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 second-order optimization method for DP-FL. The method constructs an online regularized proxy for the Fisher information matrix at the server using only privatized aggregated gradients, capturing useful curvature information without requiring 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 show that **DP-FedSOFIM** converges faster and consistently achieves higher accuracy than DP-FedGD, DP-SCAFFOLD, and DP-FedFC across a range of privacy budgets, with particularly pronounced gains under stringent privacy constraints.

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) remains plagued by a fundamental convergence privacy tension. 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 is far from straightforward. 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 demonstrate that DP-FedSOFIM consistently outperforms all baselines, DP-FedGD, DP-FedFC, and DP-SCAFFOLD, across all privacy budgets and both client scales ($K \in \{20, 100\}$). On CIFAR-10, we observe absolute accuracy improvements over DP-FedGD of up to +4.05% in the non-private setting and +1.26% to +1.45% under stringent privacy ($\epsilon = 1$). On PathMNIST, the gains are substantially larger, reaching up to +4.86% at $\epsilon = 5$ with 100 clients, highlighting the particular benefit of curvature-aware preconditioning on ill-conditioned medical imaging tasks. Notably, DP-FedSOFIM maintains superiority even under the most stringent privacy constraints ($\epsilon = 0.5$), and converges faster than all baselines, surpassing DP-FedGD's final accuracy up to 20 rounds earlier.

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.

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.

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

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

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:

$$\bar{g}(x, y) = g(x, y) \cdot \min\left(1, \frac{C_g}{\|g(x, y)\|_2}\right). \tag{2}$$

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

$$g_{i,t} = \frac{1}{|\mathcal{D}_i|}\left(S_{i,t} + E_{i,t}\right), \qquad E_{i,t} \sim \mathcal{N}\left(0, \frac{(C_g\sigma_g)^2}{n} I_d\right). \tag{3}$$

The noise multiplier σ_g controls the privacy–utility tradeoff.

Server aggregation. The server aggregates client updates as

$$G_t = \frac{1}{n}\sum_{i=1}^{n} g_{i,t}. \tag{4}$$

The global model is updated via

$$\theta_{t+1} = \theta_t - \eta_t G_t. \tag{5}$$

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 4.7. 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 4.7 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

$$M_t = \beta M_{t-1} + (1-\beta) G_t, \qquad M_{-1} = \mathbf{0}_d, \quad \beta \in [0,1). \tag{6}$$

This momentum buffer stabilizes curvature estimation by smoothing stochastic noise introduced by both sampling variability and privacy perturbations.

We construct the curvature proxy

$$\widehat{\mathcal{I}}_t = M_t M_t^\top + \rho I_d, \tag{7}$$

where $\rho > 0$ ensures positive definiteness and numerical stability.

The rank-one structure is motivated by empirical observations in stochastic optimization that gradient covariance matrices often exhibit a strongly spiked spectrum, with a small number of dominant eigenvalues capturing most of the curvature information. In such regimes, the leading eigenvector of the empirical Fisher matrix frequently aligns closely with the running gradient direction, a phenomenon documented in empirical studies of natural-gradient and Fisher-based optimization methods (Martens, 2020). The exponential moving average M_t therefore provides a stable estimate of this dominant direction across rounds. The resulting proxy $M_t M_t^\top$ captures the principal curvature component while avoiding the $O(d^2)$ storage and computation

required for full Fisher approximations. Preconditioning by this low-rank Fisher proxy rescales gradient updates along the dominant curvature direction. If the anisotropic preconditioning induced by $\widehat{\mathcal{I}}_t^{-1}$ is removed, the update reduces to a scaled first-order gradient step. The momentum buffer M_t therefore serves primarily to construct a curvature proxy rather than directly driving the optimization direction.

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

$$H_t = \widehat{\mathcal{I}}_t^{-1} = \frac{1}{\rho} I_d - \frac{M_t M_t^\top}{\rho^2 + \rho \|M_t\|_2^2}, \tag{8}$$

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.

Preconditioned server update. The global model is updated via

$$\theta_{t+1} = \theta_t - \eta_t H_t G_t. \tag{9}$$

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.

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 e actly those defined in Section 3; in particular, the server receives the privatized aggregated gradient

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

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

Algorithm 1 DP-FedSOFIM

Input: θ_0, learning rates η_t, 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
  for each client $i \in [n]$ **do**
    for each $(x, y) \in \mathcal{D}_i$ **do**
      $g(x, y) = \nabla \ell(\theta_t; x, y)$
      $\bar{g}(x, y) = g(x, y) / \max(1, \|g(x, y)\|_2 / C_g)$
    end for
    $S_{i,t} = \sum \bar{g}(x, y)$
    Sample $E_{i,t} \sim \mathcal{N}(0, (C_g \sigma_g)^2 / n I_d)$
    $g_{i,t} = (S_{i,t} + E_{i,t}) / |\mathcal{D}_i|$
  end for
  Server side
  $G_t = \frac{1}{n} \sum_{i=1}^{n} g_{i,t}$
  $M_t = \beta M_{t-1} + (1 - \beta) G_t$
  $H_t = \frac{1}{\rho} I_d - \frac{M_t M_t^\top}{\rho^2 + \rho \|M_t\|^2}$
  $\theta_{t+1} = \theta_t - \eta_t H_t G_t$
end for

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

$$g_{\text{clip}}(\theta_t) := \frac{1}{n} \sum_{i=1}^{n} \frac{1}{|\mathcal{D}_i|} \sum_{(x,y) \in \mathcal{D}_i} \bar{g}(x, y), \tag{10}$$

where $\bar{g}(x, y)$ denotes the clipped per-example gradient defined in (2). The server aggregate can therefore be written as

$$G_t = g_{\text{clip}}(\theta_t) + \xi_t = \nabla F(\theta_t) + \zeta_t + \xi_t, \tag{11}$$

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 (3), the noise term admits the explicit representation

$$\xi_t = \frac{1}{n} \sum_{i=1}^{n} \frac{E_{i,t}}{|\mathcal{D}_i|}, \tag{12}$$

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

$$\mathbb{E}[\xi_t \xi_t^\top \mid \mathcal{F}_t] = \nu_t^2 I_d, \tag{13}$$

where

$$\nu_t^2 = \frac{(C_g \sigma_g)^2}{n^3} \sum_{i=1}^{n} \frac{1}{|\mathcal{D}_i|^2}. \tag{14}$$

Lemma 4.1 (Conditional mean under preconditioning)**.** *Let*

$$G_t = g_{\text{clip}}(\theta_t) + \xi_t, \qquad H_t = (\rho I_d + M_t M_t^\top)^{-1}.$$

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 (11),

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

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

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

□

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:

$$\|\nabla F(\theta) - \nabla F(\theta')\|_2 \le L\|\theta - \theta'\|_2, \qquad \forall\, \theta, \theta' \in \mathbb{R}^d. \tag{15}$$

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

$$F(\theta + \Delta) \le F(\theta) + \langle \nabla F(\theta), \Delta \rangle + \frac{L}{2}\|\Delta\|_2^2. \tag{16}$$

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

$$F(\theta') \ge F(\theta) + \langle \nabla F(\theta), \theta' - \theta \rangle + \frac{\mu}{2}\|\theta' - \theta\|_2^2, \qquad \forall\, \theta, \theta' \in \mathbb{R}^d. \tag{17}$$

In particular,

$$\|\nabla F(\theta)\|_2^2 \ge 2\mu\big(F(\theta) - F(\theta^\star)\big), \qquad \forall\, \theta \in \mathbb{R}^d, \tag{18}$$

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

$$\|\zeta_t\|_2 \le \zeta_{\max}, \qquad \forall\, t \ge 0. \tag{19}$$

Assumption 4.6 (Bounded gradient noise)**.** Let ξ_t denote the stochastic noise defined in (12). There exists a constant $\nu^2 < \infty$ such that

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

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

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

In the common case where all clients have equal dataset size $|\mathcal{D}_i| = m$, this simplifies to

$$\nu_t^2 = \frac{(C_g\sigma_g)^2}{n^2 m^2}.$$

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

$$\|\nabla F(\theta_t)\|_2 \le G_{\max}, \qquad \forall\, t \ge 0. \tag{20}$$

Remark 4.9 (On Assumption 4.8)*.* 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.

The following lemma shows that the DP noise mechanism satisfies the bounded-noise assumption used in the optimization analysis.

Lemma 4.10 (Uniform bound on DP noise variance)**.** *Let ξ_t denote the noise defined in* (12)*. Assume that client dataset sizes satisfy*

$$|\mathcal{D}_i| \ge m_{\min} > 0 \quad \textit{for all } i.$$

Then the variance parameter in (14) *satisfies*

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

Consequently,

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

Proof. From (14),

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

Since $|\mathcal{D}_i| \ge m_{\min}$,

$$\frac{1}{|\mathcal{D}_i|^2} \le \frac{1}{m_{\min}^2}.$$

Therefore

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

Substituting into the expression for ν_t^2 gives

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

The bound on the conditional second moment follows from

$$\mathbb{E}[\|\xi_t\|_2^2 \mid \mathcal{F}_t] = \operatorname{tr}\left(\mathbb{E}[\xi_t\xi_t^\top|\mathcal{F}_t]\right) = d\nu_t^2 \le d\nu^2.$$

□

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 $t \ge 0$,*

$$\|g_{\text{clip}}(\theta_t)\|_2 \le C_g. \tag{21}$$

Consequently,

$$\mathbb{E}\big[\|G_t\|_2^2 \mid \mathcal{F}_t\big] = \|g_{\text{clip}}(\theta_t)\|_2^2 + d\nu_t^2 \le C_g^2 + d\nu^2. \tag{22}$$

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

$$\mathbb{E}\big[\|M_t\|_2^2 \mid \mathcal{F}_t\big] \le \beta\|M_{t-1}\|_2^2 + (1-\beta)C_g^2 + (1-\beta)^2 d\nu^2. \tag{23}$$

Define

$$u_t := \mathbb{E}\|M_t\|_2^2. \tag{24}$$

Then

$$u_t \le \beta u_{t-1} + (1-\beta)C_g^2 + (1-\beta)^2 d\nu^2, \qquad t \ge 0, \tag{25}$$

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

$$u_t \le \bar{M}^2 := C_g^2 + (1-\beta)d\nu^2, \qquad t \ge 0. \tag{26}$$

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 $t \ge 0$,*

$$H_t = \frac{1}{\rho} I_d - \frac{M_t M_t^\top}{\rho(\rho + \|M_t\|_2^2)}. \tag{27}$$

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

$$\|H_t v\|_2 \le \frac{1}{\rho}\|v\|_2, \tag{28}$$

$$\|H_t v\|_2^2 \le \frac{1}{\rho^2}\|v\|_2^2, \tag{29}$$

$$v^\top H_t v \ge \frac{1}{\rho}\|v\|_2^2 - \frac{\|M_t\|_2^2}{\rho^2}\|v\|_2^2. \tag{30}$$

In particular, for $v = \nabla F(\theta_t)$,

$$\nabla F(\theta_t)^\top H_t \nabla F(\theta_t) \geq \frac{1}{\rho}\|\nabla F(\theta_t)\|_2^2 - \frac{G_{\max}^2}{\rho^2}\|M_t\|_2^2. \tag{31}$$

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.

Lemma 4.15 (Conditional one-step descent)**.** *Fix any* $\tau_1, \tau_2 > 0$*, and define*

$$c_\nabla := \frac{1}{\rho} - \frac{\tau_1 + \tau_2}{2}. \tag{32}$$

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

$$\begin{aligned}\mathbb{E}[F(\theta_{t+1}) \mid \mathcal{F}_t] \leq F(\theta_t) - \eta c_\nabla \|\nabla F(\theta_t)\|_2^2 + \frac{\eta G_{\max}^2}{\rho^2}\mathbb{E}\big[\|M_t\|_2^2 \mid \mathcal{F}_t\big] \\ + \frac{\eta\,\zeta_{\max}^2}{2\tau_1\rho^2} + \frac{\eta\, d\,\nu^2}{2\tau_2\rho^2} + \frac{L\eta^2}{2\rho^2}\big(C_g^2 + d\nu^2\big).\end{aligned} \tag{33}$$

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 4.16 (Uniform unconditional one-step descent)**.** *Fix* $\tau_1, \tau_2 > 0$ *and define* c_∇ *as in* (32)*. Under Assumptions 4.2, 4.5, 4.6, and 4.8, for every* $t \geq 0$,

$$\mathbb{E}[F(\theta_{t+1})] \leq \mathbb{E}[F(\theta_t)] - \eta c_\nabla\, \mathbb{E}\|\nabla F(\theta_t)\|_2^2 + \Gamma, \tag{34}$$

where

$$\Gamma := \frac{\eta G_{\max}^2}{\rho^2}\bar{M}^2 + \frac{\eta\,\zeta_{\max}^2}{2\tau_1\rho^2} + \frac{\eta\, d\,\nu^2}{2\tau_2\rho^2} + \frac{L\eta^2}{2\rho^2}\big(C_g^2 + d\nu^2\big), \tag{35}$$

and $\bar{M}^2$ *is given by* (26)*.*

Proofs of both the above lemmas are in Appendix A.

4.5 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 4.17 (Linear convergence to a bias-and-noise neighborhood)**.** *Fix* $\tau_1, \tau_2 > 0$ *such that*

$$c_\nabla = \frac{1}{\rho} - \frac{\tau_1 + \tau_2}{2} > 0. \tag{36}$$

Suppose Assumptions 4.2, 4.3, 4.5, 4.6, and 4.8 hold. Define

$$r := 1 - 2\mu\eta c_\nabla. \tag{37}$$

If $r \in (0, 1)$*, equivalently* $0 < \eta < 1/(2\mu c_\nabla)$*, then for every* $T \geq 0$,

$$\mathbb{E}[F(\theta_T) - F(\theta^\star)] \leq r^T\big(F(\theta_0) - F(\theta^\star)\big) + \frac{1 - r^T}{1 - r}\,\Gamma \tag{38}$$

$$\leq r^T\big(F(\theta_0) - F(\theta^\star)\big) + \frac{\Gamma}{2\mu\eta c_\nabla}, \tag{39}$$

where Γ *is defined in* (35)*.*

Remark 4.18 (Interpretation of the error floor). The limiting neighborhood in (39) 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 4.19 (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.,

$$\tau_1 + \tau_2 < \frac{2}{\rho}.$$

In practice one may choose τ_1 and τ_2 as fixed small constants satisfying this condition.

4.6 Extension to the Polyak–Łojasiewicz Condition

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

Assumption 4.20 (Polyak–Łojasiewicz condition)**.** There exists $\mu_{\mathrm{PL}} > 0$ such that for all $\theta \in \mathbb{R}^d$,

$$\frac{1}{2}\|\nabla F(\theta)\|_2^2 \geq \mu_{\mathrm{PL}}\big(F(\theta) - F(\theta^\star)\big). \tag{40}$$

Remark 4.21. The PL condition is strictly weaker than strong convexity. In particular, μ-strong convexity implies (40) with $\mu_{\mathrm{PL}} = \mu$, but the converse need not hold.

Theorem 4.22 (DP-FedSOFIM under the PL condition)**.** *Fix $\tau_1, \tau_2 > 0$ such that $c_\nabla > 0$ in* (36)*. Suppose Assumptions 4.2, 4.5, 4.6, 4.8, and 4.20 hold. Define*

$$r_{\mathrm{PL}} := 1 - 2\mu_{\mathrm{PL}}\eta c_\nabla. \tag{41}$$

If $r_{\mathrm{PL}} \in (0,1)$, equivalently $0 < \eta < 1/(2\mu_{\mathrm{PL}}c_\nabla)$, then for every $T \geq 0$,

$$\mathbb{E}[F(\theta_T) - F(\theta^\star)] \leq r_{\mathrm{PL}}^T\big(F(\theta_0) - F(\theta^\star)\big) + \frac{1 - r_{\mathrm{PL}}^T}{1 - r_{\mathrm{PL}}}\,\Gamma \tag{42}$$

$$\leq r_{\mathrm{PL}}^T\big(F(\theta_0) - F(\theta^\star)\big) + \frac{\Gamma}{2\mu_{\mathrm{PL}}\eta c_\nabla}, \tag{43}$$

where Γ is defined in (35)*.*

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

Remark 4.23 (Relationship between the strong-convexity and PL results). Since strong convexity implies the PL condition,Theorem 4.22 strictly generalizes Theorem 4.17. 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.

4.7 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 (3).

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 4.24 (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

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

Lemma 4.25 (Privacy of the aggregated gradient)**.** *Under Assumption 4.24, the aggregated gradient G_t is $(\varepsilon_t, \delta_t)$-differentially private.*

Proof. 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. □

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 4.26 (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.*

Proof. 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.

□

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 4.27 (End-to-end privacy of DP-FedSOFIM)**.** *Suppose Assumption 4.24 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

$$(T\varepsilon_0,\ T\delta_0)\text{-DP}.$$

Proof. By Lemma 4.25, the aggregated gradient G_t is $(\varepsilon_t, \delta_t)$-differentially private. Lemma 4.26 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.

□

Instantiating the privacy parameters. Appendix F derives the concrete privacy parameters for the Gaussian release used in (3) 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, heorem 4.27 implies that the SOFIM momentum and preconditioning steps incur no additional privacy loss.

Remark 4.28 (Privacy cost relative to the baseline method)*.* If DP-FedSOFIM and the baseline DP-FL algorithm use the same client-side clipping rule, noise multiplier, participation pattern, and privacy accountant, then both algorithms incur exactly the same privacy cost. The difference between the methods lies only in the server-side optimization step.

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 established in recent differentially private federated learning work (Andrew et al., 2021), we employ a pre-trained ResNet-20 feature extractor (frozen during training) for CIFAR-10 and a pre-trained feature extractor for PathMNIST, optimizing only the final linear classification head in both cases. This setup yields a low-dimensional parameter space conducive to second-order optimization while preserving the essential challenges of private federated learning.

Federated Setting. We simulate federated environments with $n \in \{20, 100\}$ clients, with full client participation in every round. Data is partitioned uniformly at random across clients (IID setting). Each client performs local gradient computation with batch size 64. Training proceeds for $T = 70$ federated rounds with a single local iteration per round, representing a communication-constrained scenario typical of cross-device federated learning.

Privacy Parameters. We evaluate five privacy regimes: no differential privacy (baseline upper bound), and

$$\varepsilon \in \{0.5, 1, 2, 5, 10\} \quad \text{with} \quad \delta = 10^{-5},$$

spanning from stringent ($\varepsilon = 0.5$) to moderate ($\varepsilon = 10$) privacy guarantees. The noise multiplier σ_g for each ε is calibrated using Hockey-Stick divergence composition (Zhu et al., 2022; Dong et al., 2022) to achieve the target privacy 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.

Hyperparameters. All methods use batch size 64 and momentum parameter $\beta = 0.9$ for exponential moving average where applicable. DP-FedSOFIM uses regularization parameter $\rho = 0.5$ for stabilizing the Fisher proxy inverse. Gradient clipping thresholds are tuned per method and dataset from $C_g \in \{5, 10\}$. Learning rates are determined via grid search over

$$\eta \in \{0.01, 0.05, 0.1, 0.2, 0.3, \ldots, 0.9, 1.0, 1.5, 2.0, \ldots, 10.0\},$$

with finer resolution of 0.1 increments in the range $[0.1, 0.9]$, independently for each method, dataset, client count, and privacy regime. The configuration yielding optimal final test accuracy is selected. For DP-SCAFFOLD, we additionally tune the number of local iterations and client and server learning rates independently. Final hyperparameters are reported in Table 1.

Table 1: Hyperparameter search grids for all methods. Fixed parameters shared across all methods: batch size 64, $T = 70$ rounds, $\delta = 10^{-5}$. For DP-FedSOFIM, $\rho = 0.5$ and $\beta = 0.9$ are fixed across all experiments.

Method	Hyperparameter	Search Grid	Grid Size
DP-FedGD	Learning rate η	$\{0.01, 0.05, 0.1, 0.2, 0.3, 0.5, 0.7, 1.0, 2.0, 5.0\}$	20
	Grad. clip C_g	$\{5, 10\}$	
DP-FedFC	Learning rate η	$\{0.01, 0.05, 0.1, 0.2, 0.3, 0.5, 0.7, 1.0, 2.0, 5.0\}$	20
	Grad. clip C_g	$\{5, 10\}$	
DP-FedSOFIM	Learning rate η	$\{0.01, 0.05, 0.1, 0.2, 0.3, 0.5, 0.7, 1.0, 2.0, 5.0\}$	20
	Grad. clip C_g	$\{5, 10\}$	
	Regularization ρ	0.5 (fixed)	
DP-SCAFFOLD	Local iterations	$\{1, 3, 5\}$	36
	Client lr η_c	$\{0.01, 0.05, 0.1\}$	
	Server lr η_s	$\{0.5, 1.0, 3.0, 5.0\}$	
	Grad. clip C_g	$\{5, 10\}$	

Baselines. We compare DP-FedSOFIM against three 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; **DP-FedFC** (Krouka et al., 2025), a second-order method that computes local feature covariance matrices at each client as an approximation to the Fisher Information Matrix, incurring $O(d^2)$ client-side memory; and **DP-SCAFFOLD** (Karimireddy et al., 2020), a variance-reduction method that uses control variates to correct for client drift. All baselines are calibrated to the same (ε, δ) privacy budget as DP-FedSOFIM using identical privacy accounting.

5.2 Results

Tables 2–3 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 2: Test accuracy (%) on CIFAR-10 across federated rounds for different privacy budgets. Results shown at 10-round intervals. Best result per privacy regime is in **bold**.

Method	Federated Round						
	10	20	30	40	50	60	70
(a) 20 clients							
$\varepsilon =$ *No DP*							
DP-FedGD	47.84	56.58	60.52	62.74	63.91	64.87	65.48
DP-FedFC	47.57	56.29	60.50	62.75	63.79	64.80	65.52
DP-SCAFFOLD	**60.52**	64.87	66.39	66.96	67.40	67.75	68.03
DP-FedSOFIM	60.32	**66.01**	**67.17**	**68.42**	**69.08**	**69.37**	**69.53**
$\varepsilon = 0.5$							
DP-FedGD	52.96	**59.25**	61.73	61.84	61.08	60.09	59.98
DP-FedFC	**53.51**	58.83	60.48	60.80	59.79	59.30	58.50
DP-SCAFFOLD	37.89	47.20	52.44	54.69	54.86	54.56	54.69
DP-FedSOFIM	50.76	59.20	**61.84**	**62.93**	**62.27**	**61.41**	**61.24**
$\varepsilon = 1$							
DP-FedGD	53.46	59.99	63.24	64.58	64.90	64.82	64.60
DP-FedFC	**54.37**	60.46	62.56	63.51	63.60	64.25	64.20
DP-SCAFFOLD	48.15	56.43	60.00	61.28	60.90	60.97	60.74
DP-FedSOFIM	53.07	**60.52**	**63.45**	**65.15**	**65.29**	**65.25**	**65.38**
$\varepsilon = 2$							
DP-FedGD	53.42	60.26	63.99	65.33	66.02	66.21	66.30
DP-FedFC	56.98	61.44	63.62	65.07	65.84	66.60	66.36
DP-SCAFFOLD	53.53	60.05	63.28	64.39	64.65	64.48	64.33
DP-FedSOFIM	**58.31**	**64.28**	**66.07**	**66.77**	**67.19**	**67.35**	**67.18**
$\varepsilon = 5$							
DP-FedGD	53.38	60.84	64.03	65.48	66.09	66.56	67.02
DP-FedFC	57.15	62.42	64.22	65.55	66.76	67.58	67.86
DP-SCAFFOLD	53.43	60.43	64.14	65.43	66.01	66.35	66.49
DP-FedSOFIM	**62.16**	**66.15**	**67.62**	**68.06**	**68.32**	**68.60**	**68.64**
$\varepsilon = 10$							
DP-FedGD	53.37	61.03	64.03	65.46	66.39	66.64	66.92
DP-FedFC	57.27	61.83	64.78	66.03	66.89	67.70	68.00
DP-SCAFFOLD	53.44	60.86	64.09	65.45	66.17	66.72	66.98
DP-FedSOFIM	**62.20**	**66.31**	**67.44**	**68.12**	**68.47**	**68.78**	**68.77**
(b) 100 clients							
$\varepsilon =$ *No DP*							
DP-FedGD	47.78	58.12	61.83	63.62	64.88	65.70	66.04
DP-FedFC	47.31	57.96	61.86	63.58	64.78	65.68	65.99
DP-SCAFFOLD	54.38	61.81	64.11	65.70	66.14	66.65	67.15
DP-FedSOFIM	**63.72**	**66.83**	**67.54**	**68.91**	**69.33**	**69.45**	**69.62**
$\varepsilon = 0.5$							
DP-FedGD	**37.15**	48.00	52.46	53.55	53.95	54.04	52.12

Continued on next page

Continued from previous page

Method	Federated Round						
	10	20	30	40	50	60	70
DP-FedFC	37.00	48.17	**52.57**	52.29	53.10	53.22	50.65
DP-SCAFFOLD	26.57	35.50	39.36	38.93	40.82	41.30	40.74
DP-FedSOFIM	36.41	**48.78**	52.46	**54.36**	**54.84**	**55.16**	**54.61**
$\varepsilon = 1$							
DP-FedGD	**47.09**	**56.16**	59.43	60.56	60.81	60.84	60.08
DP-FedFC	46.70	55.47	**59.90**	60.61	60.59	61.33	59.75
DP-SCAFFOLD	30.50	42.89	48.66	49.68	51.21	51.97	51.88
DP-FedSOFIM	42.50	54.83	59.75	**60.89**	**61.69**	**61.34**	**61.53**
$\varepsilon = 2$							
DP-FedGD	54.39	61.03	64.08	64.57	64.54	64.73	64.13
DP-FedFC	47.65	57.04	61.21	63.10	63.63	64.48	**64.47**
DP-SCAFFOLD	47.01	56.15	58.54	59.32	59.65	59.59	58.80
DP-FedSOFIM	**57.80**	**63.61**	**64.52**	**64.96**	**64.88**	**64.09**	63.86
$\varepsilon = 5$							
DP-FedGD	53.80	61.27	63.94	65.42	65.47	66.34	66.58
DP-FedFC	48.12	57.69	61.85	63.81	64.31	65.29	65.83
DP-SCAFFOLD	48.16	57.91	61.43	63.17	64.04	64.13	64.13
DP-FedSOFIM	**61.30**	**65.79**	**66.84**	**67.15**	**66.99**	**66.27**	**66.15**
$\varepsilon = 10$							
DP-FedGD	53.95	61.50	63.98	65.52	65.86	66.41	67.04
DP-FedFC	55.95	62.42	64.82	65.85	66.20	66.92	67.39
DP-SCAFFOLD	48.41	58.20	62.15	63.80	64.69	65.07	65.34
DP-FedSOFIM	**62.32**	**66.06**	**67.67**	**67.75**	**67.76**	**67.55**	**67.50**

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

Method	Federated Round						
	10	20	30	40	50	60	70
(a) 20 clients							
$\varepsilon =$ *No DP*							
DP-FedGD	47.24	54.48	57.01	59.22	60.78	62.38	63.37
DP-FedFC	54.14	59.07	62.33	64.29	65.35	66.50	67.14
DP-SCAFFOLD	**52.40**	60.61	64.11	65.95	66.94	68.01	68.62
DP-FedSOFIM	27.34	**61.77**	**66.81**	**68.87**	**68.98**	**70.42**	**71.48**
$\varepsilon = 0.5$							
DP-FedGD	52.05	57.21	58.73	61.39	62.42	63.05	63.27
DP-FedFC	**53.51**	58.83	60.48	60.80	59.79	59.30	58.50
DP-SCAFFOLD	49.36	53.93	56.10	58.26	58.69	59.53	61.14

Continued on next page

Continued from previous page

Method	Federated Round						
	10	20	30	40	50	60	70
DP-FedSOFIM	52.60	**61.52**	**66.57**	**68.43**	**67.90**	**66.27**	**65.35**
$\varepsilon = 1$							
DP-FedGD	51.95	57.01	58.76	61.20	62.40	63.51	64.35
DP-FedFC	**54.37**	60.46	62.56	63.51	63.60	64.25	64.20
DP-SCAFFOLD	50.85	55.54	57.67	59.93	60.65	61.89	63.22
DP-FedSOFIM	52.45	**62.53**	**67.40**	**68.68**	**68.38**	**68.18**	**67.92**
$\varepsilon = 2$							
DP-FedGD	51.84	56.96	58.66	60.81	62.51	63.73	64.60
DP-FedFC	**56.98**	61.44	63.62	65.07	65.84	66.60	66.36
DP-SCAFFOLD	55.17	58.68	61.28	63.91	64.72	65.92	66.77
DP-FedSOFIM	52.31	**62.73**	**67.33**	**68.65**	**68.57**	**68.70**	**68.98**
$\varepsilon = 5$							
DP-FedGD	51.84	56.89	58.84	60.67	62.44	63.66	64.94
DP-FedFC	**57.15**	62.42	64.22	65.55	66.76	67.58	67.86
DP-SCAFFOLD	55.24	58.79	61.53	63.89	65.18	66.25	67.65
DP-FedSOFIM	52.14	**62.80**	**67.33**	**68.61**	**68.69**	**69.09**	**69.67**
$\varepsilon = 10$							
DP-FedGD	51.84	56.80	58.80	60.58	62.49	63.73	64.94
DP-FedFC	57.27	61.83	64.78	66.03	66.89	67.70	68.00
DP-SCAFFOLD	55.28	58.69	61.57	63.70	65.25	66.17	67.08
DP-FedSOFIM	**61.24**	**67.24**	**70.17**	**70.01**	**71.13**	**71.27**	**71.41**
(b) 100 clients							
$\varepsilon =$ *No DP*							
DP-FedGD	50.95	56.56	59.99	61.75	63.26	64.32	65.17
DP-FedFC	56.43	61.73	64.30	65.74	66.67	67.21	67.84
DP-SCAFFOLD	58.65	65.75	67.23	67.35	67.17	67.52	67.73
DP-FedSOFIM	**64.43**	**68.52**	**69.60**	**70.18**	**71.56**	**71.39**	**71.70**
$\varepsilon = 0.5$							
DP-FedGD	47.01	53.16	55.97	57.28	58.68	59.04	60.17
DP-FedFC	46.03	53.13	55.60	56.87	58.59	59.43	60.57
DP-SCAFFOLD	29.36	44.07	49.12	51.63	53.98	55.17	56.24
DP-FedSOFIM	**57.40**	**62.17**	**64.01**	**64.39**	**64.97**	**65.36**	**65.19**
$\varepsilon = 1$							
DP-FedGD	47.47	53.87	56.13	57.27	58.59	59.32	60.54
DP-FedFC	53.77	56.82	59.14	61.21	63.20	64.22	65.17
DP-SCAFFOLD	55.31	57.83	59.28	61.28	63.59	63.98	64.42
DP-FedSOFIM	**61.75**	**65.89**	**67.21**	**67.94**	**68.20**	**68.66**	**67.86**
$\varepsilon = 2$							
DP-FedGD	54.39	58.01	60.65	62.38	64.60	64.69	66.20
DP-FedFC	53.91	56.96	59.29	61.25	63.05	64.16	65.31
DP-SCAFFOLD	55.99	59.12	61.21	62.91	64.21	65.31	65.61
DP-FedSOFIM	**62.58**	**67.76**	**68.91**	**69.43**	**69.60**	**69.76**	**69.15**
$\varepsilon = 5$							
DP-FedGD	54.50	58.01	60.96	62.59	64.48	65.11	66.41
DP-FedFC	57.94	62.37	65.38	66.67	67.94	68.75	69.15

Continued on next page

Continued from previous page

Method	Federated Round						
	10	20	30	40	50	60	70
DP-SCAFFOLD	53.43	60.43	64.14	65.43	66.01	66.35	66.49
DP-FedSOFIM	**61.95**	**68.44**	**69.11**	**69.57**	**70.13**	**70.58**	**70.88**
$\varepsilon = 10$							
DP-FedGD	54.57	57.98	61.02	62.59	64.32	65.18	66.34
DP-FedFC	58.08	62.53	65.24	66.56	67.92	68.68	69.26
DP-SCAFFOLD	61.02	65.10	67.06	67.73	68.89	69.30	69.35
DP-FedSOFIM	**61.60**	**68.22**	**68.94**	**69.51**	**70.28**	**70.84**	**71.11**

5.3 Analysis

5.3.1 Phenomenon 1: Consistent Superiority of DP-FedSOFIM Across All Methods and Datasets

DP-FedSOFIM achieves the highest final accuracy across all four methods, both datasets, and all privacy budgets tested. On CIFAR-10 with 20 clients, DP-FedSOFIM outperforms DP-FedGD by +0.78% at $\varepsilon = 1$ to +4.05% in the non-private setting, and outperforms DP-FedFC by similar margins despite DP-FedFC also employing second-order information. With 100 clients on CIFAR-10, DP-FedSOFIM's advantage over DP-FedGD ranges from +1.45% at $\varepsilon = 1$ to +3.58% in the non-private setting, demonstrating robustness to federation scale.

The gains are substantially larger on PathMNIST, where the medical imaging domain appears to present a more ill-conditioned optimization landscape that benefits more from curvature-aware preconditioning. With 20 clients, DP-FedSOFIM outperforms DP-FedGD by +2.08% to +6.47% across $\varepsilon \in \{0.5, \ldots, 10\}$, and achieves a final accuracy of 71.48% in the non-private setting versus 68.62% for the next best method (DP-SCAFFOLD). With 100 clients, DP-FedSOFIM reaches 71.70% (No DP) and 65.19% ($\varepsilon = 0.5$), outperforming all baselines by large margins. Among the baselines, DP-FedFC is the closest competitor to DP-FedSOFIM at relaxed privacy budgets on PathMNIST (e.g., 100 clients, $\varepsilon = 5$: DP-FedFC 69.15% vs. DP-FedSOFIM 70.88%), while DP-FedGD and DP-SCAFFOLD consistently trail by larger margins across both datasets.

5.3.2 Phenomenon 2: Early-Round Instability Under Tight Privacy (CIFAR-10)

A characteristic pattern emerges on CIFAR-10 at $\varepsilon \in \{0.5, 1\}$: DP-FedSOFIM exhibits *lower* accuracy than DP-FedGD and DP-FedFC at round 10 (deficits of 2.20% and 0.39%, respectively, against DP-FedGD with 20 clients), but rapidly overtakes all baselines by round 30. This behavior is specific to CIFAR-10 and is not observed on PathMNIST, where DP-FedSOFIM leads or matches from early rounds across all privacy regimes except the non-private 20-client setting. At $\varepsilon = 2$ on CIFAR-10, DP-FedSOFIM already leads from round 10 (58.31% vs. 53.42% for DP-FedGD), indicating that the early instability threshold shifts with the noise level.

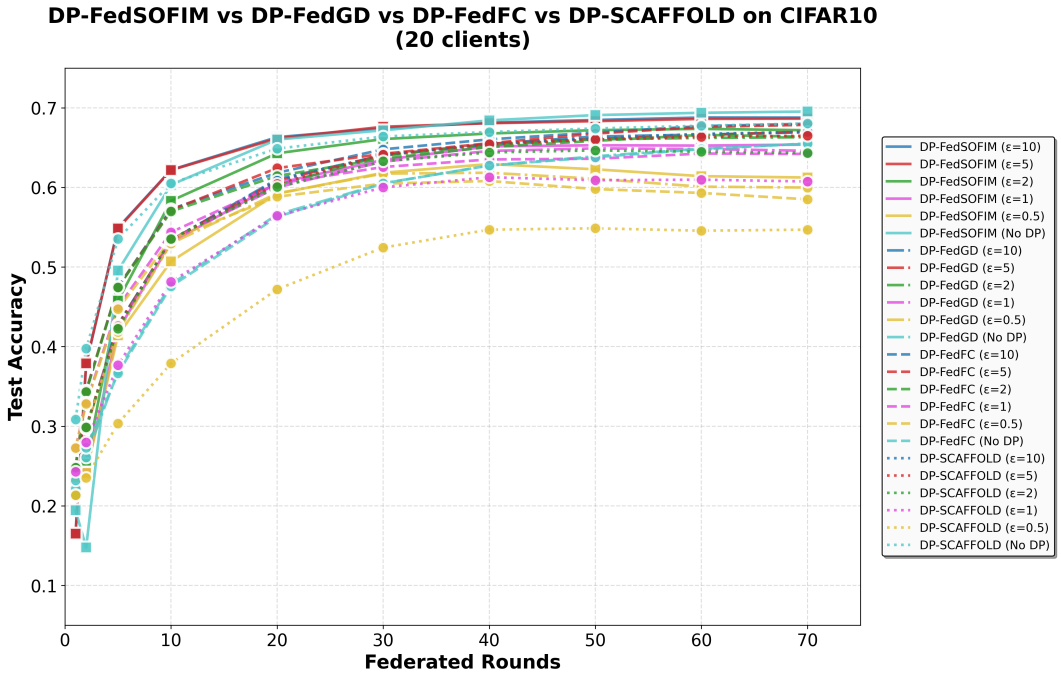
Mechanistic Explanation. The momentum buffer is initialized as $M_{-1} = 0_d$. Therefore the initial curvature proxy is purely regularization-driven,

$$\widehat{\mathcal{F}}_{-1} = \rho I_d.$$

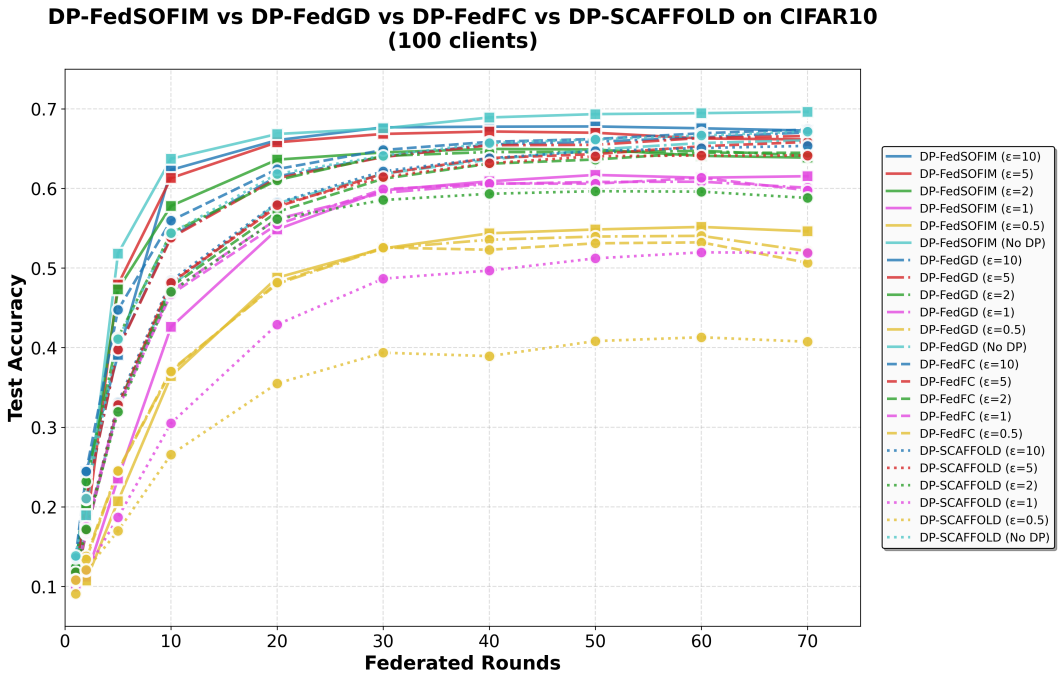
After the first communication round the buffer becomes $M_0 = (1 - \beta)G_0$, and the Fisher proxy begins to incorporate observed gradient information. The corresponding preconditioner

$$H_{-1} = \frac{1}{\rho} I_d$$

applies isotropic scaling without curvature adaptation. Under high noise ($\sigma_g \gg 1$ for tight ε), the early noisy gradients G_t corrupt the momentum buffer, leading to poorly conditioned Fisher proxy estimates in the initial rounds. The anisotropic rescaling amplifies noise in directions with spurious high curvature, temporarily degrading performance below the baselines. However, as M_t accumulates signal through exponential moving

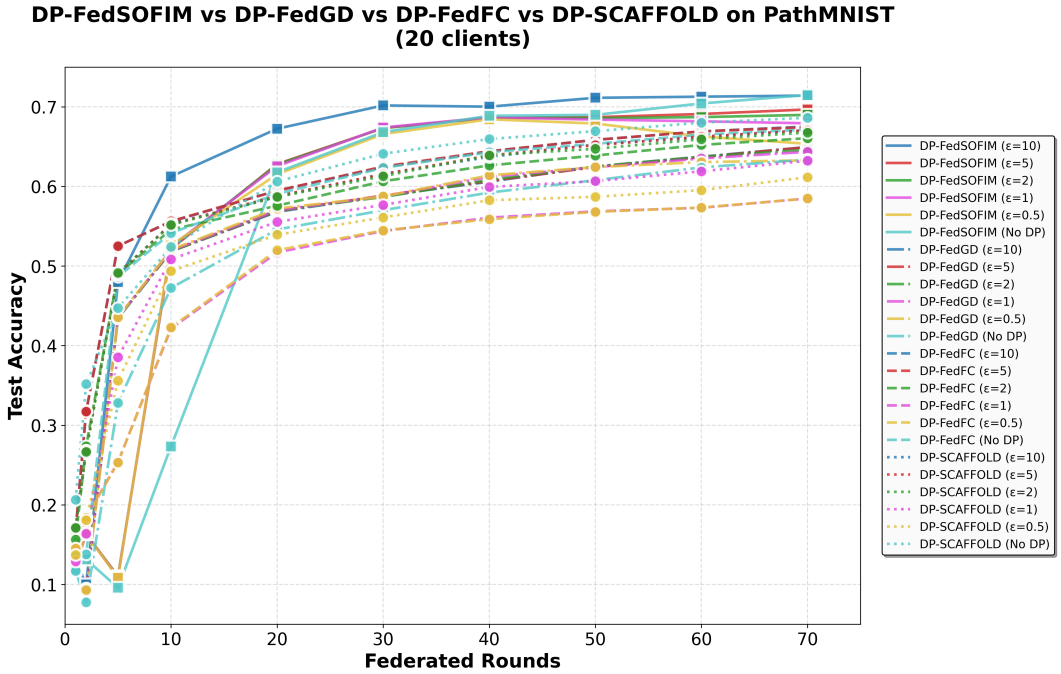


(a) CIFAR-10, 20 clients

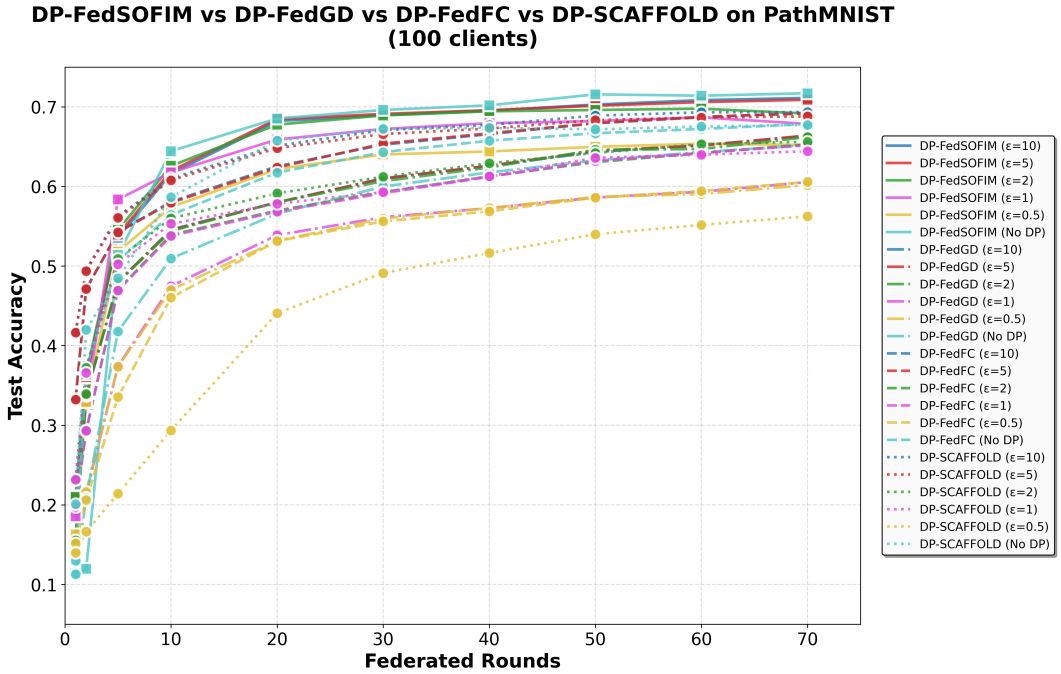


(b) CIFAR-10, 100 clients

Figure 1: Convergence trajectories on CIFAR-10 across privacy regimes. DP-FedSOFIM (solid lines) consistently outperforms all baselines. Under tight privacy ($\varepsilon \leq 2$), early deficits are overcome by round 30, while relaxed privacy ($\varepsilon \geq 5$) enables immediate dominance from round 10.



(a) PathMNIST, 20 clients



(b) PathMNIST, 100 clients

Figure 2: Convergence trajectories on PathMNIST across privacy regimes. DP-FedSOFIM (solid lines) demonstrates substantial improvements over all baselines. The pronounced early deficit in the non-private 20-client setting resolves rapidly by round 20, with DP-FedSOFIM achieving the highest final accuracy across all privacy budgets.

average (achieving a variance reduction factor of $\frac{1-\beta}{1+\beta} = \frac{1}{19}$ for $\beta = 0.9$), the Fisher proxy stabilizes by round 20. The preconditioner then correctly identifies the dominant gradient direction, applying aggressive step sizes proportional to

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

along high-curvature modes, enabling DP-FedSOFIM to rapidly overtake all baselines by round 30. The reason this instability is absent on PathMNIST is that the stronger curvature signal in the medical imaging feature space allows the momentum buffer to stabilize faster, even under high noise.

5.3.3 Phenomenon 3: Immediate Dominance Under Relaxed Privacy and at Scale

At $\varepsilon \in \{5, 10\}$ and in the non-private setting, DP-FedSOFIM dominates all baselines from round 10 onward across both datasets and both client counts. On CIFAR-10 with 20 clients, early leads over DP-FedGD range from +8.78% to +12.48% at round 10, substantially larger than the corresponding final-round advantages of +1.62% to +4.05%, indicating rapid early exploitation of curvature that partially consolidates as training progresses. On PathMNIST with 100 clients, this dominance is even more pronounced: at $\varepsilon = 1$, DP-FedSOFIM leads by +14.28% at round 10 (61.75% vs. 47.47% for DP-FedGD), maintaining a +7.32% advantage at round 70. DP-FedFC, while competitive at relaxed privacy on PathMNIST, consistently falls behind DP-FedSOFIM from round 10 onward, suggesting that server-side Fisher proxy preconditioning provides a more effective curvature estimate than client-side feature covariance at scale.

Beyond final accuracy, DP-FedSOFIM demonstrates *faster convergence* to near-optimal performance. Define the target accuracy as 95% of DP-FedGD's final accuracy. At $\varepsilon = 2$ on CIFAR-10 with 20 clients: **(i)** DP-FedGD reaches 62.99% (95% of 66.30%) at round 30, **(ii)** DP-FedSOFIM already exceeds 62.99% at round 20 (64.28%), and **(iii)** DP-FedSOFIM surpasses DP-FedGD's *final* accuracy of 66.30% by round 40 (66.77%), continuing to improve to 67.18% at round 70. This acceleration is critical in communication-constrained federated settings where minimizing the number of rounds directly reduces training time and energy consumption on client devices.

5.3.4 Phenomenon 4: SCAFFOLD's Sensitivity to Privacy Noise

DP-SCAFFOLD exhibits a distinct failure mode under tight privacy budgets that sets it apart from the other baselines. On CIFAR-10 with 100 clients at $\varepsilon = 0.5$, DP-SCAFFOLD achieves only 40.74% final accuracy, far below DP-FedGD (52.12%), DP-FedFC (50.65%), and DP-FedSOFIM (54.61%). Similarly, on CIFAR-10 with 20 clients at $\varepsilon = 0.5$, DP-SCAFFOLD reaches only 54.69% versus 61.24% for DP-FedSOFIM. This degradation arises because SCAFFOLD's variance reduction mechanism relies on accurate 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.

Interestingly, DP-SCAFFOLD recovers competitiveness at relaxed privacy budgets and on PathMNIST with 100 clients, where it achieves 69.35% at $\varepsilon = 10$, competitive with DP-FedFC (69.26%) but still below DP-FedSOFIM (71.11%). This suggests that SCAFFOLD's variance reduction is effective when the noise level is sufficiently low to preserve the integrity of control variates, but its tight privacy performance makes it unsuitable for applications requiring strong privacy guarantees. DP-FedSOFIM's server-side preconditioning, by contrast, degrades more gracefully under tight privacy, maintaining consistent advantages even at $\varepsilon = 0.5$.

5.3.5 Phenomenon 5: Dataset-Dependent Sensitivity to Curvature Under Differential Privacy

The results reveal a consistent asymmetry between the two datasets: DP-FedSOFIM achieves immediate dominance on PathMNIST across all *DP* privacy regimes, yet exhibits transient early-round instability on CIFAR-10 under tight privacy before recovering by round 30. With 100 clients at $\varepsilon = 0.5$, DP-FedSOFIM leads on PathMNIST from round 10 with 57.40% versus 47.01% for DP-FedGD (+10.39%), and at $\varepsilon = 1$ the early lead widens to +14.28% (61.75% vs. 47.47%). On CIFAR-10 under the same settings, DP-FedSOFIM trails slightly in early rounds before overtaking all baselines by round 30. We offer a geometric interpretation of this contrast, noting that it is qualitatively consistent with the anisotropic scaling induced by the preconditioner in Lemma 4.14 and with the bias-and-noise neighborhood convergence guarantee in Theorem 4.17.

Geometric Interpretation. The magnitude of benefit from Fisher proxy preconditioning depends on the condition number of the loss landscape: larger condition numbers imply greater anisotropy in gradient directions and thus larger potential gains from curvature-aware step scaling, as reflected in the operator and quadratic-form bounds of Lemma 4.14. The consistently larger gains observed on PathMNIST, particularly under tight privacy and at scale, suggest that the frozen ResNet-20 representation of medical imaging data may induce a more ill-conditioned effective loss landscape, where a smaller number of dominant curvature directions correspond to discriminative tissue-level features. Under this hypothesis, the momentum buffer M_t concentrates signal along these dominant directions more rapidly, enabling reliable preconditioning even under stringent privacy noise. In contrast, the comparatively modest gains on CIFAR-10 suggest a more diffuse effective curvature structure, where the gradient signal is spread across parameter dimensions and therefore more susceptible to corruption by privacy noise in early rounds.

Early Instability as a Diagnostic. This interpretation is further supported by the presence of early-round instability on CIFAR-10 but not on PathMNIST under DP regimes. When the effective curvature is concentrated, the momentum buffer stabilizes quickly and the preconditioner provides accurate anisotropic rescaling from early rounds. When the effective curvature is more diffuse, the momentum buffer requires additional rounds to accumulate sufficient signal, during which time the preconditioner may transiently amplify noise in poorly estimated directions. The exponential averaging in M_t implies that the momentum estimate achieves substantial variance suppression within roughly 15–20 rounds (with reduction factor $\frac{1-\beta}{1+\beta} = \frac{1}{19}$ for $\beta = 0.9$), after which DP-FedSOFIM consistently overtakes all baselines. The absence of this instability on CIFAR-10 under relaxed privacy ($\varepsilon \geq 5$) and on PathMNIST across all DP regimes is consistent with the hypothesis that the noise-to-curvature ratio governs the onset of early instability.

These observations suggest a practical guideline for deploying DP-FedSOFIM: the method is particularly well-suited to domains where the pretrained feature extractor induces a structured, anisotropic loss landscape, such as medical imaging and other specialized domains, where the curvature signal remains informative even under the high noise levels required by stringent privacy budgets.

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, DP-FedSOFIM consistently outperforms all baselines, DP-FedGD, DP-FedFC, and DP-SCAFFOLD, across both CIFAR-10 and PathMNIST datasets, two client scales (20 and 100), and all privacy budgets tested ($\epsilon \in \{0.5, 1, 2, 5, 10\}$ and the non-private setting). On CIFAR-10, gains over DP-FedGD range from +0.78% at $\epsilon = 1$ to +4.05% in the non-private setting, while on PathMNIST the advantages are substantially larger, reaching up to +4.47% at $\epsilon = 5$ with 100 clients, highlighting the benefit of curvature-aware preconditioning on more ill-conditioned medical imaging tasks. Our analysis identified five key phenomena: consistent final accuracy superiority across all methods and datasets; an early-round instability on CIFAR-10 under tight privacy that resolves by round 30 as the momentum buffer accumulates sufficient signal; immediate dominance from round 10 under relaxed privacy, enabling faster convergence and reduced communication overhead; and a characteristic failure mode of DP-SCAFFOLD under tight privacy budgets, where corrupted control variates amplify rather than reduce variance, rendering it unsuitable for strong privacy guarantees. 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 demonstrates a unique "catch-up" phenomenon: while tight privacy budgets ($\epsilon \leq 2$) initially introduce instability in the Fisher Information Matrix (FIM) estimate, the accumulation of signal in the momentum buffer allows the optimizer to rapidly overtake first-order baselines. This suggests 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.

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.

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.

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.

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.

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 4.11. Fix $t \geq 0$. By (2), 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 (21).

Next, by (11),

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

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 (21) 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

$$\mathbb{E}\big[\|M_t\|_2^2 \mid \mathcal{F}_t\big] = \|\beta M_{t-1} + (1 - \beta) g_{\text{clip}}(\theta_t)\|_2^2 + (1 - \beta)^2 \mathbb{E}[\|\xi_t\|_2^2 \mid \mathcal{F}_t]. \tag{44}$$

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

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

Substituting this and $\mathbb{E}[\|\xi_t\|_2^2 \mid \mathcal{F}_t] = d\nu^2$ into (44) proves (23).

Taking total expectation yields (25). To solve this recursion, let

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

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

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

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

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

This proves (26). □

Proof of Lemma 4.14. The identity (27) 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 (27).

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 (28) follows immediately:

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

Squaring both sides gives (29).

For the quadratic lower bound, use (27) directly:

$$\begin{aligned} v^\top H_t v &= \frac{1}{\rho}\|v\|_2^2 - \frac{(M_t^\top v)^2}{\rho(\rho + \|M_t\|_2^2)} \\ &\ge \frac{1}{\rho}\|v\|_2^2 - \frac{(M_t^\top v)^2}{\rho^2} \\ &\ge \frac{1}{\rho}\|v\|_2^2 - \frac{\|M_t\|_2^2 \|v\|_2^2}{\rho^2}, \end{aligned}$$

where the last step is Cauchy–Schwarz. This proves (30).

Finally, substituting $v = \nabla F(\theta_t)$ and invoking Assumption 4.8 yields (31). □

Proof of Lemma 4.15. Set

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

By L-smoothness, (16) gives

$$F(\theta_{t+1}) \leq F(\theta_t) + \langle \nabla F(\theta_t), \Delta_t \rangle + \frac{L}{2}\|\Delta_t\|_2^2. \tag{45}$$

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

$$F(\theta_{t+1}) \leq F(\theta_t) - \eta \langle \nabla F(\theta_t), H_t G_t \rangle + \frac{L\eta^2}{2}\|H_t G_t\|_2^2. \tag{46}$$

We bound the linear and quadratic terms separately.

Step 1: decomposition of the linear term.

Using (11),

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

Therefore

$$\langle \nabla F(\theta_t), H_t G_t \rangle = \nabla F(\theta_t)^\top H_t \nabla F(\theta_t) + \nabla F(\theta_t)^\top H_t \zeta_t + \nabla F(\theta_t)^\top H_t \xi_t. \tag{47}$$

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.

By Lemma 4.14,

$$\nabla F(\theta_t)^\top H_t \nabla F(\theta_t) \geq \frac{1}{\rho}\|\nabla F(\theta_t)\|_2^2 - \frac{G_{\max}^2}{\rho^2}\|M_t\|_2^2. \tag{48}$$

Step 3: control of the bias cross-term.

Using Cauchy–Schwarz, (28), and Assumption 4.5,

$$\left|\nabla F(\theta_t)^\top H_t \zeta_t\right| \leq \|\nabla F(\theta_t)\|_2 \, \|H_t \zeta_t\|_2 \leq \frac{\|\nabla F(\theta_t)\|_2 \, \|\zeta_t\|_2}{\rho} \leq \frac{\|\nabla F(\theta_t)\|_2 \, \zeta_{\max}}{\rho}. \tag{49}$$

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

$$\left|\nabla F(\theta_t)^\top H_t \zeta_t\right| \leq \frac{\tau_1}{2}\|\nabla F(\theta_t)\|_2^2 + \frac{\zeta_{\max}^2}{2\tau_1 \rho^2}. \tag{50}$$

Step 4: control of the noise cross-term.

Similarly,

$$\left|\nabla F(\theta_t)^\top H_t \xi_t\right| \leq \|\nabla F(\theta_t)\|_2 \, \|H_t \xi_t\|_2 \leq \frac{\|\nabla F(\theta_t)\|_2 \, \|\xi_t\|_2}{\rho}. \tag{51}$$

Applying Young's inequality with

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

gives

$$\left|\nabla F(\theta_t)^\top H_t \xi_t\right| \le \frac{\tau_2}{2}\|\nabla F(\theta_t)\|_2^2 + \frac{\|\xi_t\|_2^2}{2\tau_2\rho^2}. \tag{52}$$

Combining (47), (48), (50), and (52), we obtain the pathwise lower bound

$$\begin{aligned}\langle \nabla F(\theta_t), H_t G_t\rangle \ge \left(\frac{1}{\rho} - \frac{\tau_1+\tau_2}{2}\right)\|\nabla F(\theta_t)\|_2^2 - \frac{G_{\max}^2}{\rho^2}\|M_t\|_2^2 \\ - \frac{\zeta_{\max}^2}{2\tau_1\rho^2} - \frac{\|\xi_t\|_2^2}{2\tau_2\rho^2}.\end{aligned} \tag{53}$$

Step 5: upper bound on the quadratic term.

By (29),

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

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

$$\mathbb{E}\big[\|H_t G_t\|_2^2 \mid \mathcal{F}_t\big] \le \frac{1}{\rho^2}\mathbb{E}\big[\|G_t\|_2^2 \mid \mathcal{F}_t\big] \le \frac{C_g^2 + d\nu^2}{\rho^2}. \tag{54}$$

Step 6: combine the bounds and take conditional expectation.

Substitute (53) into (46):

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

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 (54) 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 4.16. Take total expectation in (33):

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

By Lemma 4.12,

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

Substituting this bound yields (34). □

Proof of Theorem 4.17. By Lemma 4.16,

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

By (18), 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

$$\mathbb{E}[F(\theta_{t+1}) - F(\theta^\star)] \le \big(1 - 2\mu\eta c_\nabla\big)\mathbb{E}[F(\theta_t) - F(\theta^\star)] + \Gamma. \tag{55}$$

Define

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

Then (55) 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 (38). Since $1 - r = 2\mu\eta c_\nabla$ and $1 - r^T \le 1$, we obtain (39). □

Proof of Theorem 4.22. Lemma 4.16 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 (40),

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

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

$$\mathbb{E}[F(\theta_{t+1}) - F(\theta^\star)] \le \big(1 - 2\mu_{\mathrm{PL}}\eta c_\nabla\big)\mathbb{E}[F(\theta_t) - F(\theta^\star)] + \Gamma. \tag{56}$$

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 (42); the simplified bound (43) follows from $1 - r_{\mathrm{PL}} = 2\mu_{\mathrm{PL}}\eta c_\nabla$. □

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 (2), 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.

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.

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 $1+v^\top A^{-1}u \neq 0$,

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

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 (3); 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 (3).

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

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

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 4.27 yields the overall differential privacy guarantee for the full DP-FedSOFIM training procedure.