the bias radius B.

4.6 Convergence under Strong Convexity

We now combine the one-step descent bound with strong convexity. We provide the proof of this result in Appendix A.

Theorem 4.21 (Linear convergence to a bias-and-noise neighborhood). *Fix $\tau_1, \tau_2 > 0$ such that c_∇ as defined in* (17) *is positive. Suppose Assumptions 4.2, 4.3, 4.5, 4.6, and 4.8 hold. Define $r := 1 - 2\mu\eta c_\nabla$. If $r \in (0,1)$, equivalently $0 < \eta < 1/(2\mu c_\nabla)$, then for every $T \geq 0$,*

$$\mathbb{E}[F(\theta_T) - F(\theta^\star)] \leq r^T\big(F(\theta_0) - F(\theta^\star)\big) + \frac{1-r^T}{1-r}\,\Gamma \tag{21}$$

$$\leq r^T\big(F(\theta_0) - F(\theta^\star)\big) + \frac{\Gamma}{2\mu\eta c_\nabla}, \tag{22}$$

where Γ is defined in (20).

Remark 4.22 (Interpretation of the error floor). The limiting neighborhood in (22) has three distinct sources. The term involving $d\nu^2$ is the differential-privacy noise floor. The term involving $\zeta_{\max}^2$ is the clipping-bias floor. The term involving $\bar{M}^2$ quantifies the additional cost induced by same-step coupling between the current preconditioner and the current privatized gradient. Since $\bar{M}^2 = C_g^2 + (1-\beta)d\nu^2$, the coupling penalty is itself controlled by the clipping threshold, the privacy noise scale, and the momentum parameter.

Remark 4.23 (Stepsize dependence of the error floor). Although the error floor is written as $\Gamma/(2\mu\eta c_\nabla)$, the quantity Γ itself depends on η. In particular, the clipping-bias, privacy-noise, and same-step coupling terms in Γ are of order η, whereas the smoothness term is of order η^2. Hence the limiting neighborhood consists of constant-order bias/noise/coupling terms plus an $O(\eta)$ smoothness contribution. Decreasing η slows the contraction factor but does not by itself inflate the limiting neighborhood.

Remark 4.24 (Choice of the Young parameters τ_1, τ_2). The parameters τ_1 and τ_2 arise solely from Young's inequality in the control of the bias and noise cross-terms. Smaller values improve the descent coefficient c_∇, but enlarge the additive constants in Γ; larger values do the opposite. For the convergence theorem one only needs $c_\nabla > 0$, i.e., $\tau_1 + \tau_2 < 2/\rho$. In practice one may choose τ_1 and τ_2 as fixed small constants satisfying this condition.

4.7 Comparison with the DP-FedGD Error Floor

We next compare the DP-FedSOFIM floor with the corresponding first-order baseline under the same clipping-bias and DP-noise assumptions. Consider DP-FedGD,

$$\theta_{t+1}^{\mathrm{GD}} = \theta_t^{\mathrm{GD}} - \eta G_t, \qquad G_t = \nabla F(\theta_t^{\mathrm{GD}}) + \zeta_t + \xi_t,$$

where $\|\zeta_t\|_2 \leq \zeta_{\max}$, $\mathbb{E}[\xi_t \mid \mathcal{F}_t] = 0$, and $\mathbb{E}[\|\xi_t\|_2^2 \mid \mathcal{F}_t] \leq d\nu^2$.

Lemma 4.25 (One-step descent for DP-FedGD). *Suppose F is L-smooth and Assumptions 4.5 and 4.6 hold. Then, for any $\tau > 0$,*

$$\mathbb{E}[F(\theta_{t+1}^{\mathrm{GD}}) \mid \mathcal{F}_t] \leq F(\theta_t^{\mathrm{GD}}) - \eta c_{\mathrm{GD}}\|\nabla F(\theta_t^{\mathrm{GD}})\|_2^2 + \Gamma_{\mathrm{GD}},$$

where

$$c_{\mathrm{GD}} := 1 - \frac{\tau}{2}, \qquad \Gamma_{\mathrm{GD}} := \frac{\eta\zeta_{\max}^2}{2\tau} + \frac{L\eta^2}{2}(C_g^2 + d\nu^2).$$

Theorem 4.26 (DP-FedGD error floor under strong convexity). *Suppose F is μ-strongly convex and the assumptions of Lemma 4.25 hold. If $c_{\mathrm{GD}} > 0$ and $0 < \eta < 1/(2\mu c_{\mathrm{GD}})$, then*

$$\mathbb{E}[F(\theta_T^{\mathrm{GD}}) - F(\theta^\star)] \leq (1 - 2\mu\eta c_{\mathrm{GD}})^T\{F(\theta_0) - F(\theta^\star)\} + \frac{\Gamma_{\mathrm{GD}}}{2\mu\eta c_{\mathrm{GD}}}.$$

The DP-FedGD floor is therefore $\Gamma_{\mathrm{GD}}/(2\mu\eta c_{\mathrm{GD}})$, whereas the DP-FedSOFIM floor in Theorem 4.21 is $\Gamma/(2\mu\eta c_{\nabla})$. These bounds do not imply that DP-FedSOFIM has a uniformly smaller worst-case error floor than DP-FedGD. The SOFIM bound contains additional terms arising from the same-step coupling between G_t and H_t, which is the price of using a data-dependent preconditioner constructed from the current privatized aggregate.

The theoretical advantage of DP-FedSOFIM appears when the preconditioner improves the effective descent geometry more than it increases the additive coupling term. The present analysis is conservative and does not fully exploit possible alignment between the momentum direction M_t, the gradient direction, and favorable curvature directions of the objective. Thus, our theorem proves stability and convergence of the server-side preconditioned method under DP noise, but it does not by itself prove a uniformly lower worst-case floor than DP-FedGD.

A sharper comparison is possible under an additional alignment or preconditioner-quality condition. For example, suppose that along the optimization path

$$\nabla F(\theta_t)^\top H_t \nabla F(\theta_t) \geq \alpha_H \|\nabla F(\theta_t)\|_2^2$$

for some $\alpha_H > 0$, and

$$\mathbb{E}\|H_t G_t\|_2^2 \leq \sigma_H^2 (C_g^2 + d\nu^2).$$

Then the SOFIM floor has the schematic form

$$\frac{\eta B_H + L\eta^2 \sigma_H^2 (C_g^2 + d\nu^2)}{2\mu\eta\alpha_H},$$

where B_H collects the bias and coupling terms. This floor can be smaller than the DP-FedGD floor when the effective-descent gain α_H and the variance reduction σ_H^2 dominate the additional coupling cost. This regime is plausible in ill-conditioned problems with a stable dominant direction in the privatized gradient trajectory, and is consistent with the empirical gains observed on PathMNIST and in the faster-convergence results.

4.8 Extension to the Polyak–Łojasiewicz Condition

The same descent argument extends verbatim from strong convexity to the Polyak–Łojasiewicz condition.

Assumption 4.27 (Polyak–Łojasiewicz condition)**.** There exists $\mu_{\mathrm{PL}} > 0$ such that for all $\theta \in \mathbb{R}^d$,

$$\frac{1}{2}\|\nabla F(\theta)\|_2^2 \geq \mu_{\mathrm{PL}}\big(F(\theta) - F(\theta^\star)\big). \tag{23}$$

Remark 4.28. The PL condition is strictly weaker than strong convexity. In particular, μ-strong convexity implies (23) with $\mu_{\mathrm{PL}} = \mu$, but the converse need not hold.

Theorem 4.29 (DP-FedSOFIM under the PL condition)**.** *Fix $\tau_1, \tau_2 > 0$ such that $c_\nabla > 0$ in* (17)*. Suppose Assumptions 4.2, 4.5, 4.6, 4.8, and 4.27 hold. Define $r_{\mathrm{PL}} := 1 - 2\mu_{\mathrm{PL}}\eta c_\nabla$. If $r_{\mathrm{PL}} \in (0,1)$, equivalently $0 < \eta < 1/(2\mu_{\mathrm{PL}} c_\nabla)$, then for every $T \geq 0$,*

$$\mathbb{E}[F(\theta_T) - F(\theta^\star)] \leq r_{\mathrm{PL}}^T\big(F(\theta_0) - F(\theta^\star)\big) + \frac{1 - r_{\mathrm{PL}}^T}{1 - r_{\mathrm{PL}}}\Gamma \tag{24}$$

$$\leq r_{\mathrm{PL}}^T\big(F(\theta_0) - F(\theta^\star)\big) + \frac{\Gamma}{2\mu_{\mathrm{PL}}\eta c_\nabla}, \tag{25}$$

where Γ is defined in (20)*.*

We provide the proof of the above result in Appendix A.

Remark 4.30 (Relationship between the strong-convexity and PL results). Since strong convexity implies the PL condition,Theorem 4.29 strictly generalizes Theorem 4.21. We state both versions explicitly because the strongly convex result aligns directly with the frozen linear-head regime used in the experiments, whereas the PL theorem extends the guarantee to a broader class of objectives.

4.9 Smooth Non-convex Stationarity

The preceding convergence result was stated under strong convexity, which is the most natural condition for the frozen-feature linear-head setting considered in our experiments. However, the one-step descent argument itself does not require convexity. It only requires smoothness, control of the bias in the privatized aggregate update, control of the DP noise, and a uniform bound on the gradient norm along the optimization path. Therefore, the same recursion also yields a standard stationarity guarantee for smooth non-convex objectives.

In the non-convex case, convergence to a global minimizer cannot be expected without additional geometric assumptions. We therefore measure optimization progress through the average squared gradient norm,

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla F(\theta_t)\|_2^2,$$

or equivalently through the squared gradient norm at a uniformly sampled iterate. This is the usual stationarity criterion in smooth non-convex optimization. The result below shows that DP-FedSOFIM reaches a stationary neighborhood whose size is determined by the same bias-and-noise term appearing in the one-step descent bound. Thus, the effects of clipping bias, privacy noise, and same-step preconditioner coupling remain explicit in the non-convex setting.

Theorem 4.31 (Stationarity under smooth non-convex objectives)**.** *Suppose Assumptions 4.2, 4.5, 4.6, and 4.8 hold. Let $c_\nabla = \rho^{-1} - (\tau_1+\tau_2)/2 > 0$ for some $\tau_1, \tau_2 > 0$, and suppose $F_{\inf} := \inf_{\theta\in\mathbb{R}^d} F(\theta) > -\infty$. Then, for every $T \geq 1$,*

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla F(\theta_t)\|_2^2 \leq \frac{F(\theta_0)-F_{\inf}}{\eta c_\nabla T} + \frac{\Gamma}{\eta c_\nabla},$$

where Γ is the bias-and-noise term defined in Lemma 4.17. Equivalently, if R is sampled uniformly from $\{0,\dots,T-1\}$, independently of the algorithmic randomness, then

$$\mathbb{E}\|\nabla F(\theta_R)\|_2^2 \leq \frac{F(\theta_0)-F_{\inf}}{\eta c_\nabla T} + \frac{\Gamma}{\eta c_\nabla}.$$

Corollary 4.32 (Stationarity for biased privatized aggregate updates)**.** *Under the assumptions of Proposition 4.19, suppose $F_{\inf} > -\infty$ and $c_\nabla > 0$. Then*

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla F(\theta_t)\|_2^2 \leq \frac{F(\theta_0)-F_{\inf}}{\eta c_\nabla T} + \frac{\Gamma_B}{\eta c_\nabla}.$$

Remark 4.33 (Interpretation). Theorem 4.31 does not require strong convexity or the PL condition. The first term on the right-hand side decays at the usual $O(1/T)$ rate for fixed stepsize, while the second term is the limiting stationarity floor induced by clipping, privacy noise, and the coupling between the current privatized aggregate G_t and the same-round preconditioner H_t. When the clipping bias and privacy noise are small and the preconditioner is stable, this neighborhood is correspondingly small. Fully general global convergence for deep non-convex objectives is not claimed.

4.10 Privacy Analysis

We now analyze the differential privacy guarantees of DP-FedSOFIM. The key observation is that DP-FedSOFIM introduces no additional client-side data access beyond the privatized gradient release mechanism already used in standard differentially private federated learning. All additional computations occur at the server and depend only on already privatized quantities. Accordingly, the privacy guarantees of DP-FedSOFIM follow from two standard principles of differential privacy: post-processing invariance and sequential composition.

Round-wise privatized release. Fix a communication round t. Let $\mathcal{M}_t(\mathcal{D})$ denote the randomized mechanism that maps the federated dataset $\mathcal{D} = \mathcal{D}_1, \ldots, \mathcal{D}_n)$to the set of privatized client updates released to the server at round t. In the present algorithm this mechanism produces the vector $\mathcal{M}_t(\mathcal{D}) = \big(g_{1,t}, \ldots, g_{n,t}\big)$, where each $g_{i,t}$ is the privatized update defined in (2). The exact differential privacy parameters of this mechanism depend on the clipping rule, noise scale, dataset normalization, participation pattern, and the privacy accountant used across rounds. Rather than fixing a specific accountant in the theoretical analysis, we assume that the round-wise release mechanism satisfies a valid differential privacy guarantee.

Assumption 4.34 (Per-round privacy of the released gradients)**.** For each round t, the release mechanism $\mathcal{M}_t$ satisfies $(\varepsilon_t, \delta_t)$-differential privacy with respect to the federated record-level adjacency defined in Definition 3.2.

Privacy of the aggregated gradient. The server aggregates the client updates as $G_t = n^{-1} \sum_{i=1}^{n} g_{i,t}$.

Lemma 4.35 (Privacy of the aggregated gradient)**.** *Under Assumption 4.34, the aggregated gradient G_t is $(\varepsilon_t, \delta_t)$-differentially private.*

Server-side privacy preservation. DP-FedSOFIM differs from the underlying first-order private federated optimization method only through additional server-side computations based on the privatized aggregated gradients.

Specifically, the server performs the updates

$$M_t = \beta M_{t-1} + (1-\beta) G_t, \qquad \widehat{\mathcal{I}}_t = \rho I_d + M_t M_t^\top, \qquad H_t = \widehat{\mathcal{I}}_t^{-1}, \qquad \theta_{t+1} = \theta_t - \eta_t H_t G_t.$$

These quantities depend only on G_t and the previous server state.

Lemma 4.36 (Server-side post-processing)**.** *Fix a communication round t. If the aggregated gradient G_t is $(\varepsilon_t, \delta_t)$-differentially private conditional on the previous server state, then the full server output of round t, namely (M_t, H_t, θ_{t+1}), is also $(\varepsilon_t, \delta_t)$-differentially private conditional on the same previous server state.*

Adaptive composition across rounds. Because the model parameters released at round t influence the client computations performed at round $t+1$, the overall training procedure is adaptive across rounds. Differential privacy nevertheless remains valid under adaptive sequential composition.

Theorem 4.37 (End-to-end privacy of DP-FedSOFIM)**.** *Suppose Assumption 4.34 holds for each round $t = 0, \ldots, T-1$. Then the full T-round DP-FedSOFIM algorithm satisfies*

$$\left(\sum_{t=0}^{T-1} \varepsilon_t, \sum_{t=0}^{T-1} \delta_t \right) \text{-DP}.$$

In particular, if the same per-round privacy parameters $(\varepsilon_0, \delta_0)$ are used at every round, then the overall procedure satisfies $(T\varepsilon_0, T\delta_0)$-DP.

Proof. By Lemma 4.35, the aggregated gradient G_t is $(\varepsilon_t, \delta_t)$-differentially private. Lemma 4.36 then implies that the complete server output of round t inherits the same privacy guarantee.

Applying adaptive sequential composition across $t = 0, \ldots, T-1$ yields the stated bound.

□

Note that the above bound is only a simple-composition baseline. The actual privacy guarantee of DP-FedSOFIM is accountant-agnostic: whatever final $\varepsilon, \delta)$ guarantee is certified for the underlying privatized gradient-release sequence is inherited unchanged by DP-FedSOFIM, because all SOFIM computations are post-processings of already privatized aggregate gradients.

Theorem 4.38 (Privacy under an arbitrary valid accountant)**.** *Fix the clipping rule, noise multiplier, participation mechanism, number of rounds, adjacency relation, and privacy accountant used for the underlying privatized gradient-release mechanism. Suppose this accountant certifies that the sequence of privatized releases $(G_0, \ldots, G_{T-1})$ satisfies (ε, δ)-differential privacy. Then the full DP-FedSOFIM transcript $(\theta_1, M_0, H_0, \ldots, \theta_T, M_{T-1}, H_{T-1})$ also satisfies (ε, δ)-differential privacy.*

Proof. The DP-FedSOFIM transcript is obtained by applying deterministic server-side maps to the privatized release sequence $(G_0, \ldots, G_{T-1})$. These maps include the momentum recursion, the rank-one regularized preconditioner, and the parameter update. Therefore, by post-processing invariance of differential privacy, the transcript has the same privacy guarantee certified for the underlying release sequence. □

Proposition 4.39 (Privacy preservation for general privatized local updates)**.** *Let $\widetilde{\Delta}_t$ be any aggregate client update released to the server at round t. Suppose $\widetilde{\Delta}_t$ is $(\varepsilon_t, \delta_t)$-differentially private conditional on the previous server state. Define*

$$M_t = \beta M_{t-1} + (1-\beta)\widetilde{\Delta}_t, \qquad H_t = (\rho I_d + M_t M_t^\top)^{-1}, \qquad \theta_{t+1} = \theta_t - \eta_t H_t \widetilde{\Delta}_t.$$

Then (M_t, H_t, θ_{t+1}) is also $(\varepsilon_t, \delta_t)$-differentially private conditional on the previous server state.

Proof. Condition on the previous server state (θ_t, M_{t-1}). Then M_t, H_t, and θ_{t+1} are deterministic functions of the privatized aggregate update $\widetilde{\Delta}_t$. Therefore, by the post-processing property of differential privacy, the tuple (M_t, H_t, θ_{t+1}) has the same round-wise privacy guarantee as $\widetilde{\Delta}_t$. □

Instantiating the privacy parameters. Appendix F derives the concrete privacy parameters for the Gaussian release used in (2) under replace-one record adjacency. That derivation computes the sensitivity of the clipped client update and applies the chosen privacy accountant to obtain the corresponding $(\varepsilon_t, \delta_t)$ guarantees. Once these parameters are instantiated for the client-side release mechanism, Theorem 4.37 implies that the SOFIM momentum and preconditioning steps incur no additional privacy loss.

Proposition 4.40 (Privacy-cost equivalence with DP-FedGD)**.** *Fix a clipping rule, clipping threshold C_g, noise multiplier σ_g, client participation mechanism, number of communication rounds T, and privacy accountant. Suppose DP-FedGD with these choices satisfies (ε, δ) differential privacy. Then DP-FedSOFIM with the same choices also satisfies (ε, δ) differential privacy.*

Proof. Under the stated choices, DP-FedGD and DP-FedSOFIM use the same client-side private release mechanism. In each communication round, the data-dependent quantity released to the server is the same clipped and privatized aggregate gradient G_t.

DP-FedSOFIM differs from DP-FedGD only through the server-side computations

$$M_t = \beta M_{t-1} + (1-\beta) G_t, \qquad \widehat{I}_t = \rho I_d + M_t M_t^\top, \qquad H_t = \widehat{I}_t^{-1},$$

and

$$\theta_{t+1} = \theta_t - \eta_t H_t G_t.$$

Conditional on the previous server state, these quantities are deterministic functions of the already privatized aggregate G_t. Therefore, by the post-processing property of differential privacy, they introduce no additional privacy loss in round t.

Since the sequence of private releases across the T rounds is the same as in DP-FedGD, the adaptive composition analysis under the chosen privacy accountant is also identical. Hence DP-FedSOFIM satisfies the same final (ε, δ) guarantee as DP-FedGD. □

Table 1 compares the information released by representative DP-FL methods. Standard DP-FL algorithms privatize clipped gradients or client updates and track the cumulative privacy loss through composition/accounting methods (Dwork et al., 2014; Abadi et al., 2016; McMahan et al., 2017b; Mironov, 2017). Methods that release additional client-side quantities, such as control variates or curvature/covariance statistics, require privacy accounting appropriate to those releases (Karimireddy et al., 2020). DP-FedSOFIM differs in that all curvature operations are performed at the server after the aggregate gradient has already been privatized. Therefore, the SOFIM update is a deterministic post-processing of a DP quantity and incurs no additional privacy cost (Dwork et al., 2014).

Table 1: Privacy and computational comparison. DP-FedSOFIM has the same privacy cost as DP-FedGD because all SOFIM operations are post-processings of already privatized aggregate gradients.

Method	Additional information beyond noisy aggregate	Privacy cost	Client memory
DP-FedGD	None	(ε, δ)	$O(d)$
DP-FedAvg	None, if only privatized client updates are released	Accountant-dependent	$O(d)$
DP-SCAFFOLD	Control-variate state, depending on implementation	Accountant-dependent	$O(d)$
DP-FedFC / DP-FedNew-type	Client-side curvature or covariance statistic	Requires accounting if privately released	$O(d^2)$
DP-FedSOFIM	None; SOFIM is server-side post-processing	Same as DP-FedGD	$O(d)$

Remark 4.41 (Meaning of "no additional privacy cost"). Proposition 4.40 says that DP-FedSOFIM and DP-FedGD have identical privacy cost when the private release mechanism and privacy accountant are fixed. The advantage of DP-FedSOFIM is that curvature-aware preconditioning is added after privatization, and hence does not consume additional privacy budget.

4.11 Comparison with Existing DP-FL Theory

The convergence results above should be interpreted as structural guarantees for privacy-compatible server-side curvature adaptation. We do not claim that DP-FedSOFIM improves the minimax convergence rate of DP-FL in all regimes. Rather, the theoretical advantage is that curvature information is introduced without changing the private release mechanism.

Compared with DP-FedGD, DP-FedSOFIM uses the same clipped and privatized aggregate gradient, and therefore has the same privacy cost under the same accountant. The difference lies only in the server-side update: DP-FedGD applies an isotropic first-order step, while DP-FedSOFIM applies the preconditioned step

$$\theta_{t+1} = \theta_t - \eta_t H_t G_t, \qquad H_t = (\rho I_d + M_t M_t^\top)^{-1}.$$

Thus the privacy noise source is the same as in DP-FedGD, but the update is rescaled along the curvature direction estimated from the privatized aggregate trajectory.

Compared with DP-FedFC and DP-FedNew-type methods, DP-FedSOFIM avoids releasing client-side covariance, Hessian, or Fisher-type statistics. This distinction is important because such second-order statistics generally require $O(d^2)$ client-side memory and communication, and must be accounted for if they are released privately. DP-FedSOFIM instead constructs its rank-one Fisher proxy entirely at the server from already privatized aggregate gradients.

Compared with DP-SCAFFOLD, DP-FedSOFIM does not rely on maintaining client-level control variates. In high privacy-noise regimes, control-variate estimates may themselves become noisy or require additional accounting depending on the implementation. DP-FedSOFIM uses only the server-side momentum buffer $M_t = \beta M_{t-1} + (1-\beta)G_t$, which is a deterministic function of privatized aggregates. Therefore, the main theoretical benefit of DP-FedSOFIM is the combination of same privacy cost as DP-FedGD, $O(d)$ client-side complexity, and server-side curvature-aware preconditioning. The empirical section then evaluates whether this structural advantage translates into improved optimization and predictive performance under fixed privacy budgets.

5 Experiments

5.1 Experimental Setup

Datasets and Models We evaluate DP-FedSOFIM on two datasets. **CIFAR-10** consists of 50,000 training images and 10,000 test images across 10 classes. **PathMNIST** is a medical imaging dataset from the MedMNIST benchmark (Yang et al., 2023), consisting of colorectal cancer histology patches across 9 tissue classes. Following the transfer learning protocol for differentially private learning (Tramer & Boneh, 2020; Krouka et al., 2025), we employ a ResNet-20 backbone pre-trained on CIFAR-100 as a frozen feature extractor, optimizing only the final linear classification head on both datasets. This yields a low-dimensional parameter space well-suited to second-order optimization while preserving the essential challenges of private federated learning.

Federated Setting We simulate a federated environment with $n = 20$ clients with full client participation in every round. To assess robustness to data heterogeneity, we evaluate each method under two data partitioning schemes: **IID**, where training data is partitioned uniformly at random across clients, and **non-IID**, where data is partitioned using a Dirichlet process with concentration parameter $\alpha = 0.5$, inducing moderate label heterogeneity across clients. The same set of hyperparameters is used for both partitioning schemes. Each client computes gradients over its entire local dataset (full-batch), following the full-gradient protocol of Krouka et al. (2025). Training proceeds for $T = 70$ federated rounds with a single local iteration per round (except DP-FedAvg and DP-SCAFFOLD, where local iterations are tuned as a hyperparameter), representing a communication-constrained scenario typical of cross-device federated learning.

Privacy Parameters We evaluate across four privacy regimes:

$$\varepsilon \in \{0.5,\ 1,\ 5,\ 10\} \quad \text{with} \quad \delta = 10^{-5},$$

spanning from stringent ($\varepsilon = 0.5$) to moderate ($\varepsilon = 10$) privacy guarantees. The noise multiplier σ_g for each ε is calibrated using the closed-form hockey-stick divergence formula for Gaussian mechanisms (Balle & Wang, 2018), with tight T-fold composition bounds following Zhu et al. (2022), to achieve the target (ε, δ)-DP budget after $T = 70$ rounds. Under full client participation the per-round Gaussian noise scales as $\sigma_g = \Theta(\varepsilon^{-1}\sqrt{T})$, making tight privacy budgets ($\varepsilon \leq 1$) significantly more challenging for convergence.

Baselines We compare DP-FedSOFIM against six baselines: **DP-FedGD** (McMahan et al., 2017b; Krouka et al., 2025), the standard first-order baseline applying per-example gradient clipping followed by Gaussian noise addition with a single local step; **DP-FedAvg** (McMahan et al., 2017b; Abadi et al., 2016), which extends DP-FedGD to multiple local SGD steps before aggregation; **DP-FedFC** (Krouka et al., 2025), a second-order method that computes local feature covariance matrices as an approximation to the Fisher information matrix, incurring $O(d^2)$ client-side memory; **DP-SCAFFOLD** (Karimireddy et al., 2020), a variance-reduction method using control variates to correct for client drift; **DP-FedAdam** (Reddi et al., 2020), which applies an Adam-style adaptive server optimizer atop per-client DP gradient aggregation; and **DP-FedYogi** (Reddi et al., 2020), which replaces the Adam server step with a Yogi update, improving stability under the large gradient variance induced by DP noise. All baselines are calibrated to the same (ε, δ) privacy budget as DP-FedSOFIM using identical privacy accounting.

Hyperparameters DP-FedSOFIM tunes regularization parameter ρ, EMA momentum β, and learning rate η per privacy regime, with bias correction enabled for $\varepsilon \leq 1$ where noise-induced curvature bias is most pronounced.

Hyperparameters are determined via a two-stage grid search. A coarse search over a wide logarithmically-spaced grid identifies the promising region; a subsequent fine search with finer resolution around the best coarse result yields the final configuration. Both stages use a fixed seed, $T = 50$ rounds, and are conducted on a held-out validation split to avoid overfitting to test performance. The configuration yielding the best final validation accuracy is selected independently for each method, dataset, and privacy regime, and the same hyperparameters are used for both IID and non-IID partitions. Search grids are detailed in Table 2.

For the tightest privacy regime ($\varepsilon = 0.5$), we additionally employ a *warmup* strategy for DP-FedSOFIM: the Sherman–Morrison preconditioning step is disabled for the first 20 rounds, during which the server uses

EMA-only updates. This delayed activation prevents the curvature estimate from being corrupted by the large Gaussian noise early in training, after which the signal-to-noise ratio is sufficient for reliable second-order information to accumulate. The warmup duration (20 rounds) is treated as a fixed implementation choice rather than a tuned hyperparameter, and is applied consistently across both datasets.

Table 2: Hyperparameter search grids for all methods, determined via a two-stage coarse-to-fine grid search. Each stage is run for $T = 50$ rounds with a fixed seed; the configuration achieving the best final validation accuracy is carried forward. Parameters fixed across all methods: $C_g = 10$, $T = 70$ rounds, $\delta = 10^{-5}$.

Method	Hyperparameter	Search Grid (Coarse → Fine)
DP-FedGD	Learning rate η	$\{10^{-4}, 10^{-3}, 0.01, 0.1, 1, 5, 10\} \rightarrow \{0.03, 0.05, 0.1, 0.3, 0.5\}$
DP-FedAvg	Learning rate η	$\{10^{-4}, 10^{-3}, 0.01, 0.1, 1, 5\} \rightarrow \{0.005, 0.01, 0.05, 0.1, 0.5, 1\}$
	Local iterations	$\{1, 3, 5, 10, 20\} \rightarrow \{1, 2, 5, 7, 10\}$
DP-FedFC	Learning rate η	$\{10^{-4}, 10^{-3}, 0.01, 0.1, 1, 5\} \rightarrow \{0.005, 0.01, 0.05, 0.1, 0.5\}$
	Curvature scale γ	$\{0.01, 0.1, 1, 10, 100\} \rightarrow \{0.05, 0.1, 0.5, 1, 5\}$
DP-SCAFFOLD	Server learning rate η_s	$\{0.01, 0.1, 0.5, 1, 5\} \rightarrow \{0.05, 0.1, 0.3, 0.5, 1\}$
	Client learning rate η_c	$\{0.001, 0.01, 0.1, 0.5\} \rightarrow \{0.005, 0.01, 0.05, 0.1\}$
	Local steps	$\{1, 5, 10\} \rightarrow \{1, 3, 5, 7, 10\}$
DP-FedAdam	Learning rate η	$\{10^{-4}, 10^{-3}, 0.01, 0.1, 1\} \rightarrow \{3{\times}10^{-4}, 10^{-3}, 3{\times}10^{-3}, 0.01, 0.03, 0.1\}$
	(β_1, β_2)	$(0, 0.9, 0.99){\times}(0.99, 0.999) \rightarrow (0.9, 0.95, 0.99){\times}(0.99, 0.999, 0.9999)$
	Server ϵ	$\{10^{-8}, 10^{-4}, 10^{-2}, 0.1\} \rightarrow \{10^{-6}, 10^{-4}, 10^{-3}, 0.01\}$
DP-FedYogi	Learning rate η	$\{10^{-4}, 10^{-3}, 0.01, 0.1, 1\} \rightarrow \{3{\times}10^{-4}, 10^{-3}, 3{\times}10^{-3}, 0.01, 0.03, 0.1\}$
	(β_1, β_2)	$(0, 0.9, 0.99){\times}(0.99, 0.999) \rightarrow (0.9, 0.95, 0.99){\times}(0.99, 0.999, 0.9999)$
	Server ϵ	$\{10^{-4}, 10^{-3}, 0.01, 0.1\} \rightarrow \{10^{-3}, 5{\times}10^{-3}, 0.01, 0.05\}$
DP-FedSOFIM	Learning rate η	$\{10^{-3}, 0.01, 0.1, 1, 5\} \rightarrow \{0.05, 0.1, 0.2, 0.5\}$
	Regularization ρ	$\{0.01, 0.1, 1, 5, 10\} \rightarrow \{0.1, 0.3, 0.5, 1, 2\}$
	EMA momentum β	$\{0.8, 0.9, 0.99\} \rightarrow \{0.85, 0.9, 0.93, 0.95\}$

5.2 Results

Tables 3–4 and Figures 1–2 present test accuracy trajectories across all privacy regimes for CIFAR-10 and PathMNIST with varying client counts. We organize our analysis around five key phenomena observed in the results.

Table 3: Test accuracy (%) on CIFAR-10 across federated rounds for different privacy budgets. Results shown at 10-round intervals (mean ± std over 3 seeds). Best result per privacy regime is in **bold**.

Method	Federated Round						
	10	20	30	40	50	60	70
$\varepsilon = 0.5$							
DP-FedGD	36.96±2.85	47.01±1.28	52.21±1.41	55.29±0.73	57.17±0.24	58.00±0.36	**59.36±0.67**
DP-FedAvg	36.96±2.85	47.01±1.27	52.21±1.43	55.29±0.72	57.18±0.23	58.00±0.34	59.34±0.66
DP-FedFC	23.37±5.03	34.86±3.10	43.64±2.18	48.90±0.57	52.33±1.21	53.92±0.72	55.33±1.28
DP-SCAFFOLD	30.45±2.81	38.29±1.24	43.49±1.90	45.52±0.42	47.76±1.03	48.31±1.18	49.78±1.53
DP-FedAdam	34.22±3.59	44.62±1.94	50.75±1.34	54.43±0.74	56.76±0.24	57.74±0.18	59.23±0.53
DP-FedYogi	29.77±3.43	41.35±3.01	47.80±1.20	52.69±0.85	55.38±0.40	56.90±0.27	58.60±0.24
DP-FedSOFIM	**39.60±1.61**	**47.53±1.24**	**52.53±0.89**	**55.55±0.88**	**56.93±0.43**	**57.58±0.94**	57.96±1.09
$\varepsilon = 1$							
DP-FedGD	39.53±2.50	50.26±0.69	55.47±0.56	58.55±0.68	60.68±0.28	61.82±0.42	62.75±0.26

Continued on next page

Continued from previous page

Method	Federated Round						
	10	20	30	40	50	60	70
DP-FedAvg	39.53±2.51	50.25±0.69	55.46±0.56	58.54±0.68	60.68±0.27	61.84±0.41	62.74±0.26
DP-FedFC	51.20±1.10	57.04±0.33	59.44±0.48	60.47±0.34	61.60±0.96	61.88±0.74	61.57±0.72
DP-SCAFFOLD	41.99±1.27	49.58±0.92	53.77±0.16	54.20±0.97	54.96±1.54	56.03±1.00	56.03±0.72
DP-FedAdam	49.15±2.01	58.43±0.29	60.55±0.71	61.88±0.48	62.45±0.63	62.54±0.57	62.49±0.74
DP-FedYogi	40.18±4.62	53.93±0.66	57.00±0.81	58.00±0.72	58.15±1.06	58.37±1.44	57.45±1.38
DP-FedSOFIM	**51.25±0.59**	**58.56±0.29**	**61.35±0.39**	**62.37±0.42**	**62.55±0.51**	**63.14±0.77**	**62.76±0.63**
$\varepsilon = 5$							
DP-FedGD	40.74±2.17	51.80±0.38	56.85±0.09	59.98±0.69	61.85±0.45	63.21±0.62	64.10±0.58
DP-FedAvg	40.75±2.16	51.79±0.38	56.85±0.09	59.97±0.68	61.85±0.45	63.20±0.61	64.09±0.58
DP-FedFC	46.56±3.37	49.58±0.98	55.42±1.17	58.43±0.77	60.83±0.52	62.40±0.47	63.51±0.71
DP-SCAFFOLD	55.66±0.36	62.19±0.74	64.29±0.62	65.18±0.16	65.67±0.22	65.96±0.24	66.42±0.10
DP-FedAdam	43.92±1.63	59.76±1.34	63.64±1.06	66.09±0.52	67.11±0.06	67.15±0.20	67.28±0.20
DP-FedYogi	55.37±0.43	63.72±0.40	66.06±0.33	67.06±0.13	67.35±0.33	**67.69±0.23**	**67.81±0.20**
DP-FedSOFIM	**61.07±0.94**	**65.63±0.39**	**67.09±0.15**	**67.31±0.16**	**67.40±0.41**	67.54±0.26	67.44±0.31
$\varepsilon = 10$							
DP-FedGD	40.79±2.13	51.86±0.42	56.94±0.10	59.97±0.59	61.94±0.49	63.17±0.56	64.10±0.52
DP-FedAvg	40.79±2.14	51.84±0.41	56.94±0.10	59.97±0.59	61.92±0.49	63.18±0.57	64.10±0.52
DP-FedFC	40.74±2.17	51.67±0.45	57.00±0.22	60.03±0.56	61.88±0.56	63.14±0.56	64.10±0.51
DP-SCAFFOLD	61.36±0.29	65.24±0.46	66.29±0.33	67.00±0.13	67.22±0.16	67.60±0.19	67.85±0.17
DP-FedAdam	43.51±1.73	57.62±1.46	63.46±0.45	65.02±0.41	66.91±0.43	67.51±0.29	67.77±0.27
DP-FedYogi	44.04±1.32	58.01±1.48	62.90±0.34	64.78±0.45	66.60±0.33	67.34±0.15	67.72±0.18
DP-FedSOFIM	**61.56±0.96**	**66.18±0.40**	**67.57±0.29**	**68.01±0.27**	**68.20±0.20**	**68.59±0.12**	**68.65±0.12**

Table 4: Test accuracy (%) on PathMNIST across federated rounds for different privacy budgets. Results shown at 10-round intervals (mean ± std over 3 seeds). Best result per privacy regime is in **bold**.

Method	Federated Round						
	10	20	30	40	50	60	70
$\varepsilon = 0.5$							
DP-FedGD	50.79±0.50	57.48±1.96	59.94±1.93	62.35±0.68	63.70±0.36	63.79±0.69	64.50±0.73
DP-FedAvg	50.77±0.51	57.47±1.97	59.93±1.90	62.34±0.67	63.70±0.34	63.78±0.67	64.50±0.73
DP-FedFC	55.15±1.66	58.89±1.24	62.67±0.54	64.16±0.75	64.20±1.16	64.66±1.33	64.43±2.12
DP-SCAFFOLD	49.36±1.03	55.01±2.78	56.42±4.00	59.33±1.50	60.55±1.30	59.94±1.99	61.05±1.82
DP-FedAdam	53.82±1.42	60.48±1.21	62.81±1.21	64.46±1.29	**65.23±0.97**	**65.27±1.17**	65.48±1.31
DP-FedYogi	49.39±3.52	58.08±2.31	61.95±1.16	63.52±0.93	64.66±1.06	65.26±1.12	65.41±1.34
DP-FedSOFIM	**56.13±2.01**	**60.67±2.09**	**63.66±1.02**	**64.24±1.31**	65.16±1.59	64.87±1.76	**66.26±1.36**
$\varepsilon = 1$							
DP-FedGD	51.26±1.49	58.18±1.40	61.22±0.87	63.31±0.43	64.59±0.12	65.10±0.29	65.68±0.18
DP-FedAvg	51.25±1.48	58.18±1.41	61.24±0.87	63.31±0.41	64.58±0.13	65.10±0.29	65.67±0.19
DP-FedFC	55.10±1.69	63.03±1.51	64.73±1.66	66.20±1.25	65.26±0.83	65.42±0.50	65.65±1.42
DP-SCAFFOLD	55.32±2.35	58.46±1.58	62.48±1.19	63.37±2.54	64.92±0.98	64.77±1.22	64.97±1.12
DP-FedAdam	**59.93±0.57**	**65.29±0.32**	**66.84±0.61**	**67.65±1.06**	**68.21±0.70**	**67.95±0.68**	68.12±0.90
DP-FedYogi	49.70±3.92	60.56±2.89	62.60±2.71	64.61±1.10	65.45±0.50	65.09±0.81	65.04±1.77
DP-FedSOFIM	58.49±3.13	64.67±0.97	66.98±1.29	67.32±1.18	67.47±0.94	67.88±0.95	**68.88±0.62**
$\varepsilon = 5$							

Continued on next page

Continued from previous page

Method	Federated Round						
	10	20	30	40	50	60	70
DP-FedGD	51.51±2.57	58.53±1.17	61.88±0.42	63.75±0.06	65.15±0.26	65.96±0.18	66.49±0.20
DP-FedAvg	51.50±2.58	58.52±1.16	61.86±0.42	63.77±0.05	65.16±0.27	65.94±0.19	66.50±0.18
DP-FedFC	43.48±0.53	55.09±1.29	52.10±6.59	56.43±4.22	56.48±4.71	60.87±1.72	62.04±3.69
DP-SCAFFOLD	60.73±1.13	65.79±0.22	67.64±0.20	68.60±0.45	69.20±0.26	69.43±0.30	69.77±0.37
DP-FedAdam	47.75±4.17	58.66±1.17	65.41±1.52	68.43±0.27	70.03±0.52	70.94±0.43	71.20±0.40
DP-FedYogi	59.09±0.38	66.43±1.04	68.43±0.75	69.53±0.66	70.39±0.06	70.66±0.17	71.03±0.22
DP-FedSOFIM	**64.22±2.23**	**67.34±1.66**	**68.89±0.71**	**70.30±0.51**	**70.63±0.39**	**71.10±0.31**	**71.54±0.31**
$\varepsilon = 10$							
DP-FedGD	51.49±2.64	58.51±1.23	61.86±0.44	63.89±0.10	65.17±0.22	65.98±0.28	66.62±0.23
DP-FedAvg	51.48±2.65	58.51±1.22	61.86±0.42	63.89±0.08	65.17±0.22	65.98±0.27	66.64±0.20
DP-FedFC	51.44±2.86	58.30±1.28	61.87±0.27	63.90±0.07	65.13±0.21	65.99±0.29	66.59±0.24
DP-SCAFFOLD	62.47±0.43	66.44±0.40	68.10±0.22	68.91±0.42	69.35±0.32	69.84±0.31	70.13±0.36
DP-FedAdam	56.77±4.06	63.37±3.97	66.92±0.72	67.72±0.80	70.03±0.61	70.11±0.32	70.97±0.27
DP-FedYogi	56.56±3.99	63.44±3.65	66.20±2.22	68.50±0.84	70.85±0.22	69.58±0.77	70.45±0.77
DP-FedSOFIM	**64.25±2.25**	**67.32±1.91**	**68.94±0.82**	**70.45±0.32**	**70.92±0.39**	**71.50±0.31**	**71.78±0.28**

5.3 Analysis

We analyze the round-by-round test accuracy in Tables 3 and 4 ($n = 20$ clients, non-IID Dirichlet $\alpha = 0.5$; IID results in Appendix G), organized around five phenomena.

5.3.1 Phenomenon 1: DP-FedSOFIM Converges Substantially Faster Than All Baselines

The defining advantage of DP-FedSOFIM is convergence speed, not asymptotic accuracy. It attains the best accuracy at round 10 in seven of the eight dataset/privacy regimes (the only exception being PathMNIST at $\varepsilon = 1$, where DP-FedAdam leads early), with round-10 leads over the standard first-order baseline DP-FedGD as large as +20.33% (CIFAR-10, $\varepsilon = 5$) and +20.77% (CIFAR-10, $\varepsilon = 10$). These early gaps far exceed the corresponding final-round gaps (+3.34% to +5.16% vs. DP-FedGD), indicating that the curvature-aware preconditioner front-loads its gains.

This translates directly into communication savings. Defining the target as 95% of DP-FedGD's final accuracy, on CIFAR-10 at $\varepsilon = 10$: **(i)** DP-FedGD reaches the target (60.90%, i.e. 95% of 64.10%) only at round 50; **(ii)** DP-FedSOFIM reaches it at round 10 (61.56%); and **(iii)** DP-FedSOFIM surpasses DP-FedGD's *entire-run* final accuracy by round 20 (66.18%). The same pattern holds on PathMNIST at $\varepsilon = 5$, where DP-FedSOFIM hits the 95% target at round 10 versus round 40 for DP-FedGD. A $\sim$5$\times$ reduction in rounds-to-target is decisive in cross-device settings, where each round incurs communication, latency, and on-device energy costs.

On final accuracy, DP-FedSOFIM remains competitive throughout: it attains the best round-70 accuracy in six of eight regimes (all four on PathMNIST, and CIFAR-10 at $\varepsilon \in \{1, 10\}$). In the two regimes where it is edged out at the final round (CIFAR-10 $\varepsilon = 0.5$ and $\varepsilon = 5$), it had led for the majority of training before being overtaken late, a dynamic we examine in Phenomenon 2. Final accuracy is thus not the differentiator between methods; the rate of convergence is.

The mechanism is the rapid stabilization of the EMA momentum buffer. Initialized at $M_{-1} = 0_d$, the curvature proxy begins purely isotropic ($\widehat{\mathcal{I}}_{-1} = \rho I_d$, $H_{-1} = \rho^{-1} I_d$) and accumulates gradient signal as $M_t = (1-\beta)\sum_{s \le t} \beta^{t-s} G_s$. The EMA suppresses noise variance by a factor $\frac{1-\beta}{1+\beta}$, and because we tune β upward as privacy tightens ($\beta = 0.9$ at $\varepsilon \ge 2$ rising to $\beta = 0.95$ at $\varepsilon = 0.5$), the suppression strengthens from $\sim$1/19 to $\sim$1/39 exactly where per-round noise is largest. Consequently, DP-FedSOFIM exhibits *no* early-round instability even at $\varepsilon = 0.5$: at this regime we employ a 20-round warmup in which the server applies EMA-only updates before activating the Sherman–Morrison step, ensuring the momentum buffer

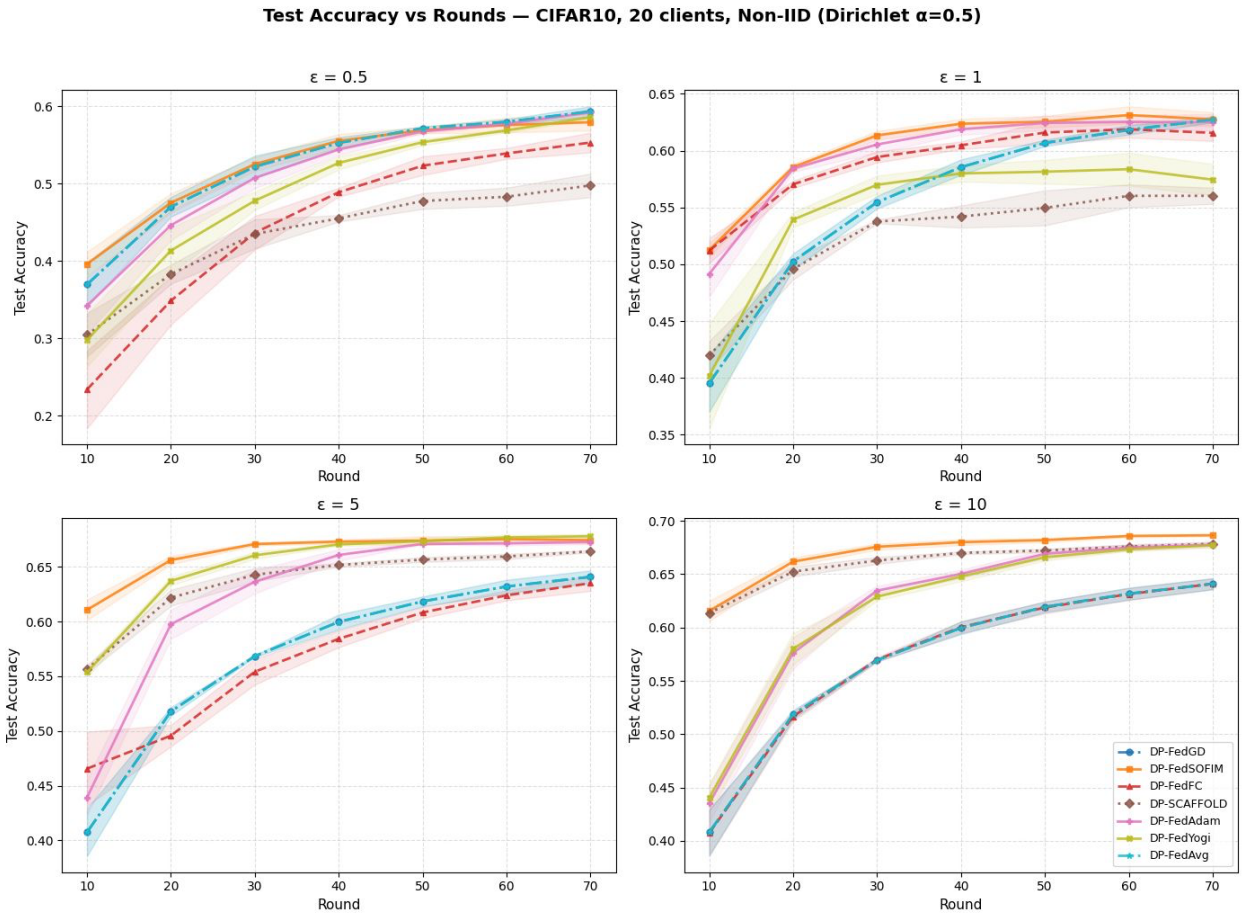

Figure 1: Convergence trajectories on CIFAR-10, 20 clients, Non-IID (Dirichlet $\alpha = 0.5$) across privacy regimes. DP-FedSOFIM consistently leads or matches the best baseline across all ε.

has accumulated sufficient signal before curvature preconditioning begins. The buffer stabilizes within the first 10–20 rounds and the preconditioner aligns with the dominant gradient directions almost immediately thereafter.

5.3.2 Phenomenon 2: Early Convergence to a Strong Solution Across Landscapes

DP-FedSOFIM front-loads its gains, reaching near-final accuracy within the first 20–30 rounds, and the subsequent trajectory differs by dataset. On PathMNIST the round-10 lead consolidates into a final-round win across all four privacy budgets. On CIFAR-10, DP-FedSOFIM likewise reaches a strong solution early and holds it, while the first-order and adaptive baselines continue to make incremental late-round progress on this dataset and converge to a comparable accuracy by round 70, drawing level at $\varepsilon = 0.5$ (DP-FedGD 59.36% vs. 57.96%) and $\varepsilon = 5$ (DP-FedYogi 67.81% vs. 67.44%). The effect is visible in the trajectory: at $\varepsilon = 5$ on CIFAR-10, DP-FedSOFIM is already at 67.09% by round 30 and ends at 67.44%, having attained essentially its final performance less than halfway through training, whereas DP-FedYogi requires the full budget to reach the same level.

This behavior is consistent with DP-FedSOFIM arriving at a strong solution in a fraction of the communication rounds the baselines require. The dataset-dependent shape of the late-round trajectory follows from the structure of the effective loss landscape, for which we give a geometric account in Phenomenon 5.

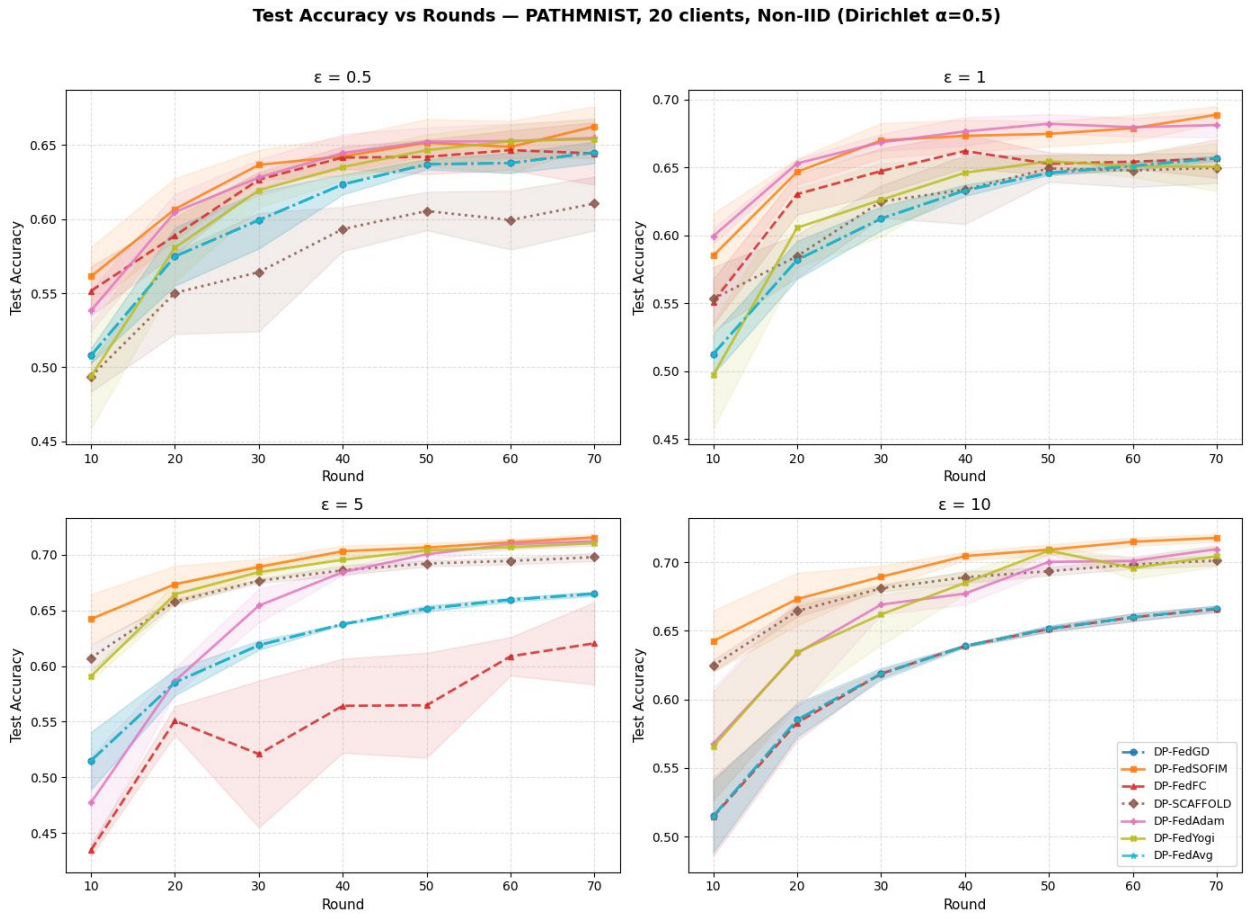

Figure 2: Convergence trajectories on PathMNIST, 20 clients, Non-IID (Dirichlet $\alpha = 0.5$) across privacy regimes. DP-FedSOFIM demonstrates substantial improvements over all baselines, with early-round dominance at $\varepsilon \geq 5$.

5.3.3 Phenomenon 3: DP-FedAdam Is the Strongest Baseline but Converges Slowly

DP-FedAdam emerges as the most competitive baseline overall, particularly at relaxed privacy: it is the second-best method at the final round on PathMNIST $\varepsilon \in \{1, 5, 10\}$ (68.12%, 71.20%, 70.97%) and is within 0.2% of DP-FedSOFIM at the final round on CIFAR-10 $\varepsilon = 5$. Its adaptive server-side second-moment scaling thus rivals explicit Fisher preconditioning once enough rounds have elapsed.

However, DP-FedAdam pays a steep early-round penalty. Its adaptive moments require several rounds of accumulation before the server step is well-scaled, producing slow, high-variance starts: at round 10 it trails DP-FedSOFIM by +17.15% (CIFAR-10 $\varepsilon = 5$), +18.05% (CIFAR-10 $\varepsilon = 10$), and +16.47% (PathMNIST $\varepsilon = 5$), with round-10 standard deviations as large as $\pm 4.17\%$. The single regime where DP-FedAdam leads early (PathMNIST $\varepsilon = 1$, round 10: 59.93% vs. 58.49%) is also one where DP-FedSOFIM overtakes it by the final round (68.88% vs. 68.12%). The distinction is therefore not asymptotic quality but *when* useful curvature information becomes available: DP-FedSOFIM's explicit proxy is informative from the first rounds, whereas DP-FedAdam must warm up its moment estimates, making it ill-suited to round-limited deployments.

5.3.4 Phenomenon 4: DP-SCAFFOLD's Sensitivity to Privacy Noise

DP-SCAFFOLD exhibits a distinct failure mode under tight privacy. On CIFAR-10 at $\varepsilon = 0.5$ it reaches only 49.78% final accuracy, the worst of all methods and far below DP-FedGD (59.36%) and DP-FedSOFIM (57.96%); at $\varepsilon = 1$ it again trails badly (56.03% vs. 62.76% for DP-FedSOFIM). The same pattern holds

on PathMNIST, where it is the weakest method at $\varepsilon \in \{0.5, 1\}$ (61.05%, 64.97%). This degradation arises because SCAFFOLD's variance reduction relies on control variates maintained across rounds; under high privacy noise these control variates become corrupted, causing the client-drift correction to amplify rather than reduce variance.

DP-SCAFFOLD recovers competitiveness once noise is low enough to preserve the control variates: at $\varepsilon \in \{5, 10\}$ it climbs to 66.42%/67.85% on CIFAR-10 and 69.77%/70.13% on PathMNIST, in both cases still below DP-FedSOFIM. Its inability to maintain accurate control variates under stringent privacy makes it unsuitable for strong-privacy applications, whereas DP-FedSOFIM's server-side preconditioning degrades gracefully and retains its advantage down to $\varepsilon = 0.5$.

5.3.5 Phenomenon 5: Dataset-Dependent Curvature Sensitivity

Phenomena 1 and 2 reveal a consistent asymmetry: DP-FedSOFIM's early lead consolidates into a final-round win on PathMNIST across all privacy budgets, yet plateaus and is overtaken late on CIFAR-10 at $\varepsilon \in \{0.5, 5\}$. We interpret this geometrically, consistent with the operator and quadratic-form bounds of Lemma 4.14 and the bias-and-noise neighborhood guarantee of Theorem 4.21.

The benefit of Fisher-proxy preconditioning scales with the condition number of the effective loss landscape: greater anisotropy implies larger gains from curvature-aware step scaling. We hypothesize that the frozen ResNet-20 representation of PathMNIST induces a *concentrated* curvature structure, where a few dominant directions carry the discriminative tissue-level signal. The momentum buffer aligns with these directions quickly, and because the remaining low-curvature directions carry little additional signal, DP-FedSOFIM's early lead is durable. On CIFAR-10 the effective curvature appears more *diffuse*: DP-FedSOFIM captures the dominant directions rapidly (the large early lead) but down-weights the many low-curvature directions that nonetheless carry slow, cumulative signal; first-order and adaptive baselines keep extracting that residual signal and close the gap in late rounds. The CIFAR-10 late plateau is thus the diagnostic of a diffuse landscape.

This yields a practical guideline: DP-FedSOFIM is most advantageous in domains whose pretrained representation induces a structured, anisotropic landscape, such as medical imaging, where curvature remains informative even under the high noise of stringent privacy budgets, and in any communication-constrained setting where rounds-to-target, not asymptotic accuracy, is the binding constraint.

6 Conclusion

In this paper, we presented DP-FedSOFIM, a novel framework for differentially private second-order federated optimization. By shifting the computational burden of Fisher Information Matrix preconditioning entirely to the server and utilizing the Sherman-Morrison formula for $O(d)$ updates, we addressed the prohibitive memory constraints of existing second-order methods such as DP-FedNew and DP-FedFC, reducing client-side complexity from $O(d^2)$ to $O(d)$. Our theoretical analysis confirms that server-side preconditioning preserves (ϵ, δ) differential privacy through the post-processing theorem, while convergence guarantees establish linear convergence to a noise-determined error floor under strongly convex objectives.

Empirically, we evaluated DP-FedSOFIM against six baselines, DP-FedGD, DP-FedAvg (whose hyperparameter search over local steps yielded one as optimal, making it numerically equivalent to DP-FedGD in this setting), DP-FedFC, DP-SCAFFOLD, DP-FedAdam, and DP-FedYogi, on CIFAR-10 and PathMNIST with $n = 20$ clients across four privacy budgets ($\epsilon \in \{0.5, 1, 5, 10\}$). The defining result is convergence speed: DP-FedSOFIM attains the best accuracy at round 10 in seven of the eight dataset/privacy regimes, reaching 95% of DP-FedGD's final accuracy in roughly a fifth of the rounds, exceeding DP-FedGD's concurrent round-20 accuracy in all eight regimes, and surpassing its entire-run final accuracy within the first 20 rounds at $\varepsilon \in \{5, 10\}$. It is also the most accurate method at the final round in six of eight regimes, with gains over DP-FedGD of up to +4.55% on CIFAR-10 ($\epsilon = 10$) and up to +5.16% on PathMNIST ($\epsilon = 10$), the larger PathMNIST margins reflecting the benefit of curvature-aware preconditioning on more ill-conditioned medical imaging tasks. In the two regimes where it is edged out at the final round (CIFAR-10 $\epsilon \in \{0.5, 5\}$), it had led for the majority of training before the baselines drew level. Our analysis identified five phenomena:

substantially faster convergence from the earliest rounds; a dataset-dependent late-round trajectory in which the early lead consolidates on PathMNIST and levels off on CIFAR-10; DP-FedAdam emerging as the strongest baseline asymptotically but slow to warm up its adaptive moments; and a characteristic failure mode of DP-SCAFFOLD under tight privacy, where corrupted control variates amplify rather than reduce variance. DP-FedSOFIM, by contrast, degrades gracefully across all privacy regimes.

These results establish DP-FedSOFIM as a scalable, communication-efficient, and privacy-preserving solution for federated learning in privacy-critical domains such as healthcare and finance, where both strong formal privacy guarantees and high model utility are essential requirements.

7 Discussion and Future Work

We have introduced DP-FedSOFIM, a server-side second-order optimization framework that bridges the gap between the communication efficiency of first-order methods and the convergence stability of natural gradient descent. By leveraging the Sherman-Morrison matrix inversion identity, we maintain $O(d)$ memory and communication complexity, effectively overcoming the $O(d^2)$ bottleneck inherent in prior second-order federated methods like DP-FedNew (Krouka et al., 2025). Our theoretical framework establishes that preconditioning acts as a post-processing step, preserving the rigorous privacy guarantees of the Gaussian mechanism while significantly reducing the convergence error floor induced by noise.

Empirically, DP-FedSOFIM exhibits strong early-round convergence: the exponential moving average in the momentum buffer suppresses the Gaussian privacy noise within the first 10–20 rounds, so the rank-one Fisher proxy yields informative curvature from the earliest rounds and the optimizer leads first-order and adaptive baselines from round 10 across nearly all privacy regimes. This stands in contrast to adaptive baselines such as DP-FedAdam, whose server-side moment estimates require several rounds to warm up before becoming effective. These observations suggest that second-order methods are not only compatible with differential privacy but are arguably necessary for maintaining utility in communication-constrained environments. Several promising directions for future research remain:

Non-Convex Generalization: While our analysis focused on the strongly convex landscape of linear heads, extending these guarantees to the non-convex landscapes of deep neural networks remains a priority. This involves analyzing the interaction between the preconditioner and the signal-to-noise ratio in deeper layers.

Structured Curvature: Investigating Kronecker-factored (K-FAC) (Martens & Grosse, 2015) or block-diagonal approximations could capture higher-order inter-layer dependencies without sacrificing the $O(d)$ server-side efficiency.

User-Level Privacy: Transitioning from record-level to user-level DP presents a different sensitivity profile. Adapting the FIM preconditioning to handle the increased noise variance required for user-level protection is a critical next step for cross-device deployments.

Adaptive Hyperparameters: Automating the selection of the regularization parameter ρ and momentum β based on the target ϵ would further reduce the need for expensive grid searches in federated settings.

Impact Statement

This work advances the field of Differentially Private Federated Learning (DP-FL), providing a scalable solution for training high-utility models on decentralized, sensitive data. The efficiency and convergence stability of DP-FedSOFIM have direct implications for high-stakes domains, such as:

- **Healthcare and Medical Imaging:** Enabling the training of diagnostic models across multiple hospitals without the need for data centralization, thereby respecting patient confidentiality and complying with regulations like GDPR or HIPAA.
- **Financial Analytics:** Facilitating collaborative fraud detection or risk assessment between institutions while protecting individual transaction records.

While our technical contribution tightens the privacy-utility tradeoff, we recognize that formal privacy guarantees are only one pillar of ethical AI. Practitioners must remain vigilant regarding algorithmic fairness, as the noise introduced by differential privacy can sometimes disproportionately affect accuracy for minority subgroups. Furthermore, the use of pre-trained feature extractors, while efficient, may inherit biases from the source data. We encourage the community to deploy DP-FedSOFIM alongside comprehensive fairness auditing and transparent data governance frameworks.

References

Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pp. 308–318, 2016.

Shun-ichi Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10(2):251–276, 1998. doi: 10.1162/089976698300017746.

Galen Andrew, Om Thakkar, Brendan McMahan, and Swaroop Ramaswamy. Differentially private learning with adaptive clipping. *Advances in neural information processing systems*, 34:17455–17466, 2021.

Borja Balle and Yu-Xiang Wang. Improving the gaussian mechanism for differential privacy: Analytical calibration and optimal denoising. In *International conference on machine learning*, pp. 394–403. PMLR, 2018.

Nicolò Dal Fabbro, Subhrakanti Dey, Michele Rossi, and Luca Schenato. SHED: A newton-type algorithm for federated learning based on incremental hessian eigenvector sharing. *Automatica*, 160:111460, 2024.

Jinshuo Dong, Aaron Roth, and Weijie J Su. Gaussian differential privacy. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 84(1):3–37, 2022.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pp. 265–284. Springer, 2006.

Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and trends® in theoretical computer science*, 9(3–4):211–407, 2014.

Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller. Inverting gradients – how easy is it to break privacy in federated learning? In *NeurIPS*, 2020.

Yonghai Gong, Yichuan Li, and Nikolaos M Freris. Fedadmm: A robust federated deep learning framework with adaptivity to system heterogeneity. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*, pp. 2575–2587. IEEE, 2022.

Simon S Haykin. *Adaptive filter theory*. Pearson Education India, 2002.

Elad Hazan, Amit Agarwal, and Satyen Kale. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69(2):169–192, 2007.

Rui Hu, Yuanxiong Guo, Hongning Li, Qingqi Pei, and Yanmin Gong. Personalized federated learning with differential privacy. *IEEE Internet of Things Journal*, 8(11):9539–9550, 2020.

Peter Kairouz and H Brendan McMahan. Advances and open problems in federated learning. *Foundations and trends in machine learning*, 14(1-2):1–210, 2021.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International conference on machine learning*, pp. 5132–5143. PMLR, 2020.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. URL https://arxiv.org/abs/1412.6980.

Mounssif Krouka, Antti Koskela, and Tejas Kulkarni. Communication efficient differentially private federated learning using second order information. *Proceedings on Privacy Enhancing Technologies*, 2025.

Guangchen Lan, Han Wang, James Anderson, Christopher Brinton, and Vaneet Aggarwal. Improved communication efficiency in federated natural policy gradient via admm-based gradient updates. *arXiv preprint arXiv:2310.19807*, 2023.

Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. *IEEE signal processing magazine*, 37(3):50–60, 2020a.

Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. In *ICLR*, 2020b.

Junxu Liu, Jian Lou, Li Xiong, Jinfei Liu, and Xiaofeng Meng. Projected federated averaging with heterogeneous differential privacy. *Proceedings of the VLDB Endowment*, 15(4):828–840, 2022.

James Martens. New insights and perspectives on the natural gradient method. *Journal of Machine Learning Research*, 21(146):1–76, 2020.

James Martens and Roger Grosse. Optimizing neural networks with kronecker-factored approximate curvature. In *International conference on machine learning*, pp. 2408–2417. PMLR, 2015.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pp. 1273–1282. PMLR, 2017a.

H Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. Learning differentially private recurrent language models. *arXiv preprint arXiv:1710.06963*, 2017b.

Ilya Mironov. Rényi differential privacy. In *2017 IEEE 30th computer security foundations symposium (CSF)*, pp. 263–275. IEEE, 2017.

Thomas Möllenhoff, Siddharth Swaroop, Finale Doshi-Velez, and Mohammad Emtiyaz Khan. Federated admm from bayesian duality. *arXiv preprint arXiv:2506.13150*, 2025.

Mohammad Naseri, Jamie Hayes, and Emiliano De Cristofaro. Local and central differential privacy for robustness and privacy in federated learning. In *Proceedings of the 29th Network and Distributed System Security Symposium (NDSS)*, 2022. doi: 10.14722/ndss.2022.23054.

Maxence Noble, Aurélien Bellet, and Aymeric Dieuleveut. Differentially private federated learning on heterogeneous data. In *International conference on artificial intelligence and statistics*, pp. 10110–10145. PMLR, 2022.

Razvan Pascanu and Yoshua Bengio. Revisiting natural gradient for deep networks, 2014. URL https://arxiv.org/abs/1301.3584.

Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečnỳ, Sanjiv Kumar, and H Brendan McMahan. Adaptive federated optimization. *arXiv preprint arXiv:2003.00295*, 2020.

Nicola Rieke, Jonny Hancox, Wenqi Li, Fausto Milletari, Holger R Roth, Shadi Albarqouni, Spyridon Bakas, Mathieu N Galtier, Bennett A Landman, Klaus Maier-Hein, et al. The future of digital health with federated learning. *NPJ digital medicine*, 3(1):119, 2020.

Mher Safaryan, Rustem Islamov, Xun Qian, and Peter Richtarik. FedNL: Making newton-type methods applicable to federated learning. In *ICML*, 2022.

Mrinmay Sen, A Kai Qin, Yen-Wei Chen, Balasubramanian Raman, et al. Sofim: Stochastic optimization using regularized fisher information matrix. In *2024 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–7. IEEE, 2024.

Micah J. Sheller, Brandon Edwards, G. Anthony Reina, Jason Martin, Sarthak Pati, Aikaterini Kotrotsou, Mikhail Milchenko, Weilin Xu, Daniel Marcus, Rivka R. Colen, and Spyridon Bakas. Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data. *Scientific Reports*, 10 (1):12598, Jul 2020. ISSN 2045-2322. doi: 10.1038/s41598-020-69250-1. URL `https://doi.org/10.1038/s41598-020-69250-1`.

Jack Sherman and Winifred J Morrison. Adjustment of an inverse matrix corresponding to a change in one element of a given matrix. *The Annals of Mathematical Statistics*, 21(1):124–127, 1950.

Florian Tramer and Dan Boneh. Differentially private learning needs better features (or much more data). *arXiv preprint arXiv:2011.11660*, 2020.

Shusen Wang, Fred Roosta, Peng Xu, and Michael W Mahoney. Giant: Globally improved approximate newton method for distributed optimization. *Advances in neural information processing systems*, 31, 2018.

Yu-Xiang Wang, Borja Balle, and Shiva Prasad Kasiviswanathan. Subsampled rényi differential privacy and analytical moments accountant. In *The 22nd international conference on artificial intelligence and statistics*, pp. 1226–1235. PMLR, 2019.

Kang Wei, Jun Li, Chuan Ma, Ming Ding, Wen Chen, Jun Wu, Meixia Tao, and H Vincent Poor. Personalized federated learning with differential privacy and convergence guarantee. *IEEE Transactions on Information Forensics and Security*, 18:4488–4503, 2023.

Max A Woodbury. *Inverting modified matrices*. Department of Statistics, Princeton University, 1950.

Jiancheng Yang, Rui Shi, Donglai Wei, Zequan Liu, Lin Zhao, Bilian Ke, Hanspeter Pfister, and Bingbing Ni. Medmnist v2-a large-scale lightweight benchmark for 2d and 3d biomedical image classification. *Scientific data*, 10(1):41, 2023.

Jinshan Zeng, Shao-Bo Lin, Yuan Yao, and Ding-Xuan Zhou. On admm in deep learning: Convergence and saturation-avoidance. *Journal of Machine Learning Research*, 22(199):1–67, 2021.

Ligeng Zhu, Zhijian Liu, and Song Han. Deep leakage from gradients. In *NeurIPS*, 2019.

Yuqing Zhu, Jinshuo Dong, and Yu-Xiang Wang. Optimal accounting of differential privacy via characteristic function. In *International Conference on Artificial Intelligence and Statistics*, pp. 4782–4817. PMLR, 2022.

Appendices

A Proofs of lemmas and main results

In this section, we provide the proofs of the lemmas and the main results.

Proof of Lemma 4.10. From (5),

$$\nu_t^2 = \frac{(C_g\sigma_g)^2}{n^3}\sum_{i=1}^{n}\frac{1}{|\mathcal{D}_i|^2}.$$

Since $|\mathcal{D}_i| \geq m_{\min}$, we have $|\mathcal{D}_i|^{-2} \leq m_{\min}^{-2}$, and hence

$$\sum_{i=1}^{n}\frac{1}{|\mathcal{D}_i|^2} \leq \frac{n}{m_{\min}^2}.$$

Substituting this into the expression for ν_t^2 gives

$$\nu_t^2 \leq \frac{(C_g\sigma_g)^2}{n^3}\cdot\frac{n}{m_{\min}^2} = \frac{(C_g\sigma_g)^2}{n^2 m_{\min}^2} =: \nu^2.$$

Finally,

$$\mathbb{E}[\|\xi_t\|_2^2 \mid \mathcal{F}_t] = \operatorname{tr}\{\mathbb{E}[\xi_t\xi_t^\top \mid \mathcal{F}_t]\} = d\nu_t^2 \leq d\nu^2,$$

which proves the claim. □

Proof of Lemma 4.11. Fix $t \geq 0$. By (1), every clipped per-example gradient satisfies

$$\|\bar{g}(x,y)\|_2 \leq C_g.$$

Therefore, for each client i,

$$\left\|\frac{1}{|\mathcal{D}_i|}\sum_{(x,y)\in\mathcal{D}_i}\bar{g}(x,y)\right\|_2 \leq \frac{1}{|\mathcal{D}_i|}\sum_{(x,y)\in\mathcal{D}_i}\|\bar{g}(x,y)\|_2 \leq C_g.$$

where we used the triangle inequality. Averaging these client-wise quantities gives

$$\begin{aligned}\|g_{\text{clip}}(\theta_t)\|_2 &= \left\|\frac{1}{n}\sum_{i=1}^{n}\frac{1}{|\mathcal{D}_i|}\sum_{(x,y)\in\mathcal{D}_i}\bar{g}(x,y)\right\|_2 \\ &\leq \frac{1}{n}\sum_{i=1}^{n}\left\|\frac{1}{|\mathcal{D}_i|}\sum_{(x,y)\in\mathcal{D}_i}\bar{g}(x,y)\right\|_2 \leq \frac{1}{n}\sum_{i=1}^{n}C_g = C_g.\end{aligned}$$

This proves (8).

Next, by (3),

$$G_t = g_{\text{clip}}(\theta_t) + \xi_t.$$

Since $g_{\text{clip}}(\theta_t)$ is $\mathcal{F}_t$-measurable and $\mathbb{E}[\xi_t \mid \mathcal{F}_t] = 0$,

$$\begin{aligned}\mathbb{E}\big[\|G_t\|_2^2 \mid \mathcal{F}_t\big] &= \mathbb{E}\big[\|g_{\text{clip}}(\theta_t) + \xi_t\|_2^2 \mid \mathcal{F}_t\big] \\ &= \|g_{\text{clip}}(\theta_t)\|_2^2 + 2\,g_{\text{clip}}(\theta_t)^\top\mathbb{E}[\xi_t \mid \mathcal{F}_t] + \mathbb{E}[\|\xi_t\|_2^2 \mid \mathcal{F}_t] \\ &= \|g_{\text{clip}}(\theta_t)\|_2^2 + \operatorname{tr}\big(\mathbb{E}[\xi_t\xi_t^\top \mid \mathcal{F}_t]\big) \\ &= \|g_{\text{clip}}(\theta_t)\|_2^2 + d\nu_t^2.\end{aligned}$$

The final bound follows from (8) and Lemma 4.10. □

Proof of Lemma 4.12. Fix $t \geq 0$. Using the recursion for M_t and the decomposition $G_t = g_{\text{clip}}(\theta_t) + \xi_t$,

$$M_t = \beta M_{t-1} + (1-\beta) g_{\text{clip}}(\theta_t) + (1-\beta)\xi_t.$$

Conditional on $\mathcal{F}_t$, both M_{t-1} and $g_{\text{clip}}(\theta_t)$ are deterministic, whereas ξ_t is mean-zero. Therefore

$$\mathbb{E}\big[\|M_t\|_2^2 \mid \mathcal{F}_t\big] = \|\beta M_{t-1} + (1-\beta) g_{\text{clip}}(\theta_t)\|_2^2 + (1-\beta)^2 \mathbb{E}[\|\xi_t\|_2^2 \mid \mathcal{F}_t]. \tag{26}$$

Indeed, if we write

$$a_t := \beta M_{t-1} + (1-\beta) g_{\text{clip}}(\theta_t),$$

then

$$\|a_t + (1-\beta)\xi_t\|_2^2 = \|a_t\|_2^2 + 2(1-\beta) a_t^\top \xi_t + (1-\beta)^2 \|\xi_t\|_2^2,$$

and the mixed term vanishes after conditioning on $\mathcal{F}_t$.

Now use the convexity of the squared Euclidean norm: for every $a, b \in \mathbb{R}^d$ and every $\beta \in [0,1]$,

$$\|\beta a + (1-\beta) b\|_2^2 \leq \beta \|a\|_2^2 + (1-\beta)\|b\|_2^2.$$

Applying this to $a = M_{t-1}$ and $b = g_{\text{clip}}(\theta_t)$, and invoking Lemma 4.11, gives

$$\|\beta M_{t-1} + (1-\beta) g_{\text{clip}}(\theta_t)\|_2^2 \leq \beta \|M_{t-1}\|_2^2 + (1-\beta) C_g^2.$$

Substituting this and $\mathbb{E}[\|\xi_t\|_2^2 \mid \mathcal{F}_t] = d\nu^2$ into (26) proves (9).

Taking total expectation yields (10). To solve this recursion, let

$$b := (1-\beta) C_g^2 + (1-\beta)^2 d\nu^2.$$

Then $u_t \leq \beta u_{t-1} + b$, so by induction,

$$u_t \leq \beta^{t+1} u_{-1} + b \sum_{j=0}^{t} \beta^j.$$

Since $M_{-1} = 0_d$, we have $u_{-1} = 0$, hence

$$u_t \leq b \, \frac{1-\beta^{t+1}}{1-\beta} = \big(1-\beta^{t+1}\big)\big(C_g^2 + (1-\beta) d\nu^2\big) \leq C_g^2 + (1-\beta) d\nu^2.$$

This proves (11). □

Proof of Lemma 4.14. The identity (12) is the Sherman–Morrison formula applied to the rank-one perturbation $\rho I_d + M_t M_t^\top$:

$$(\rho I_d + u u^\top)^{-1} = \frac{1}{\rho} I_d - \frac{u u^\top}{\rho(\rho + \|u\|_2^2)}, \qquad u \in \mathbb{R}^d.$$

Substituting $u = M_t$ yields (12).

The matrix

$$\frac{M_t M_t^\top}{\rho(\rho + \|M_t\|_2^2)}$$

is positive semidefinite, hence

$$0 \preceq H_t \preceq \frac{1}{\rho} I_d.$$

The operator norm bound (13) follows immediately:

$$\|H_t v\|_2 \leq \|H_t\|_{\text{op}} \|v\|_2 \leq \frac{1}{\rho} \|v\|_2.$$

Squaring both sides gives (14).

For the quadratic bounds, use (12) directly:

$$v^\top H_t v = \frac{1}{\rho}\|v\|_2^2 - \frac{(M_t^\top v)^2}{\rho(\rho + \|M_t\|_2^2)}.$$

The upper bound follows immediately because the second term is nonnegative:

$$v^\top H_t v \le \frac{1}{\rho}\|v\|_2^2.$$

For the lower bound, by Cauchy–Schwarz, $(M_t^\top v)^2 \le \|M_t\|_2^2\|v\|_2^2$. Hence

$$\frac{(M_t^\top v)^2}{\rho(\rho + \|M_t\|_2^2)} \le \frac{\|M_t\|_2^2\|v\|_2^2}{\rho(\rho + \|M_t\|_2^2)}.$$

Therefore

$$v^\top H_t v \ge \left\{\frac{1}{\rho} - \frac{\|M_t\|_2^2}{\rho(\rho + \|M_t\|_2^2)}\right\}\|v\|_2^2 = \frac{1}{\rho + \|M_t\|_2^2}\|v\|_2^2.$$

This proves (15). Substituting $v = \nabla F(\theta_t)$ gives

$$\nabla F(\theta_t)^\top H_t \nabla F(\theta_t) \ge \frac{1}{\rho + \|M_t\|_2^2}\|\nabla F(\theta_t)\|_2^2,$$

which proves (16). □

Proof of Corollary 4.15. By Lemma 4.12,

$$\mathbb{E}\|M_t\|_2^2 \le \bar{M}^2 \qquad \text{for every } t \ge 0.$$

Therefore, by Markov's inequality,

$$\mathbb{P}\left(\|M_t\|_2^2 > M_\delta^2\right) \le \frac{\bar{M}^2}{M_\delta^2} = \frac{\delta_M}{T}.$$

Applying the union bound over $t = 0, \dots, T-1$ gives

$$\mathbb{P}\left(\max_{0\le t\le T-1}\|M_t\|_2^2 > M_\delta^2\right) \le \sum_{t=0}^{T-1}\mathbb{P}\left(\|M_t\|_2^2 > M_\delta^2\right) \le \delta_M.$$

Thus, with probability at least $1 - \delta_M$,

$$\|M_t\|_2^2 \le M_\delta^2 \qquad \text{for all } t = 0, \dots, T-1.$$

On this event, Lemma 4.14 implies

$$\nabla F(\theta_t)^\top H_t \nabla F(\theta_t) \ge \frac{1}{\rho + \|M_t\|_2^2}\|\nabla F(\theta_t)\|_2^2 \ge \frac{1}{\rho + M_\delta^2}\|\nabla F(\theta_t)\|_2^2.$$

This proves the claim. □

Proof of Lemma 4.16. Set

$$\Delta_t := \theta_{t+1} - \theta_t = -\eta H_t G_t.$$

By L-smoothness, (6) gives

$$F(\theta_{t+1}) \le F(\theta_t) + \langle \nabla F(\theta_t), \Delta_t\rangle + \frac{L}{2}\|\Delta_t\|_2^2. \tag{27}$$

Substituting $\Delta_t = -\eta H_t G_t$ yields

$$F(\theta_{t+1}) \leq F(\theta_t) - \eta\langle \nabla F(\theta_t), H_t G_t\rangle + \frac{L\eta^2}{2}\|H_t G_t\|_2^2. \tag{28}$$

We bound the linear and quadratic terms separately.

Step 1: decomposition of the linear term.

Using (3),

$$G_t = \nabla F(\theta_t) + \zeta_t + \xi_t.$$

Therefore

$$\langle \nabla F(\theta_t), H_t G_t\rangle = \nabla F(\theta_t)^\top H_t \nabla F(\theta_t) + \nabla F(\theta_t)^\top H_t \zeta_t + \nabla F(\theta_t)^\top H_t \xi_t. \tag{29}$$

All bounds below are *pathwise*, i.e. they hold for each realized M_t and ξ_t. Hence no conditional independence between H_t and ξ_t is needed.

Step 2: lower bound on the curvature term.

Using the Sherman–Morrison representation (12), we also have the conservative bound

$$\nabla F(\theta_t)^\top H_t \nabla F(\theta_t) \geq \frac{1}{\rho}\|\nabla F(\theta_t)\|_2^2 - \frac{\|M_t\|_2^2}{\rho(\rho + \|M_t\|_2^2)}\|\nabla F(\theta_t)\|_2^2.$$

Since

$$\frac{1}{\rho(\rho + \|M_t\|_2^2)} \leq \frac{1}{\rho^2}$$

and $\|\nabla F(\theta_t)\|_2 \leq G_{\max}$, this yields

$$\nabla F(\theta_t)^\top H_t \nabla F(\theta_t) \geq \frac{1}{\rho}\|\nabla F(\theta_t)\|_2^2 - \frac{G_{\max}^2}{\rho^2}\|M_t\|_2^2. \tag{30}$$

Step 3: control of the bias cross-term.

Using Cauchy–Schwarz, (13), and Assumption 4.5,

$$\left|\nabla F(\theta_t)^\top H_t \zeta_t\right| \leq \|\nabla F(\theta_t)\|_2 \, \|H_t \zeta_t\|_2 \leq \frac{\|\nabla F(\theta_t)\|_2 \, \|\zeta_t\|_2}{\rho} \leq \frac{\|\nabla F(\theta_t)\|_2 \, \zeta_{\max}}{\rho}. \tag{31}$$

Apply Young's inequality $ab \leq \frac{\tau_1}{2}a^2 + \frac{1}{2\tau_1}b^2$ with

$$a = \|\nabla F(\theta_t)\|_2, \qquad b = \frac{\zeta_{\max}}{\rho},$$

to obtain

$$\left|\nabla F(\theta_t)^\top H_t \zeta_t\right| \leq \frac{\tau_1}{2}\|\nabla F(\theta_t)\|_2^2 + \frac{\zeta_{\max}^2}{2\tau_1\rho^2}. \tag{32}$$

Step 4: control of the noise cross-term.

Similarly,

$$\left|\nabla F(\theta_t)^\top H_t \xi_t\right| \leq \|\nabla F(\theta_t)\|_2 \, \|H_t \xi_t\|_2 \leq \frac{\|\nabla F(\theta_t)\|_2 \, \|\xi_t\|_2}{\rho}. \tag{33}$$

Applying Young's inequality with

$$a = \|\nabla F(\theta_t)\|_2, \qquad b = \frac{\|\xi_t\|_2}{\rho},$$

gives

$$\left|\nabla F(\theta_t)^\top H_t \xi_t\right| \le \frac{\tau_2}{2}\|\nabla F(\theta_t)\|_2^2 + \frac{\|\xi_t\|_2^2}{2\tau_2\rho^2}. \tag{34}$$

Combining (29), (30), (32), and (34), we obtain the pathwise lower bound

$$\begin{aligned}\langle \nabla F(\theta_t), H_t G_t\rangle \ge \left(\frac{1}{\rho} - \frac{\tau_1+\tau_2}{2}\right)\|\nabla F(\theta_t)\|_2^2 - \frac{G_{\max}^2}{\rho^2}\|M_t\|_2^2 \\ - \frac{\zeta_{\max}^2}{2\tau_1\rho^2} - \frac{\|\xi_t\|_2^2}{2\tau_2\rho^2}.\end{aligned} \tag{35}$$

Step 5: upper bound on the quadratic term.

By (14),

$$\|H_t G_t\|_2^2 \le \frac{1}{\rho^2}\|G_t\|_2^2.$$

Taking conditional expectation given $\mathcal{F}_t$ and then applying Lemma 4.11,

$$\mathbb{E}\big[\|H_t G_t\|_2^2 \mid \mathcal{F}_t\big] \le \frac{1}{\rho^2}\mathbb{E}\big[\|G_t\|_2^2 \mid \mathcal{F}_t\big] \le \frac{C_g^2 + d\nu^2}{\rho^2}. \tag{36}$$

Step 6: combine the bounds and take conditional expectation.

Substitute (35) into (28):

$$\begin{aligned}F(\theta_{t+1}) \le F(\theta_t) - \eta\left(\frac{1}{\rho} - \frac{\tau_1+\tau_2}{2}\right)\|\nabla F(\theta_t)\|_2^2 + \frac{\eta G_{\max}^2}{\rho^2}\|M_t\|_2^2 \\ + \frac{\eta\zeta_{\max}^2}{2\tau_1\rho^2} + \frac{\eta}{2\tau_2\rho^2}\|\xi_t\|_2^2 + \frac{L\eta^2}{2}\|H_t G_t\|_2^2.\end{aligned}$$

Now we condition on $\mathcal{F}_t$. Since $F(\theta_t)$ and $\nabla F(\theta_t)$ are $\mathcal{F}_t$-measurable,

$$\mathbb{E}[\|\xi_t\|_2^2 \mid \mathcal{F}_t] = d\nu_t^2 \le d\nu^2,$$

and (36) holds, we obtain

$$\begin{aligned}\mathbb{E}[F(\theta_{t+1}) \mid \mathcal{F}_t] \le F(\theta_t) - \eta\left(\frac{1}{\rho} - \frac{\tau_1+\tau_2}{2}\right)\|\nabla F(\theta_t)\|_2^2 + \frac{\eta G_{\max}^2}{\rho^2}\mathbb{E}\big[\|M_t\|_2^2 \mid \mathcal{F}_t\big] \\ + \frac{\eta\,\zeta_{\max}^2}{2\tau_1\rho^2} + \frac{\eta\,d\,\nu^2}{2\tau_2\rho^2} + \frac{L\eta^2}{2\rho^2}\big(C_g^2 + d\nu^2\big).\end{aligned}$$

This completes the proof. □

Proof of Lemma 4.17. Take total expectation in (18):

$$\begin{aligned}\mathbb{E}[F(\theta_{t+1})] \le \mathbb{E}[F(\theta_t)] - \eta c_\nabla\,\mathbb{E}\|\nabla F(\theta_t)\|_2^2 + \frac{\eta G_{\max}^2}{\rho^2}\mathbb{E}\|M_t\|_2^2 \\ + \frac{\eta\,\zeta_{\max}^2}{2\tau_1\rho^2} + \frac{\eta\,d\,\nu^2}{2\tau_2\rho^2} + \frac{L\eta^2}{2\rho^2}\big(C_g^2 + d\nu^2\big).\end{aligned}$$

By Lemma 4.12,

$$\mathbb{E}\|M_t\|_2^2 \le \bar{M}^2.$$

Substituting this bound yields (19). □

Proof of Theorem 4.21. By Lemma 4.17,

$$\mathbb{E}[F(\theta_{t+1})] \le \mathbb{E}[F(\theta_t)] - \eta c_\nabla \, \mathbb{E}\|\nabla F(\theta_t)\|_2^2 + \Gamma.$$

By (7), strong convexity implies the pointwise bound

$$\|\nabla F(\theta_t)\|_2^2 \ge 2\mu\big(F(\theta_t) - F(\theta^\star)\big).$$

Taking expectations preserves the inequality:

$$\mathbb{E}\|\nabla F(\theta_t)\|_2^2 \ge 2\mu \, \mathbb{E}[F(\theta_t) - F(\theta^\star)] \, .$$

Substituting this into the previous display gives

$$\mathbb{E}[F(\theta_{t+1}) - F(\theta^\star)] \le \big(1 - 2\mu\eta c_\nabla\big)\mathbb{E}[F(\theta_t) - F(\theta^\star)] + \Gamma. \tag{37}$$

Define

$$\Delta_t := \mathbb{E}[F(\theta_t) - F(\theta^\star)].$$

Then (37) becomes

$$\Delta_{t+1} \le r\Delta_t + \Gamma.$$

Repeatedly applying this recursion yields

$$\Delta_T \le r^T \Delta_0 + \Gamma \sum_{j=0}^{T-1} r^j = r^T \Delta_0 + \Gamma \frac{1 - r^T}{1 - r},$$

which proves (21). Since $1 - r = 2\mu\eta c_\nabla$ and $1 - r^T \le 1$, we obtain (22). □

Proof of Lemma 4.25. By L-smoothness,

$$F(\theta_{t+1}^{\mathrm{GD}}) \le F(\theta_t^{\mathrm{GD}}) - \eta\langle \nabla F(\theta_t^{\mathrm{GD}}), G_t\rangle + \frac{L\eta^2}{2}\|G_t\|_2^2.$$

Since $G_t = \nabla F(\theta_t^{\mathrm{GD}}) + \zeta_t + \xi_t$ and $\mathbb{E}[\xi_t \mid \mathcal{F}_t] = 0$,

$$\mathbb{E}[\langle \nabla F(\theta_t^{\mathrm{GD}}), G_t\rangle \mid \mathcal{F}_t] = \|\nabla F(\theta_t^{\mathrm{GD}})\|_2^2 + \langle \nabla F(\theta_t^{\mathrm{GD}}), \zeta_t\rangle.$$

Young's inequality gives

$$-\langle \nabla F(\theta_t^{\mathrm{GD}}), \zeta_t\rangle \le \frac{\tau}{2}\|\nabla F(\theta_t^{\mathrm{GD}})\|_2^2 + \frac{\zeta_{\max}^2}{2\tau}.$$

Together with $\mathbb{E}[\|G_t\|_2^2 \mid \mathcal{F}_t] \le C_g^2 + d\nu^2$, this proves the claim. □

Proof of Theorem 4.26. Taking expectations in Lemma 4.25 and using strong convexity, $\|\nabla F(\theta)\|_2^2 \ge 2\mu\{F(\theta) - F(\theta^\star)\}$, gives

$$\mathbb{E}[F(\theta_{t+1}^{\mathrm{GD}}) - F(\theta^\star)] \le (1 - 2\mu\eta c_{\mathrm{GD}})\mathbb{E}[F(\theta_t^{\mathrm{GD}}) - F(\theta^\star)] + \Gamma_{\mathrm{GD}}.$$

Iterating the recursion proves the result. □

Proof of Theorem 4.29. Lemma 4.17 gives

$$\mathbb{E}[F(\theta_{t+1})] \le \mathbb{E}[F(\theta_t)] - \eta c_\nabla \, \mathbb{E}\|\nabla F(\theta_t)\|_2^2 + \Gamma.$$

By the PL inequality (23),

$$\|\nabla F(\theta_t)\|_2^2 \ge 2\mu_{\mathrm{PL}}\big(F(\theta_t) - F(\theta^\star)\big)$$

holds pointwise, hence also after taking expectations:

$$\mathbb{E}\|\nabla F(\theta_t)\|_2^2 \ge 2\mu_{\mathrm{PL}} \, \mathbb{E}[F(\theta_t) - F(\theta^\star)] \, .$$

Substituting this into the one-step descent bound yields

$$\mathbb{E}[F(\theta_{t+1}) - F(\theta^\star)] \le \big(1 - 2\mu_{\mathrm{PL}}\eta c_\nabla\big)\mathbb{E}[F(\theta_t) - F(\theta^\star)] + \Gamma. \tag{38}$$

Defining

$$\Delta_t := \mathbb{E}[F(\theta_t) - F(\theta^\star)],$$

we obtain

$$\Delta_{t+1} \le r_{\mathrm{PL}}\Delta_t + \Gamma.$$

Iterating the recursion and summing the geometric series proves (24); the simplified bound (25) follows from $1 - r_{\mathrm{PL}} = 2\mu_{\mathrm{PL}}\eta c_\nabla$. □

Proof of Proposition 4.19. By L-smoothness,

$$F(\theta_{t+1}) \le F(\theta_t) - \eta\langle \nabla F(\theta_t), H_t G_t\rangle + \frac{L\eta^2}{2}\|H_t G_t\|_2^2.$$

Write $G_t = \nabla F(\theta_t) + b_t + \xi_t$. Then

$$-\langle \nabla F(\theta_t), H_t G_t\rangle = -\nabla F(\theta_t)^\top H_t \nabla F(\theta_t) - \nabla F(\theta_t)^\top H_t b_t - \nabla F(\theta_t)^\top H_t \xi_t.$$

Using the same preconditioner bound (30),

$$\nabla F(\theta_t)^\top H_t \nabla F(\theta_t) \ge \frac{1}{\rho}\|\nabla F(\theta_t)\|_2^2 - \frac{G_{\max}^2}{\rho^2}\|M_t\|_2^2.$$

Also,

$$|\nabla F(\theta_t)^\top H_t b_t| \le \frac{1}{\rho}\|\nabla F(\theta_t)\|_2\|b_t\|_2 \le \frac{\tau_1}{2}\|\nabla F(\theta_t)\|_2^2 + \frac{B^2}{2\tau_1\rho^2},$$

and similarly,

$$\mathbb{E}\left[|\nabla F(\theta_t)^\top H_t \xi_t| \mid \mathcal{F}_t\right] \le \frac{\tau_2}{2}\|\nabla F(\theta_t)\|_2^2 + \frac{d\nu^2}{2\tau_2\rho^2}.$$

Finally, since $\|H_t G_t\|_2 \le \rho^{-1}\|G_t\|_2$,

$$\mathbb{E}[\|H_t G_t\|_2^2 \mid \mathcal{F}_t] \le \frac{1}{\rho^2}\mathbb{E}[\|G_t\|_2^2 \mid \mathcal{F}_t] \le \frac{G_0^2 + d\nu^2}{\rho^2}.$$

Combining the preceding bounds gives the conditional inequality. Taking expectations and using $\mathbb{E}\|M_t\|_2^2 \le \overline{M}^2$ gives the unconditional bound. □

Proof of Theorem 4.31. From Lemma 4.17,

$$\mathbb{E}[F(\theta_{t+1})] \le \mathbb{E}[F(\theta_t)] - \eta c_\nabla \mathbb{E}\|\nabla F(\theta_t)\|_2^2 + \Gamma.$$

Rearranging gives

$$\eta c_\nabla \mathbb{E}\|\nabla F(\theta_t)\|_2^2 \le \mathbb{E}[F(\theta_t)] - \mathbb{E}[F(\theta_{t+1})] + \Gamma.$$

Summing from $t = 0$ to $T - 1$ yields

$$\eta c_\nabla \sum_{t=0}^{T-1} \mathbb{E}\|\nabla F(\theta_t)\|_2^2 \le F(\theta_0) - \mathbb{E}[F(\theta_T)] + T\Gamma.$$

Since $F(\theta_T) \ge F_{\inf}$, we obtain

$$\eta c_\nabla \sum_{t=0}^{T-1} \mathbb{E}\|\nabla F(\theta_t)\|_2^2 \le F(\theta_0) - F_{\inf} + T\Gamma.$$

Dividing by $\eta c_\nabla T$ gives the first claim. The randomized-iterate statement follows because

$$\mathbb{E}\|\nabla F(\theta_R)\|_2^2 = \frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla F(\theta_t)\|_2^2.$$

□

Proof of Corollary 4.32. The proof is identical to that of Theorem 4.31, using the descent recursion in Proposition 4.19 with Γ_B in place of Γ. □

Proof of Lemma 4.35. The mapping

$$(g_{1,t}, \ldots, g_{n,t}) \mapsto \frac{1}{n}\sum_{i=1}^{n} g_{i,t}$$

is deterministic. Therefore G_t is a deterministic post-processing of the released output $\mathcal{M}_t(\mathcal{D})$.

Since differential privacy is invariant under post-processing Dwork et al. (2014), the privacy guarantee is preserved. □

Proof of Lemma 4.36. Condition on the previous server state (θ_t, M_{t-1}). Once this state is fixed, the quantities

$$M_t = \beta M_{t-1} + (1-\beta)G_t, \qquad \widehat{\mathcal{I}}_t = \rho I_d + M_t M_t^\top, \qquad H_t = \widehat{\mathcal{I}}_t^{-1}, \qquad \theta_{t+1} = \theta_t - \eta_t H_t G_t$$

are deterministic functions of G_t.

Thus the mapping

$$G_t \mapsto (M_t, H_t, \theta_{t+1})$$

is deterministic. The result therefore follows from the post-processing property of differential privacy.

□

B Computational Complexity Analysis

We now quantify the computational cost of DP-FedSOFIM and compare it with matrix-based second-order alternatives. The key point is that the proposed server-side preconditioner preserves the linear-in-d structure of first-order federated optimization while introducing only a negligible additional cost beyond standard aggregation.

Lemma B.1 (Per-round complexity of DP-FedSOFIM)**.** *Let n denote the number of participating clients and let d denote the model dimension. Under full participation, one communication round of DP-FedSOFIM has total client-side computational cost $O(nd)$ and server-side computational cost $O(nd)$. The additional cost incurred by the SOFIM preconditioning step itself is only $O(d)$ per round, with $O(d)$ server memory.*

Proof. We separate the computation into the client and server components.

On the client side, each client i computes per-example gradients in $\mathbb{R}^d$, clips them according to (1), forms the sum of clipped gradients, and adds Gaussian noise before normalization. At the level of vector operations, both gradient clipping and the formation of the released update require work linear in d. Thus the computational cost per client is $O(d)$, and summing over the n participating clients yields an overall client-side cost of $O(nd)$ per round.

On the server side, the first step is the aggregation

$$G_t = \frac{1}{n}\sum_{i=1}^{n} g_{i,t},$$

which requires summing n vectors in $\mathbb{R}^d$ and therefore costs $O(nd)$. The momentum update

$$M_t = \beta M_{t-1} + (1-\beta)G_t$$

requires one vector scaling and one vector addition, hence $O(d)$ operations.

The distinctive step in DP-FedSOFIM is the application of the rank-one preconditioner. Using the Sherman–Morrison representation,

$$H_t G_t = \frac{1}{\rho} G_t - \frac{M_t^\top G_t}{\rho^2 + \rho \|M_t\|_2^2} M_t,$$

the preconditioned direction can be computed using two inner products, one scalar division, one scalar-vector multiplication, and one vector subtraction. Each of these operations is linear in d, so the total cost of forming $H_t G_t$ is $O(d)$. The parameter update

$$\theta_{t+1} = \theta_t - \eta H_t G_t$$

is again a single vector operation and therefore costs $O(d)$.

Combining these terms, the total server-side cost per round is

$$O(nd) + O(d) = O(nd),$$

with the aggregation step dominating the asymptotics. The additional curvature computation introduced by DP-FedSOFIM is therefore only $O(d)$ beyond the standard first-order federated pipeline. Finally, since the server stores only the current vectors G_t, M_t, and θ_t, the additional server memory required by the preconditioner is $O(d)$.

In contrast, matrix-based second-order methods that maintain a dense curvature surrogate require at least $O(d^2)$ storage and typically $O(d^3)$ inversion or factorization cost per round. DP-FedSOFIM avoids these costs by exploiting the rank-one structure of the SOFIM approximation. □

The preceding result shows that DP-FedSOFIM preserves the principal computational advantage of first-order federated optimization. In particular, the method does not require clients to transmit or store any matrix-valued curvature information, and all curvature-aware computation remains confined to a linear-time server-side post-processing step.

Runtime Analysis

Table 5 reports the average wall-clock time per communication round and total training time over 70 rounds on CIFAR-10 and PathMNIST ($n = 20$, non-IID, Dirichlet $\alpha = 0.5$, seed=42). All methods use $K = 1$ local step per round except DP-SCAFFOLD, which uses the K value selected by hyperparameter tuning at each privacy level. The longer per-round times on PathMNIST (≈1s vs. ≈0.6s on CIFAR-10) reflect its larger input dimensionality. DP-FedSOFIM adds negligible overhead over DP-FedGD on both datasets ($< 2\%$), confirming the $O(d)$ complexity of the Sherman–Morrison preconditioning step established in Lemma B.1. DP-SCAFFOLD incurs significantly higher total training cost at $\varepsilon \geq 1$ due to its use of multiple local steps ($K = 5$ at $\varepsilon = 1$, $K = 7$ at $\varepsilon \geq 5$), reaching up to 7× the wall-clock time of DP-FedSOFIM on PathMNIST at $\varepsilon \geq 5$.

C Variance Reduction via the Momentum Buffer

An important feature of DP-FedSOFIM is that the curvature proxy is not formed from a single privatized aggregate, but from the exponentially weighted moving average

$$M_t = \beta M_{t-1} + (1-\beta)G_t.$$

Since the aggregated gradient G_t contains Gaussian privacy noise, this recursion implicitly smooths the noise across rounds. The next result makes this effect precise.

Table 5: Average wall-clock time per round and total training time (70 rounds), $n = 20$ clients, Non-IID (Dirichlet $\alpha = 0.5$), seed=42. Runtimes for all methods except DP-SCAFFOLD are invariant to ε.

	CIFAR-10		**PathMNIST**	
Algorithm	Avg (s)	Total (s)	Avg (s)	Total (s)
DP-FedGD	0.62	44.2	1.07	75.7
DP-FedSOFIM	0.62	45.4	1.09	78.2
DP-FedAvg	0.59	42.5	1.05	74.2
DP-FedAdam	0.64	45.7	1.11	78.3
DP-FedYogi	0.62	44.5	1.10	77.9
DP-FedFC	0.60	42.9	1.07	75.4
DP-SCAFFOLD† *(per privacy level)*				
$\varepsilon = 0.5$	0.57	40.8	1.01	71.4
$\varepsilon = 1$	2.85	200.4	5.04	353.8
$\varepsilon = 5$	3.97	278.7	7.07	495.7
$\varepsilon = 10$	3.97	278.6	7.09	496.8

† DP-SCAFFOLD uses $K = 1, 5, 7, 7$ local steps at $\varepsilon = 0.5, 1, 5, 10$ respectively. All other methods use $K = 1$.

Lemma C.1 (Variance reduction induced by exponential averaging)**.** *Suppose that the aggregated gradient admits the decomposition*

$$G_t = \nabla F(\theta_t) + \xi_t,$$

where $\{\xi_t\}_{t\geq 0}$ *are independent mean-zero Gaussian vectors with covariance* $\nu^2 I_d$*. If*

$$M_t = (1-\beta)\sum_{s=0}^{t} \beta^{\,t-s} G_s, \qquad \beta \in [0,1),$$

then the noise component of M_t *has covariance*

$$\mathrm{Cov}(M_t) = \nu^2(1-\beta)^2 \sum_{s=0}^{t} \beta^{2(t-s)} I_d = \nu^2(1-\beta)^2 \frac{1-\beta^{2(t+1)}}{1-\beta^2}\, I_d.$$

Consequently, as $t \to \infty$*,*

$$\mathrm{Cov}(M_t) \longrightarrow \frac{1-\beta}{1+\beta}\,\nu^2 I_d.$$

Proof. Expanding the recursion for M_t gives

$$M_t = (1-\beta)\sum_{s=0}^{t} \beta^{\,t-s}\big(\nabla F(\theta_s) + \xi_s\big).$$

Only the Gaussian noise terms contribute to the covariance, so it suffices to consider

$$M_t^{\text{noise}} = (1-\beta)\sum_{s=0}^{t} \beta^{\,t-s}\xi_s.$$

Because the vectors ξ_s are independent and satisfy

$$\mathbb{E}[\xi_s] = 0, \qquad \mathrm{Cov}(\xi_s) = \nu^2 I_d,$$

all cross-covariance terms vanish, and we obtain

$$\begin{aligned}\mathrm{Cov}(M_t) &= (1-\beta)^2 \sum_{s=0}^{t} \beta^{2(t-s)} \mathrm{Cov}(\xi_s) \\ &= \nu^2 (1-\beta)^2 \sum_{s=0}^{t} \beta^{2(t-s)} I_d.\end{aligned}$$

Evaluating the geometric sum yields

$$\sum_{s=0}^{t} \beta^{2(t-s)} = \sum_{j=0}^{t} \beta^{2j} = \frac{1-\beta^{2(t+1)}}{1-\beta^2},$$

and therefore

$$\mathrm{Cov}(M_t) = \nu^2 (1-\beta)^2 \frac{1-\beta^{2(t+1)}}{1-\beta^2} I_d.$$

Since $1-\beta^2 = (1-\beta)(1+\beta)$, the limit as $t \to \infty$ is

$$\mathrm{Cov}(M_t) = \frac{1-\beta}{1+\beta} \nu^2 I_d.$$

This proves the claim. □

The lemma shows that the moving-average buffer attenuates the isotropic privacy noise by the factor $(1-\beta)/(1+\beta)$. For values such as $\beta = 0.9$, this factor is approximately 0.0526, corresponding to a variance reduction of about 19× relative to the raw per-round noise. This provides a useful quantitative explanation for why the rank-one preconditioner built from M_t remains stable even when the individual privatized aggregates are substantially corrupted by Gaussian noise.

D Sherman–Morrison Derivation for the Regularized SOFIM Preconditioner

For completeness, we record the derivation of the closed-form inverse used in DP-FedSOFIM. At round t, the server constructs the regularized rank-one curvature approximation

$$\widehat{\mathcal{F}}_t = M_t M_t^\top + \rho I_d,$$

with regularization parameter $\rho > 0$. The corresponding preconditioner is

$$H_t = \widehat{\mathcal{F}}_t^{-1}.$$

A direct inversion of a dense $d \times d$ matrix would cost $O(d^3)$, but the rank-one structure of $\widehat{\mathcal{F}}_t$ permits an exact closed form.

The Sherman–Morrison identity states that for any invertible matrix A and vectors u, v such that $1 + v^\top A^{-1} u \neq 0$,

$$(A + uv^\top)^{-1} = A^{-1} - \frac{A^{-1} u v^\top A^{-1}}{1 + v^\top A^{-1} u}.$$

Applying this identity with

$$A = \rho I_d, \qquad u = v = M_t,$$

we obtain

$$\begin{aligned}H_t &= (\rho I_d + M_t M_t^\top)^{-1} \\ &= \frac{1}{\rho} I_d - \frac{\frac{1}{\rho} M_t M_t^\top \frac{1}{\rho}}{1 + M_t^\top \frac{1}{\rho} M_t}.\end{aligned}$$

Since

$$1 + M_t^\top \frac{1}{\rho} M_t = \frac{\rho + \|M_t\|_2^2}{\rho},$$

this simplifies to

$$H_t = \frac{1}{\rho} I_d - \frac{M_t M_t^\top}{\rho^2 + \rho \|M_t\|_2^2},$$

which is exactly the formula used in the main text.

This representation is particularly valuable because one never needs to form the matrix H_t explicitly. Indeed, applying the preconditioner to the aggregated gradient yields

$$H_t G_t = \left(\frac{1}{\rho} I_d - \frac{M_t M_t^\top}{\rho^2 + \rho \|M_t\|_2^2} \right) G_t = \frac{1}{\rho} G_t - \frac{M_t^\top G_t}{\rho^2 + \rho \|M_t\|_2^2} M_t.$$

Thus the action of H_t on G_t reduces to two inner products and a small number of vector operations, all of which are linear in d. This is why the preconditioning step adds only $O(d)$ computation and $O(d)$ storage to the standard federated gradient pipeline.

The formula also makes the geometry transparent. Along the direction spanned by M_t, the effective scaling is

$$\frac{1}{\rho + \|M_t\|_2^2},$$

whereas any direction orthogonal to M_t is scaled by $1/\rho$. The preconditioner therefore contracts the update more strongly along the dominant historical gradient direction while leaving orthogonal directions less damped. This anisotropic rescaling is precisely the mechanism through which the rank-one SOFIM approximation captures curvature information at linear cost.

E Hockey-Stick Divergence and Privacy Accounting

We now describe the privacy accounting used to calibrate the Gaussian noise multiplier σ_g for a target privacy budget (ε, δ) over T communication rounds under full client participation. Since the client-side release mechanism in DP-FedSOFIM is identical to that of DP-FedGD, the privacy analysis is entirely determined by the noisy release of the clipped gradients; the momentum buffer, Fisher proxy construction, and Sherman–Morrison preconditioning are deterministic post-processing steps and therefore do not contribute additional privacy loss.

At round t, client i releases

$$g_{i,t} = \frac{1}{|\mathcal{D}_i|} \left(\sum_{(x,y) \in \mathcal{D}_i} \bar{g}(x, y) + E_{i,t} \right), \qquad E_{i,t} \sim \mathcal{N}\left(0, \frac{(C_g \sigma_g)^2}{n} I_d\right).$$

We adopt record-level replace-one adjacency: two federated datasets are neighboring if they differ in exactly one record in exactly one client's local dataset. Under this notion of adjacency, changing a single record can alter the sum of clipped gradients on client i by at most $2C_g$ in Euclidean norm. After normalization by $|\mathcal{D}_i|$, the ℓ_2-sensitivity of the released client update is therefore

$$\Delta_i = \frac{2C_g}{|\mathcal{D}_i|}.$$

To obtain a single conservative bound across all clients, we use

$$|\mathcal{D}_{\min}| := \min_i |\mathcal{D}_i|, \qquad \Delta := \frac{2C_g}{|\mathcal{D}_{\min}|}.$$

Because the Gaussian perturbation is added before normalization, the released vector $g_{i,t}$ has noise standard deviation

$$\sigma_{\text{release}} = \frac{C_g \sigma_g}{\sqrt{n}\, |\mathcal{D}_i|}.$$

Again, for conservative accounting, we upper bound using $|\mathcal{D}_{\min}|$, giving

$$\sigma_{\text{release}} = \frac{C_g \sigma_g}{\sqrt{n}\,|\mathcal{D}_{\min}|}.$$

For a single Gaussian mechanism with sensitivity Δ and noise standard deviation σ_{release}, the exact (ε, δ)-tradeoff under the hockey-stick divergence is

$$\delta(\varepsilon) = \Phi\left(-\frac{\varepsilon\sigma_{\text{release}}}{\Delta} + \frac{\Delta}{2\sigma_{\text{release}}}\right) - e^{\varepsilon}\Phi\left(-\frac{\varepsilon\sigma_{\text{release}}}{\Delta} - \frac{\Delta}{2\sigma_{\text{release}}}\right),$$

where Φ denotes the standard normal distribution function. Under T rounds with independent Gaussian perturbations, the corresponding composed expression becomes

$$\delta(\varepsilon) = \Phi\left(-\frac{\varepsilon\sigma_{\text{release}}}{\sqrt{T}\,\Delta} + \frac{\sqrt{T}\,\Delta}{2\sigma_{\text{release}}}\right) - e^{\varepsilon}\Phi\left(-\frac{\varepsilon\sigma_{\text{release}}}{\sqrt{T}\,\Delta} - \frac{\sqrt{T}\,\Delta}{2\sigma_{\text{release}}}\right).$$

Substituting the mechanism parameters

$$\Delta = \frac{2C_g}{|\mathcal{D}_{\min}|}, \qquad \sigma_{\text{release}} = \frac{C_g \sigma_g}{\sqrt{n}\,|\mathcal{D}_{\min}|},$$

yields

$$\delta(\varepsilon) = \Phi\left(\frac{\sqrt{nT}}{\sigma_g} - \frac{\varepsilon\sigma_g}{2\sqrt{nT}}\right) - e^{\varepsilon}\Phi\left(-\frac{\sqrt{nT}}{\sigma_g} - \frac{\varepsilon\sigma_g}{2\sqrt{nT}}\right).$$

Given a target pair (ε, δ), we determine the smallest σ_g satisfying the inequality

$$\delta(\varepsilon) \le \delta$$

by a one-dimensional numerical search, such as binary search over a suitably large interval.

A few remarks are worth making. First, because all clients participate at every round, there is no privacy amplification by subsampling in the present setting; the accounting above therefore corresponds directly to full participation. Second, the factor $1/\sqrt{n}$ appearing in the release noise scale arises solely from the client-side parameterization of the Gaussian perturbation in (2); it should not be interpreted as a consequence of subsampling or privacy amplification. Third, the hockey-stick divergence yields the exact privacy tradeoff for the Gaussian mechanism and is therefore tighter than moment-based upper bounds such as standard Rényi-DP conversions in many regimes.

Most importantly for the present paper, the privacy guarantee of DP-FedSOFIM is identical to that of the underlying DP-FedGD release mechanism. The server-side updates

$$M_t = \beta M_{t-1} + (1-\beta)G_t, \qquad H_t = (\rho I_d + M_t M_t^{\top})^{-1}, \qquad \theta_{t+1} = \theta_t - \eta H_t G_t$$

depend only on previously released privatized quantities and therefore preserve privacy by the post-processing theorem. Consequently, once the noise multiplier σ_g has been calibrated for the client-side Gaussian mechanism, the same (ε, δ)-DP guarantee carries over unchanged to DP-FedSOFIM.

F Privacy Analysis of the Gaussian Mechanism

This appendix derives the differential privacy parameters of the privatized client update defined in (2).

Sensitivity of the clipped client update Let $\mathcal{D}_i$ and $\mathcal{D}_i'$ be neighboring datasets under the replace-one adjacency of Definition 3.2, differing in exactly one record. Because each per-example gradient is clipped to have Euclidean norm at most C_g, the change in the summed gradient satisfies

$$\|S_{i,t}(\mathcal{D}_i) - S_{i,t}(\mathcal{D}_i')\|_2 \leq 2C_g.$$

Since the released update is normalized by the dataset size $|D_i|$, the ℓ_2-sensitivity of the released vector equals

$$\Delta_i = \frac{2C_g}{|D_i|}.$$

Gaussian mechanism Each client releases a privatized update obtained by adding Gaussian noise to the summed gradient prior to normalization:

$$g_{i,t} = \frac{1}{|D_i|}\big(S_{i,t} + E_{i,t}\big), \qquad E_{i,t} \sim \mathcal{N}\left(0, \frac{(C_g\sigma_g)^2}{n} I_d\right).$$

Equivalently, the released vector can be written as

$$g_{i,t} = \frac{1}{|D_i|} S_{i,t} + \zeta_{i,t}, \qquad \zeta_{i,t} \sim \mathcal{N}\left(0, \frac{(C_g\sigma_g)^2}{n|D_i|^2} I_d\right).$$

Thus the client update contains isotropic Gaussian noise with standard deviation

$$\frac{C_g\sigma_g}{\sqrt{n}\,|D_i|}.$$

Composition across rounds The above mechanism is applied independently at each training round. Let $(\varepsilon_t, \delta_t)$ denote the privacy guarantee of a single round. Applying the composition result of Theorem 4.37 yields the overall differential privacy guarantee for the full DP-FedSOFIM training procedure.

G IID Experimental Results

Tables 6–7 report test accuracy trajectories under the IID data partitioning scheme for CIFAR-10 and PathMNIST respectively. In the IID setting, data is partitioned uniformly at random across clients, eliminating label heterogeneity. Hyperparameters were retuned independently for the IID setting using the same grid as the non-IID experiments.

The IID results are broadly consistent with the non-IID findings: DP-FedSOFIM maintains competitive or superior performance across all privacy regimes. Notably, DP-FedGD and DP-FedAvg produce near-identical results in both settings, confirming that with a single local iteration ($K = 1$), FedAvg reduces to FedGD. DP-FedFC exhibits higher variance under IID at tight privacy budgets ($\varepsilon = 0.5, 1$), likely due to sensitivity of the covariance preconditioning step to the Gaussian noise scale.

Table 6: Test accuracy (%) on CIFAR-10 under IID partitioning across federated rounds. Results shown at 10-round intervals (mean ± std over 3 seeds). Best result per privacy regime is in **bold**.

Method	Federated Round						
	10	20	30	40	50	60	70
$\varepsilon = 0.5$							
DP-FedGD	38.56±1.96	49.61±1.17	54.36±1.20	57.04±1.20	58.79±0.86	60.13±0.82	61.02±0.35
DP-FedAvg	38.56±1.96	49.61±1.17	54.36±1.20	57.04±1.20	58.79±0.86	60.13±0.82	61.02±0.35

Continued on next page

Continued from previous page

Method	Federated Round						
	10	20	30	40	50	60	70
DP-FedFC	23.90±5.11	36.66±2.97	46.10±0.69	51.71±0.67	55.21±0.79	56.76±0.85	58.27±0.88
DP-SCAFFOLD	33.06±2.58	43.07±1.62	47.70±2.11	49.65±2.11	51.18±1.11	51.96±1.19	53.22±0.43
DP-FedAdam	35.78±2.54	48.05±1.25	53.85±0.68	56.68±1.04	58.89±0.88	59.90±0.77	**61.07±0.38**
DP-FedYogi	30.69±3.48	44.11±1.78	50.97±1.09	55.21±0.93	57.78±0.72	59.21±0.75	60.46±0.54
DP-FedSOFIM	**43.53±1.05**	**51.57±0.92**	**56.45±0.69**	**58.41±0.98**	**59.62±0.98**	**60.34±0.91**	60.34±0.61
$\varepsilon = 1$							
DP-FedGD	40.61±1.54	51.69±0.70	56.71±0.51	59.39±0.76	61.37±0.54	62.63±0.47	63.41±0.44
DP-FedAvg	40.61±1.54	51.69±0.70	56.71±0.51	59.39±0.76	61.37±0.54	62.63±0.47	63.41±0.44
DP-FedFC	23.22±5.51	35.48±3.85	46.44±1.21	53.32±0.97	57.49±1.31	59.80±1.05	61.37±0.69
DP-SCAFFOLD	45.61±1.87	53.17±0.52	56.78±0.46	57.54±0.54	57.76±1.31	58.79±0.95	58.89±0.43
DP-FedAdam	51.61±1.51	**60.21±0.67**	62.60±0.57	63.52±0.44	63.82±0.39	63.64±0.49	63.85±0.43
DP-FedYogi	43.82±7.72	55.46±2.39	59.29±1.00	60.31±0.43	60.13±0.82	59.80±1.18	59.18±1.13
DP-FedSOFIM	**53.36±0.72**	60.09±0.27	**62.80±0.53**	**63.72±0.77**	**64.24±0.53**	**64.61±0.45**	**64.39±0.55**
$\varepsilon = 5$							
DP-FedGD	41.55±1.39	52.73±0.44	57.65±0.14	60.50±0.51	62.35±0.48	63.44±0.48	64.30±0.51
DP-FedAvg	41.55±1.39	52.73±0.44	57.65±0.14	60.50±0.51	62.35±0.48	63.44±0.48	64.30±0.51
DP-FedFC	53.98±1.17	62.39±0.52	65.13±0.50	66.31±0.27	66.93±0.26	67.35±0.23	67.47±0.04
DP-SCAFFOLD	58.04±0.21	63.18±0.77	64.92±0.50	65.83±0.34	66.30±0.18	66.69±0.17	67.16±0.26
DP-FedAdam	42.03±5.69	58.58±2.51	63.04±0.85	66.43±0.78	67.28±0.27	68.02±0.06	67.94±0.15
DP-FedYogi	53.62±1.41	64.10±0.13	66.56±0.42	67.41±0.31	67.96±0.14	68.19±0.19	**68.39±0.13**
DP-FedSOFIM	**61.49±0.73**	**65.82±0.29**	**67.46±0.09**	**67.70±0.14**	**68.01±0.11**	**68.28±0.11**	68.11±0.08
$\varepsilon = 10$							
DP-FedGD	41.69±1.42	52.77±0.45	57.75±0.15	60.61±0.51	62.34±0.47	63.52±0.52	64.29±0.50
DP-FedAvg	41.69±1.42	52.77±0.45	57.75±0.15	60.61±0.51	62.34±0.47	63.52±0.52	64.29±0.50
DP-FedFC	23.02±5.11	36.41±3.12	47.30±0.99	53.66±1.34	57.57±1.04	59.89±0.73	61.50±0.79
DP-SCAFFOLD	59.05±0.28	64.05±0.69	65.81±0.33	66.63±0.25	67.09±0.31	67.46±0.31	67.84±0.24
DP-FedAdam	47.14±2.41	58.58±1.23	64.37±1.26	66.01±0.48	67.27±0.47	67.88±0.12	68.05±0.09
DP-FedYogi	45.48±2.66	57.83±1.56	63.90±1.45	65.80±0.56	67.37±0.47	67.62±0.18	68.12±0.40
DP-FedSOFIM	**61.68±0.76**	**66.13±0.36**	**67.73±0.13**	**68.26±0.17**	**68.56±0.04**	**68.87±0.11**	**69.00±0.20**

Table 7: Test accuracy (%) on PathMNIST under IID partitioning across federated rounds. Results shown at 10-round intervals (mean ± std over 3 seeds). Best result per privacy regime is in **bold**.

Method	Federated Round						
	10	20	30	40	50	60	70
$\varepsilon = 0.5$							
DP-FedGD	51.90±0.68	57.77±0.77	59.87±1.38	62.21±0.59	63.31±0.78	64.18±0.96	64.33±0.74
DP-FedAvg	51.90±0.68	57.77±0.77	59.87±1.38	62.21±0.59	63.31±0.78	64.18±0.96	64.33±0.74
DP-FedFC	24.73±5.66	47.06±6.77	55.71±4.63	60.99±1.90	62.92±0.43	64.26±0.45	65.33±0.43
DP-SCAFFOLD	50.51±0.39	56.79±1.46	57.73±2.24	60.72±0.22	61.73±0.67	62.05±1.00	62.12±0.59
DP-FedAdam	50.61±3.57	57.98±1.04	61.26±0.71	63.17±0.23	64.34±0.71	64.81±0.85	65.63±0.67
DP-FedYogi	43.56±4.08	56.38±0.98	58.77±1.22	61.62±0.81	63.10±0.86	64.18±0.83	64.76±0.89
DP-FedSOFIM	**54.49±2.68**	**60.34±2.30**	**63.85±0.84**	**64.86±1.73**	**66.44±0.70**	**66.38±1.14**	**67.44±0.52**
$\varepsilon = 1$							
DP-FedGD	52.36±1.19	58.05±0.64	60.39±1.13	62.49±0.93	63.64±0.98	64.77±0.91	65.14±0.58

Continued on next page

Continued from previous page

Method	Federated Round						
	10	20	30	40	50	60	70
DP-FedAvg	52.36±1.19	58.05±0.64	60.39±1.13	62.49±0.93	63.64±0.98	64.77±0.91	65.14±0.58
DP-FedFC	20.16±4.96	43.98±7.91	53.34±5.86	60.10±2.91	63.67±0.86	65.74±0.17	67.16±0.40
DP-SCAFFOLD	57.00±0.97	60.50±1.02	63.86±0.82	64.97±0.73	66.19±0.60	65.81±0.60	66.54±0.29
DP-FedAdam	58.73±2.14	**64.81±0.60**	**66.83±0.66**	**68.04±0.68**	**68.39±0.61**	**68.41±0.31**	68.91±0.42
DP-FedYogi	40.51±2.53	53.85±7.85	63.61±2.99	66.45±1.08	66.72±0.06	67.57±0.41	67.61±0.24
DP-FedSOFIM	**58.76±0.27**	64.67±0.47	66.40±0.46	67.31±0.60	67.69±0.48	68.07±0.61	**68.95±0.15**
$\varepsilon = 5$							
DP-FedGD	52.59±1.63	58.21±0.87	60.60±1.26	62.45±1.26	63.92±1.09	64.93±0.91	65.73±0.63
DP-FedAvg	52.59±1.63	58.21±0.87	60.60±1.26	62.45±1.26	63.92±1.09	64.93±0.91	65.73±0.63
DP-FedFC	59.55±0.71	64.68±1.15	67.17±0.62	68.50±0.27	69.21±0.37	69.69±0.38	70.03±0.34
DP-SCAFFOLD	61.15±1.27	65.32±0.55	67.34±0.29	68.39±0.05	69.04±0.14	69.50±0.26	69.81±0.40
DP-FedAdam	48.69±4.47	61.69±3.13	66.88±1.89	68.25±0.39	70.26±0.64	70.67±0.12	71.56±0.19
DP-FedYogi	56.46±1.59	64.47±1.57	67.22±0.20	69.05±0.20	70.06±0.50	70.36±0.24	70.76±0.29
DP-FedSOFIM	**62.76±2.37**	**67.25±2.01**	**69.31±0.83**	**69.68±0.31**	**70.34±0.69**	**71.30±0.15**	**71.65±0.14**
$\varepsilon = 10$							
DP-FedGD	52.57±1.62	58.20±0.92	60.61±1.31	62.43±1.32	63.89±1.08	64.98±0.90	65.80±0.68
DP-FedAvg	52.57±1.62	58.20±0.92	60.61±1.31	62.43±1.32	63.89±1.08	64.98±0.90	65.80±0.68
DP-FedFC	23.79±4.87	46.25±6.80	54.31±3.53	59.38±0.71	61.62±1.57	62.93±1.69	64.03±1.45
DP-SCAFFOLD	61.51±1.28	65.73±0.61	67.60±0.40	68.52±0.16	69.32±0.31	69.61±0.33	69.84±0.36
DP-FedAdam	57.62±3.14	62.45±5.07	67.95±0.44	66.90±0.88	69.32±0.78	69.80±0.49	70.73±0.21
DP-FedYogi	57.99±2.34	64.96±2.97	65.62±1.39	67.16±0.97	70.17±0.43	69.67±0.69	70.23±0.48
DP-FedSOFIM	**62.86±2.29**	**67.20±2.09**	**69.31±0.75**	**69.90±0.29**	**70.48±0.73**	**71.59±0.11**	**71.75±0.16**

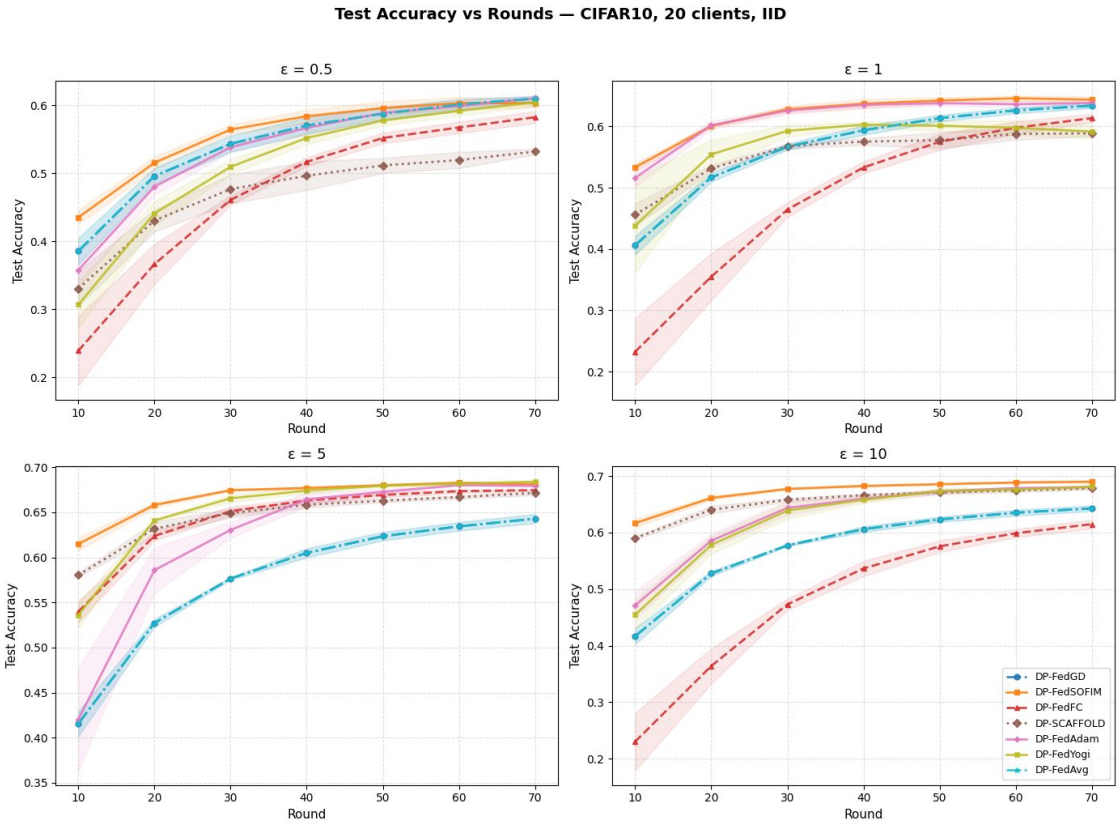

Figure 3: Convergence trajectories on CIFAR-10, 20 clients, IID across privacy regimes.

H Ablation Study and Hyperparameter Sensitivity

H.1 Component Ablation

Tables 8–9 and Figures 5–6 report the contribution of each component of DP-FedSOFIM under the non-IID (Dirichlet $\alpha = 0.5$) setting on CIFAR-10 and PathMNIST respectively, with 20 clients. We compare three variants: (i) *Grad-only*, which applies no EMA and no Sherman-Morrison (SM) preconditioning, reducing to standard DP-FedGD; (ii) *EMA-only*, which applies exponential moving average accumulation of gradients but omits the SM curvature correction; and (iii) *DP-FedSOFIM (full)*, the complete method.

At tight privacy budgets ($\varepsilon = 0.5$), all three variants perform comparably on CIFAR-10, as the noise scale dominates any curvature signal; on PathMNIST the EMA component provides a small lift even at this regime. The benefit of SM preconditioning is most pronounced at $\varepsilon = 1$, where the full method outperforms EMA-only by $\sim$7 points on CIFAR-10 and $\sim$7 points on PathMNIST. At $\varepsilon \geq 5$, EMA-only largely recovers the performance of the full method, suggesting the curvature correction contributes primarily under high noise.

Table 8: Ablation study on CIFAR-10, 20 clients, Non-IID (Dirichlet $\alpha = 0.5$). Final test accuracy (%) at round 70 (mean $\pm$ std over 3 seeds). Best result per privacy regime is in **bold**.

Method	$\varepsilon = 0.5$	$\varepsilon = 1$	$\varepsilon = 5$	$\varepsilon = 10$
Grad-only (no EMA, no SM)	58.72±1.02	52.46±2.35	57.17±0.67	54.73±1.64
EMA-only (no SM)	**59.04±0.78**	54.03±1.91	**67.97±0.15**	68.45±0.02
DP-FedSOFIM (full)	58.43±0.46	**61.35±0.94**	67.32±0.19	**68.60±0.09**

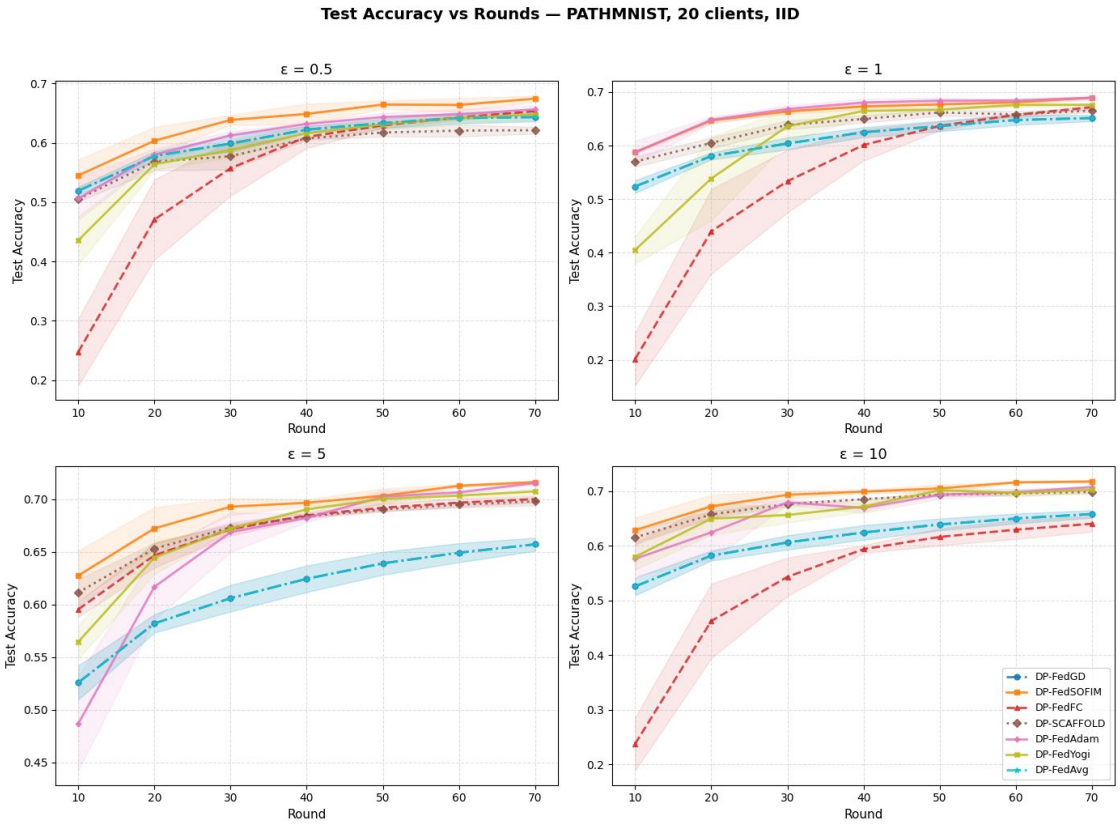

Figure 4: Convergence trajectories on PathMNIST, 20 clients, IID across privacy regimes.

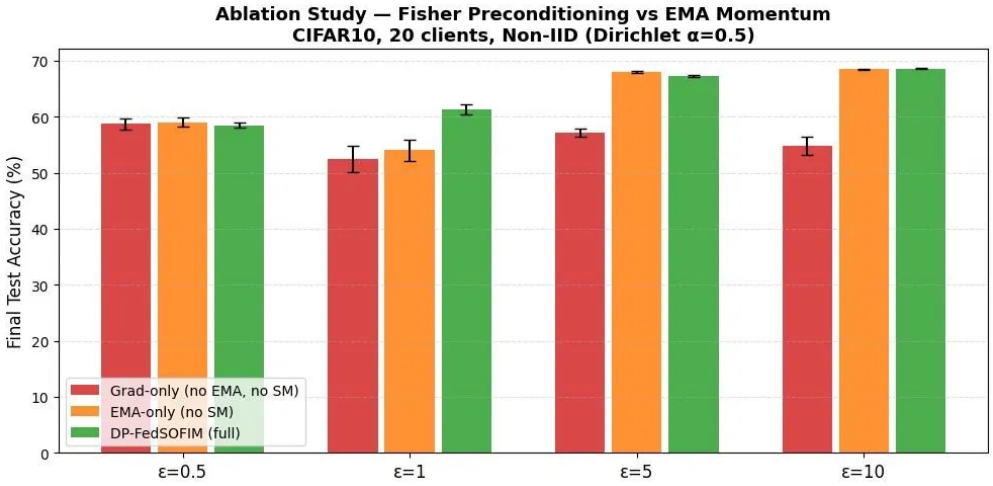

Figure 5: Ablation study — Fisher preconditioning vs. EMA momentum on CIFAR-10, 20 clients, Non-IID (Dirichlet $\alpha = 0.5$). Final test accuracy at round 70 across privacy regimes.

H.2 Sensitivity to ρ

Table 10 report final test accuracy at round 70 as a function of the EMA decay parameter $\rho \in \{0.01, 0.1, 0.5, 1.0, 2.0, 5.0, 10.0\}$ on CIFAR-10 and PathMNIST respectively, 20 clients, Non-IID (Dirichlet $\alpha = 0.5$), at $\varepsilon \in \{1, 5\}$.

Table 9: Ablation study on PathMNIST, 20 clients, Non-IID (Dirichlet $\alpha = 0.5$). Final test accuracy (%) at round 70 (mean ± std over 3 seeds). Best result per privacy regime is in **bold**.

Method	$\varepsilon = 0.5$	$\varepsilon = 1$	$\varepsilon = 5$	$\varepsilon = 10$
Grad-only (no EMA, no SM)	64.41±0.99	60.25±2.66	55.07±1.96	53.33±3.97
EMA-only (no SM)	**65.66±1.14**	61.67±2.82	70.77±0.09	70.94±0.08
DP-FedSOFIM (full)	65.24±1.40	**68.48±0.63**	**71.32±0.20**	**71.63±0.20**

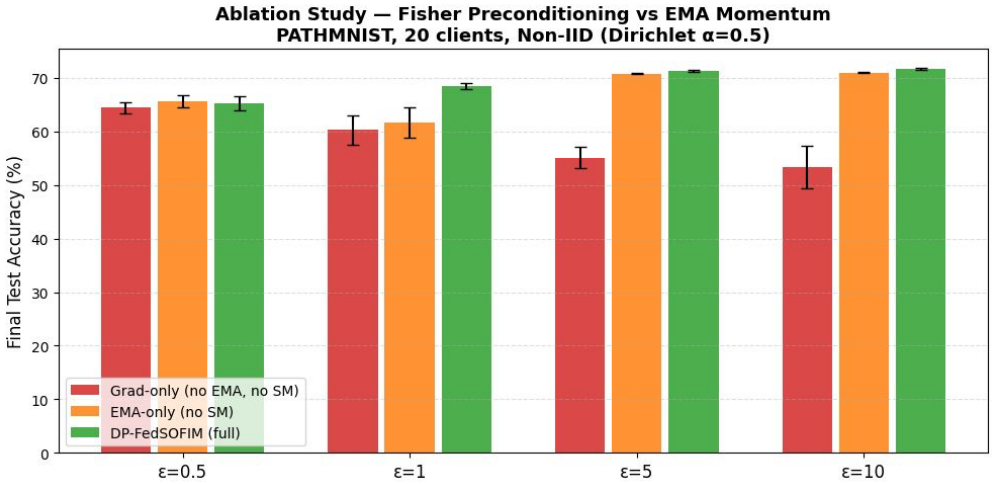

Figure 6: Ablation study — Fisher preconditioning vs. EMA momentum on PathMNIST, 20 clients, Non-IID (Dirichlet $\alpha = 0.5$). Final test accuracy at round 70 across privacy regimes.

On CIFAR-10, performance peaks near $\rho = 1.0$ at $\varepsilon = 5$ and increases monotonically with ρ at $\varepsilon = 1$, where higher momentum better suppresses the large Gaussian noise. On PathMNIST, the optimal ρ shifts to 0.5 at $\varepsilon = 5$ and again to 10.0 at $\varepsilon = 1$, consistent with the noise-dominance interpretation. In both cases the method is robust across the range $\rho \in [0.5, 5.0]$ at $\varepsilon = 5$, with degradation only at the extremes. We select $\rho = 1.0$ as the default for $\varepsilon \geq 5$ and $\rho = 10.0$ for tight privacy budgets.

Table 10: ρ sensitivity of DP-FedSOFIM. Final test accuracy (%) at round 70 (mean ± std over 3 seeds), Non-IID (Dirichlet $\alpha = 0.5$), $n = 20$ clients. Best result per ε is in **bold**.

	CIFAR-10		**PathMNIST**	
ρ	$\varepsilon = 1$	$\varepsilon = 5$	$\varepsilon = 1$	$\varepsilon = 5$
0.01	51.08±1.39	59.37±1.72	54.37±3.60	49.69±5.28
0.1	51.36±1.35	63.02±0.43	57.75±2.63	63.74±5.81
0.5	52.18±1.47	67.32±0.19	61.13±2.41	**71.32±0.20**
1.0	53.11±1.39	**68.03±0.05**	61.49±1.71	70.92±0.13
2.0	54.57±1.39	67.55±0.22	62.32±0.50	69.83±0.22
5.0	57.94±1.17	65.56±0.53	66.92±1.43	67.71±0.14
10.0	**61.35±0.94**	61.99±0.62	**68.48±0.63**	64.94±0.05