Projection-Based Federated Learning on Compact Submanifolds with Geometry-Independent Differential Privacy

Thibault Pautrel[*] *thibault.pautrel@centralesupelec.fr*
Université Paris-Saclay, CNRS, CentraleSupélec, L2S
Gif-sur-Yvette, France

Florent Bouchard *florent.bouchard@cnrs.fr*
Université Paris-Saclay, CNRS, CentraleSupélec, L2S
Gif-sur-Yvette, France

Guillaume Ginolhac *guillaume.ginolhac@univ-smb.fr*
Université Savoie Mont Blanc, LISTIC
Annecy, France

Ammar Mian *ammar.mian@univ-smb.fr*
Université Savoie Mont Blanc, LISTIC
Annecy, France

Abstract

Federated learning trains a shared model across clients without sharing their data, yet the updates they transmit still leak information about individual records. Enforcing differential privacy is harder still when the model parameters lie on a Riemannian manifold. Existing methods rebuild the noise in the geometry of each new iterate, aggregate through intrinsic maps that have no closed form, and prove convergence only in restricted regimes. We introduce DP-RFedProj, a record-level differentially private federated learning method for compact submanifolds that aggregates through closed-form projections and privatises gradients in a fixed ambient geometry. Each client may run any stochastic first-order optimiser, take several local steps, and participate only intermittently. A closed-form record-level f-DP guarantee holds for the entire training transcript, independently of the manifold, the aggregation scheme and the local optimiser. We establish convergence for Riemannian SGD and show that privacy enters the rate through a term scaling with the manifold's intrinsic dimension rather than the ambient one. On EEG motor-imagery classification with SPDNet, whose weights lie on the Stiefel manifold, the method outperforms a Euclidean baseline at every privacy budget and certifies the same guarantee as the closest prior method with roughly five times less noise.

1 Introduction

1.1 Setting and motivation

Federated learning (FL) enables a set of clients, each holding its own local data, to collaboratively train a shared model under the coordination of a central server, without exchanging raw data (McMahan et al., 2017; Konečný et al., 2016). It is well suited to settings where data are scarce or subject-specific and where centralised pooling is precluded by confidentiality or regulatory constraints, as in medical imaging (Sheller et al., 2020) or satellite imagery (Ben Youssef et al., 2024). The standard protocol proceeds over a sequence of communication rounds. In each round, the server broadcasts the current global model to the clients, each client trains it locally on its own data, and the resulting updates are transmitted back and aggregated into a

[*]Corresponding author.

new global model that seeds the next round. When the updates are Euclidean parameters, the canonical aggregation scheme is FedAvg (McMahan et al., 2017), which forms the next global model by arithmetically averaging the client updates.

In a range of applications, however, the model parameters are constrained to a Riemannian manifold. Such structure arises when signals are classified through their symmetric positive definite (SPD) spatial covariance matrices, as in EEG brain–computer interfaces (BCI) or multi-spectral satellite imagery (Barachant et al., 2012; Li et al., 2017), and it underpins architectures such as SPDNet (Huang & Van Gool, 2017; Brooks et al., 2019), whose orthonormal weight layers lie on the Stiefel manifold. Riemannian federated learning adapts the protocol to this geometry: clients optimise locally with Riemannian gradient methods, and the server aggregates parameters constrained to the manifold. Aggregation is the central difficulty. Averaging the client parameters arithmetically may leave the manifold, and even when it does not, the result need not be geometrically meaningful. The natural replacement is the Riemannian (Fréchet) mean, which rarely admits a closed form and is instead approximated through the manifold's geometric maps, as in the classical Karcher flow (Karcher, 2014).

Keeping data on-device does not by itself protect it. The updates a client transmits are computed from its records and leak information about them: reconstruction attacks recover training examples from shared gradients (Zhu et al., 2019), and membership-inference attacks reveal whether a particular record was used (Shokri et al., 2017). Randomisation severs that link: with calibrated noise, the output is nearly as likely with or without any single record. Differential privacy (DP) (Dwork et al., 2006; 2014) makes this precise, bounding how much the output distribution can change when a single record is altered. It comes at two granularities: record-level DP makes any single training example undetectable, while client-level DP conceals a client's entire dataset and thus whether that client took part at all (Geyer et al., 2017; McMahan et al., 2018). We target the former. A record influences training only through the gradients it contributes to, so protecting it means acting on those gradients, bounding each contribution's influence before masking it by adding noise.

On Riemannian manifolds, that last step is the obstruction: addition is a vector-space operation, and a manifold is not closed under it. Transferring DP therefore raises a question absent in the Euclidean case: where should the noise live? Two lines of work answer it. The first perturbs the output: the Riemannian Laplace mechanism of Reimherr et al. (2021) and the Gaussian-DP extension of Jiang et al. (2023) privatise a single released statistic, such as the Fréchet mean. Federated training releases no such statistic, but a stream of updates over many rounds, so this line does not apply. The second perturbs the gradients instead: the tangent space supplies the missing vector-space structure, so noise can be injected there at each step. Han et al. (2024) formally introduce the tangent-space Gaussian mechanism and privatise Riemannian SGD (DP-RSGD), and Utpala et al. (2023) add a variance-reduced variant (DP-RSVRG), later used by Huang et al. (2026).

Certifying such a mechanism requires accounting for what a whole training run reveals about any one record. In the Euclidean setting, privacy accounting has tightened across successive frameworks: from the original (ε, δ) bound (Dwork et al., 2014), loose under composition, through its Rényi-DP refinement (Mironov, 2017; Balle et al., 2018), which composes sharply but needs care under subsampling, to the f-DP / Gaussian-DP calculus of Dong et al. (2022), which tightens both in closed form. Zheng et al. (2021) bring the f-DP calculus to the federated setting for record-level guarantees against an adversary that is a curious client or a coalition of clients. Our adversary is the server itself, which sees every message the protocol produces.

1.2 Related work

On general manifolds, aggregation goes through the exponential and logarithmic maps and a transport. Li & Ma (2022) gives the first Riemannian FL framework, RFedSVRG, averaging by a Karcher-flow step with a variance-reduced local solver, later extended with second-order information (Xiao et al., 2024; 2025). Huang et al. (2024) instead aggregates the gradient streams accumulated along the local trajectories (AGS), which needs only a retraction and a vector transport and admits multiple local steps under partial participation, at the cost of an analysis specific to Riemannian SGD: aggregating gradients rather than iterates leaves no

Table 1: Comparison of Riemannian federated methods. The convergence regime reports the combinations of local steps (single vs. multiple) and client participation (single, full, or partial) for which convergence is established. Both PriRFed and our method are optimiser-agnostic by design; the Solver column reports the optimiser under which each convergence result is proved.

Method	Manifold	Privacy	Convergence regime	Correction term	Solver	Geom. ops.
RFedSVRG (Li & Ma, 2022)	General	×	multi-step, single-client / single-step, full†	✓	RSVRG	Exp, Log, PT
AGS (Huang et al., 2024)	General	×	multi-step, partial	×	RSGD⋆	R, VT
PriRFed (Huang et al., 2026)	General	(ε, δ)	multi-step, single-client / single-step, full†	× / ✓	RSGD/RSVRG§	R, R^{-1}
Zhang et al. (2024)	Compact	×	multi-step, full	✓	RSGD	Π
Ours	**Compact**	***f*-DP**	**multi-step, partial**	×	RSGD§	**Π**

† Multiple local steps are analysed only with a single participating client; multi-client participation only with a single local step.
⋆ Structural: aggregating gradient streams precludes a general local optimiser.
§ Optimiser-agnostic framework outside convergence analysis specifications
Exp/Log: exponential/logarithm; $\mathrm{R}/\mathrm{R}^{-1}$: retraction and inverse; PT/VT: parallel/vector transport; Π: nearest-point projection.

room for a different local optimiser. Neither addresses privacy, and both rely on maps that on Stiefel are unavailable in closed form or recovered only by an iterative solve (Table 1).

On compact submanifolds, the nearest-point projection offers an alternative to general retractions, both for keeping local iterates on the manifold and for aggregating them at the server. Zhang et al. (2024) adopt it to support multiple local epochs, and Wang et al. (2025) carry the same construction to the zeroth-order setting, substituting a gradient-free estimator for the Riemannian gradient. Neither, however, addresses privacy, and both cancel client drift with a per-client, SCAFFOLD-style control variate carried in the submission: this couples aggregation to the local optimiser, and, since every variate must be refreshed each round, confines their analyses to full participation. We follow this projection-based line but add a record-level privacy layer and drop the control variate, leaving aggregation a pure function of the final iterates (Table 1).

Bringing privacy, federation, and manifold structure together, PriRFed (Huang et al., 2026) is, to our knowledge, the only prior method to combine all three. It targets record-level privacy under an honest-but-curious server — one that follows the protocol faithfully but may try to infer clients' data from the messages it receives —, but on a general manifold: it aggregates through a retraction and its inverse, privatises with the tangent-Gaussian mechanism of Han et al. (2024) in the underlying geometry, and analyses convergence for the DP-RSGD and DP-RSVRG local solvers. The noise construction is rebuilt at every iterate, and the training run is accounted by composing per-round (ε, δ) pairs. Convergence is established only when multiple local steps and multiple clients are not combined. Aggregation, finally, needs a retraction and its inverse in tandem, which on Stiefel is available in closed form in at most one direction.

1.3 Contributions

No existing method brings together projection-based aggregation, partial participation, multiple local steps, and differential privacy. We close this gap with DP-RFedProj, a record-level differentially private Riemannian federated learning algorithm for compact submanifolds under the Euclidean-induced metric. Our contributions are fourfold.

- **A projection-based, optimiser-agnostic private FL algorithm.** Closed-form projections do the geometric work at both stages of a round: they keep each client's local steps on the manifold, and they aggregate the returned iterates at the server. We give two aggregation rules on that basis: ProjAvg, which projects the ambient mean, and RLAvg, which retracts the average of the tangent-lifted displacements at the current iterate. Both fit a common template that also covers the prior Karcher-flow and retraction-based rules, which differ from ours only in the submission and the server map (Table 2). Because private local training is decoupled from aggregation, each client can pair any of them with any stochastic first-order optimiser consuming minibatch gradients (SGD, momentum, Adam).

- **A geometry-free privacy guarantee in closed form.** We privatise each client's gradients through a fixed ambient geometry, drawing the noise ambiently and projecting it onto the tangent space at each iterate. The resulting guarantee is a closed-form expression in the f-DP/GDP calculus, with record-level subsampling amplification and no dependence on the number of participating clients. It holds across the template's aggregators and is independent of the manifold and of the local optimisers. Against the closest prior accounting, which composes per-round (ε, δ) pairs, the budget grows with the square root of the number of rounds rather than linearly.
- **Convergence under partial participation and multiple local steps.** We give the first convergence analysis on compact submanifolds combining projection-based aggregation, partial participation, and multiple local steps under data heterogeneity, bypassing the logarithmic, exponential, and inverse-retraction maps entirely. The leading privacy term enters the rate through the manifold's intrinsic dimension rather than the ambient one.
- **Numerical experiments.** We instantiate the framework on EEG motor-imagery classification with SPDNet (Huang & Van Gool, 2017) on three MOABB datasets (Chevallier et al., 2024), and compare the resulting DP-FedSPDNet against FedAvg applied to EEGNet (Lawhern et al., 2018). Privacy reverses the ranking: EEGNet is the stronger model without it, DP-FedSPDNet wins at every budget and both participation levels, losing about half as much accuracy to privacy. Among the aggregation rules, our projection-based schemes matches the retraction-based one in accuracy at an order of magnitude less compute, and against PriRFed (Huang et al., 2026) our accountant certifies the same guarantee with roughly five times less noise.

Section 2 fixes the background, Section 3 presents the algorithm, Sections 4 and 5 give its privacy guarantee and convergence analysis, and Section 6 reports the experiments.

2 Background

Section 2.1 recalls the Riemannian geometry we use, Section 2.2 the federated protocol, its threat model and differential privacy. Sections 2.3-2.4 then describes the existing PriRFed (Huang et al., 2026) pipeline and its limitations that our construction departs from.

2.1 Riemannian Geometry

This section recalls basic notions from Riemannian geometry and optimisation. We refer the reader to Absil et al. (2008); Boumal (2023) for further background.

Submanifolds and the induced metric. A smooth manifold $\mathcal{M}$ of dimension $d_{\mathcal{M}}$, embedded in a Euclidean space $(\mathcal{E}, \langle\cdot,\cdot\rangle)$ of ambient dimension $D := \dim(\mathcal{E})$, is a subset locally diffeomorphic to $\mathbb{R}^{d_{\mathcal{M}}}$ via smoothly compatible charts. The tangent space $T_x\mathcal{M}$ at $x \in \mathcal{M}$ is the $d_{\mathcal{M}}$-dimensional subspace of $\mathcal{E}$ spanned by velocities of smooth curves through x. Equipping each tangent space with an inner product $g_x(\cdot,\cdot)$, varying smoothly in x, makes $\mathcal{M}$ a Riemannian manifold. Among the possible choices, the one induced by the ambient inner product,

$$g_x(U, V) \;:=\; \langle U, V\rangle, \qquad U, V \in T_x\mathcal{M},\; x \in \mathcal{M}, \tag{1}$$

makes $\mathcal{M}$ an embedded Riemannian submanifold. We write $\|\cdot\|$ for the norm of $(\mathcal{E}, \langle\cdot,\cdot\rangle)$, $\|\cdot\|_x = \sqrt{g_x(\cdot,\cdot)}$ for the one induced on $T_x\mathcal{M}$, and $P_x : \mathcal{E} \to T_x\mathcal{M}$ for the orthogonal projector onto the tangent space. We denote by $\mathrm{conv}(\mathcal{M}) \subseteq \mathcal{E}$ the convex hull of $\mathcal{M}$.

Euclidean and Riemannian gradients. For a smooth $f : \mathcal{M} \to \mathbb{R}$, the Riemannian gradient $\operatorname{grad} f(x) \in T_x\mathcal{M}$ is the unique tangent vector representing the differential through the chosen metric:

$$\mathrm{D}f(x)[\xi] = g_x(\operatorname{grad} f(x), \xi), \qquad \xi \in T_x\mathcal{M}.$$

In the embedded setting, when f extends to a smooth function on a neighbourhood of $\mathcal{M}$ in $\mathcal{E}$ (also denoted f) with Euclidean gradient $\nabla f(x) \in \mathcal{E}$, the Riemannian gradient is given by the orthogonal projection of the ambient gradient onto $T_x\mathcal{M}$: $\operatorname{grad} f(x) = P_x \nabla f(x)$.

Geodesics and exponential map. A geodesic is a curve that locally minimises arc length, i.e. the Riemannian analogue of straight lines in curved spaces. The Riemannian distance $d(x, y)$ is the length of a shortest geodesic connecting x and y. The exponential map $\mathrm{Exp}_x : T_x\mathcal{M} \to \mathcal{M}$ sends $\xi \in T_x\mathcal{M}$ to $\gamma(1)$, where γ is the geodesic with $\gamma(0) = x$ and $\dot\gamma(0) = \xi$. Its local inverse is the logarithmic map $\mathrm{Log}_x : \mathcal{M} \to T_x\mathcal{M}$.

Retractions and projections. In Riemannian optimisation, retractions serve as computationally efficient substitutes for the exponential map. A smooth mapping $\mathrm{R}_x : T_x\mathcal{M} \to \mathcal{M}$ is a retraction if $\mathrm{R}_x(0) = x$ and $\mathrm{DR}_x(0)[u] = u$ for all $u \in T_x\mathcal{M}$, i.e. it coincides with the exponential map to first order. On compact submanifolds, a natural choice is the projection-based retraction (Absil & Malick, 2012) defined through a projection map $\Pi : \mathcal{E} \to \mathcal{M}$ via $\mathrm{R}_x(u) := \Pi(x + u)$, $u \in T_x\mathcal{M}$. The canonical example of such projection is the nearest-point map

$$\Pi(y) := \arg\min_{x\in\mathcal{M}} \tfrac{1}{2}\|y - x\|^2. \tag{2}$$

Examples. The Stiefel manifold $\mathrm{St}(d, p) = \{X \in \mathbb{R}^{d\times p} : X^\top X = I_p\}$ is a compact submanifold of $\mathbb{R}^{d\times p}$ (Appendix A.1). It carries the constrained weight layers of SPDNet (Huang & Van Gool, 2017), the architecture we train. A second example is the Grassmann manifold of p-dimensional subspaces of $\mathbb{R}^d$, which is a compact submanifold of the symmetric matrices (Appendix A.2). SPDNet's inputs are covariance descriptors, which live on the manifold $\mathcal{S}_n^{++}$ of $n \times n$ symmetric positive definite matrices. This manifold is not compact, and is commonly endowed with the log-Euclidean (Arsigny et al., 2007) or affine-invariant (Pennec et al., 2006) metric rather than the ambient one (Appendix A.3).

2.2 Federated Learning and Differential Privacy

Federated learning (FL) trains collaboratively a shared model across n clients, coordinated by a central server (McMahan et al., 2017). Each client $i \in [n]$ holds its own dataset $\mathcal{D}_i := \{z_{i,1}, \dots, z_{i,m_i}\} \subset \mathcal{Z}$ of size m_i, drawn from a client-specific distribution μ_i (possibly non-i.i.d. data). The goal is to minimise the empirical risk

$$\min_{x\in\mathcal{M}} \Big\{ F(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x) \Big\}, \qquad f_i(x) := \frac{1}{m_i}\sum_{\ell=1}^{m_i} f(x; z_{i,\ell}), \tag{3}$$

where $\mathcal{M}$ is a Riemannian manifold and $f(x; z)$ is a chosen loss function computed at model parameter $x \in \mathcal{M}$ on a data sample z. Here f_i represents client i's local empirical risk associated with the loss f on its dataset $\mathcal{D}_i$. For clarity we use uniform weights in (3). A sample-weighted objective (in which each f_i is weighted by $m_i/\sum_j m_j$) can be handled by the same arguments.

The key requirement is that raw data are never shared: to perform optimization (3), clients transmit only model updates computed locally, which the server aggregates over T communication rounds. Entering round t, the server holds the global model parameter $x_t \in \mathcal{M}$ and samples a set $S_t \subset [n]$ of k participating clients, uniformly without replacement and independently of the data. The round has two stages:

(i) **Local optimisation.** The server broadcasts the current global model x_t to the k selected clients. Each selected client i initialises $x_{t,0}^{(i)} = x_t$ and runs $\tau \geq 1$ gradient-based steps: at step j, from the current local iterate $x_{t,j}^{(i)}$, it draws a minibatch $\mathcal{B}_{t,j}^{(i)} \subset \mathcal{D}_i$ of size B uniformly without replacement and forms the corresponding stochastic Riemannian gradient of its local objective,

$$g_{t,j}^{(i)} = \frac{1}{B}\sum_{\ell\in\mathcal{B}_{t,j}^{(i)}} \mathrm{grad}\, f(x_{t,j}^{(i)}; z_{i,\ell}). \tag{4}$$

The chosen gradient-based optimiser then maps this gradient to the next iterate $x_{t,j+1}^{(i)}$, producing after τ steps the final local iterate $x_{t,\tau}^{(i)} \in \mathcal{M}$.

(ii) **Aggregation.** An aggregation scheme is fixed by two ingredients: the submission $\Delta^{(i)}$ each client forms from $(x_t, x_{t,\tau}^{(i)})$, and the rule Agg by which the server combines the submissions into a point of $\mathcal{M}$. Each selected client $i \in S_t$ sends its submission, and the server sets the next global model

to $x_{t+1} = \textsc{Agg}\big(x_t, \{\Delta^{(i)}\}_{i\in S_t}\big)$. Both ingredients depend on the geometry; Section 2.3.2 gives the schemes we consider.

Although raw data never leave the client, the messages it sends do not keep them safe: per-record gradients encode enough to reconstruct training examples (Zhu et al., 2019) or to infer whether a given record was used (Shokri et al., 2017), and a submission accumulates those gradients over the local steps. Minibatch sampling randomises the submission without privatising it: conditionally on the records drawn, the update is still determined exactly.

Threat model. Every submission is thus leak-prone, and the server collects them all, which makes it the natural adversary (see Figure 1). Following the standard threat model for private federated learning (Le et al., 2023), also used by PriRFed (Huang et al., 2026), we assume the server to be an honest-but-curious (HBC) adversary: it follows the protocol but seeks to infer clients' records from every message it receives, that is, from the full transcript

$$\mathcal{V}_T := \big(\{(x_t, S_t, \{\Delta^{(i)}\}_{i\in S_t})\}_{t=0}^{T-1},\, x_T\big). \tag{5}$$

We target record-level protection: no client's training example should be inferable from $\mathcal{V}_T$.

Differential privacy. Meeting this goal requires randomisation, the mechanism at the heart of Differential Privacy (Dwork et al., 2014). Calibrated randomness makes each release nearly as likely with or without any single record, so no given record can be identified from the transcript. Calibrating it requires bounding how much one record can change what is released. At the gradient level, a record contributes a single summand to a minibatch average, and capping that summand bounds its influence. We therefore randomise each gradient before it is fed to an optimiser (Abadi et al., 2016).

Throughout, the local datasets are assumed pairwise disjoint, so that each record belongs to exactly one client. This is the standing assumption under which record-level adjacency is stated.

Definition 1 (Record-level adjacency)**.** Two databases $D = (\mathcal{D}_1, \ldots, \mathcal{D}_n)$ and $D' = (\mathcal{D}'_1, \ldots, \mathcal{D}'_n)$ over the same cohort of clients are *record-adjacent*, written $D \bowtie D'$, if $\mathcal{D}_i = \mathcal{D}'_i$ for all but one client $i^\star$, whose dataset differs from $\mathcal{D}'_{i^\star}$ by a single substituted record (in particular $|\mathcal{D}_{i^\star}| = |\mathcal{D}'_{i^\star}|$).

Substituting a record thus perturbs one local dataset and leaves the other $n-1$ untouched.

Differential privacy compares, across such a substitution, the law of a *randomised mechanism* $\mathcal{A}$: a map sending a database D to a random output in a space $\mathcal{Y}$. Here $\mathcal{A}$ is the protocol detailed below (i.e. local privatised optimisation (2.3.1) and server aggregation (2.3.2)) and its output the full transcript $\mathcal{V}_T$ (5).

Definition 2 ((ε, δ)-DP (Dwork et al., 2014))**.** A randomised mechanism $\mathcal{A}$ from databases to an output space $\mathcal{Y}$ is (ε, δ)-differentially private if, for every adjacent pair $D \bowtie D'$ and every measurable $E \subseteq \mathcal{Y}$,

$$\Pr\big(\mathcal{A}(D) \in E\big) \le e^{\varepsilon}\, \Pr\big(\mathcal{A}(D') \in E\big) + \delta,$$

the probability being over the randomness of $\mathcal{A}$ at a fixed database. Here $\varepsilon \ge 0$ bounds the worst-case privacy loss (smaller is stronger) and $\delta \ge 0$ is a residual failure probability.

Since each client releases a privatised gradient at every local step of every round, certifying $\mathcal{V}_T$ means quantifying how the guarantee of a single such release degrades once all of them are observed jointly. This is the accounting task of Section 4.

2.3 Existing approach PriRFed

The design below is the one instantiated by PriRFed (Huang et al., 2026): both the privatisation and the aggregation are carried out in the manifold's own geometry. It is the baseline our construction departs from.

2.3.1 Local private optimisation

In the Euclidean setting (Abadi et al., 2016), gradient-level randomisation combines two operations at each local step: (i) clipping each per-sample gradient in ℓ_2-norm, to bound any single record's influence, then (ii)

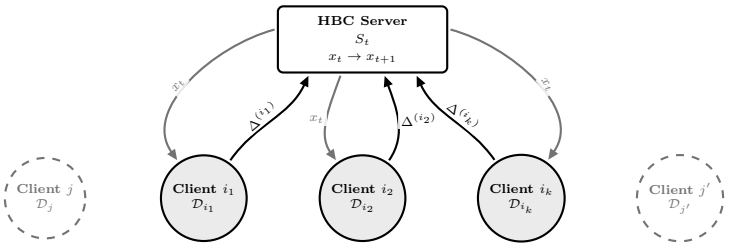


Figure 1: One round t under HBC server. The server draws a fixed-size active set $S_t \subset [n]$, sends x_t to the selected clients (shaded), and collects their submissions $\Delta^{(i)}$ from τ private local steps and aggregate them to form the next global iterate x_{t+1}. The HBC server sees $(x_t, S_t, \{\Delta^{(i)}\}_{i \in S_t})$.

adding Gaussian noise to mask it. Neither transfers directly to a Riemannian manifold, since the gradient at x is not a free Euclidean vector but an element of $T_x\mathcal{M}$. Both operations must therefore be anchored to the geometry at the current iterate.

Clipping in the Riemannian norm. Given a norm $\|\cdot\|_\bullet$ on a vector space and a threshold $C > 0$, the clipping operator rescales any vector v to have norm at most C:

$$\mathrm{clip}_C^{\|\cdot\|_\bullet}(v) := v \cdot \min\bigl(1,\, C/\,\|v\|_\bullet\bigr), \qquad \mathrm{clip}_C^{\|\cdot\|_\bullet}(0) := 0. \tag{6}$$

A tangent vector's magnitude is measured by the point-dependent $\|\cdot\|_x$ rather than by an ambient norm, so clipping is performed in $\|\cdot\|_x$: the threshold C then bounds a record's influence as the geometry at x measures it.

Tangent-space Gaussian noise. The noise is added to the clipped gradient, so it must live in the same vector space $T_x\mathcal{M}$ and follow a distribution that likewise varies with x. PriRFed uses the tangent-space Gaussian of Han et al. (2024), isotropic with respect to g_x and denoted $\mathcal{N}_x(0, \sigma_{\mathrm{dp}}^2)$: writing $(e_1, \ldots, e_{d_\mathcal{M}})$ for a g_x-orthonormal basis of $T_x\mathcal{M}$, a sample $\xi \sim \mathcal{N}_x(0, \sigma_{\mathrm{dp}}^2)$ is obtained by drawing i.i.d. Gaussian coordinates and combining them along this basis,

$$\xi = \sum_{a=1}^{d_\mathcal{M}} z_a e_a, \qquad z \sim \mathcal{N}(0, \sigma_{\mathrm{dp}}^2 I_{d_\mathcal{M}}). \tag{7}$$

The law of ξ does not depend on which basis is used. Sampling from (7) requires a g_x-orthonormal basis of $T_x\mathcal{M}$, which must be rebuilt at every iterate. Utpala et al. (2023) avoid this cost by fixing a reference point $\bar{x}$ at which the metric is Euclidean up to a constant. There, an orthonormal basis is available in closed form, so drawing $\bar{\xi} \sim \mathcal{N}_{\bar{x}}(0, \sigma_{\mathrm{dp}}^2)$ amounts to sampling a standard Gaussian vector and reshaping it into $T_{\bar{x}}\mathcal{M}$. A linear isometry $\mathcal{T}_{\bar{x} \to x}$ then maps $\bar{\xi}$ to $T_x\mathcal{M}$, where it follows $\mathcal{N}_x(0, \sigma_{\mathrm{dp}}^2)$, orthonormal bases being preserved. The per-step basis construction is thus replaced by a per-step transport.

Combining the two operations, at local step j of round t a selected client i perturbs its clipped minibatch gradient with noise drawn afresh in the geometry of the current iterate,

$$\tilde{g}_{t,j}^{(i)} := \tfrac{1}{B} \sum_{\ell \in \mathcal{B}_{t,j}^{(i)}} \mathrm{clip}_C^{\|\cdot\|_{x_{t,j}^{(i)}}}\bigl(\mathrm{grad}\, f(x_{t,j}^{(i)}; z_{i,\ell})\bigr) + \xi_{t,j}^{(i)} \in T_{x_{t,j}^{(i)}}\mathcal{M}, \qquad \xi_{t,j}^{(i)} \sim \mathcal{N}_{x_{t,j}^{(i)}}\bigl(0, \sigma_{\mathrm{dp}}^2\bigr). \tag{8}$$

This $\tilde{g}_{t,j}^{(i)}$ replaces the stochastic Riemannian gradient $g_{t,j}^{(i)}$ of (4) as the optimiser's input, producing $x_{t,j+1}^{(i)}$. Being the only point where the data enter, it carries all record-level protection. The choice of optimiser is left open.

2.3.2 Server aggregation on Riemannian manifolds

Privatisation being confined to the local step, aggregation carries no privacy role: the server's task is purely geometric, to turn the submissions $\{\Delta^{(i)}\}_{i \in S_t}$ into a global iterate x_{t+1} that is a meaningful average of

the clients' final iterates and lies on $\mathcal{M}$. In Euclidean FL (McMahan et al., 2017) both ingredients are trivial: the submission is the local iterate itself and AGG the arithmetic mean, as in FedAvg. On a manifold neither survives, since the arithmetic mean of points of $\mathcal{M}$ generally leaves $\mathcal{M}$ or carries no intrinsic meaning. Depending on the scheme, the submission becomes a point of $\mathcal{M}$ or a tangent vector at x_t [1].

On Riemannian manifolds, the natural substitute is the Fréchet mean (FM), which generalises the arithmetic mean by minimising squared Riemannian rather than Euclidean distance to the inputs. Each client submits its final iterate and the server returns

$$x_{t+1}^{\mathrm{FM}} \leftarrow \arg\min_{x\in\mathcal{M}} \frac{1}{k}\sum_{i\in S_t} d\big(x, \Delta^{(i)}\big)^2, \qquad \Delta^{(i)} := x_{t,\tau}^{(i)} \in \mathcal{M}.$$

This minimiser generally has no closed form (Pennec, 2006) and is computed iteratively by the Karcher flow (Karcher, 2014). Federated aggregation keeps a single iteration: each client lifts its iterate to $T_{x_t}\mathcal{M}$ through the logarithmic map, and the server averages these lifts and maps the result back to the manifold via the exponential map (Li & Ma, 2022),

$$x_{t+1}^{\mathrm{KFAvg}} = \mathrm{Exp}_{x_t}\left(\frac{1}{k}\sum_{i\in S_t}\Delta^{(i)}\right), \qquad \Delta^{(i)} = \mathrm{Log}_{x_t}\big(x_{t,\tau}^{(i)}\big) \in T_{x_t}\mathcal{M}. \tag{9}$$

The logarithm has no closed form on manifolds of practical interest such as Stiefel (Zimmermann & Huper, 2022). PriRFed (Huang et al., 2026) replaces Exp_{x_t} and Log_{x_t} by a retraction R_{x_t} and its inverse $\mathrm{R}_{x_t}^{-1}$, so the submission becomes the inverse-retraction lift in place of the logarithmic map:

$$x_{t+1}^{\mathrm{RetAvg}} = \mathrm{R}_{x_t}\left(\frac{1}{k}\sum_{i\in S_t}\Delta^{(i)}\right), \qquad \Delta^{(i)} = \mathrm{R}_{x_t}^{-1}\big(x_{t,\tau}^{(i)}\big) \in T_{x_t}\mathcal{M}. \tag{10}$$

This substitution eases the previous difficulties but does not remove them entirely (see Limitation 3).

2.4 Limitations of the intrinsic pipeline

The design of Sections 2.3.1–2.3.2, instantiated by PriRFed (Huang et al., 2026), carries four limitations, which the rest of the paper lifts.

(1) **A mechanism bound to a moving geometry.** Both operations of (8) are anchored at the current iterate, so the noise construction must be redone at each of them, either via a fresh g_x-orthonormal basis, or the transport of Utpala et al. (2023), itself available only on manifolds carrying a suitable reference point. Privatisation is thus inseparable from the geometry.

(2) **Loose accounting.** PriRFed builds its guarantee in three steps, with the first and third ones inflating the budget:

   (a) each client's local training is summarised by a single (ε, δ) pair, which only partly expresses what that training reveals;

   (b) the pairs of the clients active in a round are combined, by parallel composition when the datasets are disjoint and sequentially otherwise (Huang et al., 2026, Thms. 6 and 5);

   (c) the T per-round (ε, δ) pairs are combined by a generic composition theorem (Huang et al., 2026, Thm. 7), a step loose by construction which holds for the worst mechanism those pairs could describe.

   A separate gap runs the other way: PriRFed amplifies by client sampling (Huang et al., 2026, Def. 7, Lem. 1), a gain available only while the active set stays hidden from the adversary, whereas here the server draws it and keeps it (Remark 7). Section 4 re-accounts for the same mechanism in the f-DP calculus, with amplification at the record level alone.

[1] This template requires $\Delta^{(i)}$ to depend only on $(x_t, x_{t,\tau}^{(i)})$. It therefore excludes schemes folding a per-client drift correction into the local update, such as Zhang et al. (2024), where aggregation and local optimiser are coupled.

(3) **Cost of the geometric primitives.** RetAvg uses a retraction and its inverse in tandem, and on Stiefel no retraction supplies both directions in closed form: the polar and QR retractions invert only by solving a matrix equation, whereas the orthographic retraction inverts in closed form but has an implicitly defined, Riccati-based forward map (Bouchard et al., 2025; Kaneko et al., 2012). An iterative solve is therefore paid once per submission, for every participating client and every round.

(4) **Reach of the convergence guarantees.** Existing analyses cover multiple local steps only when one client participates, and several clients only when each takes one step (Li & Ma, 2022; Huang et al., 2026) but never both, the regime federated training actually runs in. The lifting map is again the obstruction: it is anchored at a base point that moves at every local step, so a client's successive increments are never directly comparable, and reconciling them drags the curvature of the manifold into the constants. The multi-step results moreover assume the data identically distributed across clients, setting heterogeneity aside.

3 Projection-Based Private Federated Learning on Compact Submanifolds

3.1 Structural properties

Lifting the limitations of Section 2.4 costs generality: where PriRFed assumes a general Riemannian manifold, we assume two structural properties.

(P1) *Euclidean-induced metric* (1): the base-point norm $\|\cdot\|_x$ is the restriction of the ambient norm $\|\cdot\|$ to $T_x\mathcal{M}$, so gradient formation, clipping, and noise sampling reduce to applying the tangent projector P_x to ambient quantities.

(P2) *Compactness of* $\mathcal{M}$: it guarantees a manifold projection $\Pi : \mathcal{E} \to \mathcal{M}$, which returns updates to $\mathcal{M}$ through the retraction $\mathrm{R}_x(u) = \Pi(x + u)$.

Each property lifts a distinct limitation. Property P1 detaches the mechanism from the geometry (Limitation 1): clipping and noise act on iterate-independent ambient quantities, so no per-iterate construction is needed. Property P2 supplies the projection the algorithm is built on, and with it the convergence analysis: it replaces the retraction and its inverse by the single map Π, applied at both the local and the aggregation stage, and the nonlinear lift by the linear tangent projection (Limitation 3). Specialising to the nearest-point projection (2) then lets multiple local steps and several clients be analysed together, under heterogeneity (Limitation 4). Limitation 2 needs neither property: it is lifted by re-accounting the mechanism in the tight f-DP/GDP calculus.

Modularity guides the design: server aggregation depends on each client's private local training only through its final iterate, so the two FL stages are fully decoupled. Sections 3.2 and 3.3 develop them in turn, and Section 3.4 assembles them into Algorithm 1.

3.2 Privatised local optimisation

Private local training follows the pipeline of Section 2.3.1, with each of its three intrinsic ingredients replaced by an ambient one: clipping moves to the ambient norm, the tangent Gaussian is drawn ambiently and projected by P_x, and the retraction becomes projection-based.

Extrinsic gradient privatisation. Under the induced metric (1), both operations of (8) can be carried out ambiently, in a single fixed geometry, on quantities that do not depend on the iterate. Extrinsic and intrinsic privatisation thereby produce the same mechanism, which Section 4 re-accounts in the f-DP/GDP calculus.

The Riemannian gradient is the orthogonal projection of the ambient Euclidean gradient, and by Property P1 the two norms $\|\cdot\|$ and $\|\cdot\|_x$ coincide on $T_x\mathcal{M}$. We may therefore clip in the ambient norm, writing $\mathrm{clip}_C := \mathrm{clip}_C^{\|\cdot\|}$ for the corresponding instance of (6). The threshold caps each per-record contribution by C

uniformly in x. For a minibatch $\mathcal{B}$ of size B, the clipped minibatch Riemannian gradient of f_i at x is

$$\widehat{\operatorname{grad}}^{C} f_i(x;\mathcal{B}) = \frac{1}{B}\sum_{\ell\in\mathcal{B}} \operatorname{clip}_C(P_x \nabla f(x; z_{i,\ell})) \in T_x\mathcal{M}. \tag{11}$$

For the noise, we draw a single isotropic sample $\xi \sim \mathcal{N}(0, \sigma_{\mathrm{dp}}^2 I_{\mathcal{E}})$ in $\mathcal{E}$ and project it onto $T_x\mathcal{M}$. This is the tangent-space Gaussian, obtained without an orthonormal basis.

Lemma 3 (Projected ambient noise is tangent Gaussian)**.** *Let $x \in \mathcal{M}$ and $\xi \sim \mathcal{N}(0, \sigma_{\mathrm{dp}}^2 I_{\mathcal{E}})$ on $\mathcal{E}$. Then $P_x\xi \sim \mathcal{N}_x(0, \sigma_{\mathrm{dp}}^2)$ in the sense of* (7)*.*

Proof. Let $(e_1, \ldots, e_{d_{\mathcal{M}}})$ be a $\|\cdot\|_x$-orthonormal basis of $T_x\mathcal{M}$; by Property P1 it is orthonormal for the ambient inner product. Since P_x is a self-adjoint projector fixing $T_x\mathcal{M}$, $\langle P_x\xi, e_a\rangle = \langle \xi, P_x e_a\rangle = \langle \xi, e_a\rangle$. These coordinates are jointly Gaussian, centred, with covariance $\sigma_{\mathrm{dp}}^2\langle e_a, e_b\rangle = \sigma_{\mathrm{dp}}^2\delta_{ab}$, hence i.i.d. $\mathcal{N}(0, \sigma_{\mathrm{dp}}^2)$, which is (7). □

Two consequences follow. First, noise sampling no longer needs a g_x-orthonormal basis rebuilt at each iterate, nor the reference point and vector transport of Utpala et al. (2023). Second, the privatised gradient is distributionally identical to PriRFed's: under Property P1 the two clipping operators coincide, and by Lemma 3 so do the two noise laws.

Gathering both operations, at local step $j \in \{0, \ldots, \tau-1\}$ of round t, client $i \in S_t$ forms its privatised Riemannian gradient

$$\tilde{g}_{t,j}^{(i)} \;:=\; \frac{1}{B}\sum_{\ell\in\mathcal{B}_{t,j}^{(i)}} \operatorname{clip}_C\big(P_{x_{t,j}^{(i)}} \nabla f(x_{t,j}^{(i)}; z_{i,\ell})\big) \;+\; P_{x_{t,j}^{(i)}}\, \xi_{t,j}^{(i)}, \qquad \xi_{t,j}^{(i)} \sim \mathcal{N}(0, \sigma_{\mathrm{dp}}^2 I_{\mathcal{E}}), \tag{12}$$

where the minibatch draws, the noise draws and the selection sets are mutually independent and independent of the data. Each minibatch is thus drawn afresh at every local step, independently of the previous ones, which is the sampling model assumed throughout Sections 4 and 5.

Optimiser-agnostic local updates. The privatised gradient (12) substitutes directly for the stochastic Riemannian gradient (4) and is the sole point at which the data enter the local run. Any optimiser whose only access to $\mathcal{D}_i$ is a minibatch gradient at the current iterate can therefore be used: client i runs $\textsc{Opt}^{(i)}$ with hyperparameters $\theta^{(i)}$ and internal state $s_{t,j}^{(i)}$, producing at step j

$$\big(u_{t,j}^{(i)},\, s_{t,j+1}^{(i)}\big) \;=\; \textsc{Opt}^{(i)}\big(\tilde{g}_{t,j}^{(i)},\, s_{t,j}^{(i)}; \theta^{(i)}\big), \qquad u_{t,j}^{(i)} \;\in\; T_{x_{t,j}^{(i)}}\mathcal{M}, \tag{13}$$

with $\textsc{Opt}^{(i)}$ and $\theta^{(i)}$ allowed to differ across clients. This covers in particular RSGD with stepsize η, which takes $u_{t,j}^{(i)} = \eta\, \tilde{g}_{t,j}^{(i)}$, and heavy-ball momentum, which takes $u_{t,j}^{(i)} = \eta\, v_{t,j}^{(i)}$, where the state $v_{t,j}^{(i)} = \beta\, P_{x_{t,j}^{(i)}} v_{t,j-1}^{(i)} + \tilde{g}_{t,j}^{(i)}$ is projected onto the current tangent space at each step. Excluded are optimisers that touch the data other than through (12), RSVRG among them: its periodic full gradient over $\mathcal{D}_i$ is a second data access which the per-step accounting of Section 4 does not cover. Privatising it would require charging it separately, and it is anyway costly for large m_i.

Projection-based local steps. It remains to map the update direction back onto $\mathcal{M}$. Compactness supplies a projection $\Pi : \mathcal{E} \to \mathcal{M}$, so the local update uses the projection-based retraction $\mathrm{R}_x(u) = \Pi(x+u)$ of Absil & Malick (2012) in place of a general one: starting from $x_{t,0}^{(i)} := x_t$, for $j = 0, \ldots, \tau-1$,

$$x_{t,j+1}^{(i)} \;\leftarrow\; \Pi\big(x_{t,j}^{(i)} - u_{t,j}^{(i)}\big), \tag{14}$$

with $u_{t,j}^{(i)}$ given by (13). After τ such steps the client holds its final local iterate $x_{t,\tau}^{(i)} \in \mathcal{M}$, the sole quantity it passes to the aggregation stage.

3.3 Server-side aggregation

As in Section 2.3.2, aggregation operates on model iterates: from the submissions $\{\Delta^{(i)}\}_{i\in S_t}$ formed by the clients from their final local iterates, the server computes the next global iterate $x_{t+1} \in \mathcal{M}$.

Building on the retracted-lifted (RL) barycenter analysis of Bouchard et al. (2025), we propose two schemes, ProjAvg and RLAvg, involving only the tangent projector P_x and the manifold projection Π supplied by Property P2.

ProjAvg. It projects the ambient mean of the iterates back onto $\mathcal{M}$:

$$x_{t+1}^{\text{ProjAvg}} \;=\; \Pi\!\left(\frac{1}{k}\sum_{i\in S_t}\Delta^{(i)}\right), \qquad \Delta^{(i)} \;=\; x_{t,\tau}^{(i)} \in \mathcal{M}. \tag{15}$$

This is a geometrically meaningful average: it is the closed-form RL-barycenter of the iterates for the projection-based retraction and the tangent-projector lifting map (Bouchard et al., 2025, Prop. 1). It also matches the project-the-mean update of Deng & Hu (2025) in the decentralised setting.

RLAvg. It implements the federated analogue of the RL-barycenter in its alternative fixed-point construction, anchored at the current iterate x_t: each client submits the tangent projection at x_t of its displacement, and the server retracts their average at the same point,

$$x_{t+1}^{\text{RLAvg}} \;=\; \Pi\!\left(x_t + \frac{1}{k}\sum_{i\in S_t}\Delta^{(i)}\right), \qquad \Delta^{(i)} \;=\; P_{x_t}\big(x_{t,\tau}^{(i)} - x_t\big) \in T_{x_t}\mathcal{M}. \tag{16}$$

This mirrors the lift–average–retract structure of KFAvg and RetAvg, with the logarithm and inverse retraction replaced by the linear projector P_{x_t}, retaining an anchored, tangent-space aggregation.

Table 2: Aggregation primitives for the four schemes. $\text{Agg.Sub}(x_t, x_{t,\tau}^{(i)})$ is the client-side submission $\Delta^{(i)}$ and $\text{Agg.Out}(x_t, \Delta_t)$ is the server-side global update x_{t+1}, where $\Delta_t = \frac{1}{k}\sum_{i\in S_t}\Delta^{(i)}$.

Scheme	$\text{Agg.Sub}(x_t, x_{t,\tau}^{(i)})$	$\text{Agg.Out}(x_t, \Delta_t)$
KFAvg	$\text{Log}_{x_t}\big(x_{t,\tau}^{(i)}\big)$	$\text{Exp}_{x_t}\big(\Delta_t\big)$
RetAvg	$\text{R}_{x_t}^{-1}\big(x_{t,\tau}^{(i)}\big)$	$\text{R}_{x_t}\big(\Delta_t\big)$
ProjAvg (ours)	$x_{t,\tau}^{(i)}$	$\Pi\big(\Delta_t\big)$
RLAvg (ours)	$P_{x_t}\big(x_{t,\tau}^{(i)} - x_t\big)$	$\Pi\big(x_t + \Delta_t\big)$

3.4 Algorithm

Algorithm 1 assembles the two stages: each client runs τ private local steps (14) with its own optimiser and sends the single submission $\Delta^{(i)}$, from which the server forms the next global iterate via the chosen aggregation scheme.

Unified template. Algorithm 1 is stated for our two schemes, but the aggregator enters it at Lines 13 and 16 only. Writing those two lines as a client-side submission Agg.Sub and a server-side output Agg.Out yields a unifying template: substituting the first two rows of Table 2 recovers the intrinsic KFAvg and RetAvg aggregation. The unification holds at the aggregation level only. At the local optimisation level our approach privatises extrinsically (12), whereas the original KFAvg and RetAvg of PriRFed privatise intrinsically, in the iterate-dependent geometry (8). Any privatised local optimiser of Section 3.2 may be paired with any of the four.

Algorithm 1 DP-RFedProj: record-level DP Riemannian federated learning

Require: n clients; k clients/round; τ local steps; batch size B; T rounds; client optimisers $\{\textsc{Opt}^{(i)}\}_{i\in[n]}$ with hyperparameters $\{\theta^{(i)}\}_{i\in[n]}$; aggregator $\textsc{Agg} \in \{\textsc{ProjAvg}, \textsc{RLAvg}\}$

DP parameters: clip C; per-step noise scale σ_{dp}

Initialise $x_0 \in \mathcal{M}$

for $t = 0$ to $T-1$ **do** ▷ communication round

  Draw $S_t \subset [n]$ uniformly without replacement, $|S_t| = k$

  Broadcast x_t to every client $i \in S_t$

  for all $i \in S_t$ **in parallel do**

    $x_{t,0}^{(i)} \leftarrow x_t$ and $s_{t,0}^{(i)} \leftarrow$ initial optimiser state

    for $j = 0$ to $\tau - 1$ **do** ▷ privatised local step

      Draw $\mathcal{B}_{t,j}^{(i)} \subset \mathcal{D}_i$ of size B, uniformly without replacement

      Draw $\xi_{t,j}^{(i)} \sim \mathcal{N}\big(0, \sigma_{\mathrm{dp}}^2 I_{\mathcal{E}}\big)$

      $\tilde{g}_{t,j}^{(i)} \leftarrow \dfrac{1}{B} \displaystyle\sum_{\ell \in \mathcal{B}_{t,j}^{(i)}} \mathrm{clip}_C\big(P_{x_{t,j}^{(i)}} \nabla f(x_{t,j}^{(i)}; z_{i,\ell})\big) \;+\; P_{x_{t,j}^{(i)}}\, \xi_{t,j}^{(i)}$ ▷ Eq. (12)

      $\big(u_{t,j}^{(i)}, s_{t,j+1}^{(i)}\big) \leftarrow \textsc{Opt}^{(i)}\big(\tilde{g}_{t,j}^{(i)},\, s_{t,j}^{(i)};\, \theta^{(i)}\big)$

      $x_{t,j+1}^{(i)} \leftarrow \Pi\big(x_{t,j}^{(i)} - u_{t,j}^{(i)}\big)$ ▷ Eq. (14)

    $\Delta^{(i)} \leftarrow \textsc{Agg.Sub}(x_t, x_{t,\tau}^{(i)})$ ▷ Table 2

    Send submission $\Delta^{(i)}$ to the server

  The server computes $\Delta_t \leftarrow \frac{1}{k} \sum_{i \in S_t} \Delta^{(i)}$

  $x_{t+1} \leftarrow \textsc{Agg.Out}(x_t, \Delta_t)$

return x_T

3.5 Scope of the guarantees

The two guarantees are scoped along two independent axes: which aggregation scheme is used (Table 2), and which manifold the parameters live on (Table 3).

Choice of aggregator. The convergence rate of Section 5 is established for ProjAvg and RLAvg only: both their primitives are linear in the submissions, which is what lets the τ local displacements telescope in the single ambient space $\mathcal{E}$ and the k client submissions be averaged rather than composed (Lemmas 18 and 19). For KFAvg and RetAvg the submission is a nonlinear lift and the output a nonlinear map of the average, and neither step goes through.

The privacy guarantee, by contrast, covers all four. The data enter a client's computation through the privatised gradient (12), where Section 4 settles the privacy cost ambiently, before any geometry is applied. Everything downstream, such as the projections P_x and Π, the local optimiser, the aggregation scheme, reshapes that vector without consulting the data again, which by post-processing cannot help an adversary. The guarantee therefore carries through untouched, on any embedded submanifold where the algorithm runs at all and for any aggregation scheme of Table 2 (Theorem 10).

Choice of manifold. With ProjAvg or RLAvg, the geometry enters Algorithm 1 only through P_x and Π, which fixes where it applies (Table 3). The two properties play distinct roles here. Property P2 is a sufficient condition which makes the algorithm runnable, by supplying Π: without that map there is neither a local step (14) nor an aggregation. Property P1 makes its mechanism the intended one: without it the injected noise is not isotropic for the manifold's own metric, and the ambient clip does not bound a record's influence as the geometry at x measures it. Both hold on the sphere and on the Stiefel and Grassmann manifolds, where the two guarantees apply together.

Two flat cases fail compactness but retain the projection it was meant to supply. When $\mathcal{M} = \mathcal{E}$ the two projection maps are the identity, the local steps (12)–(14) are DP-SGD (Abadi et al., 2016) and both

aggregators reduce to the arithmetic mean. When $\mathcal{M} = \mathcal{S}_n^{++}$, under the log-Euclidean metric (Arsigny et al., 2007), the matrix logarithm is a global isometry from $\mathcal{S}_n^{++}$ onto the flat space $\mathcal{S}_n$ of symmetric matrices, so reparametrising by the log chart puts the parameters on $\mathcal{S}_n \subset \mathbb{R}^{n\times n}$ with the induced metric, where Π and P_x both reduce to symmetrisation (Utpala et al., 2023). In both, Π is globally defined and single-valued, so Algorithm 1 runs and the privacy guarantee holds. What does not carry over is the convergence analysis, which uses compactness for the uniform bounds it relies on. The same $\mathcal{S}_n^{++}$ under the affine-invariant metric (Pennec et al., 2006) falls outside on both counts. There is no ambient projection back to the cone, so unlike the flat cases nothing substitutes for Π and Property P2 fails. Moreover, the metric is iterate-dependent, so Property P1 fails. Algorithm 1 cannot be run there at all and PriRFed (Huang et al., 2026) remains the baseline.

Table 3: Where Algorithm 1 applies. Privacy (Theorem 10) holds for all four schemes of Table 2; the convergence rate (Corollary 16) for ProjAvg and RLAvg with RSGD.

Parameter space	Metric	Privacy	Conv. rate
Sphere, Stiefel, Grassmann	induced	✓	✓
$\mathcal{E}$ (incl. $\log(\mathcal{S}_n^{++})$)	Euclidean	✓	×
$\mathcal{S}_n^{++}$	affine-inv.	algorithm not runnable	

4 Privacy Analysis

The honest-but-curious server observes the whole transcript $\mathcal{V}_T$ (5), from which no client's record should be inferable: we require $\mathcal{V}_T$ to be private under the record-level adjacency of Definition 1. Section 4.1 recalls the tools we use from the tight f-DP calculus, Section 4.2 proves the guarantee, and Section 4.3 interprets it.

4.1 Privacy accounting in the f-DP calculus

We measure privacy with the trade-off-function calculus of Dong et al. (2022), which quantifies by hypothesis testing how hard two adjacent outputs are to distinguish, and composes tightly. This subsection is theirs; the results of Section 4.2 are ours.

Definition 4 (f-DP and GDP (Dong et al., 2022, Def. 2–4))**.** For distributions P, Q on a common space, the trade-off function $\mathcal{T}(P,Q) : \alpha \mapsto \inf\{\mathbb{E}_Q[1-\phi] : \mathbb{E}_P[\phi] \le \alpha\}$ is the smallest type-II error achievable at type-I level α; it is convex, continuous and non-increasing, and larger means harder to distinguish. A mechanism $\mathcal{A}$ is *f-DP* if $\mathcal{T}(\mathcal{A}(D), \mathcal{A}(D')) \ge f$ for every adjacent pair $D \bowtie D'$. Two trade-off functions matter below: $\mathrm{Id}(\alpha) = 1-\alpha$, the largest one, attained when the two releases are indistinguishable; and $G_\mu(\alpha) := \Phi(\Phi^{-1}(1-\alpha) - \mu)$, for testing $\mathcal{N}(0,1)$ against $\mathcal{N}(\mu,1)$, with Φ the standard normal cumulative distribution function. A mechanism is *μ-GDP* if it is G_μ-DP: larger μ is weaker, and $G_0 = \mathrm{Id}$ is perfect privacy.

No interpretability is lost in leaving (ε,δ)-DP: the two representations are dual, and an f-DP guarantee is an entire family of (ε,δ) ones.

Proposition 5 (Primal–dual correspondence (Dong et al., 2022, Prop. 3, 6 and Cor. 1))**.** *A mechanism is (ε,δ)-DP if and only if it is $f_{\varepsilon,\delta}$-DP, where $f_{\varepsilon,\delta}(\alpha) := \max\{0, 1-\delta-e^\varepsilon\alpha, e^{-\varepsilon}(1-\delta-\alpha)\}$. Conversely, a mechanism is f-DP for a symmetric f if and only if it is $(\varepsilon, \delta_f(\varepsilon))$-DP for every $\varepsilon \ge 0$, with $\delta_f(\varepsilon) = 1 + f^*(-e^\varepsilon)$ and $f^*(y) := \sup_{\alpha\in[0,1]}(y\alpha - f(\alpha))$. For $f = G_\mu$,*

$$\delta_{G_\mu}(\varepsilon) = \Phi\Big(-\tfrac{\varepsilon}{\mu} + \tfrac{\mu}{2}\Big) - e^\varepsilon\,\Phi\Big(-\tfrac{\varepsilon}{\mu} - \tfrac{\mu}{2}\Big). \tag{17}$$

Four operations generate every bound below, one per stage of a release: the noise sets the budget, post-processing preserves it, subsampling amplifies it, and composition accumulates it over the many releases a client's data produce. They are stated by Dong et al. (2022) for datasets of fixed size differing in exactly one record, which is the record-level adjacency of Definition 1.

Proposition 6 (The f-DP calculus (Dong et al., 2022, Thm. 1, Prop. 4, Def. 5–6, Thm. 4 and 9))**.** *Under record-substitution adjacency:*

(i) Gaussian mechanism. *If Θ maps databases to $\mathcal{E}$ with sensitivity* $\mathrm{sens}(\Theta) := \sup_{D \bowtie D'} \|\Theta(D) - \Theta(D')\|$, *then $\Theta + \xi$ with $\xi \sim \mathcal{N}(0, \sigma^2 I_{\mathcal{E}})$ is* $(\mathrm{sens}(\Theta)/\sigma)$*-GDP.*

(ii) Post-processing. *If $\mathcal{A}$ is f-DP and h is a (possibly randomised) map not depending on the database, $h \circ \mathcal{A}$ is f-DP.*

(iii) Subsampling. *Write $f_q := qf + (1-q)\mathrm{Id}$ and $C_q(f) := \min\{f_q, f_q^{-1}\}^{**}$, with $f^{-1}(\alpha) := \inf\{t : f(t) \le \alpha\}$ and ** the greatest convex minorant. If $\mathcal{A}$ is f-DP on B-record inputs, under substitution of one record of its input, and $\texttt{Sample}_B$ draws a uniform B-subset without replacement from an m-record dataset and does not release it, then $\mathcal{A} \circ \texttt{Sample}_B$ is $C_q(f)$-DP with $q = B/m$. The result $C_q(f)$ is symmetric, and for symmetric f the operator is non-increasing in q, with $C_1(f) = f$ and $C_0(f) = \mathrm{Id}$: rarer participation means more privacy.*

(iv) Adaptive composition. *For $f = \mathcal{T}(P, Q)$ and $g = \mathcal{T}(P', Q')$, let $f \otimes g := \mathcal{T}(P \times P', Q \times Q')$ be the optimal trade-off for the two releases seen jointly; it is commutative, associative, monotone for the pointwise order, symmetry-preserving, and satisfies $f \otimes \mathrm{Id} = f$. If $\mathcal{A}_1$ is f_1-DP and $\mathcal{A}_2(\cdot, y_1)$ is f_2-DP for every output y_1 of $\mathcal{A}_1$, their adaptive composition is $(f_1 \otimes f_2)$-DP; iterating, an N-fold adaptive composition of f-DP mechanisms is $f^{\otimes N}$-DP.*

Remark 7 (What is amplified). Amplification requires that the subsample itself is not released, so that the adversary cannot tell whether the substituted record was drawn (Proposition 6(iii)). It therefore applies to the minibatch, which never leaves the client, but not to the participation set S_t, which the server draws itself and keeps in $\mathcal{V}_T$: all amplification below is record-level, at rate $q_i = B/m_i$.

4.2 Record-level guarantee

The argument runs bottom-up, from one gradient to the whole transcript (Figure 2).

Per-step guarantee. The transcript is assembled from privatised gradients of the form (12), so the analysis starts with a single one. The following lemma chains three items of Proposition 6: clipping fixes the sensitivity and Gaussian noise sets the budget (i), the tangent projection carries it by post-processing (ii), and the minibatch draw amplifies it (iii).

Lemma 8 (One subsampled local step)**.** *Fix $C, \sigma_{\mathrm{dp}} > 0$, a client i, a point $x \in \mathcal{M}$ and a batch size $B \le m_i$. Let $\mathcal{B}$ be a uniform minibatch of size B drawn from $\mathcal{D}_i$ without replacement, independently of $\xi \sim \mathcal{N}(0, \sigma_{\mathrm{dp}}^2 I_{\mathcal{E}})$, and set $\mu_0 := 2C/(B\sigma_{\mathrm{dp}})$. Then the mechanism*

$$\mathcal{D}_i \;\longmapsto\; \tilde{g} \;:=\; \widehat{\mathrm{grad}}^{\,C} f_i(x; \mathcal{B}) + P_x \xi$$

is $C_{q_i}(G_{\mu_0})$-DP under substitution of one record of $\mathcal{D}_i$, at rate $q_i = B/m_i$.

Proof. Fix the batch and consider $\Theta_x(\mathcal{B}) := \widehat{\mathrm{grad}}^{\,C} f_i(x; \mathcal{B})$ as a mechanism on B-record inputs (see (11)). Each clipped per-sample gradient has ambient norm at most C, which the induced metric (1) makes its tangent norm too, so substituting one record of $\mathcal{B}$ moves the average by at most $\mathrm{sens}(\Theta_x) = 2C/B$. By Proposition 6(i) the ambient release $\Theta_x(\mathcal{B}) + \xi$ is $(\mathrm{sens}(\Theta_x)/\sigma_{\mathrm{dp}})$-GDP, that is μ_0-GDP. The tangent projector P_x is a fixed linear map that does not depend on the data, and Θ_x is already $T_x\mathcal{M}$-valued, so

$$P_x\big(\Theta_x(\mathcal{B}) + \xi\big) \;=\; \Theta_x(\mathcal{B}) + P_x\xi \;=\; \tilde{g},$$

and $\tilde{g}$ inherits the budget μ_0 by post-processing (Proposition 6(ii)), uniformly in x. Finally the batch is a uniform B-subset drawn without replacement and never released, so Proposition 6(iii) amplifies G_{μ_0} to $C_{q_i}(G_{\mu_0})$ at rate $q_i = B/m_i$. □

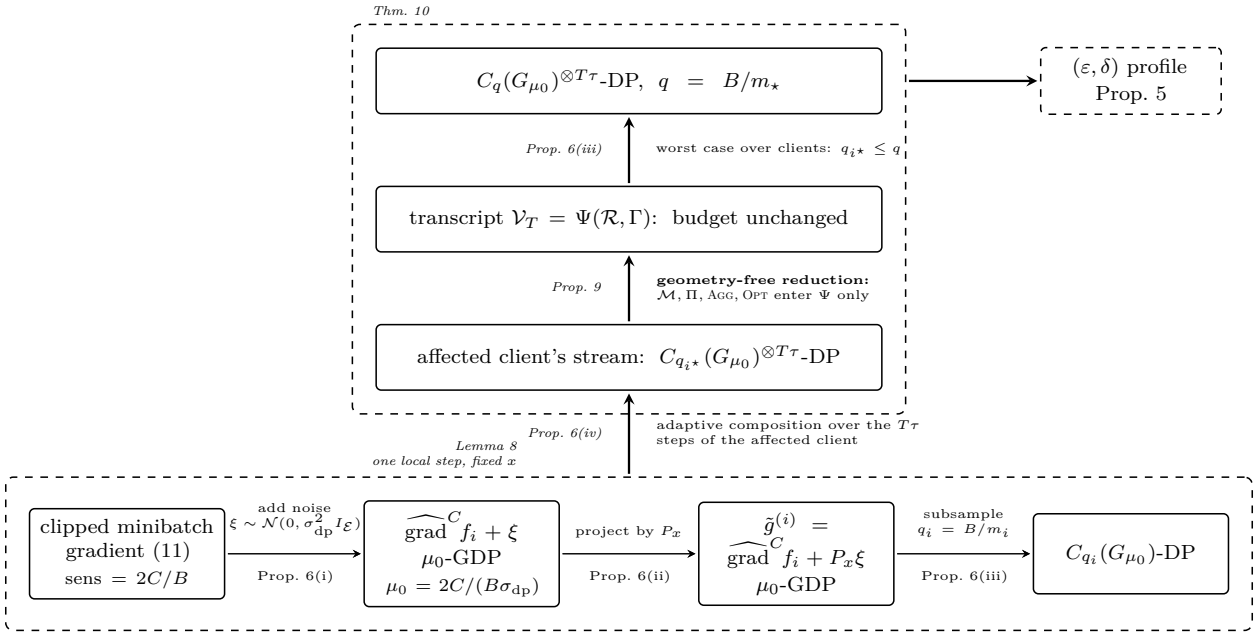


Figure 2: Structure of the privacy argument. Bottom: three operations of the f-DP calculus turn one clipped minibatch gradient into a private release at a fixed iterate x (Lemma 8). Above: three lifts carry that guarantee to the whole transcript, reconstructed by Ψ from the privatised gradients Γ and the data-independent protocol randomness $\mathcal{R}$. The middle lift is the pivot: the manifold, the aggregator and the local optimisers enter Ψ alone, so none of them appears in the budget.

Two features of this proof carry the rest of the section. First, the budget is settled on the ambient release and the projection is absorbed afterwards, so the manifold enters only as post-processing, as will every geometric map below. Second, the guarantee is uniform in x, so the lemma still applies when the iterate is produced by earlier releases.

From gradients to the transcript. It remains to lift Lemma 8 to the transcript $\mathcal{V}_T$. We split the protocol randomness into a part drawn independently of the data,

$$\mathcal{R} := \big(x_0,\ \{S_t\}_t,\ \{s_0^{(i)}\}_{i\in[n]},\ \{\omega_{t,j}^{(i)}\}_{t,j,i}\big),$$

consisting of model initialisation, client schedule, initial optimiser states and per-step optimiser randomness, and a part that depends on it: the stream of privatised gradients $\Gamma := \{\tilde{g}_{t,j}^{(i)}\}_{t,\, i\in S_t,\, j}$, each entry an instance of Lemma 8. The minibatch draws stay internal to the per-step mechanism, which is what keeps their amplification available (Remark 7).

Proposition 9 (Geometry-free reduction)**.** *There is a deterministic measurable map Ψ with $\mathcal{V}_T = \Psi(\mathcal{R}, \Gamma)$, in which the manifold $\mathcal{M}$, the maps $(P., \Pi)$, the aggregator Agg and the optimisers $\{\mathrm{Opt}^{(i)}\}$ act only as fixed, data-independent operations. Consequently, if Γ is f-DP conditionally on $\mathcal{R} = r$, uniformly in r, then $\mathcal{V}_T$ is f-DP.*

Proof. Reconstruct the protocol from $(\mathcal{R}, \Gamma)$ by induction on the rounds: at round t the active set S_t is read off $\mathcal{R}$, each selected client's local iterates follow from $x_{t,0}^{(i)} = x_t$ by (14), its update direction reading the gradient from Γ and the optimiser randomness from $\mathcal{R}$, and the submissions and next global iterate follow from Table 2. No step consults the data; this defines Ψ. For the transfer, $\mathcal{R}$ is drawn independently of the data, so its release is Id-DP and adaptive composition (Proposition 6 (iv)) makes $(\mathcal{R}, \Gamma)$ $(\mathrm{Id} \otimes f)$-DP, that is f-DP. Finally $\mathcal{V}_T$ is a post-processing of $(\mathcal{R}, \Gamma)$, hence f-DP by Proposition 6 (ii). □

In particular every iterate $x_{t,j}^{(i)}$, hence every projector $P_{x_{t,j}^{(i)}}$, is determined by $\mathcal{R}$ and the earlier releases, so Lemma 8 applies at each step with x fixed and adaptive composition chains them. The three pieces now assemble.

Theorem 10 (Privacy of DP-RFedProj under HBC)**.** *Let $m_\star := \min_{i\in[n]} m_i$, $q := B/m_\star$ and $\mu_0 := 2C/(B\sigma_{\mathrm{dp}})$. For every $T \geq 1$, the transcript $\mathcal{V}_T$ of Algorithm 1 is $C_q(G_{\mu_0})^{\otimes T\tau}$-DP under record-level adjacency. The budget depends only on $(C, B, m_\star, \sigma_{\mathrm{dp}})$ and the product $T\tau$, not on the number k of participating clients, not on $(\mathcal{M}, \Pi, \{\textsc{Opt}^{(i)}\})$, and not on the aggregator $\textsc{Agg} \in \{\textsc{ProjAvg}, \textsc{RLAvg}, \textsc{KFAvg}, \textsc{RetAvg}\}$.*

Proof. Fix $D \bowtie D'$ differing at client $i^\star$ and condition on $\mathcal{R}$. The datasets being disjoint, every other client's release has, conditionally on the prefix of the transcript, the same law under D and D': their factors are Id-DP and drop out of the tensor product, which is why k does not appear. Each release of $i^\star$ is the mechanism of Lemma 8 at an iterate fixed by the prefix, hence $C_{q_{i^\star}}(G_{\mu_0})$-DP, and adaptive composition (Proposition 6 (iv)) tensorises them. If $i^\star$ is not selected in every round it contributes fewer than $T\tau$ factors, and $f^{\otimes(N+1)} \leq f^{\otimes N}$ since $f \leq \mathrm{Id}$, so the $T\tau$-fold bound holds whatever the schedule. It cannot be reduced: the server draws S_t and records it in $\mathcal{V}_T$, so the number of selections is public and the guarantee must cover the schedule selecting $i^\star$ every round. This bound is uniform in $\mathcal{R}$, so Proposition 9 carries it to $\mathcal{V}_T$. Finally $q_{i^\star} \leq q$ and C_q is non-increasing in q (Proposition 6 (iii)), giving the cohort bound. □

Every scheme of Table 2 reads the same privatised gradients and does nothing else with the data, so the guarantee also covers Algorithm 1 run with KFAvg or RetAvg. It does not cover PriRFed (Huang et al., 2026): only the aggregation rule carries over, not the way the gradients are privatised, which there happens in the geometry of the current iterate (8) rather than in the fixed ambient one (12).

4.3 Interpreting the guarantee

Proposition 5 turns Theorem 10 into a full (ε, δ) curve, but a tensor power of subsampled trade-off functions is hard to read. We therefore report a single summary number, obtained from the f-DP central limit theorem. It is used for interpretation only, since the noise scales of Section 6 are calibrated by an exact accountant for Theorem 10 itself (Appendix C.4).

The bound is a composition of subsampled Gaussian mechanisms, which is the setting the f-DP central limit theorem covers: with $N := T\tau$, μ_0 fixed and $q\sqrt{N} \to c > 0$,

$$C_q(G_{\mu_0})^{\otimes N} \longrightarrow G_{\mu_\infty}, \qquad \mu_\infty = c\sqrt{2\chi_+^2(G_{\mu_0})}, \qquad \chi_+^2(G_{\mu_0}) = e^{\mu_0^2}\Phi\big(\tfrac{3\mu_0}{2}\big) + 3\Phi\big(-\tfrac{\mu_0}{2}\big) - 2,$$

uniformly on $[0, 1]$, where $\chi_+^2(f) := \int_0^1 (|f'| - 1)_+^2$ (Dong et al., 2022, Thm. 11 and Lem. 3). We therefore summarise a finite run by

$$\mu_{\mathrm{tot}} := q\sqrt{2N\,\chi_+^2(G_{\mu_0})}, \tag{18}$$

reporting the transcript as μ_{tot}-GDP and reading (ε, δ) off (17). Expanding at small μ_0 and using $q\mu_0 = 2C/(m_\star\sigma_{\mathrm{dp}})$ shows what drives it:

$$\mu_{\mathrm{tot}} = \frac{2C\sqrt{T\tau}}{m_\star\,\sigma_{\mathrm{dp}}}\Big(1 + \tfrac{\mu_0}{\sqrt{2\pi}} + O(\mu_0^2)\Big). \tag{19}$$

The budget grows as $\sqrt{T\tau}$ and is set by the clipping threshold relative to the noise, damped by the smallest client's dataset size. Two consequences are worth noting. The batch size cancels at leading order at fixed $T\tau$ and the growth in the number of rounds is $\sqrt{T}$ throughout.

5 Convergence Analysis

We now show that Algorithm 1 converges with several local steps, partial participation and heterogeneous clients at once, which is precisely the regime left open by Limitation 4. Three specialisations fix the algorithm under analysis.

(i) *Nearest-point projection.* We set Π to be the nearest-point map (2), which compactness makes well defined near $\mathcal{M}$ (Assumption 1) and which the linear tangent projector P_x approximates to first order at every base point (35). This linear model, available at each iterate, is what replaces the nonlinear inverse retraction of PriRFed (Huang et al., 2026).

(ii) *Local optimiser.* Each client runs plain Riemannian stochastic gradient descent (RSGD) with a rate $\eta > 0$ held constant across steps, rounds and clients, so that the local update (14) reads

$$x_{t,j+1}^{(i)} = \Pi\big(x_{t,j}^{(i)} - \eta\, \tilde{g}_{t,j}^{(i)}\big), \tag{20}$$

(iii) *Server aggregation.* We restrict to ProjAvg (15) and RLAvg (16), whose primitives are the two projections. The nonlinear lift and retract maps of KFAvg and RetAvg are precisely what the linearisation step (Lemma 18) is able to circumvent.

Throughout, $\alpha := \eta\tau$ denotes the aggregate per-round stepsize, in which the rates are stated. Prescribing α means running Algorithm 1 with $\eta = \alpha/\tau$.

We let $\mathcal{F}_t$ be the σ-algebra generated by all randomness of rounds $0, \dots, t-1$ and write $\mathbb{E}_t[\cdot] := \mathbb{E}[\cdot \mid \mathcal{F}_t]$. Conditioning on $\mathcal{F}_t$ thus fixes the current iterate x_t while leaving the round-t selection S_t, the minibatches and the noise vectors random.

5.1 Assumptions

Both the local and the aggregation stages of Algorithm 1 evaluate Π off the manifold. For the algorithm to be well-defined and the iterates controllable, Π must be single-valued throughout a neighbourhood of $\mathcal{M}$, a property guaranteed by proximal smoothness (Clarke et al., 1995; Davis et al., 2025): for $\gamma > 0$, define the open tube

$$U_{\mathcal{M}}(\gamma) := \{\, y \in \mathcal{E} : \operatorname{dist}(y, \mathcal{M}) < \gamma \,\}, \qquad \operatorname{dist}(y, \mathcal{M}) := \inf_{x \in \mathcal{M}} \|y - x\|,$$

and call $\mathcal{M}$ *γ-proximally smooth* if Π is single-valued on $U_{\mathcal{M}}(\gamma)$.

Compactness of $\mathcal{M}$ supplies the three facts the analysis relies on. First, a compact smooth embedded submanifold is proximally smooth. We fix the radius below.

Assumption 1 (Proximal smoothness)**.** $\mathcal{M}$ is 2γ-proximally smooth for some fixed $\gamma > 0$.

Second, F is continuous on the compact $\mathcal{M}$ and therefore attains finite extrema. Writing $F_\star := \inf_{x \in \mathcal{M}} F(x)$, any two points of $\mathcal{M}$ differ in objective value by at most

$$\delta_F := \sup_{\mathcal{M}} F - F_\star. \tag{21}$$

Third, each continuous ambient gradient $\nabla f(\cdot\,; z)$ is bounded on $\mathcal{M}$, uniformly in z, which lets us define the gradient bound

$$G := \max_{i \in [n]} \sup_{x \in \mathcal{M}} \max_{1 \le \ell \le m_i} \|\nabla f(x; z_{i,\ell})\| \;<\; \infty.$$

Because clipping acts on each per-record gradient before averaging, no bound on the averaged objective can be used in its place: the analysis must control every per-sample loss $f(\cdot\,; z)$ individually and uniformly in z, a stronger requirement than the per-objective smoothness of Deng & Hu (2025).

Assumption 2 (Smoothness)**.** Each per-sample loss $f(\cdot\,; z)$ has ℓ_f-Lipschitz ambient gradient on $\operatorname{conv}(\mathcal{M})$, with ℓ_f independent of z:

$$\|\nabla f(x; z) - \nabla f(y; z)\| \le \ell_f \|x - y\|, \qquad x, y \in \operatorname{conv}(\mathcal{M}),\ z \in \mathcal{Z}.$$

Because ℓ_f is uniform in z, every per-sample loss obeys the same quadratic bound, which averaging passes to F as defined in (3). Under Assumptions 1 and 2, the following descent inequality (Deng & Hu, 2025, Lem. 4.2) holds on $\mathcal{M}$, with $L_{\mathcal{M}} := \ell_f + \frac{G}{2\gamma}$:

$$F(y) \le F(x) + \langle \operatorname{grad} F(x),\, y - x \rangle + \frac{L_{\mathcal{M}}}{2} \|y - x\|^2. \tag{22}$$

Local steps (Algorithm 1, Line 10) use minibatch gradients, so the analysis must control their dispersion.

Assumption 3 (Stochastic gradients)**.** There exists $\sigma_{\mathrm{st}}^2 < \infty$ such that, for all $i \in [n]$ and $x \in \mathcal{M}$,

$$\frac{1}{m_i} \sum_{\ell=1}^{m_i} \| \operatorname{grad} f(x; z_{i,\ell}) - \operatorname{grad} f_i(x) \|^2 \le \sigma_{\mathrm{st}}^2. \tag{23}$$

For a threshold $C > 0$, a client i, and $x \in \mathcal{M}$, let us write the clipped per-client full gradient as

$$\operatorname{grad}^C f_i(x) := \frac{1}{m_i} \sum_{\ell=1}^{m_i} \operatorname{clip}_C\big(\operatorname{grad} f(x; z_{i,\ell})\big), \tag{24}$$

whose minibatch estimator is no other than (11). Assumption 3 bounds the within-client variance, which suffices to control the variance of the clipped minibatch gradient (11) under sampling without replacement.

Lemma 11 (Clipped stochastic-gradient bias and variance)**.** *Under Assumption 3, the minibatch estimator* $\widehat{\operatorname{grad}}^C f_i(x; \mathcal{B})$ *of* (11) *is unbiased for the clipped per-client gradient* (24)*, with*

$$\mathbb{E}_{\mathcal{B}}\Big[\big\| \widehat{\operatorname{grad}}^C f_i(x; \mathcal{B}) - \operatorname{grad}^C f_i(x) \big\|^2 \Big] \;\le\; \rho_i \, \sigma_{\mathrm{st}}^2, \qquad \rho_i := \frac{m_i - B}{B(m_i - 1)}, \tag{25}$$

the expectation taken over the minibatch randomness.

Our next assumption bounds the heterogeneity driving the drift of the client gradients from the gradient of the objective $\operatorname{grad} F = \frac{1}{n} \sum_{i=1}^n \operatorname{grad} f_i$.

Assumption 4 (Client-gradient heterogeneity)**.** There exists $\sigma_{\mathrm{het}}^2 \ge 0$ such that

$$\sup_{x \in \mathcal{M}} \frac{1}{n} \sum_{i=1}^n \| \operatorname{grad} f_i(x) - \operatorname{grad} F(x) \|^2 \le \sigma_{\mathrm{het}}^2.$$

Clipping perturbs each $\operatorname{grad} f_i(x)$ by a bias $b_i^C(x) := \operatorname{grad}^C f_i(x) - \operatorname{grad} f_i(x)$. These biases differ across clients, which adds to the heterogeneity already present. We measure their spread by

$$\sigma_{\mathrm{clip}}^2 := \sup_{x \in \mathcal{M}} \frac{1}{n} \sum_{i=1}^n \Big\| b_i^C(x) - \frac{1}{n} \sum_{j=1}^n b_j^C(x) \Big\|^2. \tag{26}$$

This constant is explicit: $\|\operatorname{clip}_C(v) - v\| = (\|v\| - C)_+$ and $\|\operatorname{grad} f(x; z)\| \le G$ give $\sigma_{\mathrm{clip}} \le (G - C)_+$, which vanishes whenever $C \ge G$.

Lemma 12 (Clipped-gradient heterogeneity)**.** *Under Assumption 4,*

$$\sup_{x \in \mathcal{M}} \frac{1}{n} \sum_{i=1}^n \Big\| \operatorname{grad}^C f_i(x) - \tfrac{1}{n} \sum_{j=1}^n \operatorname{grad}^C f_j(x) \Big\|^2 \le (\sigma_{\mathrm{het}} + \sigma_{\mathrm{clip}})^2. \tag{27}$$

The per-client clipping biases also have a nonzero mean, which shifts the aggregated signal away from $\operatorname{grad} F$. It is quantified by the clipping-bias floor

$$\beta_C := \sup_{x \in \mathcal{M}} \Big\| \tfrac{1}{n} \sum_{i=1}^n b_i^C(x) \Big\| \;<\; \infty. \tag{28}$$

5.2 Per-round noise control

The injected DP noise (Algorithm 1, Lines 9–10) is Gaussian and hence unbounded: with small but positive probability a single draw ejects the pre-projection point from the tube $U_{\mathcal{M}}(2\gamma)$, where Π ceases to be single-valued. We handle this by conditioning each round on a high-probability event on which every Gaussian noise vector $\xi_{t,j}^{(i)}$ is bounded, with the threshold calibrated so that the event fails only with small probability. On the rare failure, the projection still returns a point of $\mathcal{M}$, so the objective increases by at most δ_F over the round.

Definition 13 (Per-round noise control event)**.** Fix a noise-control margin $c_0 > 0$ and define the noise threshold

$$R := \sigma_{\mathrm{dp}}\big(\sqrt{D} + \sqrt{c_0}\big), \tag{29}$$

together with the per-round noise control event

$$\mathcal{G}_t := \big\{\, \|\xi_{t,j}^{(i)}\| \le R \text{ for all } (i,j) \in S_t \times \{0,\dots,\tau-1\} \,\big\}, \quad t \in \{0,\dots,T\}. \tag{30}$$

On $\mathcal{G}_t$, each privatised gradient (12) is bounded,

$$\|\tilde{g}_{t,j}^{(i)}\| \le C + \|\xi_{t,j}^{(i)}\| \le G_{\mathrm{eff}}, \qquad G_{\mathrm{eff}} := C + R, \tag{31}$$

which is the bound the convergence analysis relies on. Whichever clients are selected, $\mathcal{G}_t$ constrains the same number $k\tau$ of i.i.d. noise draws, so the probability of the complement $\mathcal{G}_t^c$ does not depend on the round index t. Gaussian norm concentration at the threshold (29), followed by a union bound over the $k\tau$ draws, gives the following, proved in Appendix B.6.

Lemma 14 (Per-round failure probability)**.** *The failure probability $\pi := \Pr(\mathcal{G}_t^c)$ is the same for every round $t \in \{0,\dots,T-1\}$ and satisfies $\pi \le k\tau\, e^{-c_0/2}$.*

The margin c_0 trades failure probability against the gradient bound: larger c_0 shrinks $\Pr(\mathcal{G}_t^c)$ but inflates R, and with it the effective gradient bound $G_{\mathrm{eff}} = C + R$. It is a free parameter, held fixed in the finite-T bound (Theorem 15) and tuned as $c_0 \asymp \log T$ for the asymptotic rate (Corollary 16).

5.3 Main results

Our main guarantee is a finite-horizon stationarity bound for Algorithm 1. Because ProjAvg and RLAvg induce the same server update to first order (see Lemma 19), a single analysis covers both, and they attain identical rates. We state the result once for the two. Here and below, $a \lesssim b$ means $a \le c\, b$ for a universal numerical constant c, made explicit in Appendix B.7.

Theorem 15. *Under Assumptions 1–4, for every $T \ge 1$ and every constant stepsize $\alpha \le \min\{\gamma/(4G_{\mathrm{eff}}),\, 1/(4L_{\mathcal{M}})\}$,*

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\big[\|\operatorname{grad} F(x_t)\|^2\big] \;\lesssim\; \frac{F(x_0) - F_\star}{(1-\pi)\,\alpha\, T} + \frac{L_{\mathcal{M}}\,\alpha}{k}\left[\frac{n-k}{n-1}(\sigma_{\mathrm{het}} + \sigma_{\mathrm{clip}})^2 + \frac{1}{\tau}\big(\bar{\rho}\,\sigma_{\mathrm{st}}^2 + d_{\mathcal{M}}\sigma_{\mathrm{dp}}^2\big)\right] + \beta_C^2 + K\,\alpha^2 + \frac{\pi\,\delta_F}{(1-\pi)\,\alpha}. \tag{32}$$

where $\bar{\rho} := \frac{1}{n}\sum_{i=1}^n \rho_i$ and the constant $K > 0$ depends on the problem constants and on c_0.

Unlike prior analyses (Li & Ma, 2022; Huang et al., 2026), the bound holds for an arbitrary number of local updates and partial participation at once, the regime Limitation 4 left open. The projection-based linearisation makes both accessible together, without the curvature-dependent constants of intrinsic aggregation. Two dimensions enter (32) at different orders: the intrinsic manifold dimension $d_{\mathcal{M}}$ sits in the leading DP-noise term, the ambient dimension D only in the remainder $K\alpha^2$ and the stepsize ceiling, through G_{eff} (29).

Theorem 15 holds for any fixed margin c_0; letting c_0 grow with the horizon yields the asymptotic rate of Corollary 16.

Corollary 16. *Under Assumptions 1–4, for T large enough,*

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\big[\|\operatorname{grad} F(x_t)\|^2\big] = O\left(\frac{1}{\sqrt{T}}\left[1 + \frac{1}{k}\left(\frac{n-k}{n-1}\,(\sigma_{\mathrm{het}} + \sigma_{\mathrm{clip}})^2 + \frac{\bar{\rho}\,\sigma_{\mathrm{st}}^2 + d_{\mathcal{M}}\,\sigma_{\mathrm{dp}}^2}{\tau}\right)\right] + \beta_C^2\right). \tag{33}$$

Proof. Taking $\alpha \asymp 1/\sqrt{T}$ collapses the optimisation and noise terms into $O(1/\sqrt{T})$. The choice $c_0 = 4\log T + 2\log(k\tau)$ then forces $\pi \le T^{-2}$ (Lemma 14), so it is $O(T^{-3/2})$. This margin leaves the heterogeneity, stochastic-gradient, DP-noise, and clipping-bias terms untouched, inflating only (through G_{eff}) the subdominant remainder $K\alpha^2$ to $o(1/\sqrt{T})$. Absorbing these into the $O(\cdot)$ gives the rate, valid for T large enough for the stepsize constraint to be satisfied. □

Interpretation. The bound splits into a vanishing part and a T-independent floor β_C^2, so the iterates converge to a β_C-neighbourhood of stationarity rather than to a stationary point unless $C \geq G$. Term by term:

(i) *Optimisation error*, $O(1/\sqrt{T})$: the standard nonconvex stationarity rate, independent of k, τ, and the privacy parameters.

(ii) *Client heterogeneity*, $(\sigma_{\text{het}} + \sigma_{\text{clip}})^2$, scaled by $\frac{1}{k} \cdot \frac{n-k}{n-1}$: the price of partial participation. It combines the intrinsic gradient dissimilarity σ_{het} (Assumption 4) with the clipping-induced dispersion σ_{clip} of the per-client biases. The factor $\frac{n-k}{n-1}$ is the finite-population sampling correction: it vanishes at full participation ($k = n$) and is largest when a single client is sampled ($k = 1$).

(iii) *Stochastic-gradient noise*, $\bar{\rho}\,\sigma_{\text{st}}^2/(k\tau)$: the minibatch variance, averaged over the cohort ($1/k$) and the local trajectory ($1/\tau$). The factor $\bar{\rho} = \frac{1}{n}\sum_i \frac{m_i - B}{B(m_i - 1)}$ is the without-replacement sampling factor (Lemma 11); it decreases as B grows and vanishes at full local batches ($B = m_i$), removing this term entirely.

(iv) *DP noise*, $d_{\mathcal{M}}\,\sigma_{\text{dp}}^2/(k\tau)$: the dominant privacy cost. Like the stochastic term it is damped by $1/(k\tau)$, but it cannot be removed by tuning: only a smaller σ_{dp}, hence weaker privacy, shrinks it. It scales with the intrinsic dimension $d_{\mathcal{M}}$, the concrete benefit of the projection-based design. Substituting the noise scale that Section 4 requires for a target budget μ_{tot}, given by (19), turns this term into

$$\frac{d_{\mathcal{M}}\,\sigma_{\text{dp}}^2}{k\tau} \;\simeq\; \frac{4\,d_{\mathcal{M}}\,C^2\,T}{k\,\mu_{\text{tot}}^2\,m_\star^2}. \tag{34}$$

At a fixed budget, the privacy cost thus grows linearly in the number of rounds and decreases with the cohort size and with the square of the smallest dataset size. It does not depend on τ: the budget spent on extra local steps is offset by the $1/\tau$ damping, so local work is free in privacy–utility terms.

(v) *Clipping-bias floor*, β_C^2: the systematic error from gradient clipping (Eq. (28)), the only term that does not decay with T, k, or τ. It is zero when the clip is inactive ($C \geq G$) and otherwise a fixed positive floor. Enlarging C shrinks it but raises the noise scale needed for a given privacy level, exposing a clip–noise trade-off.

5.4 Outline of the proof of Theorem 15

The aggregate stepsize $\alpha = \eta\tau$ is the central quantity governing both tube stability and the convergence rate. We write $\mathbb{E}_t^{\mathcal{G}}[\cdot] := \mathbb{E}[\cdot \mid \mathcal{F}_t, \mathcal{G}_t]$ for the conditional expectation $\mathbb{E}_t$ further restricted to the noise-control event $\mathcal{G}_t$ (Def. 13).

Our convergence analysis proceeds in three main steps: (i) a tube-stability and linearisation analysis on the noise-control event $\mathcal{G}_t$; (ii) a first-order decomposition of the server update common to both aggregators on $\mathcal{G}_t$; and (iii) a one-round expected-descent inequality, which we telescope over the T rounds.

Tube stability and linearisation on $\mathcal{G}_t$. By Assumption 1, the nearest-point projection Π is single-valued on the tube $U_{\mathcal{M}}(2\gamma)$. Each projection step displaces its argument off $\mathcal{M}$ by at most ηG_{eff} (see (31)), so a stepsize constraint of the form $\alpha \leq \gamma/(4G_{\text{eff}})$ keeps every pre-projection point within the inner tube $\overline{U_{\mathcal{M}}(\gamma)} \subset U_{\mathcal{M}}(2\gamma)$, where Π is single-valued. The following lemma makes this precise.

Lemma 17 (Tube stability on $\mathcal{G}_t$)**.** *On $\mathcal{G}_t$, if $\alpha \leq \gamma/(4G_{\text{eff}})$, then for every $t \in \{0, \ldots, T-1\}$, and $i \in S_t$:*

(a) ***Local-iterate bound:***

$$\|x_{t,j}^{(i)} - x_t\| \leq 2G_{\text{eff}}\,\eta j \leq \gamma/2, \quad j = 0, \ldots, \tau$$

(b) ***Pre-projection tube containment:*** *for $j \leq \tau - 1$, the pre-projection point $x_{t,j}^{(i)} - \eta\,\tilde{g}_{t,j}^{(i)}$ lies in $U_{\mathcal{M}}(\gamma)$, so its projection $x_{t,j+1}^{(i)}$ is unique.*

(c) ***Aggregation feasibility:*** *for the two averaged server increments*

$$\Delta_t^{\textsc{ProjAvg}} := \tfrac{1}{k}\sum_{i\in S_t}\big(x_{t,\tau}^{(i)} - x_t\big), \qquad \Delta_t^{\textsc{RLAvg}} := \tfrac{1}{k}\sum_{i\in S_t} P_{x_t}\big(x_{t,\tau}^{(i)} - x_t\big),$$

one has $\|\Delta_t^{\textsc{RLAvg}}\| \le \|\Delta_t^{\textsc{ProjAvg}}\| \le 2G_{\mathrm{eff}}\alpha \le \gamma/2$*, so that the server pre-projection point* $x_t + \Delta_t^{\textsc{Agg}}$ *(*$\textsc{Agg} \in \{\textsc{ProjAvg}, \textsc{RLAvg}\}$*) lies in* $U_{\mathcal{M}}(\gamma)$ *and its projection* x_{t+1} *is unique.*

With the iterates confined to the tube, we can linearise each client's local trajectory, even when $\tau > 1$.

The exponential-map scheme of Li & Ma (2022) and the retraction scheme of Huang et al. (2026) both represent each local step as a tangent vector at the *current* local iterate $x_{t,j}^{(i)}$,

$$\mathrm{Log}_{x_{t,j}^{(i)}}\big(x_{t,j+1}^{(i)}\big) \ \Big(\text{resp. } \mathrm{R}^{-1}_{x_{t,j}^{(i)}}\big(x_{t,j+1}^{(i)}\big)\Big) = -\eta\, v_{t,j}^{(i)} \in T_{x_{t,j}^{(i)}}\mathcal{M},$$

where $v_{t,j}^{(i)}$ is the local update direction — variance-reduced for Li & Ma (2022), privatised for Huang et al. (2026). Because Log and R^{-1} are nonlinear and anchored at the base point $x_{t,j}^{(i)}$, which moves from one step to the next, the τ increments live in τ different tangent spaces and cannot be added directly. Reconciling them requires transporting between these spaces, a curvature-dependent argument (Zhang et al., 2016) that prior analyses carry out only when a single client participates ($k = 1$) — the $\tau > 1$ side of Limitation 4.

The nearest-point projection removes this obstruction through a single property: its first-order behaviour at every base point is given by the tangent projector. Precisely, there is a constant $C_{\mathrm{qr}} > 0$, depending only on γ and the second-order geometry of $\mathcal{M}$, such that (Deng & Hu, 2025, Lem. 4.3)

$$\Pi(x+u) = x + P_x\, u + w(x,u), \qquad \|w(x,u)\| \le C_{\mathrm{qr}}\|u\|^2, \qquad x \in \mathcal{M},\ \|u\| \le \gamma. \tag{35}$$

The local step (20) now moves the iterate by the ambient displacement $x_{t,j+1}^{(i)} - x_{t,j}^{(i)}$, a difference of two points of $\mathcal{E}$ rather than a tangent vector pinned to $x_{t,j}^{(i)}$. As all such displacements lie in the single space $\mathcal{E}$, the τ local steps telescope to the total displacement $x_{t,\tau}^{(i)} - x_t$, leaving no per-step tangent spaces to reconcile. Each client's submission is in turn an ambient displacement from the common point x_t, so the server aggregation reduces to an arithmetic mean of these displacements, not a composition of nonlinear maps over the sampled clients (made precise in Lemma 19). The two sides of Limitation 4 — many local steps and many clients — are thus handled together.

Passing from the exact step to this linearisation replaces each local projector $P_{x_{t,j}^{(i)}}$ by the common P_{x_t}. Consecutive base points lie $O(\eta j G_{\mathrm{eff}}) = O(\alpha G_{\mathrm{eff}})$ apart in the tube (Lemma 17(a)), so each replacement costs $O(\eta\alpha)$; summed with the quadratic remainder of (35) over the τ steps, the total error is $O(\alpha^2)$. The multi-step case therefore incurs no τ-dependent penalty, and we obtain the following linearisation.

Lemma 18 (Linearisation on $\mathcal{G}_t$)**.** *On* $\mathcal{G}_t$*, under* $\alpha \le \gamma/(4G_{\mathrm{eff}})$*, every client* $i \in S_t$ *satisfies*

$$\begin{aligned} x_{t,\tau}^{(i)} - x_t = &-\alpha\ \mathrm{grad}^C f_i(x_t) \\ &- \eta\sum_{j=0}^{\tau-1} P_{x_t}\big(\widehat{\mathrm{grad}}^C f_i(x_{t,j}^{(i)}; \mathcal{B}_{t,j}^{(i)}) - \mathrm{grad}^C f_i(x_{t,j}^{(i)})\big) \\ &- \eta\sum_{j=0}^{\tau-1} P_{x_t}\big(\xi_{t,j}^{(i)}\big) + O(\alpha^2), \end{aligned} \tag{36}$$

where the remainder is independent of T *(for fixed* c_0*) and depends only on* ℓ_f, G, C, R *and the geometry of* $\mathcal{M}$*.*

Server update decomposition on $\mathcal{G}_t$. Although they aggregate differently, ProjAvg and RLAvg induce the same server update to first order. Both take the form $x_{t+1} = \Pi(x_t + \Delta_t^{\text{Agg}})$, $\text{Agg} \in \{\text{ProjAvg}, \text{RLAvg}\}$. Since $\|\Delta_t^{\text{Agg}}\| \le 2G_{\text{eff}}\alpha \le \gamma/2$ on $\mathcal{G}_t$ (Lemma 17(c)), the quadratic-remainder expansion (35) gives

$$x_{t+1} - x_t = P_{x_t}\Delta_t^{\text{Agg}} + O(\alpha^2).$$

The leading terms coincide: $P_{x_t}\Delta_t^{\text{ProjAvg}} = \Delta_t^{\text{RLAvg}}$ by linearity of P_{x_t}, and $P_{x_t}\Delta_t^{\text{RLAvg}} = \Delta_t^{\text{RLAvg}}$ since $\Delta_t^{\text{RLAvg}} \in T_{x_t}\mathcal{M}$. Averaging the linearisation (36) over $i \in S_t$ then expresses this common update as a clipped-gradient signal, a stochastic-gradient noise term ζ_t^{st}, a DP-noise term ζ_t^{dp}, and a quadratic remainder.

Lemma 19 (Server update decomposition on $\mathcal{G}_t$)**.** *On $\mathcal{G}_t$, under $\alpha \le \gamma/(4G_{\text{eff}})$, both ProjAvg and RLAvg satisfy*

$$x_{t+1} - x_t = -\alpha\,\frac{1}{k}\sum_{i\in S_t}\operatorname{grad}^C f_i(x_t) + \zeta_t^{\text{st}} + \zeta_t^{\text{dp}} + O(\alpha^2), \tag{37}$$

where the aggregated noise terms are

$$\zeta_t^{\text{st}} := -\frac{\eta}{k}\sum_{i\in S_t}\sum_{j=0}^{\tau-1} P_{x_t}\big(\widehat{\operatorname{grad}}^C f_i(x_{t,j}^{(i)};\mathcal{B}_{t,j}^{(i)}) - \operatorname{grad}^C f_i(x_{t,j}^{(i)})\big),$$

$$\zeta_t^{\text{dp}} := -\frac{\eta}{k}\sum_{i\in S_t}\sum_{j=0}^{\tau-1} P_{x_t}\big(\xi_{t,j}^{(i)}\big),$$

and the $O(\alpha^2)$ remainder is independent of T (for fixed c_0), depending only on ℓ_f, G, C, R and the geometry of $\mathcal{M}$.

Conditionally on $\mathcal{F}_t$ and $\mathcal{G}_t$, the three terms of (37) have controlled moments. The clipped-gradient signal $\frac{1}{k}\sum_{i\in S_t}\operatorname{grad}^C f_i(x_t)$ has $\mathbb{E}_t^{\mathcal{G}}$-mean equal to $\operatorname{grad} F(x_t)$ up to an additive clipping bias of norm at most β_C (defined in (28)), and client-sampling variance at most $\frac{n-k}{k(n-1)}(\sigma_{\text{het}} + \sigma_{\text{clip}})^2$ by Lemma 12. The two noise terms are $\mathbb{E}_t^{\mathcal{G}}$-zero-mean, with

$$\mathbb{E}_t^{\mathcal{G}}\big[\|\zeta_t^{\text{st}}\|^2\big] \le \frac{\alpha^2}{k\tau}\,\bar\rho\,\sigma_{\text{st}}^2, \qquad \mathbb{E}_t^{\mathcal{G}}\big[\|\zeta_t^{\text{dp}}\|^2\big] \le \frac{\alpha^2}{k\tau}\,d_{\mathcal{M}}\sigma_{\text{dp}}^2.$$

The first follows from Lemma 11; the second from the rank-$d_{\mathcal{M}}$ projector P_{x_t} collapsing the isotropic DP noise onto the tangent space, so the variance scales with the intrinsic dimension $d_{\mathcal{M}}$ rather than the ambient D. Finally, the signal and the two noise terms are mutually uncorrelated in $\mathbb{E}_t^{\mathcal{G}}$-expectation (see Lemma 21 in Appendix B.7).

One-round expected descent. We turn the server-update decomposition into a descent guarantee through the ambient smoothness inequality (22), applied to the consecutive iterates $x_t, x_{t+1} \in \mathcal{M}$:

$$F(x_{t+1}) \le F(x_t) + \langle \operatorname{grad} F(x_t),\, x_{t+1} - x_t\rangle + \tfrac{L_{\mathcal{M}}}{2}\|x_{t+1} - x_t\|^2.$$

This is the extrinsic counterpart of the geodesic and retraction smoothness inequalities used by Li & Ma (2022) and Huang et al. (2026), and shared by Zhang et al. (2024); Wang et al. (2025). As in the linearisation step, it lets us bypass exponential, logarithmic, and inverse-retraction maps.

Controlled-noise round. On $\mathcal{G}_t$, substituting (37), taking $\mathbb{E}_t^{\mathcal{G}}[\cdot]$, and using that $\zeta_t^{\text{st}}, \zeta_t^{\text{dp}}$ are zero-mean with the second-moment bounds of Lemma 21, the cross terms cancel and we obtain, for $\alpha \le \min\{\gamma/(4G_{\text{eff}}),\, 1/(4L_{\mathcal{M}})\}$,

$$\mathbb{E}_t^{\mathcal{G}}[F(x_{t+1})] \;\lesssim\; F(x_t) - \alpha\,\|\operatorname{grad} F(x_t)\|^2 + V, \tag{38}$$

where the variance term V separates the clipping bias, the client heterogeneity and clipping dispersion, and the stochastic and DP noise:

$$V = (2\alpha + L_{\mathcal{M}}\alpha^2)\,\beta_C^2 + L_{\mathcal{M}}\alpha^2\,\frac{n-k}{k(n-1)}\,(\sigma_{\text{het}} + \sigma_{\text{clip}})^2 + \frac{L_{\mathcal{M}}\alpha^2}{k\tau}\big(\bar\rho\,\sigma_{\text{st}}^2 + d_{\mathcal{M}}\sigma_{\text{dp}}^2\big) + K\,\alpha^3,$$

with K a constant independent of T and α (for fixed c_0).

Failure round and telescoping. On $\mathcal{G}_t^c$, the projection still returns a point of $\mathcal{M}$, so the worst-case bound $F(x_{t+1}) - F(x_t) \le \delta_F$ holds by (21). Writing $\pi := \Pr(\mathcal{G}_t^c)$ and combining the two cases,

$$\begin{aligned}\mathbb{E}_t[F(x_{t+1})] &= (1-\pi)\,\mathbb{E}_t^{\mathcal{G}}[F(x_{t+1})] + \pi\,\mathbb{E}\big[F(x_{t+1}) \mid \mathcal{F}_t, \mathcal{G}_t^c\big] \\ &\lesssim (1-\pi)\big[F(x_t) - \alpha\|\operatorname{grad} F(x_t)\|^2 + V\big] + \pi\,\big[F(x_t) + \delta_F\big].\end{aligned}$$

Taking full expectations, summing over $t = 0, \dots, T-1$, and telescoping yields Theorem 15.

6 Numerical experiments

We evaluate the framework on EEG motor-imagery classification, a setting in which data confidentiality motivates private federated learning and in which geometry is central: spatial covariance matrices are highly effective EEG descriptors (Barachant et al., 2012), motivating methods that respect their SPD structure. We compare SPDNet (Huang & Van Gool, 2017), whose BiMap weight lies on the Stiefel manifold, with EEGNet (Lawhern et al., 2018), a Euclidean convolutional baseline on raw signals.

Each record is a labelled trial $z = (X, y)$ with $y \in \{1, \dots, n_{\mathrm{cl}}\}$, given to the model as the raw signal X for EEGNet and as its sample covariance matrix Σ for SPDNet. Client i holds the m_i trials of a fixed set of subjects and never shares them, and the n clients solve

$$\min_{x \in \mathcal{M}} \Big\{ F(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x) \Big\}, \qquad f_i(x) := \frac{1}{m_i}\sum_{\ell=1}^{m_i} f(x; z_{i,\ell}), \qquad f(x; z) = -\log\big[h_x(z)\big]_y, \tag{39}$$

where $h_x(z)$ is the vector of class probabilities the model with parameters x assigns to z, making f the cross-entropy loss. The two architectures differ only in $\mathcal{M}$: Euclidean for EEGNet, a product of a Stiefel factor and a Euclidean one for SPDNet.

One model is trained across all subjects and tested on their held-out trials, under a class-stratified 75/10/15 split. Every reported number is a mean over the same 10 seeds; each seed redraws the split and the initialisation, and every arm sees the same split within a seed.

Sections 6.3 and 6.4 describe the centralized baselines and the federated protocol; Sections 6.5 to 6.7 report the privacy–utility trade-off, the aggregation rules and the comparison with PriRFed.

6.1 Datasets and preprocessing

We use three datasets from the MOABB benchmark (Chevallier et al., 2024), summarized in Table 4. They differ in electrode count, which sets the size of the covariances SPDNet operates on, and in cohort size, which sets the number of clients. BNCI2014_001 (Tangermann et al., 2012) has the fewest electrodes, over four classes. Weibo2014 (Yi et al., 2014) and Cho2017 (Cho et al., 2017) have comparable electrode counts: the first is a seven-class task on ten subjects, the second binary on 52, allowing a moderate scalability evaluation.

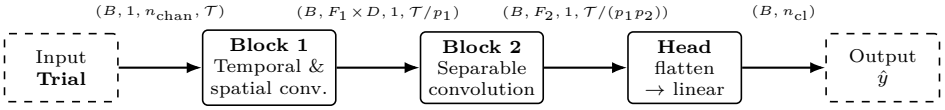
Table 4: The three EEG motor-imagery datasets.

Dataset	n_{subj}	n_{chan}	n_{cl}	Native S.R. (Hz)	$\mathcal{T}$	Trials/class
BNCI2014_001	9	22	4	250	513	144
Weibo2014	10	60	7	200	513	80
Cho2017	52	64	2	512	384	100

All signals are band-pass filtered to $[8, 32]$ Hz and resampled from their native rate to $f_s = 128$ Hz, leaving $\mathcal{T}$ time samples per trial. Trial counts are per subject and approximate; within a dataset subjects contribute comparable numbers, so clients hold comparable amounts of data. Each trial is then normalized on its own, either with per-channel standardization for EEGNet or unit-trace covariance for SPDNet, which removes the recording gain and keeps every trial independent of the rest.

6.2 Model architectures

EEGNet. EEGNet (Lawhern et al., 2018) is a compact convolutional network operating directly on raw trials (Figure 3). Its first block applies a temporal convolution, then a depthwise spatial convolution that learns electrode combinations for each temporal filter; its second block refines the result with a separable convolution. Each block ends with a batch normalization, an ELU nonlinearity, average pooling and dropout, and a linear layer maps the flattened features to class scores. We retain the published architecture unchanged, including the max-norm constraints on the depthwise spatial and classifier weights and the temporal kernel sizes, which scale with f_s.



F_1=8 (temporal filters), D=2 (depth multiplier), F_2=$F_1 \times D$=16, p_1=4, p_2=8 (pooling factors).
$K_t = \max(\lfloor 0.5 f_s \rceil, 32)$ (temporal, ≈500 ms), $K_s = \max(\lfloor f_s/(2p_1) \rceil, 8)$.

Figure 3: EEGNet pipeline. Shapes above each arrow, with B the batch size and $\mathcal{T}$ the number of time samples.

SPDNet. SPDNet (Huang & Van Gool, 2017) operates on the per-trial sample covariance $\Sigma_j = (\mathcal{T} - 1)^{-1} \bar{X}_j \bar{X}_j^\top \in \mathcal{S}^{++}_{n_{\text{chan}}}$, with $\bar{X}_j$ the centered trial (Figure 4). A BiMap layer $\Sigma \mapsto W^\top \Sigma W$, with $W \in \text{St}(n_{\text{chan}}, d)$, reduces dimension while preserving SPD structure; a ReEig layer rectifies eigenvalues below a floor ε_{R}; a LogEig layer $U \Lambda U^\top \mapsto U \log(\Lambda) U^\top$ maps to the tangent space at the identity; and a softmax classifier acts on the $d(d+1)/2$ half-vectorized features. The parameters are optimized end-to-end with Adam (Kingma & Ba, 2015), the Stiefel weight W via tangent-space local trivialization (Lezcano-Casado & Martínez-Rubio, 2019; Ablin et al., 2024) and backpropagation through the eigendecomposition following Ionescu et al. (2015).

The floor ε_{R} is a property of the covariance spectra rather than a free hyperparameter, and is fixed before any training as the 15th percentile of the trace-normalized spectrum pooled over three subjects per dataset (Appendix C.1).

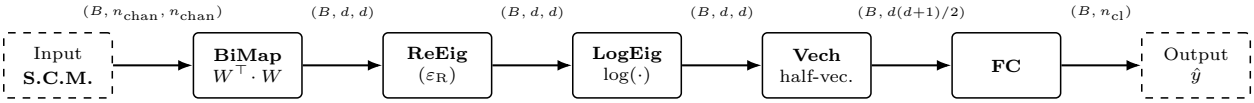


Figure 4: SPDNet pipeline. Shapes above each arrow, with B the batch size and d the dimension after BiMap.

6.3 Centralized baselines

We first establish a centralized, non-private baseline representing unrestricted data sharing: all subjects' recordings are pooled and trained on under the settings of Table 5. Validation accuracy governs training throughout, as it schedules the learning rate, stops the run and selects the checkpoint, so every run ends at convergence and its peak epoch measures the time the model needed. The BiMap dimension d and the ReEig floor ε_{R} are specific to SPDNet, whose d is constrained so that it has fewer parameters than its EEGNet counterpart; no comparison below therefore favours SPDNet through capacity.

Applying no privacy mechanism and pooling every subject, these runs upper-bound what the federated and private runs can reach. EEGNet is the stronger model on all three datasets, by 1.9 to 6.0 points (Table 6).

Table 5: Centralized configurations, selected by grid search (Appendix C.2). Epochs to converge is the 75th percentile over seeds of the peak epoch.

Dataset	Model	d	ε_{R}	#params	Batch size	Epochs to converge
BNCI2014_001	SPDNet	16	$8 \cdot 10^{-5}$	900	24	154
	EEGNet	—	—	2 484	64	236
Weibo2014	SPDNet	20	$2 \cdot 10^{-5}$	2 677	16	172
	EEGNet	—	—	3 863	64	202
Cho2017	SPDNet	16	$6 \cdot 10^{-5}$	1 298	32	106
	EEGNet	—	—	2 514	32	207

Shared: Adam optimiser, learning rate $3 \cdot 10^{-3}$, halved after 20 epochs without validation improvement; training stops after 75 such epochs; the best-validation checkpoint is the one evaluated on the test set.

Table 6: Centralized test accuracy (%), mean ± standard deviation over 10 seeds, best in bold, with chance level in parentheses.

Dataset	Architecture	Test accuracy
BNCI2014_001 (chance 25.0)	SPDNet	55.4 ± 3.0
	EEGNet	$\mathbf{61.4 \pm 1.5}$
Weibo2014 (chance 14.3)	SPDNet	48.3 ± 1.4
	EEGNet	$\mathbf{50.2 \pm 1.8}$
Cho2017 (chance 50.0)	SPDNet	63.5 ± 1.3
	EEGNet	$\mathbf{67.7 \pm 1.4}$

6.4 Federated protocol

Setup. Each client holds the recordings of three subjects on BNCI2014_001 and two on Weibo2014 and Cho2017. We study full ($k = n$) and half ($k = \lceil n/2 \rceil$) client participation, giving $k = 2$ on BNCI2014_001, $k = 3$ on Weibo2014 and $k = 13$ on Cho2017. To limit client drift, each client performs $\tau = \lceil m_{\max}/B \rceil$ local steps per round (i.e. one epoch's worth of gradient computations), so that a round is the federated counterpart of a centralized epoch. The τ minibatches of a round are drawn afresh and independently, each uniformly without replacement from the client's full training set, as (12) specifies. The number of rounds is the centralized peak epoch of Table 5. Batch sizes and learning rates are selected by a grid search at $\varepsilon = 20$ (Appendix C.3) and held fixed across all privacy levels, non-private runs included (Table 7).

Relation to the analysed algorithm. Each local step draws its own size-B minibatch uniformly without replacement, independently of the others, as Theorem 10 requires. Clients hold slightly different numbers of trials, so the accounting takes the worst case on both sides: $q = B/m_\star$ from the smallest client, $\tau = \lceil m_{\max}/B \rceil$ steps from the largest (Appendix C.4). The clipping threshold is fixed a priori at $C = 1$ (De et al., 2022) and applies to the per-record gradient concatenated across all federated parameters. The parameter space is a product: the Stiefel factor of the BiMap layer, compact and carrying the induced metric, times the Euclidean classifier weights. Clients run Adam under a cosine schedule rather than RSGD, which the privacy guarantee permits.

Normalization. EEGNet's batch normalization layers standardize each record using the whole minibatch's statistics, so replacing one record moves every per-record gradient and the sensitivity bound of Lemma 8 fails. Keeping the layer client-local, as FedBN (Li et al., 2021) does, does not fix this: its statistics still reach the server through the gradients of the shared parameters. We keep EEGNet's normalization layers local as under FedBN and, under privacy, recompute their statistics once per round on each client's validation split, treated

as public, with the BN parameters frozen at $(1, 0)$. The privacy guarantee then covers the training records. Appendix C.5 reports this variant alongside a group-normalization alternative that needs no public data.

Aggregation. EEGNet's parameters are all Euclidean and averaged arithmetically (FedAvg), except its normalization layers, which stay local as described above. For SPDNet, the Stiefel weight of the BiMap layer is aggregated by one of ProjAvg, RLAvg or RetAvg, its Euclidean classifier weights arithmetically in every case, under the single learning rate of Table 7. RetAvg is screened separately at each participation fraction, since its domain constraint bears on each client's drift from the aggregate, and halving participation changes which clients form it.

Reported model. We report a non-private run and three budgets, $\varepsilon \in \{20, 10, 5\}$ at $\delta = 10^{-5}$, for both architectures. For each, σ_{dp} is set by an exact privacy-loss-distribution accountant for Theorem 10, returning a certified upper bound on ε. The reported figures are the cost of the privacy pipeline as a whole. We report the final iterate x_T on the test set, which carries the guarantee of Section 4.

Table 7: Federated configurations. T is the round budget, carried over from Table 5; the SPDNet architectures are those of that table, unchanged.

				Learning rate η_0				
						RetAvg		
Dataset	Model	Clients	Batch size	FedAvg	ProjAvg, RLAvg	full	half	T
BNCI2014_001	SPDNet	3	96	—	$6 \cdot 10^{-2}$	$3 \cdot 10^{-2}$	$2 \cdot 10^{-2}$	154
	EEGNet		96	$3 \cdot 10^{-3}$	—	—	—	236
Weibo2014	SPDNet	5	96	—	$3 \cdot 10^{-2}$	$1 \cdot 10^{-2}$	$1.5 \cdot 10^{-2}$	172
	EEGNet		64	$3 \cdot 10^{-3}$	—	—	—	202
Cho2017	SPDNet	26	32	—	$1 \cdot 10^{-1}$	$1.5 \cdot 10^{-2}$	$1.5 \cdot 10^{-2}$	106
	EEGNet		32	$3 \cdot 10^{-3}$	—	—	—	207
Shared: Adam optimizer; one local epoch per round; cosine schedule over the budget (see Appendix C.3).								

6.5 Privacy–utility trade-off

We now compare the two architectures across the privacy budgets, at both full and half client participation levels.

Without privacy EEGNet is the stronger model, by a few points both centrally (Table 6) and federated (Table 8). Under privacy the ordering flips, on all three datasets and at every budget. The reversal holds throughout the range we test, and it follows from what privacy costs each architecture rather than from any change in their non-private ranking: measured against its own non-private run, SPDNet gives up roughly half as much accuracy as EEGNet, at every budget and on every dataset. Tightening ε narrows that advantage only slightly.

Halving the clients sampled per round changes none of this. It costs the best SPDNet arm between 1.0 and 2.0 points, uniformly across the three budgets and the three datasets, and its margin over EEGNet survives everywhere (Table 9).

6.6 Riemannian aggregation rules

The three Riemannian rules ProjAvg, RLAvg and RetAvg separate on cost, not accuracy. They land within a point or two of one another in every cell of Tables 8 and 9, and the two projection-based ones differ by at most 1.5 points, smaller than the seed-to-seed standard deviation almost everywhere. RetAvg matches them and is occasionally best, but costs 8 to 13 times more per round (Figure 7) and needs its own learning rate calibration, since the rates selected for the other Riemannian strategies drive clients outside the domain of its inverse retraction. The projection-based rules therefore buy the same accuracy without either cost.

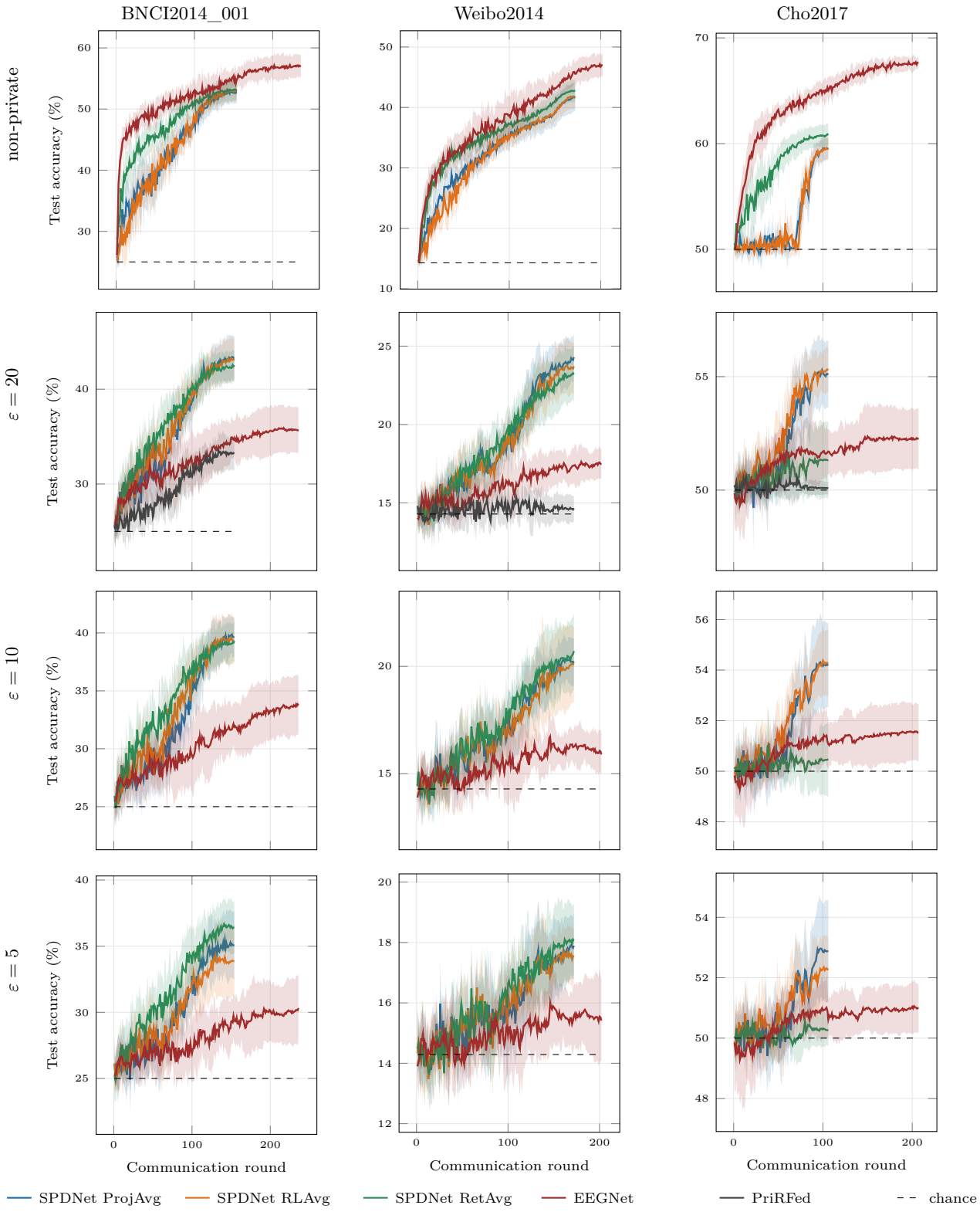


Figure 5: Test accuracy against communication round at full client participation, mean over 10 seeds with a band spanning ±1 standard deviation. Rows are privacy levels, columns datasets; the dashed line marks chance. Curves differ in length because each architecture runs to its own round budget (Table 7).

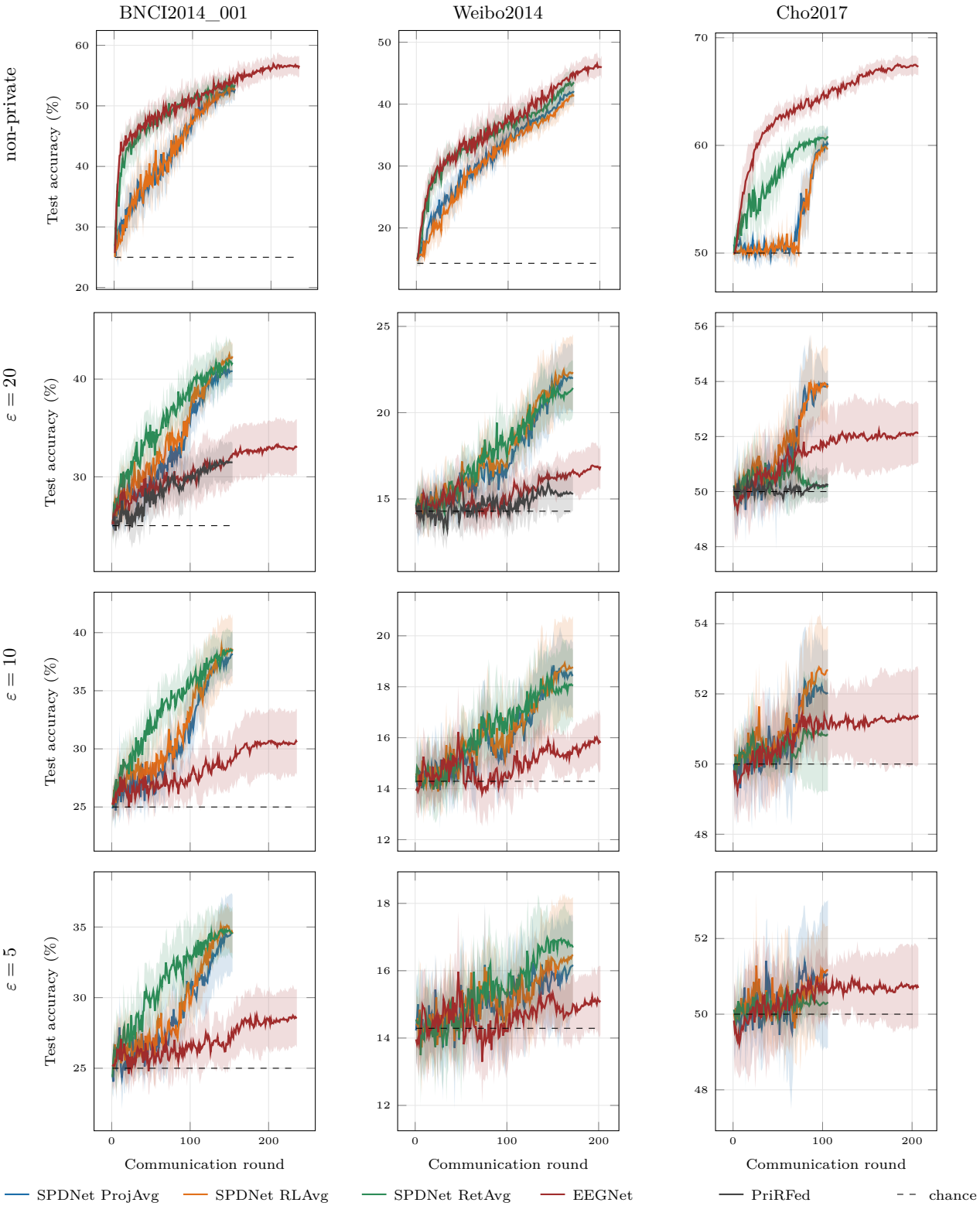


Figure 6: Test accuracy against communication round at half client participation, mean over 10 seeds with a band spanning ±1 standard deviation. Rows are privacy levels, columns datasets; the dashed line marks chance. Curves differ in length because each architecture runs to its own round budget (Table 7).

Table 8: Federated test accuracy (%) at full client participation, mean ± standard deviation over 10 seeds; best per column within a dataset in bold, second best underlined, ties marked jointly. Where two arms tie for best, both are bolded and the next distinct value is underlined.

			Private		
Dataset	Arm	Non-private	$\varepsilon = 20$	$\varepsilon = 10$	$\varepsilon = 5$
BNCI2014_001 chance 25.0	SPDNet ProjAvg	52.8 ± 1.8	$\mathbf{43.3 \pm 2.4}$	$\mathbf{39.6 \pm 1.7}$	35.1 ± 2.6
	SPDNet RLAvg	52.9 ± 1.6	43.1 ± 2.3	39.3 ± 2.0	33.9 ± 2.8
	SPDNet RetAvg	53.2 ± 1.7	42.5 ± 1.5	39.3 ± 1.7	$\mathbf{36.4 \pm 2.0}$
	EEGNet	$\mathbf{57.0 \pm 2.0}$	35.7 ± 2.4	33.8 ± 2.6	30.3 ± 2.5
Weibo2014 chance 14.3	SPDNet ProjAvg	41.6 ± 2.2	$\mathbf{24.3 \pm 1.5}$	20.2 ± 1.1	17.9 ± 1.1
	SPDNet RLAvg	41.7 ± 1.9	23.7 ± 1.7	20.1 ± 1.8	17.5 ± 0.9
	SPDNet RetAvg	42.8 ± 1.9	23.2 ± 1.8	$\mathbf{20.7 \pm 1.8}$	$\mathbf{18.1 \pm 1.2}$
	EEGNet	$\mathbf{47.0 \pm 1.9}$	17.4 ± 0.9	15.9 ± 1.0	15.4 ± 1.3
Cho2017 chance 50.0	SPDNet ProjAvg	59.5 ± 1.0	55.1 ± 1.5	54.2 ± 1.6	$\mathbf{52.9 \pm 1.7}$
	SPDNet RLAvg	59.6 ± 1.1	$\mathbf{55.3 \pm 1.0}$	$\mathbf{54.3 \pm 1.3}$	52.3 ± 1.0
	SPDNet RetAvg	60.8 ± 1.0	51.3 ± 1.7	50.5 ± 1.5	50.3 ± 0.5
	EEGNet	$\mathbf{67.7 \pm 0.6}$	52.2 ± 1.3	51.5 ± 1.1	51.0 ± 0.8

Table 9: Federated test accuracy (%) at half client participation, mean ± standard deviation over 10 seeds; best per column within a dataset in bold, second best underlined, ties marked jointly. Where two arms tie for best, both are bolded and the next distinct value is underlined.

			Private		
Dataset	Arm	Non-private	$\varepsilon = 20$	$\varepsilon = 10$	$\varepsilon = 5$
BNCI2014_001 chance 25.0	SPDNet ProjAvg	52.8 ± 1.9	40.8 ± 1.5	38.1 ± 1.8	34.5 ± 2.7
	SPDNet RLAvg	52.8 ± 1.7	$\mathbf{42.3 \pm 1.4}$	$\mathbf{38.6 \pm 3.0}$	$\mathbf{34.6 \pm 1.5}$
	SPDNet RetAvg	53.5 ± 1.4	41.7 ± 2.0	38.4 ± 1.8	$\mathbf{34.6 \pm 1.9}$
	EEGNet	$\mathbf{56.6 \pm 1.7}$	33.0 ± 2.9	30.6 ± 2.8	28.5 ± 2.1
Weibo2014 chance 14.3	SPDNet ProjAvg	41.9 ± 1.8	22.0 ± 1.9	18.5 ± 1.3	16.2 ± 1.4
	SPDNet RLAvg	41.3 ± 1.7	$\mathbf{22.3 \pm 2.2}$	$\mathbf{18.8 \pm 2.0}$	16.5 ± 1.7
	SPDNet RetAvg	43.5 ± 2.3	21.4 ± 1.6	18.0 ± 1.4	$\mathbf{16.7 \pm 0.8}$
	EEGNet	$\mathbf{45.9 \pm 1.9}$	16.9 ± 1.3	15.8 ± 1.1	15.1 ± 1.0
Cho2017 chance 50.0	SPDNet ProjAvg	60.1 ± 1.4	$\mathbf{53.9 \pm 0.5}$	52.0 ± 1.2	51.1 ± 2.0
	SPDNet RLAvg	59.7 ± 1.1	$\mathbf{53.9 \pm 1.4}$	$\mathbf{52.7 \pm 1.3}$	$\mathbf{51.2 \pm 1.2}$
	SPDNet RetAvg	60.8 ± 0.9	50.2 ± 0.4	50.8 ± 1.6	50.3 ± 0.4
	EEGNet	$\mathbf{67.3 \pm 0.8}$	52.1 ± 1.1	51.3 ± 1.4	50.7 ± 1.0

6.7 Comparison with PriRFed

The noise PriRFed must add to claim $\varepsilon = 20$ suffices for $\varepsilon = 2.22$ under our accounting: the same mechanism runs at a fifth of the noise for the same budget. Under Property P1 the two privatised gradients have the same law (Lemma 3); the aggregation rules differ, but Section 6.6 shows them accuracy-equivalent, so the accounting is what separates the two runs. PriRFed is moreover credited with our own exact per-round profile in place of the looser closed form it uses as published (Appendix C.4).

In accuracy this is worth 5.0 to 10.1 points at full participation and 3.7 to 9.3 at half (Table 10). On Weibo2014 and Cho2017 their runs land within a point of chance, so the comparison saturates there and it is the budget statement that carries the conclusion.

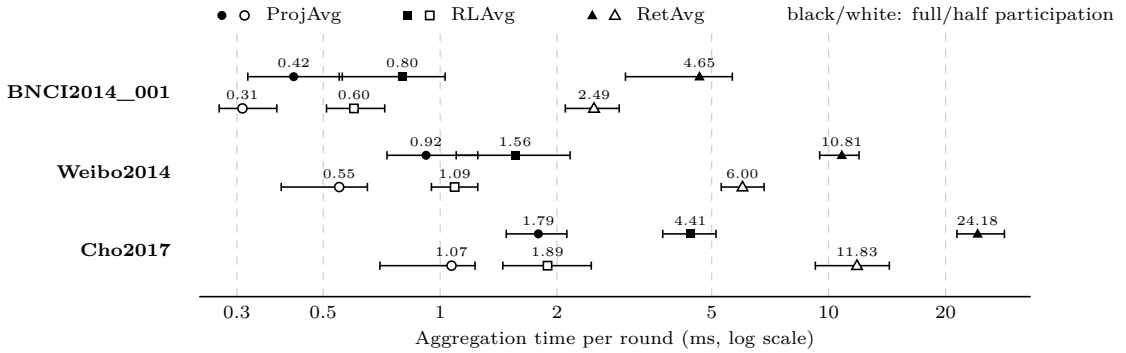


Figure 7: Per-round Riemannian aggregation time, median in milliseconds above each marker with whiskers spanning the interquartile range over rounds. RetAvg costs an order of magnitude more than the projection-based rules at both participation levels.

Table 10: Comparison with PriRFed on SPDNet, at $\varepsilon = 20$, $\delta = 10^{-5}$ and over 10 seeds. Ours is ProjAvg under our accounting; PriRFed is RetAvg at the noise its own round composition requires for the same budget, on the same architecture, data and number of rounds, and best-cased with our per-round accountant (Appendix C.4). Neither is credited with amplification by client subsampling, which an honest-but-curious server with observable participation cannot claim; the noise ratio is therefore the same at both participation levels.

	Full participation		Half participation		
Dataset	Ours	PriRFed	Ours	PriRFed	Noise ratio
BNCI2014_001	$\mathbf{43.3 \pm 2.4}$	33.2 ± 1.8	$\mathbf{40.8 \pm 1.5}$	31.5 ± 1.9	$4.91\times$
Weibo2014	$\mathbf{24.3 \pm 1.5}$	14.7 ± 0.8	$\mathbf{22.0 \pm 1.9}$	15.3 ± 1.0	$5.25\times$
Cho2017	$\mathbf{55.1 \pm 1.5}$	50.1 ± 0.3	$\mathbf{53.9 \pm 0.5}$	50.2 ± 0.6	$5.06\times$

7 Conclusion

We introduced DP-RFedProj, a record-level differentially private federated learning algorithm for compact submanifolds. Working in a fixed ambient geometry, and aggregating through closed-form projections, detaches the privacy accounting from the manifold, the aggregator and the local optimiser, and opens the convergence analysis to the regime federated training actually runs in: many local steps, partial participation, heterogeneous clients. On EEG motor-imagery classification, privacy reverses the ranking between a Riemannian and a Euclidean architecture, and our projection-based aggregation matches the retraction-based one at a fraction of the compute. Our accounting certifies the same budget at a fraction of the noise the closest prior method requires.

Four limitations bound this. The framework needs a compact submanifold under the Euclidean-induced metric, leaving out iterate-dependent metrics with no ambient projection, such as SPD under the affine-invariant one. The convergence rate is proved for Riemannian SGD at a constant stepsize while the experiments run Adam under a schedule. The accounting charges the full stream of privatised gradients rather than the submissions the server observes, which buys optimiser-agnosticism but leaves room for a tighter, optimiser-specific analysis. Finally, the clipping-bias floor does not vanish with the horizon unless the clip is inactive, so the iterates reach a neighbourhood of stationarity rather than a stationary point.

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.

Pierre Ablin, Simon Vary, Bin Gao, and Pierre-Antoine Absil. Infeasible deterministic, stochastic, and variance-reduction algorithms for optimization under orthogonality constraints. *Journal of Machine Learning Research*, 25(389):1–38, 2024.

P-A Absil and Jérôme Malick. Projection-like retractions on matrix manifolds. *SIAM Journal on Optimization*, 22(1):135–158, 2012.

P-A Absil, Robert Mahony, and Rodolphe Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2008.

Vincent Arsigny, Pierre Fillard, Xavier Pennec, and Nicholas Ayache. Geometric means in a novel vector space structure on symmetric positive-definite matrices. *SIAM journal on matrix analysis and applications*, 29(1):328–347, 2007.

Borja Balle, Gilles Barthe, and Marco Gaboardi. Privacy amplification by subsampling: Tight analyses via couplings and divergences. *Advances in neural information processing systems*, 31, 2018.

Alexandre Barachant, Stéphane Bonnet, Marco Congedo, and Christian Jutten. Multiclass brain–computer interface classification by riemannian geometry. *IEEE transactions on biomedical engineering*, 59(4): 920–928, 2012.

Belgacem Ben Youssef, Lamyaa Alhmidi, Yakoub Bazi, and Mansour Zuair. Federated learning approach for remote sensing scene classification. *Remote Sensing*, 16(12):2194, 2024.

Florent Bouchard, Nils Laurent, Salem Said, and Nicolas Le Bihan. Beyond r-barycenters: an effective averaging method on stiefel and grassmann manifolds. *arXiv preprint arXiv:2501.11555*, 2025.

Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Oxford University Press, 2013.

Nicolas Boumal. *An introduction to optimization on smooth manifolds*. Cambridge University Press, 2023.

Daniel Brooks, Olivier Schwander, Frédéric Barbaresco, Jean-Yves Schneider, and Matthieu Cord. Riemannian batch normalization for spd neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

Sylvain Chevallier, Igor Carrara, Bruno Aristimunha, Pierre Guetschel, Sara Sedlar, Bruna Lopes, Sébastien Velut, Salim Khazem, and Thomas Moreau. The largest eeg-based bci reproducibility study for open science: the moabb benchmark. *arXiv preprint arXiv:2404.15319*, 2024.

Hohyun Cho, Minkyu Ahn, Sangtae Ahn, Moonyoung Kwon, and Sung Chan Jun. Eeg datasets for motor imagery brain–computer interface. *GigaScience*, 6(7):gix034, 2017.

Francis H Clarke, Ronald J Stern, and Peter R Wolenski. Proximal smoothness and the lower-c2 property. *J. Convex Anal*, 2(1-2):117–144, 1995.

William G. Cochran. *Sampling Techniques*. Wiley, 3rd edition, 1977.

Damek Davis, Dmitriy Drusvyatskiy, and Zhan Shi. Stochastic optimization over proximally smooth sets. *SIAM Journal on Optimization*, 35(1):157–179, 2025.

Soham De, Leonard Berrada, Jamie Hayes, Samuel L Smith, and Borja Balle. Unlocking high-accuracy differentially private image classification through scale. *arXiv preprint arXiv:2204.13650*, 2022.

Kangkang Deng and Jiang Hu. Decentralized projected riemannian gradient method for smooth optimization on compact submanifolds embedded in the euclidean space. *Numerische Mathematik*, pp. 1–38, 2025.

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.

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.

Robin C Geyer, Tassilo Klein, and Moin Nabi. Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557*, 2017.

Andi Han, Bamdev Mishra, Pratik Jawanpuria, and Junbin Gao. Differentially private riemannian optimization. *Machine Learning*, 113(3):1133–1161, 2024.

Zhenwei Huang, Wen Huang, Pratik Jawanpuria, and Bamdev Mishra. Riemannian federated learning via averaging gradient stream. *arXiv preprint arXiv:2409.07223*, 2024.

Zhenwei Huang, Wen Huang, Pratik Jawanpuria, and Bamdev Mishra. Federated learning on riemannian manifolds with differential privacy. *Machine Learning*, 115(4):84, 2026.

Zhiwu Huang and Luc Van Gool. A riemannian network for spd matrix learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017.

Catalin Ionescu, Orestis Vantzos, and Cristian Sminchisescu. Matrix backpropagation for deep networks with structured layers. In *Proceedings of the IEEE international conference on computer vision*, pp. 2965–2973, 2015.

Yangdi Jiang, Xiaotian Chang, Yi Liu, Lei Ding, Linglong Kong, and Bei Jiang. Gaussian differential privacy on riemannian manifolds. *Advances in Neural Information Processing Systems*, 36:14665–14684, 2023.

Tetsuya Kaneko, Simone Fiori, and Toshihisa Tanaka. Empirical arithmetic averaging over the compact stiefel manifold. *IEEE Transactions on Signal Processing*, 61(4):883–894, 2012.

Hermann Karcher. Riemannian center of mass and so called karcher mean. *arXiv preprint arXiv:1407.2087*, 2014.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations (ICLR)*, 2015.

Jakub Konečný, H Brendan McMahan, Daniel Ramage, and Peter Richtárik. Federated optimization: Distributed machine learning for on-device intelligence. *arXiv preprint arXiv:1610.02527*, 2016.

Vernon J Lawhern, Amelia J Solon, Nicholas R Waytowich, Stephen M Gordon, Chou P Hung, and Brent J Lance. Eegnet: a compact convolutional neural network for eeg-based brain–computer interfaces. *Journal of neural engineering*, 15(5):056013, 2018.

Junqing Le, Di Zhang, Xinyu Lei, Long Jiao, Kai Zeng, and Xiaofeng Liao. Privacy-preserving federated learning with malicious clients and honest-but-curious servers. *IEEE Transactions on Information Forensics and Security*, 18:4329–4344, 2023.

Mario Lezcano-Casado and David Martínez-Rubio. Cheap orthogonal constraints in neural networks: A simple parametrization of the orthogonal and unitary group. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3794–3803. PMLR, 09–15 Jun 2019. URL `https://proceedings.mlr.press/v97/lezcano-casado19a.html`.

Jiaxiang Li and Shiqian Ma. Federated learning on riemannian manifolds. *arXiv preprint arXiv:2206.05668*, 2022.

Peihua Li, Jiangtao Xie, Qilong Wang, and Wangmeng Zuo. Is second-order information helpful for large-scale visual recognition? In *Proceedings of the IEEE international conference on computer vision*, pp. 2070–2078, 2017.

Xiaoxiao Li, Meirui Jiang, Xiaofei Zhang, Michael Kamp, and Qi Dou. Fedbn: Federated learning on non-iid features via local batch normalization. *arXiv preprint arXiv:2102.07623*, 2021.

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, 2017.

H Brendan McMahan, Galen Andrew, Ulfar Erlingsson, Steve Chien, Ilya Mironov, Nicolas Papernot, and Peter Kairouz. A general approach to adding differential privacy to iterative training procedures. *arXiv preprint arXiv:1812.06210*, 2018.

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

Xavier Pennec. Intrinsic statistics on riemannian manifolds: Basic tools for geometric measurements. *Journal of Mathematical Imaging and Vision*, 25(1):127–154, 2006.

Xavier Pennec, Pierre Fillard, and Nicholas Ayache. A riemannian framework for tensor computing. *International Journal of computer vision*, 66:41–66, 2006.

Matthew Reimherr, Karthik Bharath, and Carlos Soto. Differential privacy over riemannian manifolds. *Advances in Neural Information Processing Systems*, 34:12292–12303, 2021.

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, 2020.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, pp. 3–18. IEEE, 2017.

Michael Tangermann, Klaus-Robert Müller, Ad Aertsen, Niels Birbaumer, Christoph Braun, Clemens Brunner, Robert Leeb, Carsten Mehring, Kai J Miller, Gernot R Müller-Putz, et al. Review of the bci competition iv. *Frontiers in neuroscience*, 6:55, 2012.

Saiteja Utpala, Andi Han, Pratik Jawanpuria, and Bamdev Mishra. Improved differentially private riemannian optimization: Fast sampling and variance reduction. *Transactions on Machine Learning Research*, 2023.

Hongye Wang, Zhaoye Pan, Chang He, Jiaxiang Li, and Bo Jiang. Federated learning on riemannian manifolds: A gradient-free projection-based approach. *arXiv preprint arXiv:2507.22855*, 2025.

Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 3–19, 2018.

He Xiao, Tao Yan, and Kai Wang. Riemannian svrg using barzilai–borwein method as second-order approximation for federated learning. *Symmetry*, 16(9):1101, 2024.

He Xiao, Tao Yan, and Shimin Zhao. Riemannian svrg with barzilai-borwein scheme for federated learning. *Journal of Industrial and Management Optimization*, 21(2):1546–1567, 2025.

Weibo Yi, Shuang Qiu, Kun Wang, Hongzhi Qi, Lixin Zhang, Peng Zhou, Feng He, and Dong Ming. Evaluation of eeg oscillatory patterns and cognitive process during simple and compound limb motor imagery. *PloS one*, 9(12):e114853, 2014.

Hongyi Zhang, Sashank J Reddi, and Suvrit Sra. Riemannian svrg: Fast stochastic optimization on riemannian manifolds. *Advances in Neural Information Processing Systems*, 29, 2016.

Jiaojiao Zhang, Jiang Hu, Anthony Man-Cho So, and Mikael Johansson. Nonconvex federated learning on compact smooth submanifolds with heterogeneous data. *Advances in Neural Information Processing Systems*, 37:109817–109844, 2024.

Qinqing Zheng, Shuxiao Chen, Qi Long, and Weijie Su. Federated f-differential privacy. In *International conference on artificial intelligence and statistics*, pp. 2251–2259. PMLR, 2021.

Ligeng Zhu, Zhijian Liu, and Song Han. Deep leakage from gradients. *Advances in neural information processing systems*, 32, 2019.

Ralf Zimmermann and Knut Huper. Computing the riemannian logarithm on the stiefel manifold: Metrics, methods, and performance. *SIAM Journal on Matrix Analysis and Applications*, 43(2):953–980, 2022.

A The Stiefel, Grassmann and SPD manifolds

A.1 The Stiefel manifold

The Stiefel manifold $\mathrm{St}(d,p) = \{X \in \mathbb{R}^{d\times p} : X^\top X = I_p\}$, $p \le d$, is the set of matrices with orthonormal columns, a compact embedded submanifold of $\mathbb{R}^{d\times p}$ of dimension $dp - \frac{p(p+1)}{2}$. It reduces to the orthogonal group when $p = d$ and to the unit sphere when $p = 1$. Compactness follows from $\|X\|_F^2 = p$.

The tangent space at X is

$$T_X\mathrm{St}(d,p) = \{\xi \in \mathbb{R}^{d\times p} : X^\top\xi + \xi^\top X = 0\}, \tag{40}$$

that is, the matrices whose X-component is skew-symmetric. Equipping $\mathrm{St}(d,p)$ with the metric induced by the ambient Frobenius inner product $\langle\xi,\eta\rangle = \mathrm{tr}(\xi^\top\eta)$, the orthogonal projector onto (40) is

$$P_X(\xi) = \xi - X\,\mathrm{sym}(X^\top\xi), \qquad \mathrm{sym}(A) := \tfrac{1}{2}(A + A^\top), \tag{41}$$

and the Riemannian gradient of a smooth f is $\mathrm{grad}f(X) = P_X(\nabla f(X))$ with ∇f the ambient gradient.

For $A \in \mathbb{R}^{d\times p}$ of full column rank, two retractions are standard. The polar retraction uses the orthogonal polar factor

$$\Pi(A) = A(A^\top A)^{-1/2} = UV^\top, \qquad A = USV^\top \text{ (thin SVD)}, \tag{42}$$

which is also the nearest-point projection onto $\mathrm{St}(d,p)$, unique and smooth on that set (Absil & Malick, 2012; Bouchard et al., 2025). The QR retraction instead takes the Q factor of a thin QR decomposition, at the same $O(dp^2 + p^3)$ cost but without the minimality property. Both are idempotent on $\mathrm{St}(d,p)$ and define projection-based retractions $\mathrm{R}_X(\xi) = \Pi(X + \xi)$. We use the polar factor throughout, since it is the projection our aggregation rules require.

The Riemannian logarithm on $\mathrm{St}(d,p)$ requires an iterative algorithm (Zimmermann & Huper, 2022), and so does the inverse polar retraction, which solves the Sylvester equation $MS + SM^\top = 2I_p$ with $M = X^\top Y$ at $O(dp^2 + p^3)$ per iteration. That solution exists only when $M + M^\top$ is positive definite, so the inverse retraction is moreover defined only in a neighbourhood of X.

A.2 The Grassmann manifold

We represent the Grassmann manifold $\mathrm{Gr}(d,p)$ of p-dimensional subspaces of $\mathbb{R}^d$ by the orthogonal projectors onto them

$$\mathrm{Gr}(d,p) \;\cong\; \{P \in \mathcal{S}_d : P^2 = P,\ \mathrm{tr}\,P = p\}. \tag{43}$$

It is a compact embedded submanifold of the symmetric matrices $\mathcal{S}_d$ of dimension $p(d-p)$.

Under the metric induced by the ambient Frobenius inner product, the tangent space at P is

$$T_P\mathrm{Gr}(d,p) = \{\xi \in \mathcal{S}_d : P\xi P = 0 \text{ and } (I-P)\xi(I-P) = 0\}, \tag{44}$$

that is, the symmetric matrices supported off the two diagonal blocks, and the orthogonal projector onto (44) keeps exactly those blocks,

$$P_P(\xi) = P\xi(I-P) + (I-P)\xi P. \tag{45}$$

For $A \in \mathcal{S}_d$ with eigenvalues $\lambda_1 \ge \cdots \ge \lambda_d$ and associated orthonormal eigenvectors $u_1, \ldots, u_d$, the nearest-point projection is the spectral projector onto a top-p eigenspace,

$$\Pi(A) = \sum_{i=1}^{p} u_i u_i^\top, \tag{46}$$

unique when $\lambda_p > \lambda_{p+1}$, at the cost of one symmetric eigendecomposition.

A.3 The manifold of SPD matrices

Let $\mathcal{S}_n$ denote the space of symmetric $n \times n$ matrices. The set

$$\mathcal{S}_n^{++} = \{\Sigma \in \mathcal{S}_n : x^\top \Sigma x > 0 \text{ for all } x \in \mathbb{R}^n \setminus \{0\}\} \tag{47}$$

of symmetric positive definite matrices is an open convex cone in $\mathcal{S}_n$, hence a smooth manifold of dimension $\frac{n(n+1)}{2}$ whose tangent space at every Σ is $\mathcal{S}_n$ itself. It is not compact. Two metrics are standard.

Affine-invariant. With $\langle \xi, \eta \rangle_\Sigma = \mathrm{tr}(\Sigma^{-1}\xi\,\Sigma^{-1}\eta)$ the manifold is complete with non-positive curvature, and

$$\mathrm{Exp}_\Sigma(\xi) = \Sigma^{1/2} \exp\big(\Sigma^{-1/2}\xi\Sigma^{-1/2}\big)\Sigma^{1/2}. \tag{48}$$

The metric is invariant under $\Sigma \mapsto G\Sigma G^\top$ for any invertible G, which for covariance descriptors means invariance to linear mixing of the sensors—the property that makes it the standard choice in EEG classification.

Log-Euclidean. Pulling the Euclidean metric of $\mathcal{S}_n$ back through the matrix logarithm gives

$$\langle \xi, \eta \rangle_\Sigma = \big\langle D_\Sigma \log(\xi),\, D_\Sigma \log(\eta) \big\rangle_F, \tag{49}$$

with $D_\Sigma \log$ the differential of the matrix logarithm at Σ. Under this metric log is an isometry onto $\mathcal{S}_n$, so the manifold is flat and all operations reduce to Euclidean ones in the logarithmic domain. It is cheaper than the affine-invariant metric and invariant to orthogonal transformations and scaling, but not to general linear mixing.

B Proofs for the convergence analysis

B.1 Geometric consequences of proximal smoothness

Let $\mathcal{M}$ be a smooth compact manifold, assumed to be 2γ-proximally smooth. The nearest-point projection $\Pi : \mathcal{E} \to \mathcal{M}$ is single-valued on $U_{\mathcal{M}}(2\gamma)$ and enjoys two regularity properties on $\overline{U_{\mathcal{M}}(\gamma)}$, collected from (Clarke et al., 1995; Deng & Hu, 2025).

Step-to-projection bound. A consequence of the 2-Lipschitz continuity of the nearest-point projection Π on $\overline{U_{\mathcal{M}}(\gamma)}$ (Clarke et al., 1995, Thm. 4.8), applied with anchor $\Pi(x) = x$ is that, for $x \in \mathcal{M}$ and $u \in \mathcal{E}$ with $\|u\| \le \gamma$,

$$\|\Pi(x+u) - x\| \le 2\|u\|, \tag{50}$$

Lipschitz tangent projectors. The map $x \mapsto P_x$ is smooth on the compact $\mathcal{M}$, hence globally Lipschitz (Deng & Hu, 2025, Lem. 4.2). More precisely, there exists $L_P > 0$, depending only on $\mathcal{M}$, such that

$$\|P_x - P_y\|_{\mathrm{op}} \le L_P\, \|x - y\| \qquad \forall\, x, y \in \mathcal{M}. \tag{51}$$

B.2 Properties of the clipping operator

The clipping operator (6) satisfies, for all $u, v \in \mathcal{E}$,

$$\|\mathrm{clip}_C(v)\| \le C \qquad \text{and} \qquad \|\mathrm{clip}_C(u) - \mathrm{clip}_C(v)\| \le \|u - v\|, \tag{52}$$

the first by definition and the second because clip_C is the metric projection onto the convex ball $\overline{B}(0, C)$.

B.3 Auxiliary lemmas for the convergence analysis

Lemma 20 (Projected Gaussian second moment under norm truncation)**.** *Let $\xi \sim \mathcal{N}(0, s^2 I_{\mathcal{E}})$ be isotropic on $\mathcal{E}$ (with $\dim \mathcal{E} = D$) and let P be an orthogonal projector of rank $d_{\mathcal{M}} \le D$. Then $\mathbb{E}\|P\xi\|^2 = d_{\mathcal{M}} s^2$, and for every $R > 0$,*

$$\mathbb{E}\big[\|P\xi\|^2 \;\big|\; \|\xi\| \le R\big] \;\le\; d_{\mathcal{M}}\, s^2.$$

Proof. Write $\xi = \rho U$ with $\rho := \|\xi\|$ and $U := \xi/\|\xi\|$. By rotational invariance U is uniform on the unit sphere and independent of ρ, and $\|P\xi\|^2 = \rho^2\|PU\|^2$ with $\|PU\|^2 \sim \mathrm{Beta}(d_{\mathcal{M}}/2, (D-d_{\mathcal{M}})/2)$, so $\mathbb{E}\|PU\|^2 = d_{\mathcal{M}}/D$ (for $d_{\mathcal{M}} = D$, $\|PU\|^2 \equiv 1$). Taking expectations and using $\mathbb{E}[\rho^2] = Ds^2$ gives the first claim. The event $\{\rho \le R\}$ is a function of ρ alone, so U remains uniform and independent under it, whence

$$\mathbb{E}\big[\|P\xi\|^2 \,\big|\, \rho \le R\big] = \frac{d_{\mathcal{M}}}{D}\,\mathbb{E}\big[\rho^2 \,\big|\, \rho \le R\big] \le \frac{d_{\mathcal{M}}}{D}\,\mathbb{E}[\rho^2] = d_{\mathcal{M}} s^2,$$

the inequality because upper truncation of a positive random variable decreases its second moment. □

B.4 Proof of Lemma 11

Fix $i \in [n]$ and $x \in \mathcal{M}$, and abbreviate $c_\ell := \mathrm{clip}_C\big(\mathrm{grad}\, f(x; z_{i,\ell})\big)$ for $\ell \in [m_i]$, so that $\mathrm{grad}^C f_i(x) = \frac{1}{m_i}\sum_{\ell=1}^{m_i} c_\ell$ is the population mean of $\{c_\ell\}_{\ell=1}^{m_i}$ and $\widehat{\mathrm{grad}}^C f_i(x;\mathcal{B}) = \frac{1}{B}\sum_{\ell\in\mathcal{B}} c_\ell$ is its mean over a size-B minibatch $\mathcal{B} \subset \mathcal{D}_i$ drawn uniformly without replacement.

Unbiasedness. Each index ℓ belongs to $\mathcal{B}$ with marginal probability B/m_i, so by linearity

$$\mathbb{E}_{\mathcal{B}}\big[\widehat{\mathrm{grad}}^C f_i(x;\mathcal{B})\big] = \frac{1}{B}\sum_{\ell=1}^{m_i} \Pr(\ell \in \mathcal{B})\, c_\ell = \frac{1}{m_i}\sum_{\ell=1}^{m_i} c_\ell = \mathrm{grad}^C f_i(x).$$

Variance. The finite-population variance identity for sampling without replacement (Cochran, 1977, Thm. 2.2) gives

$$\mathbb{E}_{\mathcal{B}}\big[\|\widehat{\mathrm{grad}}^C f_i(x;\mathcal{B}) - \mathrm{grad}^C f_i(x)\|^2\big] = \frac{m_i - B}{m_i - 1}\cdot\frac{V_i^C(x)}{B} = \rho_i\, V_i^C(x), \tag{53}$$

where $V_i^C(x) := \frac{1}{m_i}\sum_{\ell=1}^{m_i}\big\|c_\ell - \mathrm{grad}^C f_i(x)\big\|^2$, with $\rho_i = \frac{m_i - B}{B(m_i-1)}$. It remains to bound the within-client clipped variance $V_i^C(x)$. Since the population mean minimises the mean-square deviation, evaluating instead at the centre $\mathrm{clip}_C\big(\mathrm{grad}\, f_i(x)\big)$ can only increase it. Combined with the 1-Lipschitz continuity of clip_C (see (52)) and Assumption 3,

$$\begin{aligned} V_i^C(x) &\le \frac{1}{m_i}\sum_{\ell=1}^{m_i}\big\|\mathrm{clip}_C\big(\mathrm{grad}\, f(x; z_{i,\ell})\big) - \mathrm{clip}_C\big(\mathrm{grad}\, f_i(x)\big)\big\|^2 \\ &\le \frac{1}{m_i}\sum_{\ell=1}^{m_i}\big\|\mathrm{grad}\, f(x; z_{i,\ell}) - \mathrm{grad}\, f_i(x)\big\|^2 \;\le\; \sigma_{\mathrm{st}}^2. \end{aligned}$$

Substituting into (53) yields $\mathbb{E}_{\mathcal{B}}\|\widehat{\mathrm{grad}}^C f_i(x;\mathcal{B}) - \mathrm{grad}^C f_i(x)\|^2 \le \rho_i\,\sigma_{\mathrm{st}}^2$.

B.5 Proof of Lemma 12

Write $\bar b^C(x) := \frac{1}{n}\sum_{i=1}^n b_i^C(x)$ for the cohort-average clipping bias, so that $\beta_C = \sup_{x\in\mathcal{M}}\|\bar b^C(x)\|$ by (28).

Fix $x \in \mathcal{M}$ (all gradients are evaluated at x). With $\overline{\mathrm{grad}^C f} := \frac{1}{n}\sum_{j=1}^n \mathrm{grad}^C f_j$ and $\frac{1}{n}\sum_i \mathrm{grad}\, f_i = \mathrm{grad}\, F$, decompose

$$\mathrm{grad}^C f_i - \overline{\mathrm{grad}^C f} = (\mathrm{grad}\, f_i - \mathrm{grad}\, F) + (b_i^C - \bar b^C).$$

By Young's inequality $\|a+b\|^2 \le (1+\lambda)\|a\|^2 + (1+\lambda^{-1})\|b\|^2$ ($\lambda > 0$), averaging over i and invoking Assumption 4 together with the definition (26) of σ_{clip}^2,

$$\frac{1}{n}\sum_{i=1}^n \|\,\mathrm{grad}^C f_i - \overline{\mathrm{grad}^C f}\|^2 \;\le\; (1+\lambda)\,\sigma_{\mathrm{het}}^2 + (1+\lambda^{-1})\,\sigma_{\mathrm{clip}}^2.$$

Minimising the right-hand side over $\lambda > 0$ yields $(\sigma_{\mathrm{het}} + \sigma_{\mathrm{clip}})^2$ (attained at $\lambda^\star = \sigma_{\mathrm{clip}}/\sigma_{\mathrm{het}}$ when both are positive, and as a limit otherwise). Since the bound is uniform in x, taking $\sup_x$ gives (27).

B.6 Proof of Lemma 14

Fix $t \in \{0, \dots, T-1\}$. The $k\tau$ noise vectors $\{\xi_{t,j}^{(i)}\}_{(i,j)\in S_t\times\{0,\dots,\tau-1\}}$ are i.i.d. $\mathcal{N}(0, \sigma_{\mathrm{dp}}^2 I_{\mathcal{E}})$ and independent of $\mathcal{F}_t$, so

$$\Pr(\mathcal{G}_t \mid \mathcal{F}_t) = \Pr_{\xi\sim\mathcal{N}(0,\sigma_{\mathrm{dp}}^2 I_{\mathcal{E}})}\big(\|\xi\| \le R\big)^{k\tau},$$

a deterministic constant depending on neither t nor S_t; hence $\pi := \Pr(\mathcal{G}_t^c)$ is well-defined and independent of t.

For a single draw, rescaling gives $\|\xi\|/\sigma_{\mathrm{dp}} \overset{d}{=} \|W\|$ with $W \sim \mathcal{N}(0, I_{\mathcal{E}})$. The map $w \mapsto \|w\|$ is 1-Lipschitz with $\mathbb{E}\|W\| \le \sqrt{D}$ (Jensen), so the one-sided Gaussian Lipschitz inequality (Boucheron et al., 2013, Thm. 5.6) with deviation $\sqrt{c_0}$ yields

$$\Pr\big(\|W\| \ge \sqrt{D} + \sqrt{c_0}\big) \le \Pr\big(\|W\| \ge \mathbb{E}\|W\| + \sqrt{c_0}\big) \le e^{-c_0/2}.$$

Since $R = \sigma_{\mathrm{dp}}(\sqrt{D} + \sqrt{c_0})$ by (29), this reads $\Pr(\|\xi_{t,j}^{(i)}\| > R) \le e^{-c_0/2}$. A union bound over the $k\tau$ pairs $(i,j) \in S_t \times \{0, \dots, \tau-1\}$ gives $\pi = \Pr(\mathcal{G}_t^c) \le k\tau\, e^{-c_0/2}$.

B.7 Proof of Theorem 15

Step 1: Tube stability and linearisation on $\mathcal{G}_t$

Proof of Lemma 17.

(a) By induction on j. The base $j = 0$ is trivial. For the inductive step, set $u_j := -\eta\, \tilde{g}_{t,j}^{(i)}$; on $\mathcal{G}_t$ we have $\|u_j\| \le \eta G_{\mathrm{eff}}$ by (31), and since $\alpha \le \gamma/(4G_{\mathrm{eff}})$, $\eta G_{\mathrm{eff}} = \alpha G_{\mathrm{eff}}/\tau \le \gamma/(4\tau) < \gamma$. As $x_{t,j}^{(i)} \in \mathcal{M}$, the step-to-projection bound (50) gives

$$\|x_{t,j+1}^{(i)} - x_{t,j}^{(i)}\| = \|\Pi(x_{t,j}^{(i)} + u_j) - x_{t,j}^{(i)}\| \le 2\|u_j\| \le 2\eta G_{\mathrm{eff}}.$$

Telescoping over $j' = 0, \dots, j$ yields $\|x_{t,j+1}^{(i)} - x_t\| \le 2G_{\mathrm{eff}}\, \eta(j+1)$; at $j = \tau - 1$ this is $\|x_{t,\tau}^{(i)} - x_t\| \le 2G_{\mathrm{eff}}\, \alpha \le \gamma/2$.

(b) For $j \le \tau - 1$, the pre-projection point $x_{t,j}^{(i)} - \eta\, \tilde{g}_{t,j}^{(i)}$ satisfies

$$\mathrm{dist}\big(x_{t,j}^{(i)} - \eta\, \tilde{g}_{t,j}^{(i)},\, \mathcal{M}\big) \le \eta\, \|\tilde{g}_{t,j}^{(i)}\| \le \eta G_{\mathrm{eff}} \le \gamma/4 < \gamma,$$

so it lies in $U_{\mathcal{M}}(\gamma)$ and its projection $x_{t,j+1}^{(i)}$ is unique by Assumption 1.

(c) Define the two server increments

$$\Delta_t^{\textsc{ProjAvg}} := \frac{1}{k}\sum_{i\in S_t}\big(x_{t,\tau}^{(i)} - x_t\big), \qquad \Delta_t^{\mathrm{RLAvg}} := \frac{1}{k}\sum_{i\in S_t} P_{x_t}\big(x_{t,\tau}^{(i)} - x_t\big) = P_{x_t}\Delta_t^{\textsc{ProjAvg}},$$

the last equality by linearity of P_{x_t}. By the triangle inequality and (a), $\|\Delta_t^{\textsc{ProjAvg}}\| \le \frac{1}{k}\sum_{i\in S_t}\|x_{t,\tau}^{(i)} - x_t\| \le 2G_{\mathrm{eff}}\, \alpha$; and since P_{x_t} is an orthogonal projector, $\|\Delta_t^{\mathrm{RLAvg}}\| = \|P_{x_t}\Delta_t^{\textsc{ProjAvg}}\| \le \|\Delta_t^{\textsc{ProjAvg}}\|$. Hence

$$\|\Delta_t^{\mathrm{RLAvg}}\| \le \|\Delta_t^{\textsc{ProjAvg}}\| \le 2G_{\mathrm{eff}}\, \alpha \le \gamma/2,$$

so for either aggregator the server pre-projection point $x_t + \Delta_t^{\mathrm{Agg}}$ lies in $U_{\mathcal{M}}(\gamma)$ and its projection x_{t+1} is unique.

Proof of Lemma 18. Throughout we abbreviate $P_{t,j}^{(i)} := P_{x_{t,j}^{(i)}}$ and work on $\mathcal{G}_t$, where Lemma 17 applies. Let $\mathcal{F}_{t,j}^{(i)}$ denote the σ-algebra generated by $\mathcal{F}_t$, S_t, and all minibatch and noise draws of client i at local steps $0, \dots, j-1$, so that $x_{t,j}^{(i)}$ is $\mathcal{F}_{t,j}^{(i)}$-measurable while the step-j draws $\mathcal{B}_{t,j}^{(i)}$ and $\xi_{t,j}^{(i)}$ are independent of it.

Step 1: per-step expansion. Set $u_j := -\eta\,\tilde{g}_{t,j}^{(i)} \in T_{x_{t,j}^{(i)}}\mathcal{M}$; by (31), $\|u_j\| \le \eta G_{\text{eff}} \le \gamma$ on $\mathcal{G}_t$, so the quadratic-remainder expansion (35) at base point $x_{t,j}^{(i)}$ (where $P_{t,j}^{(i)} u_j = u_j$) gives

$$x_{t,j+1}^{(i)} - x_{t,j}^{(i)} = -\eta\,\tilde{g}_{t,j}^{(i)} + w_j, \qquad \|w_j\| \le C_{\text{qr}}\,\eta^2 G_{\text{eff}}^2. \tag{54}$$

Step 2: signal–noise split of the clipped minibatch gradient. Write

$$\widehat{\text{grad}}^{C} f_i(x_{t,j}^{(i)}; \mathcal{B}_{t,j}^{(i)}) = \text{grad}^{C} f_i(x_{t,j}^{(i)}) + \delta_{t,j}^{(i)}, \qquad \delta_{t,j}^{(i)} := \widehat{\text{grad}}^{C} f_i(x_{t,j}^{(i)}; \mathcal{B}_{t,j}^{(i)}) - \text{grad}^{C} f_i(x_{t,j}^{(i)}) \;\in\; T_{x_{t,j}^{(i)}}\mathcal{M}.$$

Since the step-j minibatch is fresh, Lemma 11 gives

$$\mathbb{E}\big[\delta_{t,j}^{(i)} \mid \mathcal{F}_{t,j}^{(i)}\big] = 0, \qquad \mathbb{E}\big[\|\delta_{t,j}^{(i)}\|^2 \mid \mathcal{F}_{t,j}^{(i)}\big] \le \rho_i\,\sigma_{\text{st}}^2. \tag{55}$$

Step 3: signal drift. Using $\text{grad}\, f(x;z) = P_x \nabla f(x;z)$, the 1-Lipschitz continuity of clip_C (see (52)), and adding and subtracting $P_{t,j}^{(i)} \nabla f(x_t; z_{i,\ell})$,

$$\begin{aligned}
\|\,\text{grad}^{C} f_i(x_{t,j}^{(i)}) - \text{grad}^{C} f_i(x_t)\| &\le \frac{1}{m_i}\sum_{\ell=1}^{m_i} \big\| P_{t,j}^{(i)} \nabla f(x_{t,j}^{(i)}; z_{i,\ell}) - P_{x_t} \nabla f(x_t; z_{i,\ell}) \big\| \\
&\le \frac{1}{m_i}\sum_{\ell=1}^{m_i} \Big[\|\nabla f(x_{t,j}^{(i)}; z_{i,\ell}) - \nabla f(x_t; z_{i,\ell})\| + \|P_{t,j}^{(i)} - P_{x_t}\|_{\text{op}}\, \|\nabla f(x_t; z_{i,\ell})\| \Big] \\
&\le \big(\ell_f + L_P\, G\big)\, \|x_{t,j}^{(i)} - x_t\|,
\end{aligned}$$

by Assumption 2, the projector Lipschitz bound (51), $\|\nabla f(x_t;\cdot)\| \le G$, and $\|P_{t,j}^{(i)}\|_{\text{op}} \le 1$. With Lemma 17(a) ($\|x_{t,j}^{(i)} - x_t\| \le 2G_{\text{eff}}\eta j$) and $\eta^2 \sum_{j=0}^{\tau-1} j = \eta^2 \tau(\tau-1)/2 \le \alpha^2/2$,

$$\eta \sum_{j=0}^{\tau-1} \|\,\text{grad}^{C} f_i(x_{t,j}^{(i)}) - \text{grad}^{C} f_i(x_t)\| \;\le\; (\ell_f + L_P G)\, G_{\text{eff}}\, \alpha^2. \tag{56}$$

Step 4: re-anchoring the SG noise. Since $\delta_{t,j}^{(i)} \in T_{x_{t,j}^{(i)}}\mathcal{M}$, i.e. $P_{t,j}^{(i)} \delta_{t,j}^{(i)} = \delta_{t,j}^{(i)}$, we have $(I - P_{x_t})\delta_{t,j}^{(i)} = (P_{t,j}^{(i)} - P_{x_t})\delta_{t,j}^{(i)}$, so with $\zeta_{t,j}^{(i)} := P_{x_t}\delta_{t,j}^{(i)}$,

$$\delta_{t,j}^{(i)} = \zeta_{t,j}^{(i)} + \big(P_{t,j}^{(i)} - P_{x_t}\big)\delta_{t,j}^{(i)}, \qquad \big\|\big(P_{t,j}^{(i)} - P_{x_t}\big)\delta_{t,j}^{(i)}\big\| \le L_P\, \|x_{t,j}^{(i)} - x_t\| \cdot 2C \le 4L_P C G_{\text{eff}}\, \eta j,$$

using (51), $\|\delta_{t,j}^{(i)}\| \le 2C$ (see (52)), and Lemma 17(a). Summing,

$$\eta \sum_{j=0}^{\tau-1} \big\|\big(P_{t,j}^{(i)} - P_{x_t}\big)\delta_{t,j}^{(i)}\big\| \;\le\; 2L_P C G_{\text{eff}}\, \alpha^2. \tag{57}$$

By (55) and $\|P_{x_t}\|_{\text{op}} \le 1$, the anchored component $\zeta_{t,j}^{(i)} = P_{x_t}(\widehat{\text{grad}}^{C} f_i(x_{t,j}^{(i)}; \mathcal{B}_{t,j}^{(i)}) - \text{grad}^{C} f_i(x_{t,j}^{(i)}))$ obeys $\mathbb{E}[\zeta_{t,j}^{(i)} \mid \mathcal{F}_{t,j}^{(i)}] = 0$ and $\mathbb{E}[\|\zeta_{t,j}^{(i)}\|^2 \mid \mathcal{F}_{t,j}^{(i)}] \le \rho_i\,\sigma_{\text{st}}^2$.

Step 5: re-anchoring the DP noise. Likewise $P_{t,j}^{(i)}\xi_{t,j}^{(i)} = P_{x_t}\xi_{t,j}^{(i)} + (P_{t,j}^{(i)} - P_{x_t})\xi_{t,j}^{(i)}$, and on $\mathcal{G}_t$, $\|\xi_{t,j}^{(i)}\| \le R$, so $\|(P_{t,j}^{(i)} - P_{x_t})\xi_{t,j}^{(i)}\| \le 2L_P G_{\text{eff}} R\,\eta j$ by (51) and Lemma 17(a). Summing,

$$\eta \sum_{j=0}^{\tau-1} \|(P_{t,j}^{(i)} - P_{x_t})\xi_{t,j}^{(i)}\| \;\le\; L_P\, G_{\text{eff}}\, R\, \alpha^2. \tag{58}$$

Step 6: assembly. Summing (54) over $j = 0, \dots, \tau - 1$ and substituting $\tilde{g}_{t,j}^{(i)} = \operatorname{grad}^C f_i(x_{t,j}^{(i)}) + \delta_{t,j}^{(i)} + P_{t,j}^{(i)} \xi_{t,j}^{(i)}$ (Step 2 and the privatised-gradient definition (12)), then re-anchoring $\delta_{t,j}^{(i)}$ and $\xi_{t,j}^{(i)}$ to x_t via Steps 4–5,

$$x_{t,\tau}^{(i)} - x_t = -\eta \sum_j \operatorname{grad}^C f_i(x_{t,j}^{(i)}) - \eta \sum_j \zeta_{t,j}^{(i)} - \eta \sum_j P_{x_t} \xi_{t,j}^{(i)} + r_t^{(i)}.$$

Replacing $-\eta \sum_j \operatorname{grad}^C f_i(x_{t,j}^{(i)})$ by $-\alpha \operatorname{grad}^C f_i(x_t)$ folds the signal drift into $r_t^{(i)}$, which then collects the drift (56), the SG re-anchoring (57), the DP re-anchoring (58), and the projection remainder $\sum_j w_j$ ($\|\sum_j w_j\| \le \tau\, C_{\mathrm{qr}} \eta^2 G_{\mathrm{eff}}^2 = C_{\mathrm{qr}} G_{\mathrm{eff}}^2 \alpha^2 / \tau \le C_{\mathrm{qr}} G_{\mathrm{eff}}^2 \alpha^2$). This yields (36) with $\|r_t^{(i)}\| \le K_1 \alpha^2$, where

$$K_1 := (\ell_f + L_P G) G_{\mathrm{eff}} + 2 L_P C G_{\mathrm{eff}} + L_P G_{\mathrm{eff}} R + C_{\mathrm{qr}} G_{\mathrm{eff}}^2, \tag{59}$$

a constant independent of T for fixed c_0.

Step 2: Server update decomposition on $\mathcal{G}_t$

Proof of Lemma 19. Recall from Lemma 17(c) the two server increments $\Delta_t^{\textsc{ProjAvg}} = \frac{1}{k} \sum_{i \in S_t} (x_{t,\tau}^{(i)} - x_t)$ and $\Delta_t^{\textsc{RLAvg}} = P_{x_t} \Delta_t^{\textsc{ProjAvg}}$. Write

$$s_t := x_t + \Delta_t^{\textsc{ProjAvg}} = \frac{1}{k} \sum_{i \in S_t} x_{t,\tau}^{(i)}, \qquad u_t := \Delta_t^{\textsc{RLAvg}} = P_{x_t}(s_t - x_t) \in T_{x_t}\mathcal{M},$$

and set the clipped-gradient signal $h_t := \frac{1}{k} \sum_{i \in S_t} \operatorname{grad}^C f_i(x_t)$.

Step 1: average the linearisation. Averaging (36) over $i \in S_t$,

$$s_t - x_t = -\alpha\, h_t + \zeta_t^{\mathrm{st}} + \zeta_t^{\mathrm{dp}} + \bar{r}_t, \qquad \bar{r}_t := \frac{1}{k} \sum_{i \in S_t} r_t^{(i)},$$

where

$$\zeta_t^{\mathrm{st}} := -\frac{\eta}{k} \sum_{i \in S_t} \sum_{j=0}^{\tau-1} \zeta_{t,j}^{(i)}, \qquad \zeta_t^{\mathrm{dp}} := -\frac{\eta}{k} \sum_{i \in S_t} \sum_{j=0}^{\tau-1} P_{x_t} \xi_{t,j}^{(i)},$$

and $\|\bar{r}_t\| \le \frac{1}{k} \sum_{i \in S_t} \|r_t^{(i)}\| \le K_1 \alpha^2$ by (59).

Step 2: the three leading terms are tangent at x_t. Each $\operatorname{grad}^C f_i(x_t)$ is an average of clipped Riemannian gradients $\operatorname{clip}_C(\operatorname{grad} f(x_t; z_{i,\ell}))$; since $\operatorname{grad} f(x_t; z) = P_{x_t} \nabla f(x_t; z) \in T_{x_t}\mathcal{M}$ and clip_C only rescales its argument, $\operatorname{grad}^C f_i(x_t) \in T_{x_t}\mathcal{M}$, hence $h_t \in T_{x_t}\mathcal{M}$. Likewise ζ_t^{st} and ζ_t^{dp} lie in $T_{x_t}\mathcal{M}$ through their explicit P_{x_t} factors. Thus P_{x_t} fixes all three, and applying P_{x_t} to Step 1,

$$u_t = P_{x_t}(s_t - x_t) = -\alpha\, h_t + \zeta_t^{\mathrm{st}} + \zeta_t^{\mathrm{dp}} + P_{x_t} \bar{r}_t. \tag{60}$$

Step 3: pass through the projection. By Lemma 17(c) both pre-projection points lie in $U_{\mathcal{M}}(\gamma)$, so the quadratic-remainder expansion (35) gives, for the two aggregators,

$$\begin{aligned} \textsc{ProjAvg}: &\quad x_{t+1} - x_t = \Pi(s_t) - x_t = P_{x_t}(s_t - x_t) + w(x_t, s_t - x_t) = u_t + w(x_t, s_t - x_t), \\ \textsc{RLAvg}: &\quad x_{t+1} - x_t = \Pi(x_t + u_t) - x_t = u_t + w(x_t, u_t), \end{aligned}$$

the second line using $P_{x_t} u_t = u_t$. Denote by w_t^{agg} the relevant remainder. Since $\|s_t - x_t\| \le 2 G_{\mathrm{eff}} \alpha$ and $\|u_t\| \le 2 G_{\mathrm{eff}} \alpha$ (Lemma 17(c)),

$$\|w_t^{\mathrm{agg}}\| \le C_{\mathrm{qr}} (2 G_{\mathrm{eff}} \alpha)^2 = 4 C_{\mathrm{qr}} G_{\mathrm{eff}}^2 \alpha^2$$

in either case.

Step 4: assembly. Substituting (60),

$$x_{t+1} - x_t = u_t + w_t^{\mathrm{agg}} = -\alpha\, h_t + \zeta_t^{\mathrm{st}} + \zeta_t^{\mathrm{dp}} + r_t, \qquad r_t := P_{x_t}\bar{r}_t + w_t^{\mathrm{agg}}.$$

By $\|P_{x_t}\|_{\mathrm{op}} \le 1$ and the triangle inequality, $\|r_t\| \le \|\bar{r}_t\| + \|w_t^{\mathrm{agg}}\| \le K_1\alpha^2 + 4C_{\mathrm{qr}}G_{\mathrm{eff}}^2\alpha^2 = K_2\alpha^2$ with $K_2 := K_1 + 4C_{\mathrm{qr}}G_{\mathrm{eff}}^2$, which is (37).

Lemma 21 (Moment bounds on $\mathcal{G}_t$)**.** *Adopt the notation of Lemma 19 and recall* $\mathbb{E}_t[\cdot] = \mathbb{E}[\cdot \mid \mathcal{F}_t]$ *and* $\mathbb{E}_t^{\mathcal{G}}[\cdot] = \mathbb{E}[\cdot \mid \mathcal{F}_t, \mathcal{G}_t]$ *from Section 5.1.*

(a) ***Signal:***

$$\mathbb{E}_t^{\mathcal{G}}[h_t] = \operatorname{grad} F(x_t) + \bar{b}^C(x_t), \tag{61}$$

$$\mathbb{E}_t^{\mathcal{G}}\big[\|h_t - \overline{\operatorname{grad}^C f}(x_t)\|^2\big] \le \frac{n-k}{k(n-1)}\,(\sigma_{\mathrm{het}} + \sigma_{\mathrm{clip}})^2, \tag{62}$$

$$\mathbb{E}_t^{\mathcal{G}}[\|h_t\|^2] \le 2\|\operatorname{grad} F(x_t)\|^2 + 2\beta_C^2 + \frac{n-k}{k(n-1)}\,(\sigma_{\mathrm{het}} + \sigma_{\mathrm{clip}})^2. \tag{63}$$

(b) ***SG noise:*** $\mathbb{E}_t^{\mathcal{G}}[\zeta_t^{st}] = 0$ *and*

$$\mathbb{E}_t^{\mathcal{G}}[\|\zeta_t^{st}\|^2] \;\le\; \frac{\alpha^2}{k\tau}\cdot\bar{\rho}\,\sigma_{\mathrm{st}}^2. \tag{64}$$

(c) ***DP noise:*** $\mathbb{E}_t^{\mathcal{G}}[\zeta_t^{dp}] = 0$ *and*

$$\mathbb{E}_t^{\mathcal{G}}[\|\zeta_t^{dp}\|^2] \;\le\; \frac{\alpha^2}{k\tau}\cdot d_{\mathcal{M}}\sigma_{\mathrm{dp}}^2. \tag{65}$$

(d) ***Cross-terms vanish:***

$$\mathbb{E}_t^{\mathcal{G}}\big[\langle\zeta_t^{st}, \zeta_t^{dp}\rangle\big] = \mathbb{E}_t^{\mathcal{G}}\big[\langle h_t, \zeta_t^{st}\rangle\big] = \mathbb{E}_t^{\mathcal{G}}\big[\langle h_t, \zeta_t^{dp}\rangle\big] = 0. \tag{66}$$

Moreover, every cross-term involving the residual r_t *is* $O(\alpha^3)$.

Proof. The noise vectors $\xi_{t,j}^{(i)} \sim \mathcal{N}(0, \sigma_{\mathrm{dp}}^2 I_{\mathcal{E}})$ are i.i.d. across all $(i,j) \in [n] \times \{0, \dots, \tau-1\}$, so the probability that all selected clients' noise draws lie in $\{\|\cdot\| \le R\}$ is $\Pr(\mathcal{G}_t \mid S_t, \mathcal{F}_t) = \Pr_{\xi\sim\mathcal{N}(0,\sigma_{\mathrm{dp}}^2 I_{\mathcal{E}})}\big(\|\xi\| \le R\big)^{k\tau}$, independent of which clients are in S_t. Hence $S_t \perp \mathcal{G}_t \mid \mathcal{F}_t$, and conditioning on $\mathcal{G}_t$ leaves the conditional law of S_t given $\mathcal{F}_t$ unchanged. We use this freely below.

(a) Signal. h_t is a function of (x_t, S_t) alone; by the independence noted above, its conditional law given $\mathcal{F}_t$ is unchanged by conditioning on $\mathcal{G}_t$, so all three displays may be computed under $\mathbb{E}_t$ and transfer verbatim to $\mathbb{E}_t^{\mathcal{G}}$. Each client is selected with probability k/n (uniform without-replacement sampling), so $\mathbb{E}_t[h_t] = \frac{1}{n}\sum_i \operatorname{grad}^C f_i(x_t) = \overline{\operatorname{grad}^C f}(x_t)$, giving (61) after substituting $\overline{\operatorname{grad}^C f}(x_t) = \operatorname{grad} F(x_t) + \bar{b}^C(x_t)$.

For (62), the finite-population variance of the sample mean of $\{\operatorname{grad}^C f_i(x_t)\}_{i=1}^n$ under uniform without-replacement sampling of size k is

$$\mathbb{E}_t\big[\|h_t - \overline{\operatorname{grad}^C f}(x_t)\|^2\big] = \frac{n-k}{n-1}\cdot\frac{S^2}{k},$$

where $S^2 := \frac{1}{n}\sum_i \|\operatorname{grad}^C f_i(x_t) - \overline{\operatorname{grad}^C f}(x_t)\|^2 \le (\sigma_{\mathrm{het}} + \sigma_{\mathrm{clip}})^2$ by Lemma 12.

For (63), the bias-variance decomposition with $\mathbb{E}_t[h_t] = \overline{\operatorname{grad}^C f}(x_t)$ gives

$$\mathbb{E}_t[\|h_t\|^2] = \|\overline{\operatorname{grad}^C f}(x_t)\|^2 + \mathbb{E}_t\big[\|h_t - \overline{\operatorname{grad}^C f}(x_t)\|^2\big].$$

We then bound $\|\overline{\operatorname{grad}^C f}(x_t)\|^2 = \|\operatorname{grad} F(x_t) + \bar{b}^C(x_t)\|^2 \le 2\|\operatorname{grad} F(x_t)\|^2 + 2\beta_C^2$ via Young's inequality and $\|\bar{b}^C(x_t)\|^2 \le \beta_C^2$ and combine it with (62).

(b) SG noise. Fix (i,j). Conditional on $\mathcal{F}_{t,j}^{(i)}$ the trajectory point $x_{t,j}^{(i)}$ is determined, and the batch $\mathcal{B}_{t,j}^{(i)}$ is uniform and independent of $\mathcal{F}_{t,j}^{(i)}$ and of all noise draws — in particular of $\mathcal{G}_t$ — so $\mathbb{E}[\zeta_{t,j}^{(i)} \mid \mathcal{F}_{t,j}^{(i)}, \mathcal{G}_t] = 0$. Taking total expectation gives $\mathbb{E}_t^{\mathcal{G}}[\zeta_t^{\mathrm{st}}] = 0$.

For the second moment, cross-terms across different (i,j) vanish: within a client, by the tower property and the zero conditional mean at the finer σ-algebra; across clients, by independence of the fresh minibatches. Hence, using the tower property and $\mathbb{E}[\|\zeta_{t,j}^{(i)}\|^2 \mid \mathcal{F}_{t,j}^{(i)}, \mathcal{G}_t] \le \rho_i\, \sigma_{\mathrm{st}}^2$ (Lemma 11; the fresh batch is independent of the noise draws, hence of $\mathcal{G}_t$),

$$\mathbb{E}\big[\|\zeta_t^{\mathrm{st}}\|^2 \;\big|\; \mathcal{F}_t, \mathcal{G}_t, S_t\big] = \frac{\eta^2}{k^2} \sum_{(i,j)\in S_t\times\{0,\dots,\tau-1\}} \mathbb{E}\big[\|\zeta_{t,j}^{(i)}\|^2 \;\big|\; \mathcal{F}_t, \mathcal{G}_t, S_t\big] \;\le\; \frac{\eta^2\tau\,\sigma_{\mathrm{st}}^2}{k^2} \sum_{i\in S_t} \rho_i.$$

Averaging over S_t, which is uniform without replacement so that $\mathbb{E}_t^{\mathcal{G}}\big[\sum_{i\in S_t}\rho_i\big] = k\bar\rho$ with $\bar\rho := \frac{1}{n}\sum_{i=1}^n \rho_i$, and using $\eta^2\tau/k = \alpha^2/(k\tau)$,

$$\mathbb{E}_t^{\mathcal{G}}\big[\|\zeta_t^{\mathrm{st}}\|^2\big] \;\le\; \frac{\alpha^2}{k\tau}\,\bar\rho\,\sigma_{\mathrm{st}}^2.$$

(c) DP noise. P_{x_t} is $\mathcal{F}_t$-measurable, so the vectors $\{P_{x_t}\xi_{t,j}^{(i)}\}_{(i,j)\in S_t\times\{0,\dots,\tau-1\}}$ are deterministic linear images of i.i.d. Gaussian draws and are themselves i.i.d. given $\mathcal{F}_t$. The event $\mathcal{G}_t = \bigcap_{(i,j)\in S_t\times\{0,\dots,\tau-1\}}\{\|\xi_{t,j}^{(i)}\| \le R\}$ is an intersection of one event per draw, each depending on that draw alone. Since the draws are independent given $(\mathcal{F}_t, S_t)$, the conditional law factorises,

$$\mathcal{L}\big(\{\xi_{t,j}^{(i)}\}_{(i,j)} \;\big|\; \mathcal{F}_t, S_t, \mathcal{G}_t\big) = \bigotimes_{(i,j)} \mathcal{L}\big(\xi_{t,j}^{(i)} \;\big|\; \|\xi_{t,j}^{(i)}\| \le R\big),$$

so conditional on $(\mathcal{F}_t, S_t, \mathcal{G}_t)$ each $\xi_{t,j}^{(i)}$ is $\mathcal{N}(0, \sigma_{\mathrm{dp}}^2 I_{\mathcal{E}})$ truncated to $\{\|\xi\| \le R\}$, independently across (i,j). This is the single-draw setting of Lemma 20.

Sign-symmetry of the truncated Gaussian under $\xi \mapsto -\xi$ gives $\mathbb{E}[P_{x_t}\xi_{t,j}^{(i)} \mid \mathcal{F}_t, \mathcal{G}_t] = 0$, hence $\mathbb{E}_t^{\mathcal{G}}[\zeta_t^{\mathrm{dp}}] = 0$.

For the second moment, Lemma 20 applied with noise scale σ_{dp} and projector P_{x_t} (rank $d_{\mathcal{M}}$) gives

$$\mathbb{E}[\|P_{x_t}\xi_{t,j}^{(i)}\|^2 \mid \mathcal{F}_t, \mathcal{G}_t] \le d_{\mathcal{M}}\,\sigma_{\mathrm{dp}}^2.$$

Summing the $k\tau$ independent contributions (cross-terms vanish by conditional independence and zero mean; the tower property over S_t removes the extra conditioning) and using $\eta^2\tau/k = \alpha^2/(k\tau)$:

$$\mathbb{E}_t^{\mathcal{G}}[\|\zeta_t^{\mathrm{dp}}\|^2] \le \frac{\eta^2}{k^2}\cdot k\tau\cdot d_{\mathcal{M}}\sigma_{\mathrm{dp}}^2 = \frac{\alpha^2}{k\tau}\cdot d_{\mathcal{M}}\sigma_{\mathrm{dp}}^2.$$

Tangent projection thus replaces the ambient dimension D by the intrinsic dimension $d_{\mathcal{M}}$ in the DP noise budget.

(d) Cross-terms.

Expand $\langle \zeta_t^{\mathrm{st}}, \zeta_t^{\mathrm{dp}}\rangle$ into pairs $\langle \zeta_{t,j}^{(i)},\, P_{x_t}\xi_{t,j'}^{(i')}\rangle$. Fix a pair and condition on $\mathcal{A} :=$ the σ-algebra generated by $\mathcal{F}_t$, S_t, all round-t noise draws, and all round-t batches *except* $\mathcal{B}_{t,j}^{(i)}$. Then $\mathcal{G}_t$ is $\mathcal{A}$-measurable, $x_{t,j}^{(i)}$ and $P_{x_t}\xi_{t,j'}^{(i')}$ are $\mathcal{A}$-measurable, and the fresh batch $\mathcal{B}_{t,j}^{(i)}$ is independent of $\mathcal{A}$, so $\mathbb{E}[\zeta_{t,j}^{(i)} \mid \mathcal{A}] = 0$ and the pair has zero conditional expectation on $\mathcal{G}_t$. Summing pairs gives $\mathbb{E}_t^{\mathcal{G}}[\langle \zeta_t^{\mathrm{st}}, \zeta_t^{\mathrm{dp}}\rangle] = 0$.

h_t is $(\mathcal{F}_t, S_t)$-measurable, so conditioning further on S_t and using $\mathbb{E}[\zeta_t^{\bullet} \mid \mathcal{F}_t, \mathcal{G}_t, S_t] = 0$ from (b)–(c),

$$\mathbb{E}_t^{\mathcal{G}}[\langle h_t, \zeta_t^{\bullet}\rangle] = \mathbb{E}_t^{\mathcal{G}}\big[\langle h_t,\; \mathbb{E}[\zeta_t^{\bullet} \mid \mathcal{F}_t, \mathcal{G}_t, S_t]\rangle\big] = 0$$

for $\bullet \in \{\mathrm{st}, \mathrm{dp}\}$.

For the residual: by Cauchy–Schwarz, $|\mathbb{E}_t^{\mathcal{G}}[\langle X, r_t\rangle]| \le \sqrt{\mathbb{E}_t^{\mathcal{G}}\|X\|^2}\cdot\sqrt{\mathbb{E}_t^{\mathcal{G}}\|r_t\|^2}$. Each of $-\alpha h_t, \zeta_t^{\mathrm{st}}, \zeta_t^{\mathrm{dp}}$ has L^2-norm $O(\alpha)$ under $\mathbb{E}_t^{\mathcal{G}}$ (by (a)–(c)), while $\|r_t\| \le K_2\alpha^2$ deterministically on $\mathcal{G}_t$ (Lemma 19). The product is $O(\alpha^3)$. □

Step 3: One-round expected descent

Lemma 22 (One-round descent on $\mathcal{G}_t$)**.** *On $\mathcal{G}_t$, under $\alpha \le \min\{\gamma/(4G_{\text{eff}}),\, 1/(4L_{\mathcal{M}})\}$,*

$$\begin{aligned}\mathbb{E}_t^{\mathcal{G}}[F(x_{t+1})] \le F(x_t) - \frac{9\alpha}{16}\|\operatorname{grad} F(x_t)\|^2 + (2\alpha + L_{\mathcal{M}}\alpha^2)\,\beta_C^2 \\ + \frac{L_{\mathcal{M}}\alpha^2}{2}\cdot\frac{n-k}{k(n-1)}\,(\sigma_{\text{het}}+\sigma_{\text{clip}})^2 + \frac{L_{\mathcal{M}}\alpha^2}{2k\tau}\big(\bar\rho\sigma_{\text{st}}^2 + d_{\mathcal{M}}\sigma_{\text{dp}}^2\big) + K_3\,\alpha^3,\end{aligned} \tag{67}$$

where

$$K_3 := 5K_2^2 + 3L_{\mathcal{M}}K_2G_{\text{eff}}, \tag{68}$$

with $K_2 := K_1 + 4C_{\text{qr}}G_{\text{eff}}^2$ from Lemma 19.

Proof. Throughout the proof, $\operatorname{grad} F$ and $\bar b^C$ are evaluated at x_t. The descent inequality (22) applied at $x_t, x_{t+1} \in \mathcal{M}$ gives

$$F(x_{t+1}) \le F(x_t) + \langle \operatorname{grad} F(x_t),\, x_{t+1}-x_t\rangle + \frac{L_{\mathcal{M}}}{2}\|x_{t+1}-x_t\|^2. \tag{69}$$

We take $\mathbb{E}_t^{\mathcal{G}}[\cdot]$ on both sides and substitute the server-update decomposition (37).

Linear term. Using $\mathbb{E}_t^{\mathcal{G}}[\zeta_t^{\text{st}}] = \mathbb{E}_t^{\mathcal{G}}[\zeta_t^{\text{dp}}] = 0$ (Lemma 21(b)–(c)) and $\mathbb{E}_t^{\mathcal{G}}[h_t] = \operatorname{grad} F(x_t) + \bar b^C(x_t)$ (Lemma 21(a)):

$$\mathbb{E}_t^{\mathcal{G}}[\langle \operatorname{grad} F, x_{t+1}-x_t\rangle] = -\alpha\|\operatorname{grad} F\|^2 - \alpha\langle \operatorname{grad} F,\, \bar b^C\rangle + \langle \operatorname{grad} F,\, \mathbb{E}_t^{\mathcal{G}}[r_t]\rangle.$$

Bound the bias cross-term by Cauchy–Schwarz (with $\|\bar b^C(x_t)\| \le \beta_C$) then Young's inequality ($ab \le \frac{1}{8}a^2 + 2b^2$):

$$|\alpha\langle \operatorname{grad} F, \bar b^C\rangle| \le \frac{\alpha}{8}\|\operatorname{grad} F\|^2 + 2\alpha\,\beta_C^2.$$

Similarly, by $\|r_t\| \le K_2\alpha^2$ (Lemma 19) and Young's inequality with $\epsilon = \alpha/8$:

$$|\langle \operatorname{grad} F, \mathbb{E}_t^{\mathcal{G}}[r_t]\rangle| \le \|\operatorname{grad} F\|\cdot K_2\alpha^2 \le \frac{\alpha}{16}\|\operatorname{grad} F\|^2 + 4K_2^2\alpha^3.$$

Combining:

$$\mathbb{E}_t^{\mathcal{G}}[\langle \operatorname{grad} F, x_{t+1}-x_t\rangle] \le -\frac{13\alpha}{16}\|\operatorname{grad} F\|^2 + 2\alpha\,\beta_C^2 + 4K_2^2\alpha^3. \tag{70}$$

Quadratic term. Expanding $\|x_{t+1}-x_t\|^2$ from (37) and applying Lemma 21(d) for the cross-term cancellations:

$$\mathbb{E}_t^{\mathcal{G}}[\|x_{t+1}-x_t\|^2] = \alpha^2\mathbb{E}_t^{\mathcal{G}}[\|h_t\|^2] + \mathbb{E}_t^{\mathcal{G}}[\|\zeta_t^{\text{st}}\|^2] + \mathbb{E}_t^{\mathcal{G}}[\|\zeta_t^{\text{dp}}\|^2] + R_t,$$

where R_t collects all residual cross-terms involving r_t, plus $\|r_t\|^2$:

$$R_t = -2\alpha\mathbb{E}_t^{\mathcal{G}}[\langle h_t, r_t\rangle] + 2\mathbb{E}_t^{\mathcal{G}}[\langle \zeta_t^{\text{st}} + \zeta_t^{\text{dp}},\, r_t\rangle] + \mathbb{E}_t^{\mathcal{G}}[\|r_t\|^2].$$

By Cauchy–Schwarz with $\|h_t\| \le C$ (each $\|\operatorname{grad}^C f_i\| \le C$ pointwise by (52)); with $\|\zeta_t^{\text{st}}\| \le 2C\alpha$, since $\|\zeta_{t,j}^{(i)}\| \le \|\delta_{t,j}^{(i)}\| \le 2C$ (Step 4 of the proof of Lemma 18) and $\eta\tau = \alpha$; with $\|\zeta_t^{\text{dp}}\| \le R\alpha$ on $\mathcal{G}_t$, since $\|P_{x_t}\xi_{t,j}^{(i)}\| \le \|\xi_{t,j}^{(i)}\| \le R$ there (so $\|\zeta_t^{\text{st}} + \zeta_t^{\text{dp}}\| \le 2G_{\text{eff}}\alpha$); and with $\|r_t\| \le K_2\alpha^2$:

$$|R_t| \le 2\alpha\cdot C\cdot K_2\alpha^2 + 2\cdot 2G_{\text{eff}}\alpha\cdot K_2\alpha^2 + K_2^2\alpha^4 \le 2K_2(C+2G_{\text{eff}})\alpha^3 + K_2^2\alpha^4.$$

Using (63), (64), (65), multiplying by $L_{\mathcal{M}}/2$, and applying $L_{\mathcal{M}}\alpha \le 1/4$:

$$\begin{aligned}\frac{L_{\mathcal{M}}}{2}\mathbb{E}_t^{\mathcal{G}}[\|x_{t+1}-x_t\|^2] \le \frac{\alpha}{4}\|\operatorname{grad} F\|^2 + L_{\mathcal{M}}\alpha^2\beta_C^2 + \frac{L_{\mathcal{M}}\alpha^2}{2}\cdot\frac{n-k}{k(n-1)}(\sigma_{\text{het}}+\sigma_{\text{clip}})^2 \\ + \frac{L_{\mathcal{M}}\alpha^2}{2k\tau}\big(\bar\rho\sigma_{\text{st}}^2 + d_{\mathcal{M}}\sigma_{\text{dp}}^2\big) + L_{\mathcal{M}}K_2(C+2G_{\text{eff}})\alpha^3 + \tfrac{1}{8}K_2^2\,\alpha^3.\end{aligned} \tag{71}$$

where we bounded $\frac{L_{\mathcal{M}}}{2}K_2^2\alpha^4 \le \frac{1}{8}K_2^2\alpha^3$ using $L_{\mathcal{M}}\alpha \le 1/4$ and folded it into the α^3 remainder.

Combining. Adding (70) and (71), the $\|\operatorname{grad} F\|^2$ coefficient is $-13/16 + 1/4 = -9/16$. The β_C^2 coefficient is $(2\alpha + L_{\mathcal{M}}\alpha^2)$. The α^3 remainder is $4K_2^2 + \frac{1}{8}K_2^2 + L_{\mathcal{M}}K_2(C + 2G_{\text{eff}}) \le 5K_2^2 + 3L_{\mathcal{M}}K_2G_{\text{eff}}$ (using $C \le G_{\text{eff}}$), giving K_3 as defined in (68). □

Step 4: Bad rounds and telescoping

Almost-sure feasibility of the iterates. Outside the tube $U_{\mathcal{M}}(2\gamma)$ the nearest-point map may be multi-valued; we fix a measurable selection of it, which exists since $\mathcal{M}$ is compact. Every iterate then lies in $\mathcal{M}$ surely, so the descent inequality (22) applies on every round, and off $\mathcal{G}_t$ we use only $F(x_{t+1}) - F(x_t) \le \delta_F$.

Combining the two cases. By the factorisation in the proof of Lemma 14, $\Pr(\mathcal{G}_t^c \mid \mathcal{F}_t) = \pi \le k\tau e^{-c_0/2}$ is a deterministic constant, the same for every t. On $\mathcal{G}_t^c$ the iterate stays in $\mathcal{M}$, so $F(x_{t+1}) - F(x_t) \le \delta_F$ by (21). Decomposing on $\mathcal{G}_t$ and applying Lemma 22,

$$\begin{aligned}\mathbb{E}_t[F(x_{t+1})] &= (1-\pi)\,\mathbb{E}_t^{\mathcal{G}}[F(x_{t+1})] + \pi\,\mathbb{E}\big[F(x_{t+1}) \mid \mathcal{F}_t, \mathcal{G}_t^c\big] \\ &\le F(x_t) - (1-\pi)\frac{9\alpha}{16}\|\operatorname{grad} F(x_t)\|^2 + (1-\pi)\,V + \pi\,\delta_F, \end{aligned} \tag{72}$$

where

$$V := (2\alpha + L_{\mathcal{M}}\alpha^2)\beta_C^2 + \frac{L_{\mathcal{M}}\alpha^2}{2}\cdot\frac{n-k}{k(n-1)}(\sigma_{\text{het}} + \sigma_{\text{clip}})^2 + \frac{L_{\mathcal{M}}\alpha^2}{2k\tau}\big(\bar{\rho}\,\sigma_{\text{st}}^2 + d_{\mathcal{M}}\sigma_{\text{dp}}^2\big) + K_3\,\alpha^3. \tag{73}$$

Proof of Theorem 15. Take full expectations in (72) and sum from $t = 0$ to $T-1$. The tower property gives $\sum_t \mathbb{E}[F(x_{t+1}) - F(x_t)] = \mathbb{E}[F(x_T)] - F(x_0)$, and $\mathbb{E}[F(x_T)] \ge F_\star$ since $x_T \in \mathcal{M}$, so

$$(1-\pi)\frac{9\alpha}{16}\sum_{t=0}^{T-1}\mathbb{E}\big[\|\operatorname{grad} F(x_t)\|^2\big] \;\le\; F(x_0) - F_\star + (1-\pi)\,T\,V + T\,\pi\,\delta_F.$$

Dividing by $(1-\pi)\frac{9\alpha}{16}T$,

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\big[\|\operatorname{grad} F(x_t)\|^2\big] \;\le\; \frac{16(F(x_0) - F_\star)}{9(1-\pi)\alpha T} + \frac{16V}{9\alpha} + \frac{16\,\pi\,\delta_F}{9(1-\pi)\alpha}.$$

Expanding V/α and using $L_{\mathcal{M}}\alpha \le 1/4$, which gives $\frac{16(2+L_{\mathcal{M}}\alpha)}{9} \le 4$,

$$\frac{16V}{9\alpha} \;\le\; 4\beta_C^2 + \frac{8L_{\mathcal{M}}\alpha}{9k}\left(\frac{n-k}{n-1}(\sigma_{\text{het}} + \sigma_{\text{clip}})^2 + \frac{1}{\tau}\big(\bar{\rho}\,\sigma_{\text{st}}^2 + d_{\mathcal{M}}\sigma_{\text{dp}}^2\big)\right) + \frac{16K_3}{9}\alpha^2.$$

Substituting back and setting $K := 16K_3/9$ yields (32). □

B.8 Proof of Corollary 16

We substitute $c_0 = 4\log T + 2\log(k\tau)$ and $\alpha = c/\sqrt{T}$ into (32).

Bad-round cost is subdominant. With this c_0, $\pi \le k\tau \cdot e^{-c_0/2} = k\tau \cdot T^{-2}(k\tau)^{-1} = T^{-2}$. The bad-round term satisfies

$$\frac{16\,\pi\,\delta_F}{9(1-\pi)\alpha} \;\le\; \frac{16\,\delta_F}{9c(1-T^{-2})}\cdot T^{-3/2} \;=\; O(T^{-3/2}) \;=\; o(T^{-1/2}).$$

For $T \ge 4$, $\pi \le 1/16$ and $1/(1-\pi) \le 16/15 < 2$.

Logarithmic inflation of G_{eff}. $G_{\text{eff}} = C + \sigma_{\text{dp}}(\sqrt{D} + \sqrt{c_0}) = O(\sqrt{\log T})$, so $K_3 = O((\log T)^2)$ and the residual $K\alpha^2 = O((\log T)^2/T) = o(1/\sqrt{T})$ is subdominant.

Stepsize-constraint regime. The constraint $\alpha = c/\sqrt{T}$ is binding (as opposed to $\gamma/(4G_{\text{eff}})$ or $1/(4L_{\mathcal{M}})$) once $T \geq \max\{16c^2G_{\text{eff}}^2/\gamma^2,\, 16c^2L_{\mathcal{M}}^2\}$. Since $G_{\text{eff}}^2 = O(\log T)$, the first inequality reads $T \geq C_1(1+\log T)$ for $C_1 = O\big(c^2(C^2 + \sigma_{\text{dp}}^2(D + \log(k\tau)))/\gamma^2\big)$. Define T_0 as the smallest integer such that $T \geq C_1(1+\log T) + 16c^2L_{\mathcal{M}}^2$ for all $T \geq T_0$; such T_0 exists because the RHS is $O(\log T) = o(T)$, and depends only on geometric and algorithmic constants. For $T \geq T_0$, the finite-T inequality (32) simplifies to the rate (33), completing the proof of Corollary 16.

The hidden constants in (33) depend on $(F(x_0) - F_\star,\ \delta_F,\ \ell_f,\ L_{\mathcal{M}},\ \gamma,\ C,\ G,\ C_{\text{qr}},\ \sigma_{\text{dp}})$ but not on T; since $G_{\text{eff}} = O(\sqrt{D + \log T})$, the ambient dimension enters only through a subdominant remainder.

C Experimental details and additional results

C.1 Calibrating the ReEig floor

The ReEig layer rectifies the spectrum of its input, inducing non-linearity:

$$U\Lambda U^\top \longmapsto U\,\mathrm{diag}\big(\max(\lambda_1, \varepsilon_{\text{R}}), \ldots, \max(\lambda_d, \varepsilon_{\text{R}})\big)U^\top, \tag{74}$$

with $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_d)$ the eigenvalues of the BiMap output. We fix ε_{R} before any training: for each dataset we pool the trace-normalized covariance spectra of its first three subjects, at most 2 000 matrices, and take the 15th percentile, so that ReEig rectifies roughly the bottom sixth of the spectrum.

C.2 Centralized grid search

The batch size and the learning rate are selected beforehand by a grid search on half of each dataset's subjects, retaining the best mean validation accuracy. For SPDNet, the BiMap output dimension d is selected the same way but among the values whose parameter count stays below its EEGNet counterpart's. Table 11 reports the grids explored by the centralized sweeps of Section 6.3. Each cell is run with the three seeds $\{7, 42, 123\}$ on roughly half of the subjects, for at most 300 epochs (400 on Weibo2014), and the selected configuration is the smallest one within one standard deviation of the best mean validation accuracy. The final centralized runs then use all subjects, ten seeds $\{7, 42, 123, 400, 800, 1200, 1600, 2000, 2400, 2800\}$ and a cap of 500 epochs.

Table 11: Centralized hyperparameter grids.

Dataset	Model	d	Batch size	Learning rate
BNCI2014_001	SPDNet	$\{16, 22\}$	$\{12, 16, 24\}$	$\{10^{-3}, 3\cdot10^{-3}, 10^{-2}\}$
	EEGNet	—	$\{32, 64, 128\}$	$\{3\cdot10^{-4}, 10^{-3}, 3\cdot10^{-3}\}$
Weibo2014	SPDNet	$\{20, 30, 45\}$	$\{16, 32\}$	$\{3\cdot10^{-4}, 10^{-3}, 3\cdot10^{-3}\}$
	EEGNet	—	$\{32, 64, 128\}$	$\{3\cdot10^{-4}, 10^{-3}, 3\cdot10^{-3}\}$
Cho2017	SPDNet	$\{16, 32\}$	$\{32, 64\}$	$\{3\cdot10^{-4}, 10^{-3}, 3\cdot10^{-3}\}$
	EEGNet	—	$\{32, 64, 128\}$	$\{3\cdot10^{-4}, 10^{-3}, 3\cdot10^{-3}\}$

C.3 Federated grid search

Batch sizes and learning rates are selected at $\varepsilon = 20$, $\delta = 10^{-5}$ and $C = 1$, with the cosine schedule [2], full participation and three seeds per cell. Each axis is swept at the incumbent value of the other, and we retain the rate beating both its neighbours, extending the grid where the initial range left the optimum at an edge. For SPDNet the batch size is swept under ProjAvg and shared by all three rules, as is the learning rate by ProjAvg and RLAvg.

Learning rates are drawn from a equally-spaced grid, with values in $[3\cdot10^{-2}, 3\cdot10^{-1}]$ for SPDNet and in $[3\cdot10^{-4}, 10^{-1}]$ for EEGNet, the two architectures operating in disjoint ranges. Batch sizes span $\{16, \ldots, 128\}$, bounded above by the client sizes. The retained values are those of Table 7.

[2] $\eta_t = \eta_{\min} + \frac{1}{2}(\eta_0 - \eta_{\min})\big(1 + \cos(\pi t/T)\big),\ t = 0, \ldots, T-1,\ \eta_{\min} = 10^{-4}$.

Selecting RetAvg on feasibility. RetAvg's learning rate is selected on feasibility instead of accuracy: the rates above drive clients outside the domain of its inverse retraction, so we descend a dataset-specific grid in $[6 \cdot 10^{-3}, 10^{-1}]$ at ten seeds and keep the largest rate at which every seed completes in every mode. Feasibility depends on how far each client drifts from the aggregate. The aggregate is formed by the clients sampled that round, so halving participation changes it, and a rate that was feasible may no longer be. We therefore re-screen RetAvg at half participation, over five values in $[6 \cdot 10^{-3}, 3 \cdot 10^{-2}]$. Two of the three rates move, in opposite directions (Table 7). All other configurations are held fixed across $\varepsilon \in \{20, 10, 5\}$ and in the non-private runs.

C.4 Privacy settings

The privacy mechanism is fixed by the batch size and the client training sets: the accounting charges a subsampling rate $q = B/m_\star$ at the smallest client and $\tau = \lceil m_{\max}/B \rceil$ steps per round at the largest, composing over $T\tau$ steps in total, from which σ_{dp} is calibrated to the target ε. The clipping constant is pinned a priori at $C = 1$ for every run, so no data is consulted to set it: at a fixed budget σ_{dp} calibrates linearly in C, so the choice rescales signal and noise together.

Calibrating the noise scale. For each target (ε, δ), σ_{dp} is the smallest noise scale whose ε under an exact accountant for $C_q(G_{\mu_0})^{\otimes T\tau}$ meets the target. The per-step profile is closed form and composition uses a privacy-loss distribution built to dominate it, so the reported ε is a certified upper bound; (18) only interprets the budget and never sets σ_{dp}.

PriRFed accounting. Table 10 holds the mechanism (C, B, q, τ) fixed and changes only the outer accounting: PriRFed's Theorem 7 (Huang et al., 2026) scalarises each round into $(\varepsilon_r, \delta_r)$ and advanced-composes the T rounds, minimised over the splits $\delta = \hat{\delta} + T\delta_r$. We feed it our own exact per-round $C_q(G_{\mu_0})^{\otimes\tau}$ in place of the looser closed form it uses as published, so the gap measures that wrapper alone and understates the improvement. Amplification by client sampling is disabled ($\rho = 1$) for both methods, the server drawing and recording S_t (Remark 7); the ratio is then invariant in C and equals 1 at $T = 1$.

C.5 Normalization under record-level DP

The sensitivity $\mathrm{sens} = 2C/B$ behind (12) rests on one property: substituting a single record must change exactly one summand of the averaged clipped gradient, which requires each per-sample gradient to be a function of its own record alone. Batch normalization in training mode computes, for a layer input h,

$$\mu_{\mathcal{B}} = \frac{1}{B}\sum_{l=1}^{B} h(z_l), \qquad \varsigma^2_{\mathcal{B}} = \frac{1}{B}\sum_{l=1}^{B}\big(h(z_l) - \mu_{\mathcal{B}}\big)^2, \tag{75}$$

so that every per-sample gradient depends on every record of the minibatch through $(\mu_{\mathcal{B}}, \varsigma^2_{\mathcal{B}})$. A one-record substitution then perturbs all B summands, the decomposition fails, and $\mathrm{sens} = 2C/B$ does not hold.

Writing $\mathcal{P}_i$ for the held-out validation split of client i, treated as public, we consider three ways of replacing (75) by statistics independent of the private batch; the third uses no public data at all:

$$\begin{aligned}
&\text{(a) public, frozen affine} && y = \frac{h - \mu_{\mathcal{P}_i}}{\sqrt{\varsigma^2_{\mathcal{P}_i} + \epsilon_0}}, && \gamma \equiv 1,\ \beta \equiv 0 \text{ fixed}, \\
&\text{(b) public, trained affine} && y = \gamma\,\frac{h - \mu_{\mathcal{P}_i}}{\sqrt{\varsigma^2_{\mathcal{P}_i} + \epsilon_0}} + \beta, && (\gamma, \beta) \text{ trained, clipped and noised}, \\
&\text{(c) per-record} && y_{c,s} = \frac{h_{c,s}(z) - \mu_{g(c)}(z)}{\sqrt{\varsigma^2_{g(c)}(z) + \epsilon_0}}, && \gamma \equiv 1,\ \beta \equiv 0 \text{ fixed},
\end{aligned} \tag{76}$$

where $\epsilon_0 > 0$ is a fixed numerical stabiliser. In (c), $h(z) \in \mathbb{R}^{C \times S}$ is the layer input on record z with C channels and S spatial positions, $\mathcal{G}_1, \ldots, \mathcal{G}_G$ partition the channels into G groups, $g(c)$ is the group containing

channel c, and, for $m = |\mathcal{G}_g|\, S$,

$$\mu_g(z) = \frac{1}{m} \sum_{c \in \mathcal{G}_g} \sum_{s=1}^{S} h_{c,s}(z), \qquad \varsigma_g^2(z) = \frac{1}{m} \sum_{c \in \mathcal{G}_g} \sum_{s=1}^{S} \big(h_{c,s}(z) - \mu_g(z)\big)^2. \tag{77}$$

Variants (a) and (b) recompute the statistics at the start of a round and hold them fixed throughout it. In (b) the affine is trained from the clipped and noised gradient rather than the raw one: (γ, β) sit inside the shared forward pass, so training them on the private batch would make every subsequent release depend on that batch by a route the accounting does not model. Trained from the privatized gradient they are a function of the privatized stream alone, and the guarantee holds by post-processing. In both variants the statistics and the affine stay on the client and are never communicated, as under FedBN.

Variant (c) is group normalization (Wu & He, 2018). Comparing (77) with (75), the average runs over the channels and spatial positions of the single record z instead of over the minibatch, so y depends on z alone, the decomposition holds unmodified, and no public data is needed. We use $G = 4$ groups at each of EEGNet's three normalization layers, and keep the affine local and frozen at $(1, 0)$, matching the treatment of the batch-normalization affine in (a). Variant (c) is an ablation, not the published EEGNet architecture.

Table 12: Centralized (non-federated, non-private) test accuracy (%) over 10 seeds. The GroupNorm row uses identical settings to the BatchNorm row.

Dataset	Architecture	Test accuracy
BNCI2014_001 (ch. 25.0%)	EEGNet BatchNorm	$\mathbf{61.4 \pm 1.5}$
	EEGNet GroupNorm	54.8 ± 3.6
Cho2017 (ch. 50.0%)	EEGNet BatchNorm	$\mathbf{67.7 \pm 1.4}$
	EEGNet GroupNorm	67.5 ± 1.4
Weibo2014 (ch. 14.3%)	EEGNet BatchNorm	$\mathbf{50.2 \pm 1.8}$
	EEGNet GroupNorm	48.0 ± 1.9

Variant (a) leads eleven of the eighteen private cells against six, with one tie, but the mean gap is under 0.1 points and almost every individual gap falls inside one standard deviation. We report (a) throughout Section 6.4, as the variant that spends no privacy budget on the normalization layer.

Variant (c) scores below both at almost every budget, but it is already the weaker layer before privacy enters, trailing (a) both centrally (Table 12) and federated without privacy. That gap narrows in all eighteen private cells, so public calibration appears to buy EEGNet little under privacy beyond an architectural difference. SPDNet leads all three variants at every budget and both participation levels. Variant (c) inherits the hyperparameters swept for (a) rather than receiving its own sweep, so a separately tuned arm could only improve it.

C.6 Decomposing the cost of privacy

The main text reports the cost of the privacy pipeline as a whole, which is what a practitioner pays. That total mixes two effects. Privacy adds noise, but it also imposes requirements that have nothing to do with noise: minibatches must be drawn uniformly without replacement, gradients must be clipped, and EEGNet's normalization is frozen (Appendix C.5). Each of these costs accuracy on its own.

To separate them we run each arm at $\sigma_{\mathrm{dp}} = 0$: the complete mechanism, with the perturbation switched off. The gap from the non-private run to this anchor is the cost of DP-compatibility, the gap from the anchor to a private run the cost of the noise. Figure 8 reports both at $\varepsilon = 20$, each arm measured against its own anchor, since at $\sigma_{\mathrm{dp}} = 0$ the aggregation rules remain three different algorithms and a normalization variant changes its own ceiling.

Table 13: Federated test accuracy (%) for three DP-compatible normalization strategies for EEGNet, at full and half client participation: mean ± standard deviation over 10 seeds, best arm per privacy budget in bold, ties marked jointly.

		Non-private	$\varepsilon = 20$	$\varepsilon = 10$	$\varepsilon = 5$
BNCI2014_001	BN (a) – full	**57.0 ± 2.0**	35.7 ± 2.4	**33.8 ± 2.6**	**30.3 ± 2.5**
	BN (b) – full		**36.5 ± 3.0**	33.5 ± 3.1	29.9 ± 2.6
	GN (c) – full	52.9 ± 1.9	32.6 ± 2.8	30.1 ± 2.4	28.0 ± 2.3
	BN (a) – half	**56.6 ± 1.7**	33.0 ± 2.9	30.6 ± 2.8	28.5 ± 2.1
	BN (b) – half		**35.4 ± 2.1**	**32.0 ± 2.0**	**28.7 ± 1.9**
	GN (c) – half	51.8 ± 1.7	30.5 ± 2.9	27.9 ± 2.2	26.4 ± 1.1
Cho2017	BN (a) – full	**67.7 ± 0.6**	52.2 ± 1.3	**51.5 ± 1.1**	**51.0 ± 0.8**
	BN (b) – full		**52.4 ± 1.4**	**51.5 ± 1.3**	50.6 ± 1.1
	GN (c) – full	66.2 ± 0.8	51.8 ± 1.3	51.1 ± 0.9	50.6 ± 0.9
	BN (a) – half	**67.3 ± 0.8**	**52.1 ± 1.1**	**51.3 ± 1.4**	**50.7 ± 1.0**
	BN (b) – half		51.6 ± 0.9	51.2 ± 1.1	50.4 ± 0.7
	GN (c) – half	65.7 ± 0.4	51.7 ± 0.8	50.9 ± 0.8	50.5 ± 0.5
Weibo2014	BN (a) – full	**47.0 ± 1.9**	**17.4 ± 0.9**	15.9 ± 1.0	**15.4 ± 1.3**
	BN (b) – full		16.9 ± 1.8	**16.0 ± 1.7**	15.1 ± 1.1
	GN (c) – full	42.3 ± 1.5	16.7 ± 1.2	15.8 ± 1.3	15.2 ± 1.3
	BN (a) – half	**45.9 ± 1.9**	**16.9 ± 1.3**	**15.8 ± 1.1**	**15.1 ± 1.0**
	BN (b) – half		15.9 ± 1.9	14.9 ± 0.8	14.4 ± 0.8
	GN (c) – half	39.7 ± 1.3	16.0 ± 1.2	15.6 ± 1.0	**15.1 ± 1.1**

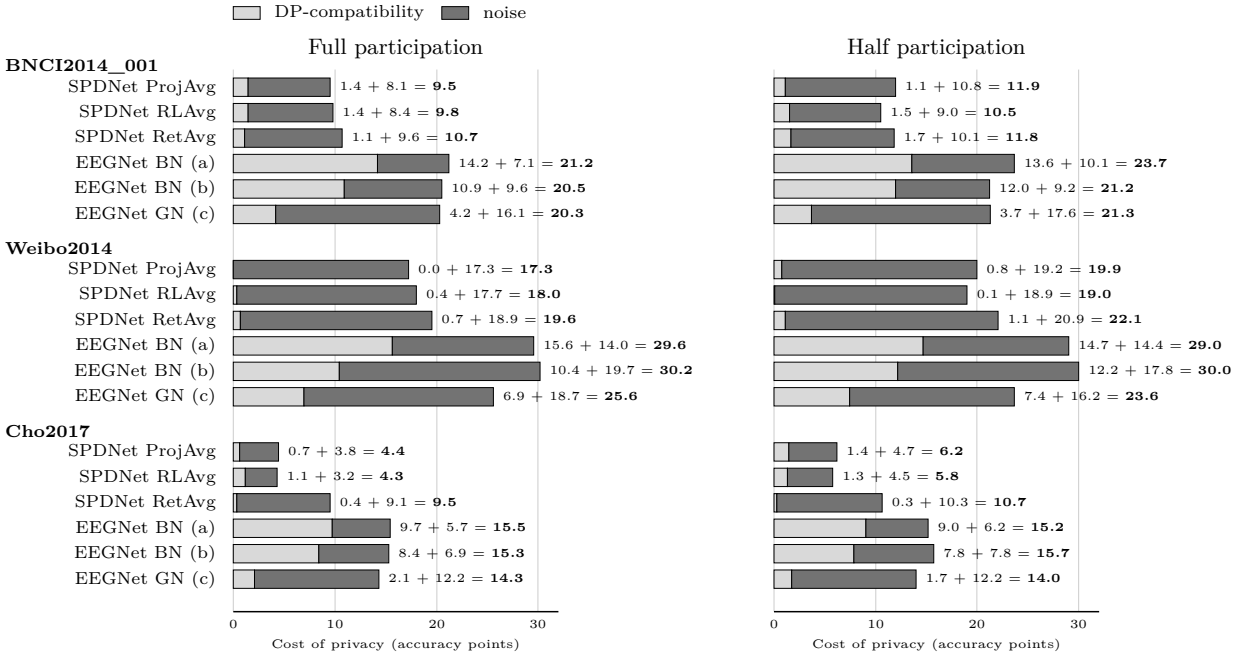


Figure 8: Cost of privacy at $\varepsilon = 20$ in accuracy points over 10 seeds, split into DP-compatibility and noise, with the total in bold.