RIPPLE in Still Water: Zero-Shot Clustering in Federated Learning with Wavelet Scattering Transform

Alessandro Licciardi
Department of Mathematical Sciences
Politecnico di Torino, Turin, Italy
alessandro.licciardi@polito.it

Abstract

Clustered Federated Learning (FL) partitions a client population into groups of similar local distributions and trains one specialized model per cluster, mitigating client drift that degrades single-model methods under non-IID data. Prior methods discover cluster structure inside the training loop through gradient similarity, loss evaluation, or EM-style updates, thus increasing communication overhead, exposing gradients to inversion attacks, and providing no mechanism to assign clients absent from training. We propose RIPPLE, a clustered FL framework in which cluster assignment is computed entirely offline from a spectral characterization of each client's local data: a variance-weighted principal-component prototype embedded via the Wavelet Scattering Transform and decoded by a Gaussian Mixture VAE trained server-side on synthetic client populations before federation begins. Per-round communication cost matches FEDAVG exactly, and a client absent from training obtains a personalized model from a single forward pass, without gradient computation, model evaluation, or extra communication round. We prove that the gap between RIPPLE's surrogate clustered objective and the oracle is bounded by a computable quantity decaying with client sample size and independent of federation duration; per-cluster convergence matches the minimax-optimal rate for non-convex smooth objectives. Across five benchmarks spanning controlled and realistic heterogeneity, RIPPLE consistently outperforms all baselines, with margins growing on the most realistic partitions. Code is publicly available at https://github.com/alelicciardi99/ripple_clustered_FL

1 Introduction

Ripple in still water,
when there is no pebble tossed,
nor wind to blow.
— R. Hunter & J. Garcia (1970)

The latent clustered structure of a federation need not be stirred into being by training, but already ripples through the spectral geometry of its data.

Federated Learning (FL) (McMahan et al., 2017; Kairouz et al., 2021) trains shared models without pooling private data, but its canonical algorithm FEDAVG (McMahan et al., 2017) silently assumes that all clients sample from the same distribution. Under the statistical heterogeneity that is ubiquitous in real deployments (Zhao et al., 2018; Hsieh et al., 2020), gradient updates from clients with structurally distinct data point in conflicting directions, which Karimireddy et al. (2020) called *client drift*,degrading every client's accuracy at once. Variance-reduction and adaptive methods (Karimireddy et al., 2020; Acar et al., 2021; Reddi et al., 2020; Li et al., 2020) attenuate the symptom

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

but do not address the cause: a single global model cannot represent clients whose data lie in qualitatively different regimes.

Clustered FL addresses the cause directly, training one specialized model per group of compatible clients (Sattler et al., 2020; Ghosh et al., 2020; Long et al., 2023; Fenoglio et al., 2025). Yet every prior method discovers cluster structure *inside* the training loop, e.g., through gradient cosine similarity (Sattler et al., 2020), repeated loss evaluation (Ghosh et al., 2020), or EM-style gradient statistics (Long et al., 2023). These methods accumulate communication overhead with each round, exposing updates to gradient-inversion attacks (Zhang et al., 2022), and offering no pathway to assign a client that was not part of training. Moreover, while prior methods such as IFCA and FESEM require the number of clusters to be specified a priori, RIPPLE needs only a loose upper bound, and RIPPLE-HDB then automatically adapts the cluster count via density-based clustering on the posterior simplex.

We propose a different route, motivated by recent work showing that the Wavelet Scattering Transform (WST) (Mallat, 2012; Bruna and Mallat, 2013) can identify anomalous clients in a federation from spectral signatures of their local data (Licciardi et al., 2026; Licciardi, 2026; Deriu and Licciardi, 2026), with no federated communication whatsoever. The same spectral characterization extends naturally to distinguishing *semantically distinct* client populations: framing the problem as probabilistic inference over a simplex-valued mixture model yields a soft cluster-weight vector for each client from a single forward pass, computed from local data statistics alone. This is the central design principle of RIPPLE: cluster assignment is a property of the data, not an artifact of training. The assignment is fixed before the first communication round and never revised; per-round cost matches FEDAVG exactly; and a client absent from training receives a personalized model by transmitting nothing but a small spectral embedding of its own data.

Contributions. We summarize the contributions as follows.

- We reframe clustered FL routing as an offline statistic computed from local data, eliminating per-round communication overhead and fixing the routing decision before the first federation round.
- We prove a surrogate-fidelity certificate: the gap between RIPPLE's clustered objective and the oracle is bounded by a computable quantity that decays with client sample size and is independent of the number of rounds.
- A client absent from training obtains a personalized model from a single WST embedding and one encoder forward pass, without any gradient computation or model evaluation required.
- RIPPLE outperforms all baselines across five benchmarks, with the largest margins on the most realistic partitions (Venkateswara et al., 2017; Weyand et al., 2020); zero-shot assignment matches or exceeds other clustered FL methods.

2 Related Work

FL under data heterogeneity. Non-IID client data has been central to FL research since Zhao et al. (2018) quantified how label skew degrades FEDAVG (McMahan et al., 2017) convergence. Variance-reduction methods address client drift directly: SCAFFOLD (Karimireddy et al., 2020) corrects local-SGD bias via control variates, while FEDDYN (Acar et al., 2021) adds a dynamic regularization term penalizing divergence from the global model. Adaptive-optimizer variants (Reddi et al., 2020) improve convergence under heterogeneous gradients; FedProx (Li et al., 2020) constrains local updates via a proximal term. All these methods pursue a *single* global model, which cannot fully resolve clients whose data lie in fundamentally distinct semantic regimes (Hsieh et al., 2020). Personalized FL sidesteps this by training per-client models via meta-learning (PER-FEDAVG (Fallah et al., 2020), FedALA (Zhang et al., 2023)), Moreau-envelope regularization (PFEDME (T Dinh et al., 2020)), or shared representations (Collins et al., 2021); yet when clients share significant within-group structure, per-client models discard exploitable inter-client information that clustered training would leverage.

WST in FL. Scattering descriptors of client data have been used to characterize clients offline in FL, for the detection of malicious or anomalous clients from a one-shot spectral summary (Licciardi et al., 2026; Licciardi, 2026), where the WST representation proved more reliable than its Fourier counterpart under the deformations typical of client data, and the same construction has been deployed

in a domain setting with unsupervised scattering autoencoders (Deriu and Licciardi, 2026). RIPPLE carries this offline characterization from detection to cluster assignment.

Clustered FL. Clustering has proven to be an effective strategy in FL for handling client heterogeneity and improving personalization (Huang et al., 2022; Duan et al., 2021; Briggs et al., 2020; Caldarola et al., 2021; Ye et al., 2023). Sattler et al. (2020) group clients by cosine gradient similarity, but the splitting threshold is sensitive and prone to over- or under-fragmentation under strong non-IID. IFCA (Ghosh et al., 2020) alternates hard cluster assignment with gradient descent, requiring each client to evaluate all C models per round. FESEM (Long et al., 2023) uses EM-style soft assignments but still iterates gradient updates from every candidate center. FedEM (Marfoq et al., 2021) models each client as a mixture of shared components with soft weights learned by EM during federation, so that weights and models co-evolve, every client evaluates all components at each round, and a client absent from training obtains no weights. Graph and hierarchy based methods (Ye et al., 2023; Huang et al., 2022; Briggs et al., 2020; Duan et al., 2021) discover collaboration structure adaptively yet tie cluster inference to in-training gradient statistics. FedGWC (Licciardi et al., 2025) and FLUX (Fenoglio et al., 2025) refine aggregation with interaction-aware weighting and data descriptors, respectively, the first by modeling interaction between clients' empirical losses, while the latter infer cluster structure from gradient feedback exchanged *during* training. PACFL (Vahidian et al., 2023) assigns clients before training, from the principal angles between the subspaces spanned by the leading singular vectors of their data, but the assignment is a hard partition of the observed population, a newcomer requires re-clustering, the transmitted singular vectors expose the local data subspace, and no bound relates the partition to the data geometry, whereas RIPPLE maps each client independently to soft weights through a descriptor with quantified stability (Section 5).

Prior methods share three structural limitations. *(i) Hard or in-training assignment*: membership is either binary (Sattler et al., 2020; Ghosh et al., 2020) or requires ongoing gradient communication (Long et al., 2023; Licciardi et al., 2025; Fenoglio et al., 2025), coupling cluster discovery to the training loop. *(ii) No zero-shot capability*: a new client must participate in at least one communication round or evaluate all C models locally to obtain an assignment. *(iii) Communication overhead above* FEDAVG: discovering cluster structure consumes extra bandwidth every round. RIPPLE overcomes these limitations, as cluster weights $\mathbf{w}_k \in \Delta^{C-1}$ are inferred *once*, before any federated round, from a frozen GMM-VAE requiring no gradient exchange; per-round communication matches FEDAVG exactly; and a new client is assigned via a single forward pass of the frozen encoder.

3 Background

In cross-silo FL, a central server coordinates K clients, indexed $k \in [K] := \{1, \dots, K\}$. Each client k holds a private local dataset $\mathcal{D}_k = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n_k}$ of n_k labeled samples drawn i.i.d. from a local distribution $\mathbb{P}_k$ on $\mathbb{R}^D \times \mathcal{Y}$, never shared directly (Kairouz et al., 2021; Bonawitz et al., 2019). The pre-federation pipeline operates on features $\mathbf{x}_i$ alone and requires no labels, which enter only through the loss $\ell(\theta; \mathbf{x}_i, y_i)$ during federated training. Let $p_k = n_k / \sum_{j=1}^{K} n_j$ denote the normalized data weight of client k. The canonical algorithm, FEDAVG (McMahan et al., 2017), minimizes a weighted empirical risk over a single global parameter vector $\theta \in \mathbb{R}^{|\theta|}$:

$$\min_{\theta} \; \mathcal{L}(\theta) \; := \; \sum_{k=1}^{K} p_k \, F_k(\theta), \qquad F_k(\theta) \; = \; \frac{1}{n_k} \sum_{i=1}^{n_k} \ell(\theta; \mathbf{x}_i, y_i), \tag{1}$$

where $\ell : \mathbb{R}^{|\theta|} \times \mathbb{R}^D \times \mathcal{Y} \to \mathbb{R}$ is a differentiable loss. Each round, the server broadcasts the current model to a sampled client subset; each client runs E steps of local SGD from that initialization, and the server aggregates updates via a data-weighted average. A full probabilistic derivation of this objective and its clustered extension is given in Appendix A.1.

3.1 A Probabilistic Model of Client Heterogeneity

We depart from the binary cluster-membership assumption and model each client's data distribution as a *convex mixture* of C underlying semantic distributions.

Definition 3.1 Client Mixture Distribution

Let $\{\mathbb{P}_1, \ldots, \mathbb{P}_C\}$ be C semantic distributions on $\mathbb{R}^D \times \mathcal{Y}$. Each client $k \in [K]$ observes its local data i.i.d. from a latent mixture distribution:

$$\mathbb{P}_k^* = \sum_{c=1}^{C} \pi_{k,c}^* \mathbb{P}_c, \tag{2}$$

where the *true cluster-weight vector* $\boldsymbol{\pi}_k^* = (\pi_{k,1}^*, \ldots, \pi_{k,C}^*)^\top \in \Delta^{C-1} = \{\boldsymbol{\pi} \in \mathbb{R}_{\geq 0}^C : \sum_c \pi_c = 1\}$.

If $\{\boldsymbol{\pi}_k^*\}_{k=1}^K$ were known, the theoretically optimal strategy minimizes the *oracle clustered objective*:

$$\mathcal{L}_{\text{true}}(\Theta) = \sum_{k=1}^{K} p_k \sum_{c=1}^{C} \pi_{k,c}^* F_k(\theta_c), \tag{3}$$

over cluster models $\Theta = \{\theta_1, \ldots, \theta_C\}$.

Remark 3.1 (Generality of the mixture model)**.** *The oracle objective* (3) *subsumes classical FL as special cases:* FEDAVG *is recovered when* $C = 1$*; fully personalized FL corresponds to* $C = K$ *with* $\pi_{k,k}^* = 1$*. The regime* $1 < C < K$ *with genuinely fractional* $\boldsymbol{\pi}_k^*$ *strictly generalizes prior clustered FL (Ghosh et al., 2020; Sattler et al., 2020; Long et al., 2023; Licciardi et al., 2025), all of which restrict* $\boldsymbol{\pi}_k^* \in \{0, 1\}^C$.

The cluster assignment problem. In practice, $\boldsymbol{\pi}_k^*$ is entirely latent. Inferring client similarities via gradient or parameter exchange incurs $O(K^2|\theta|)$ communication overhead and creates a privacy surface, since gradient updates can leak local data information (Bonawitz et al., 2017; Kairouz et al., 2021). Our goal is instead to estimate a *surrogate weight vector* $\mathbf{w}_k \approx \boldsymbol{\pi}_k^*$ from each client's *local, pre-federation statistics alone*, and substitute it into the training objective:

$$\mathcal{L}_{\text{surrogate}}(\Theta) = \sum_{k=1}^{K} p_k \sum_{c=1}^{C} w_{k,c} F_k(\theta_c). \tag{4}$$

As we show in Theorem 5.2, controlling the ℓ_1 discrepancy $\|\mathbf{w}_k - \boldsymbol{\pi}_k^*\|_1$ uniformly over all clients is both necessary and sufficient to bound the gap between $\mathcal{L}_{\text{surrogate}}$ and $\mathcal{L}_{\text{true}}$.

Remark 3.2 (Zero-shot cluster assignment)**.** *We use* cluster assignment *to refer to the map* $k \mapsto \mathbf{w}_k \in \Delta^{C-1}$*, generalizing hard partitioning to soft probabilistic allocations. When computed before any federation round using only local pre-federation statistics, without any gradient exchange or cluster-model evaluations. We call this procedure* zero-shot cluster assignment*, borrowing the convention from Bengio et al. (2017).*

3.2 Wavelet Scattering Transform

The Wavelet Scattering Transform (WST) (Mallat, 2012; Bruna and Mallat, 2013) builds a parameter-free signal representation by cascading wavelet convolutions with pointwise modulus nonlinearities, followed by lowpass averaging. The resulting second-order representation $\boldsymbol{\Phi}(\mathbf{x}) \in \mathbb{R}^P$ satisfies three properties that underpin our theoretical guarantees: it is *non-expansive* ($\|S_J\mathbf{x} - S_J\mathbf{x}'\|_2 \leq \|\mathbf{x} - \mathbf{x}'\|_2$), *translation-invariant* at scale 2^J, and *Lipschitz-stable to diffeomorphisms* with constant $C_2 = 2\|\nabla\psi\|_1$. Together these properties ensure that two clients whose distributions differ only through sensor miscalibrations, object translations, or mild domain shifts produce representations $\boldsymbol{\Phi}_k$ at a distance proportional to the magnitude of those perturbations, thus making $\boldsymbol{\Phi}$ a stable, label-free fingerprint for pre-federation cluster assignment. The full construction, formal proofs, and a sensitivity study are given in Appendix A.2.

We refer to Appendix A for a self-contained probabilistic derivation of the federated and clustered objectives from first principles, and for the full WST construction and stability properties that underpin Theorems 5.1–5.2.

4 The RIPPLE Algorithm

RIPPLE solves the cluster assignment problem zero-shot: before any federation round, each client transmits a single deformation-stable summary of its local distribution, and the server maps it to a soft cluster-weight vector $\mathbf{w}_k \in \Delta^{C-1}$. This supports clients with genuinely mixed data modalities and generalizes to unseen clients via a single encoder forward pass, following one architectural choice: a Gaussian Mixture VAE trained server-side on synthetic client populations before federation begins.

4.1 Pre-Federation Distribution Characterization

Spectral prototype. Let $\hat{\boldsymbol{\mu}}_k$ and $\hat{\boldsymbol{\Sigma}}_k$ be the empirical mean and covariance of each client's data distribution, with eigendecomposition $\hat{\boldsymbol{\Sigma}}_k = \sum_j \hat{\lambda}_j \hat{\mathbf{v}}_j \hat{\mathbf{v}}_j^\top$. We summarize each client by the variance-weighted principal-component prototype:

$$\hat{\mathbf{x}}_k \;\coloneqq\; \hat{\boldsymbol{\mu}}_k \;+\; \sum_{j=1}^{M} \frac{\hat{\lambda}_j}{\sum_{l=1}^{M} \hat{\lambda}_l}\, \hat{\mathbf{v}}_j, \tag{5}$$

where M is selected by the explained-variance criterion $M^\star = \min\{M : \sum_{j\le M} \hat{\lambda}_j / \sum_{j\le D} \hat{\lambda}_j \ge \rho\}$, where $\rho \in [0.90, 0.99]$ is the explained variance threshold.

Wavelet Scattering embedding. The spectral prototype remains sensitive to domain shifts endemic to non-IID federated data: translations, sensor miscalibrations, or affine skews can produce very different prototypes for semantically identical distributions. We resolve this by mapping $\hat{\mathbf{x}}_k$ through the Wavelet Scattering Transform (Appendix A.2):

$$\boldsymbol{\Phi}_k \;\coloneqq\; S_J(\hat{\mathbf{x}}_k) \;\in\; \mathbb{R}^P. \tag{6}$$

WST stability properties guarantee that $\|\boldsymbol{\Phi}_k - \boldsymbol{\Phi}(\mathbf{x}_k^*)\|_2$ decomposes additively into a sample-noise term and a domain-shift term, both made precise in Theorem 5.1. We use $J = 3$ scales and $L = 8$ orientations per octave (Kymatio (Andreux et al., 2020)). Each client transmits only $\boldsymbol{\Phi}_k$ to the server, once, before any federation round.

Cluster-weight inference. Given $\boldsymbol{\Phi}_k$, a GMM-VAE maps it to Δ^{C-1}: its mixture prior induces a structured latent space whose categorical posterior directly parameterizes the simplex (Wang et al., 2024), with softmax classifier $q_\phi(c \mid \boldsymbol{\Phi}) = \mathrm{softmax}(f_\phi(\boldsymbol{\Phi}))$ and cluster-conditional Gaussian encoder $q_\phi(\mathbf{z} \mid \boldsymbol{\Phi}, c) = \mathcal{N}(\boldsymbol{\mu}_\phi, \mathrm{diag}(\boldsymbol{\sigma}_\phi^2))$. The cluster-weight estimate is:

$$\mathbf{w}_k \;\coloneqq\; q_\phi(\,\cdot \mid \boldsymbol{\Phi}_k) \;\in\; \Delta^{C-1}. \tag{7}$$

Training minimizes the GMM-VAE ELBO augmented with a supervised alignment term that prevents posterior collapse (Appendix A):

$$\mathcal{L}_{\text{total}} \;=\; \mathcal{L}_{\text{ELBO}} \;-\; \lambda_{\text{sup}}\, D_{\text{KL}}(q_\phi(c \mid \boldsymbol{\Phi}) \,\|\, \mathbf{y}_{\text{target}})\,. \tag{8}$$

We denote by $\phi_\star$ the minimizer of (8) over the encoder class, i.e., the parameters returned by the server-side training described below in the infinite-sample limit of the synthetic population. At inference, $\mathbf{w}_k$ is computed in a single forward pass through the frozen $q_{\phi_\star}$.

Server-Side VAE Offline Training The GMM-VAE is trained before federation on synthetic clients generated from a small auxiliary dataset $\mathcal{D}_{\text{aux}}$ (Sattler et al., 2021; Lin et al., 2020). At each step, we sample $\alpha^{(t)} \sim \mathrm{Uniform}(\alpha_{\min}, \alpha_{\max})$, draw a target mixture $\mathbf{y}^{(t)} \sim \mathrm{Dirichlet}(\alpha^{(t)} \cdot \mathbf{1}_C)$, and subsample $\mathcal{D}_{\text{syn}}^{(t)} \subset \mathcal{D}_{\text{aux}}$ with matching class proportions. By Proposition D.1, the sampled targets $\mathbf{y}^{(t)}$ cover the full open simplex $\mathrm{int}(\Delta^{C-1})$, therefore the encoder learns the conditional distribution of cluster membership given the descriptor for every admissible mixture. Clustering the received $\boldsymbol{\Phi}_k$ directly only partitions the observed population, and Appendix G.5 compares the two. The full procedure is in Algorithm 1 (Appendix C).

4.2 Federated Posterior Mixing

At the start of federation, the server computes $\mathbf{w}_k = q_\phi(\cdot|\boldsymbol{\Phi}_k) \in \Delta^{C-1}$ once for each client, before the first round.

RIPPLE-STRAT and RIPPLE-UNIF. At round t, client k receives $\theta_k^{(\text{init})} = \sum_c w_{k,c}\,\theta_c^{(t)}$ and returns $\Delta\theta_k$ after E local SGD steps. The server updates each cluster model via normalized weighted aggregation:

$$\theta_c^{(t+1)} \;\leftarrow\; \theta_c^{(t)} + \eta_s \cdot \frac{\sum_{k\in\mathcal{S}_t} w_{k,c}\,\Delta\theta_k}{\sum_{k\in\mathcal{S}_t} w_{k,c} + \epsilon}, \tag{9}$$

where $\epsilon = 10^{-8}$. RIPPLE-STRAT guarantees every cluster receives at least one gradient per round via stratified sampling; RIPPLE-UNIF samples uniformly. Per-round communication matches FEDAVG exactly since $\mathbf{\Phi}_k$ is transmitted only once and $P \ll |\theta|$.

Topological Refinement RIPPLE-HDB. When C is mis-specified, RIPPLE-HDB applies HDBSCAN (Campello et al., 2013) to the pairwise distance matrix with entries

$$D_{ij} \;=\; \sqrt{\frac{1}{2}D_{\mathrm{KL}}\Big(\mathbf{w}_i \,\Big\|\, \tfrac{\mathbf{w}_i+\mathbf{w}_j}{2}\Big) + \frac{1}{2}D_{\mathrm{KL}}\Big(\mathbf{w}_j \,\Big\|\, \tfrac{\mathbf{w}_i+\mathbf{w}_j}{2}\Big)}, \tag{10}$$

recovering $\hat{C} \le C$ meta-clusters and collapsing weights via

$$\hat{w}_{k,m} \;=\; \frac{\sum_{j\in\mathcal{C}_m} w_{k,j}}{\sum_{m'=1}^{\hat{C}}\sum_{j\in\mathcal{C}_{m'}} w_{k,j}}. \tag{11}$$

Noise points are assigned to their nearest meta-cluster. This provides automatic model selection without additional hyperparameter search; the square-root Jensen–Shannon divergence is a valid metric on Δ^{C-1} by Proposition D.2.

5 Theoretical Analysis

We establish two guarantees for RIPPLE. The first controls the *routing error* $\|\mathbf{w}_k - \boldsymbol{\pi}_k^*\|_1$ — how faithfully the pre-federation pipeline recovers each client's true cluster weights from finite, domain-shifted samples. The second shows that soft-weighted aggregation converges at the minimax-optimal rate for non-convex SGD, with variance controlled by the effective participation mass. Together they yield a computable end-to-end excess-risk bound (Corollary 5.1). All proofs are in Appendix D.

Assumptions. Let $\mathbf{x}_k^*$ denote the population spectral prototype of client k, $\mathbf{\Phi}_k^* = S_J(\mathbf{x}_k^*)$, and $\phi_\star$ the minimizer of (8) introduced in Section 4.

Assumption 5.1 (Data generation)**.** *Each client observes n_k i.i.d. sub-Gaussian samples (σ^2 proxy), subject to a fixed translation $|v| \le v_{\max}$ and a C^2 diffeomorphism $\|\nabla\tau\|_\infty \le \epsilon_{\mathrm{dom}}$. The true covariance $\mathbf{\Sigma}_k^*$ has eigengap $\Delta\lambda := \lambda_M(\mathbf{\Sigma}_k^*) - \lambda_{M+1}(\mathbf{\Sigma}_k^*) > 0$.*

Assumption 5.2 (Loss regularity)**.** *Each F_k is bounded by $M_F < \infty$ and each per-cluster objective $\mathcal{L}_c$ is L-smooth.*

Assumption 5.3 (VAE realizability and stability)**.** *The frozen encoder satisfies* (i) $q_{\phi_\star}(\,\cdot \mid \mathbf{\Phi}_k^*) = \boldsymbol{\pi}_k^*$ *for all k, and* (ii) $\|q_{\phi_\star}(\,\cdot \mid \mathbf{\Phi}) - q_{\phi_\star}(\,\cdot \mid \mathbf{\Phi}')\|_1 \le 2L_\phi\|\mathbf{\Phi} - \mathbf{\Phi}'\|_2$ *for all $\mathbf{\Phi}, \mathbf{\Phi}'$.*

The two parts have a different status. Part (i) is a realizability assumption which states that the optimal encoder recovers the true mixture weights on population descriptors. Part (ii) is a property of the architecture rather than of the data, since spectral normalization (Miyato et al., 2018) bounds the spectral norm of each encoder layer and the composed encoder is Lipschitz with a known constant L_ϕ. The guarantees below are therefore conditional on (i) only, and Corollary D.2 characterizes the limit in which it holds exactly.

5.1 Routing Error and Embedding Stability

The routing error $\|\mathbf{w}_k - \boldsymbol{\pi}_k^*\|_1$ has two sources: finite-sample noise in the spectral prototype and domain shift from translation and diffeomorphism. Theorem 5.1 bounds their combined effect by composing the Davis–Kahan spectral perturbation bound with the WST stability properties **(P1)–(P3)** (Appendix A.2).

Theorem 5.1 Embedding stability

Under Assumption 5.1, with probability at least $1-\delta$,

$$\|\mathbf{\Phi}_k - \mathbf{\Phi}_k^*\|_2 \;\le\; \underbrace{\frac{C\sigma^2}{\Delta\lambda}\sqrt{\frac{D+\log(1/\delta)}{n_k}}}_{\epsilon_{\text{sample}}(\delta)} + \underbrace{\Big(C_2\,\epsilon_{\text{dom}} + C_1\,\tfrac{v_{\max}}{2^J}\Big)\|\mathbf{x}_k^*\|_2}_{\epsilon_{\text{shift}}} =:\; \epsilon_{\text{embed}}. \quad (12)$$

The two terms admit a clean interpretation. ϵ_{sample} decays at $O(n_k^{-1/2})$, scaled by $1/\Delta\lambda$: clients with well-separated covariance structure — the typical non-IID regime — are routed more accurately. ϵ_{shift} is a *model-independent floor*: it is a property of the data geometry alone, and quantifies precisely when pre-federation assignment becomes unreliable regardless of the routing model used. The translation cost $C_1 v_{\max}/2^J$ decreases with WST resolution J, giving a direct design knob.

Theorem 5.2 Surrogate-objective fidelity

Under Assumptions 5.1–5.3, for all Θ,

$$\big|\mathcal{L}_{\text{surrogate}}(\Theta) - \mathcal{L}_{\text{true}}(\Theta)\big| \;\le\; 2\,M_F\,L_\phi\,\epsilon_{\text{embed}}. \quad (13)$$

This is the first objective-level guarantee that a pre-federation routing step faithfully approximates the oracle clustered objective (3). Prior CFL methods (Ghosh et al., 2020; Sattler et al., 2020; Long et al., 2023) prove convergence *given* correct assignments but offer no bound on the assignment error itself. Here the two are unified: the right-hand side is computable before federation begins, giving a concrete pre-training quality certificate.

5.2 Federated Convergence

Given fixed weights $\{\mathbf{w}_k\}$, the per-cluster aggregation rule defines a stochastic gradient step on $\mathcal{L}_c$. The key CFL-specific quantity is the *effective participation mass* $S_{\text{eff}}^{(c)} := \sum_{k\in\mathcal{S}_t} w_{k,c}$, which acts as an adaptive sample-size proxy. Unlike hard-assignment CFL where $S_{\text{eff}}^{(c)}$ simply counts assigned clients, soft weights allow clients with diffuse posteriors to contribute fractionally to every cluster — preventing the gradient starvation that hard-assignment methods suffer when a cluster is underrepresented in $\mathcal{S}_t$.

Theorem 5.3 Per-cluster convergence

Under Assumption 5.2, with step size $\eta \le 1/L$ and unbiased per-cluster gradient with variance $\mathbb{E}\|\mathbf{g}_c^{(t)} - \nabla\mathcal{L}_c(\theta_c^{(t)})\|^2 \le \sigma_c^2/S_{\text{eff}}^{(c)}$,

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\big\|\nabla\mathcal{L}_c\big(\theta_c^{(t)}\big)\big\|^2 \;\le\; \frac{2\big(\mathcal{L}_c(\theta_c^{(0)}) - \mathcal{L}_c^*\big)}{\eta T} \;+\; \frac{\eta L\,\sigma_c^2}{S_{\text{eff}}^{(c)}}. \quad (14)$$

With $\eta = O(1/\sqrt{T})$ this decays at the minimax-optimal rate $O(1/\sqrt{T})$ for non-convex L-smooth objectives.

Soft assignment strictly dominates hard assignment in variance: a client with $\mathbf{w}_k = (0.6, 0.4)$ contributes fractionally to both clusters, increasing $S_{\text{eff}}^{(c)}$ for the minority cluster and reducing its gradient variance. This also justifies the stratified sampling strategy of RIPPLE-STRAT: by guaranteeing each cluster receives at least one gradient signal per round, $S_{\text{eff}}^{(c)} > 0$ is maintained and the variance term remains finite.

Corollary 5.1 End-to-end excess risk

Let $\hat{\Theta}^{(T)}$ be returned by Algorithm 1 after T rounds and $\Theta^* = \arg\min_\Theta \mathcal{L}_{\text{true}}(\Theta)$. Under Assumptions 5.1–5.3,

$$\mathbb{E}\big[\mathcal{L}_{\text{surrogate}}\big(\hat{\Theta}^{(T)}\big)\big] - \mathcal{L}_{\text{true}}(\Theta^*) \;\le\; \underbrace{2\,M_F\,L_\phi\,\epsilon_{\text{embed}}}_{\text{routing}} + \underbrace{\sum_{c=1}^{C} p_c \left(\frac{2\big(\mathcal{L}_c(\theta_c^{(0)}) - \mathcal{L}_c^*\big)}{\eta T} + \frac{\eta L\,\sigma_c^2}{S_{\text{eff}}^{(c)}} \right)}_{\text{optimization}} \tag{15}$$

where $p_c = \sum_k p_k \pi_{k,c}^*$ is the marginal cluster weight.

The bound reveals a clean separation of concerns. The routing term is frozen at $t = 0$ by data geometry and independent of T: it decreases with more local data, larger spectral gaps, finer WST resolution, and smaller domain shift. The optimization term decays at $O(1/\sqrt{T})$ with variance suppressed by $S_{\text{eff}}^{(c)}$. Together, Corollary 5.1 shows that RIPPLE converges to a neighborhood of $\mathcal{L}_{\text{true}}(\Theta^*)$ whose radius is set before a single gradient is exchanged, which is a guarantee that no prior CFL method provides. In the exact-recovery limit, $\epsilon_{\text{embed}} = 0$ with one-hot true mixtures, the routing term vanishes, $S_{\text{eff}}^{(c)}$ counts the sampled members of cluster c, and the update reduces to per-cluster FEDAVG at the classical non-convex rate (Corollary D.2, Appendix D.9).

When pre-federation routing fails. The failure modes are the two terms of Theorem 5.1. Routing degrades when the eigengap $\Delta\lambda$ is small relatively to the sampling noise, so that client covariances are not well separated and ϵ_{sample} inflates, or when the domain shift exceeds what the WST stability absorbs, which raises the floor ϵ_{shift} that no routing model can lower. A third failure is a violation of Assumption 5.3(i), when the auxiliary set does not represent part of the client population and the encoder is confidently wrong rather than diffuse. All three are visible to the server before any round, through the separability of the received descriptors, measured by the eigengap or a silhouette score on $\{\mathbf{\Phi}_k\}$, and through the normalized posterior entropy $H(\mathbf{w}_k)/\log C$, which does not require any labels. Appendix G.5 exhibits the three regimes empirically and shows that the entropy flags the one condition that affects accuracy.

6 Experiments

Implementation Details. We evaluate on five benchmarks spanning controlled and realistic heterogeneity. MNIST (LeCun, 1998), CIFAR-10, and CIFAR-100 (Krizhevsky et al., 2009) use a symmetric Dirichlet partition with concentration α over class labels (Hsu et al., 2020), where lower α induces sharper class imbalance and stronger client drift. For CIFAR-10 and CIFAR-100 the concentration acts on superclass labels, so that each client concentrates on semantically coherent groups of classes and the induced heterogeneity is feature-wise rather than purely label-wise. Office-Home (Venkateswara et al., 2017) ($K = 50$, 65 classes, four visual domains) and GLDv2-23k (Weyand et al., 2020) ($K = 233$, 203 landmark classes, natural photographer partition) reflect real-world covariate shift and long-tail distributions. MNIST and CIFAR-10 use a lightweight CNN (LeCun et al., 1998); CIFAR-100 a ResNet-9; Office-Home and GLDv2 an ImageNet-pretrained ResNet-18 (Deng et al., 2009). Full dataset statistics and architectures are in Section G.1. All methods run for 1000 rounds with $E = 1$ local epoch, $\eta_c = 0.01$ (SGD, momentum 0.9), $\eta_s = 1.0$, and $S = 10$ clients per round. Results are means over seeds $\{42, 7, 123\}$, and we report $\pm 2\sigma$ confidence intervals.

Baselines. We compare against 11 methods: *global* — FEDAVG (McMahan et al., 2017), FEDDYN (Acar et al., 2021), SCAFFOLD (Karimireddy et al., 2020), FEDADAM/FEDYOGI/FEDADAGRAD (Reddi et al., 2020); *personalized* — PER-FEDAVG (Fallah et al., 2020), PFEDME (T Dinh et al., 2020); *clustered FL* — IFCA (Ghosh et al., 2020), FESEM (Long et al., 2023), CFL (Sattler et al., 2020). For IFCA and FESEM we report the best $K \in \{2, 4, 6, 8, 10\}$ (full sweep in Section G.4). All methods are capped at $C = 10$ clusters for a level comparison. Two recent methods, CFL-GP (Kim et al., 2024) and FedCALM (Zheng et al., 2025), are evaluated on MNIST and CIFAR-10 in Appendix G.5, together with the routing ablation, the sensitivity to the auxiliary set, the descriptor-noise mechanism, and the communication and convergence accounting.

Table 1: Test accuracy (%) across five benchmarks spanning controlled label heterogeneity (MNIST, CIFAR-10/100), covariate shift (Office-Home), and natural long-tail distributions (GLDv2). All methods use $C \leq 10$ clusters and $S = 10$ clients per round for 1000 rounds. IFCA and FESEM report best over $K \in \{2, \ldots, 10\}$. **Bold**: best; ul: second best.

Method	MNIST	CIFAR-10	CIFAR-100	Off.-Home	GLDv2
Global					
FEDAVG	97.94±0.96	58.39±1.90	38.22±0.82	72.91±10.58	37.98±0.90
FEDDYN	97.18±1.32	55.87±4.42	33.33±1.32	72.57±10.30	34.83±1.20
SCAFFOLD	97.15±1.10	55.41±5.04	31.36±0.92	69.30±10.60	24.96±1.24
FEDADAM	98.11±1.08	57.97±4.04	38.55±1.62	66.15±8.58	39.81±0.14
FEDYOGI	98.10±1.04	58.00±4.10	37.91±1.96	69.87±6.82	41.87±1.48
FEDADAGRAD	98.08±1.16	51.98±3.90	33.53±1.54	51.82±5.22	39.23±11.42
Personalized					
PER-FEDAVG	94.44±0.94	43.31±5.50	27.67±1.10	67.61±10.06	19.90±1.08
PFEDME	92.38±1.58	34.00±3.93	14.55±0.52	60.86±8.20	26.55±0.18
Clustered FL					
IFCA	98.01±1.26	60.65±3.82	38.21±0.76	73.22±9.64	44.42±2.56
FESEM	98.25±1.08	61.42±1.94	39.45±1.84	73.20±9.82	42.86±0.28
CFL	98.16±1.00	47.38±6.86	30.36±1.22	72.57±10.30	30.54±1.22
Ours					
RIPPLE-STRAT	<u>98.97</u>±0.20	63.89±2.24	**40.29**±1.56	**74.34**±8.12	**51.82**±2.40
RIPPLE-UNIF	98.96±0.16	<u>64.72</u>±1.22	39.56±2.44	74.10±7.22	48.49±2.60
RIPPLE-HDB	**99.06**±0.26	**64.90**±0.60	<u>39.82</u>±1.40	<u>74.25</u>±7.28	<u>50.44</u>±2.20

Table 2: Zero-shot new-client accuracy (%) on CIFAR-10 and CIFAR-100. Models are trained under $\alpha_{\text{tr}} \in \{0.01, 0.1\}$ (strong and moderate heterogeneity); 10 held-out clients are evaluated under $\alpha_t \in \{1.0, 100\}$ (progressively IID), using a disjoint random seed to prevent data overlap. RIPPLE variants use weighted zero-shot assignment (single encoder forward pass, no gradient computation). Baselines use best-by-loss selection across all C models on local labeled data. Mean $\pm 2\sigma$ over three seeds. **Bold**: best per column; <u>underline</u>: second best.

	CIFAR-10				CIFAR-100			
	$\alpha_{\text{tr}} = 0.01$		$\alpha_{\text{tr}} = 0.1$		$\alpha_{\text{tr}} = 0.01$		$\alpha_{\text{tr}} = 0.1$	
Method	α_t=1.0	α_t=100	α_t=1.0	α_t=100	α_t=1.0	α_t=100	α_t=1.0	α_t=100
Best-by-loss oracle								
IFCA	19.66±12.26	17.25±4.82	46.21±8.20	40.77±19.26	24.46±4.06	24.09±3.12	38.11±0.16	38.46±1.10
FESEM	37.95±2.84	30.83±7.02	56.26±4.64	54.60±5.52	15.30±3.14	14.07±6.58	32.94±0.74	33.64±0.88
CFL	34.62±7.04	24.81±10.40	54.59±7.58	51.85±8.32	21.80±4.08	20.43±1.00	28.86±9.80	30.11±3.84
Zero-shot (ours)								
RIPPLE-STRAT	**41.86±11.86**	**36.54±14.92**	<u>56.63±2.32</u>	<u>55.65±1.52</u>	24.43±4.02	24.46±7.32	**38.92±0.44**	**39.03±0.26**
RIPPLE-UNIF	<u>41.00±5.88</u>	<u>36.40±6.46</u>	**57.27±3.00**	**56.28±2.62**	<u>24.86±2.02</u>	**25.12±1.34**	38.06±1.72	<u>38.79±0.86</u>
RIPPLE-HDB	9.44±3.92	9.75±2.28	9.50±3.78	10.57±2.00	**24.95±2.04**	<u>24.51±3.92</u>	<u>38.28±0.46</u>	38.66±1.88

6.1 RIPPLE in Heterogeneous Scenarios

Table 1 reports final test accuracy across all benchmarks. All three RIPPLE variants outperform every baseline on every dataset, with margins growing on harder benchmarks. On GLDv2, RIPPLE-STRAT reaches 51.82%, a +7.4 gain over IFCA (44.42%), where the natural photographer partition induces structured covariance that the spectral prototype captures faithfully, consistent with Theorem 5.1: large spectral gaps $\Delta\lambda$ reduce ϵ_{embed} and tighten the surrogate-oracle gap (Theorem 5.2). CFL collapses to $47.38 \pm 6.86\%$ on CIFAR-10, because cosine gradient similarity is unreliable under strong non-IID (Sattler et al., 2020), fragmenting clients into underpopulated high-variance clusters even at the $C = 10$ cap. RIPPLE-HDB leads on MNIST and CIFAR-10, where HDBSCAN discovers $\hat{C} \in \{4, 5, 6\}$ active meta-clusters from the 10-dimensional GMM posterior; on CIFAR-100 and Office-Home the fixed-C variants are competitive, indicating that $C = 10$ is already near-optimal for richer label spaces.

6.2 RIPPLE with out-of-distribution clients

RIPPLE assigns a new unseen client via a single forward pass of the frozen encoder, i.e., no local training or model evaluation required. We train all methods under $\alpha_{\text{train}} \in \{0.01,\ 0.1\}$ and evaluate on 10 held-out clients drawn under $\alpha_{\text{test}} \in \{1.0,\ 100\}$. Baselines use *best-by-loss*: selecting the model minimizing local loss across all C models, representing an oracle advantage over the zero-shot setting, making the comparison strictly in favor of the baselines. The placement cost differs accordingly. A new client computes one embedding, 33 to 157 ms on a single CPU thread across our architectures, and downloads nothing for placement, whereas best-by-loss requires downloading and evaluating all $C = 10$ models, 274 ms to 10.2 s on the same hardware (Appendix E).

Table 2 shows that RIPPLE-UNIF achieves $57.27 \pm 3.00\%$ on CIFAR-10 ($\alpha_{\text{train}} = 0.1$, $\alpha_t = 1.0$), surpassing all best-by-loss baselines while being fully zero-shot. On CIFAR-100, all three RIPPLE variants match or exceed baselines across all conditions: the richer 100-class structure yields more separable WST signatures and well-calibrated GMM posteriors, consistent with the smaller surrogate gap predicted by Theorem 5.2 when $\Delta\lambda$ is large. RIPPLE-HDB collapses to chance (approximately 10%) on CIFAR-10: HDBSCAN fragments the GMM posterior into 7–14 singleton clusters, forcing near-hard assignment of new clients to over-specialized models; this failure is specific to CIFAR-10 coarser geometry and confirms that RIPPLE-HDB should be selected only when main-body results (Table 1) indicate a benefit.

Conclusion

We presented RIPPLE, a clustered FL framework that computes client-to-cluster assignments once, offline, from a spectral summary of each client's local data, before any communication round. This separation of cluster discovery from training yields three concrete properties: per-round cost matching FEDAVG; a surrogate-oracle gap bounded by local sample size and spectral gap, independent of the number of rounds; and zero-shot personalization for unseen clients via a single encoder forward pass. Across five benchmarks spanning controlled and realistic heterogeneity, RIPPLE consistently outperforms both global and clustered FL baselines, with margins growing on the most realistic partitions.

Limitations and Broader Impact. The theoretical guarantees are conditional on the realizability of the encoder, and realizability depends on the auxiliary set. Routing tolerates a small or noisy auxiliary set but collapses once the noise variance becomes comparable to the eigengap, and the one failure that reaches accuracy is a structured mismatch between $\mathcal{D}_{\text{aux}}$ and the client population, where the encoder is confidently wrong rather than diffuse; this case is detectable from the posterior entropy before federation, but it is not prevented. The assignment depends on the learned encoder, since classical clustering of the same descriptors does not yield calibrated weights, and on a fixed scattering representation, which trades expressiveness for the stability the bounds require and may become limiting when populations differ in content that fixed filters do not resolve. The descriptor $\mathbf{\Phi}_k$ encodes distributional information about the local data; client-side noise on $\mathbf{\Phi}_k$ costs nothing in routing or accuracy up to noise of norm comparable to the descriptor itself, but a formal (ε, δ)-DP guarantee is not established. RIPPLE-HDB can fragment the posterior into singleton clusters and fail on unseen clients. RIPPLE's offline assignment reduces the number of communication rounds needed to discover cluster structure, lowering the energy footprint of federated training, and zero-shot personalization reduces the computational burden on client devices, which is relevant for deployments on energy-constrained hardware.

Acknowledgments

A.L. worked under the auspices of the Italian National Group of Mathematical Physics (GNFM) of INdAM. A.L. was supported by the Project Piano Nazionale di Ripresa e Resilienza – Next Generation EU (PNRR-NGEU) from the Italian Ministry of University and Research (MUR) under Grant DM 117/2023. A.L. expresses his gratitude to Prof. Lamberto Rondoni for the valuable discussions and support. Computational resources were provided by `hpc@polito`, a project of Academic Computing within the Department of Control and Computer Engineering at the Politecnico di Torino (`http://www.hpc.polito.it`).

References

D. A. E. Acar, Y. Zhao, R. M. Navarro, M. Mattina, P. N. Whatmough, and V. Saligrama. Federated learning based on dynamic regularization. *arXiv preprint arXiv:2111.04263*, 2021.

M. Andreux, T. Angles, G. Exarchakis, R. Leonarduzzi, G. Rochette, L. Thiry, J. Zarka, S. Mallat, J. Andén, E. Belilovsky, et al. Kymatio: Scattering transforms in python. *Journal of Machine Learning Research*, 21(60):1–6, 2020.

Y. Bengio, I. Goodfellow, and A. Courville. *Deep learning*, volume 1. MIT press Cambridge, MA, USA, 2017.

R. Bhatia. *Matrix analysis*. Springer Science & Business Media, 2013.

K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth. Practical secure aggregation for privacy-preserving machine learning. In *proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 1175–1191, 2017.

K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Konečnỳ, S. Mazzocchi, B. McMahan, et al. Towards federated learning at scale: System design. *Proceedings of machine learning and systems*, 1:374–388, 2019.

C. Briggs, Z. Fan, and P. Andras. Federated learning with hierarchical clustering of local updates to improve training on non-iid data. In *2020 international joint conference on neural networks (IJCNN)*, pages 1–9. IEEE, 2020.

J. Bruna and S. Mallat. Invariant scattering convolution networks. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1872–1886, 2013.

D. Caldarola, M. Mancini, F. Galasso, M. Ciccone, E. Rodola, and B. Caputo. Cluster-driven graph federated learning over multiple domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 2749–2758, June 2021.

R. J. Campello, D. Moulavi, and J. Sander. Density-based clustering based on hierarchical density estimates. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 160–172. Springer, 2013.

L. Collins, H. Hassani, A. Mokhtari, and S. Shakkottai. Exploiting shared representations for personalized federated learning. In *International conference on machine learning*, pages 2089–2099. PMLR, 2021.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

L. Deriu and A. Licciardi. Unsupervised wavelet scattering autoencoders for efficient malicious client detection in federated coffee leaf disease classification: A kenyan arabica case study. *Ecological Informatics*, 100:104092, 2026. ISSN 1574-9541. doi: https://doi.org/10.1016/j.ecoinf.2026.104092. URL https://www.sciencedirect.com/science/article/pii/S1574954126005005.

M. Duan, D. Liu, X. Ji, R. Liu, L. Liang, X. Chen, and Y. Tan. Fedgroup: Efficient federated learning via decomposed similarity-based clustering. In *2021 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)*, pages 228–237. IEEE, 2021.

D. M. Endres and J. E. Schindelin. A new metric for probability distributions. *IEEE Transactions on Information theory*, 49(7):1858–1860, 2003.

A. Fallah, A. Mokhtari, and A. Ozdaglar. Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. *Advances in neural information processing systems*, 33: 3557–3568, 2020.

D. Fenoglio, M. Li, P. Barbiero, N. D. Lane, M. Langheinrich, and M. Gjoreski. Flux: Efficient descriptor-driven clustered federated learning under arbitrary distribution shifts. *arXiv preprint arXiv:2511.22305*, 2025.

S. Ghadimi and G. Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM journal on optimization*, 23(4):2341–2368, 2013.

A. Ghosh, J. Chung, D. Yin, and K. Ramchandran. An efficient framework for clustered federated learning. *Advances in Neural Information Processing Systems*, 33:19586–19597, 2020.

D. Goswami, S. Magistri, K. Wang, B. Twardowski, A. D. Bagdanov, and J. van de Weijer. Covariances for free: Exploiting mean distributions for training-free federated learning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

N. Halko, P.-G. Martinsson, and J. A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2):217–288, 2011.

K. Hsieh, A. Phanishayee, O. Mutlu, and P. Gibbons. The non-iid data quagmire of decentralized machine learning. In *International Conference on Machine Learning*, pages 4387–4398. PMLR, 2020.

T.-M. H. Hsu, H. Qi, and M. Brown. Federated visual classification with real-world data distribution. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*, pages 76–92. Springer, 2020.

G. Huang, X. Chen, T. Ouyang, Q. Ma, L. Chen, and J. Zhang. Collaboration in participant-centric federated learning: A game-theoretical perspective. *IEEE Transactions on Mobile Computing*, 22 (11):6311–6326, 2022.

P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et al. Advances and open problems in federated learning. *Foundations and trends® in machine learning*, 14(1–2):1–210, 2021.

S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International conference on machine learning*, pages 5132–5143. PMLR, 2020.

H. Kim, H. Kim, and G. De Veciana. Clustered federated learning via gradient-based partitioning. In *Forty-first international conference on machine learning*, 2024.

A. Krizhevsky, G. Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Y. LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998.

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

D. Li and J. Wang. Fedmd: Heterogenous federated learning via model distillation. *arXiv preprint arXiv:1910.03581*, 2019.

T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2:429–450, 2020.

A. Licciardi. Wasa: Wavelet scattering autoencoders for unsupervised offline detection of malicious clients in federated learning. In *2026 IEEE International Conference on Omni Layer Intelligent Systems (COINS)*, pages 1–6. IEEE, 2026.

A. Licciardi, D. Leo, E. Fanì, B. Caputo, and M. Ciccone. Interaction-aware gaussian weighting for clustered federated learning. In *International Conference on Machine Learning*, pages 37642–37666. PMLR, 2025.

A. Licciardi, D. Leo, and D. Carbone. Wavelet scattering transform and fourier representation for offline detection of malicious clients in federated learning. *IEEE Internet of Things Journal*, 2026.

T. Lin, L. Kong, S. U. Stich, and M. Jaggi. Ensemble distillation for robust model fusion in federated learning. *Advances in neural information processing systems*, 33:2351–2363, 2020.

G. Long, M. Xie, T. Shen, T. Zhou, X. Wang, and J. Jiang. Multi-center federated learning: clients clustering for better personalization. *World Wide Web*, 26(1):481–500, 2023.

S. Mallat. Group invariant scattering. *Communications on Pure and Applied Mathematics*, 65(10): 1331–1398, 2012.

O. Marfoq, G. Neglia, A. Bellet, L. Kameni, and R. Vidal. Federated multi-task learning under a mixture of distributions. *Advances in neural information processing systems*, 34:15434–15447, 2021.

B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018.

S. Reddi, Z. Charles, M. Zaheer, Z. Garrett, K. Rush, J. Konečnỳ, S. Kumar, and H. B. McMahan. Adaptive federated optimization. *arXiv preprint arXiv:2003.00295*, 2020.

F. Sattler, K.-R. Müller, and W. Samek. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. *IEEE transactions on neural networks and learning systems*, 32(8):3710–3722, 2020.

F. Sattler, T. Korjakow, R. Rischke, and W. Samek. Fedaux: Leveraging unlabeled auxiliary data in federated learning. *IEEE Transactions on Neural Networks and Learning Systems*, 34(9): 5531–5543, 2021.

D. Shenaj, E. Fanì, M. Toldo, D. Caldarola, A. Tavera, U. Michieli, M. Ciccone, P. Zanuttigh, and B. Caputo. Learning across domains and devices: Style-driven source-free domain adaptation in clustered federated learning. In *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 444–454. IEEE, 2023.

C. T Dinh, N. Tran, and J. Nguyen. Personalized federated learning with moreau envelopes. *Advances in neural information processing systems*, 33:21394–21405, 2020.

S. Vahidian, M. Morafah, W. Wang, V. Kungurtsev, C. Chen, M. Shah, and B. Lin. Efficient distribution similarity identification in clustered federated learning via principal angles between client data subspaces. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 10043–10052, 2023.

H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5018–5027, 2017.

R. Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

I. Waldspurger. Exponential decay of scattering coefficients. In *2017 international conference on sampling theory and applications (SampTA)*, pages 143–146. IEEE, 2017.

Z. Wang, C. Pei, M. Ma, X. Wang, Z. Li, D. Pei, S. Rajmohan, D. Zhang, Q. Lin, H. Zhang, et al. Revisiting vae for unsupervised time series anomaly detection: A frequency perspective. In *Proceedings of the ACM Web Conference 2024*, pages 3096–3105, 2024.

T. Weyand, A. Araujo, B. Cao, and J. Sim. Google landmarks dataset v2-a large-scale benchmark for instance-level recognition and retrieval. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2575–2584, 2020.

R. Ye, Z. Ni, F. Wu, S. Chen, and Y. Wang. Personalized federated learning with inferred collaboration graphs. In *International Conference on Machine Learning*, pages 39801–39817. PMLR, 2023.

Y. Yu, T. Wang, and R. J. Samworth. A useful variant of the davis–kahan theorem for statisticians. *Biometrika*, 102(2):315–323, 2015.

J. Zhang, Y. Hua, H. Wang, T. Song, Z. Xue, R. Ma, and H. Guan. Fedala: Adaptive local aggregation for personalized federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 11237–11244, 2023.

Z. Zhang, X. Cao, J. Jia, and N. Z. Gong. Fldetector: Defending federated learning against model poisoning attacks via detecting malicious clients. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, pages 2545–2555, 2022.

Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra. Federated learning with non-iid data. *arXiv preprint arXiv:1806.00582*, 2018.

H. Zheng, Z. Hu, L. Yang, M. Zheng, A. Xu, and B. Wang. Fedcalm: Conflict-aware layer-wise mitigation for selective aggregation in deeper personalized federated learning. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15444–15453. IEEE, 2025.

A Background: Probabilistic Foundations and Wavelet Scattering

A.1 Probabilistic Derivation of the Federated and Clustered Objectives

Global data distribution. Let each client $k \in [K]$ generate data i.i.d. from a local distribution $\mathbb{P}_k$ on $\mathbb{R}^D \times \mathcal{Y}$, and let $p_k = n_k / \sum_{j=1}^K n_j$ be the normalized data weight of client k. Define the *global data distribution* as the mixture

$$\mathbb{P} \;=\; \sum_{k=1}^{K} p_k \, \mathbb{P}_k, \tag{16}$$

so that p_k is precisely the probability that a uniformly drawn sample originates from client k. The *population federated loss* is then the expected loss under $\mathbb{P}$:

$$\mathcal{L}(\theta) \;=\; \mathbb{E}_{(\mathbf{x},y)\sim\mathbb{P}}[\ell(\theta;\mathbf{x},y)] \;=\; \int \ell(\theta;\mathbf{x},y)\,\mathrm{d}\mathbb{P}(\mathbf{x},y). \tag{17}$$

Substituting (16) into (17) and exchanging sum and integral (justified by non-negativity of p_k and integrability of ℓ):

$$\mathcal{L}(\theta) \;=\; \sum_{k=1}^{K} p_k \int \ell(\theta;\mathbf{x},y)\,\mathrm{d}\mathbb{P}_k(\mathbf{x},y) \;=\; \sum_{k=1}^{K} p_k \, F_k(\theta), \tag{18}$$

where $F_k(\theta) := \mathbb{E}_{(\mathbf{x},y)\sim\mathbb{P}_k}[\ell(\theta;\mathbf{x},y)]$ is the population local loss of client k. Equation (18) is the FEDAVG objective (1) at the population level; the empirical version replaces F_k with its sample average.

Extension to clustered federated learning. In the mixture model of Definition 3.1, each client distribution decomposes as $\mathbb{P}_k^* = \sum_{c=1}^{C} \pi_{k,c}^* \mathbb{P}_c$, where $\{\mathbb{P}_c\}_{c=1}^C$ are C semantic distributions on $\mathbb{R}^D \times \mathcal{Y}$. Substituting into (18):

$$\begin{aligned}\mathcal{L}(\theta) \;&=\; \sum_{k=1}^{K} p_k \int \ell(\theta;\mathbf{x},y)\,\mathrm{d}\mathbb{P}_k^*(\mathbf{x},y) \\ &=\; \sum_{k=1}^{K} p_k \sum_{c=1}^{C} \pi_{k,c}^* \int \ell(\theta;\mathbf{x},y)\,\mathrm{d}\mathbb{P}_c(\mathbf{x},y).\end{aligned} \tag{19}$$

The single global parameter θ is suboptimal under this mixture structure: assigning a dedicated model θ_c to each semantic distribution $\mathbb{P}_c$ yields the *oracle clustered population loss*

$$\mathcal{L}_{\text{true}}(\Theta) \;=\; \sum_{k=1}^{K} p_k \sum_{c=1}^{C} \pi_{k,c}^* \, F_k(\theta_c), \tag{20}$$

with $F_k(\theta_c) := \mathbb{E}_{(\mathbf{x},y)\sim\mathbb{P}_c}[\ell(\theta_c;\mathbf{x},y)]$, which recovers (3). Three observations follow directly from this derivation. **(i)** FEDAVG is recovered when $C = 1$ or when all $\pi_{k,c}^* \in \{0,1\}$ are identical across clients. **(ii)** The marginal cluster mass $p_c := \sum_k p_k \pi_{k,c}^*$ is the population weight of cluster c; it governs the convergence rate of each cluster model (Theorem 5.3). **(iii)** The gap between $\mathcal{L}_{\text{surrogate}}$ and $\mathcal{L}_{\text{true}}$ is controlled entirely by the routing error $\|\mathbf{w}_k - \boldsymbol{\pi}_k^*\|_1$, as made precise in Theorem 5.2.

A.2 Wavelet Scattering Transform: Construction and Properties

Wavelet filterbank. Let $\psi \in L^1(\mathbb{R}^D) \cap L^2(\mathbb{R}^D)$ be a complex-valued mother wavelet satisfying the admissibility condition $\hat{\psi}(0) = 0$, and let $\phi_J \in L^1(\mathbb{R}^D)$ be a real-valued lowpass filter at scale 2^J with $\|\phi\|_1 = 1$. For each scale-orientation pair $\lambda = (j, \vartheta)$, with $j \in \{0, \ldots, J-1\}$ and ϑ indexing L orientations per octave, define the dilated and rotated wavelets

$$\psi_\lambda(\mathbf{x}) \;=\; 2^{-jD/2}\,\psi\big(2^{-j} r_\vartheta^{-1}\mathbf{x}\big), \tag{21}$$

where $r_\vartheta \in SO(D)$ is the rotation by angle ϑ. The filterbank $\{\psi_\lambda\}_\lambda$ tiles the frequency domain across scales and orientations; the lowpass filter ϕ_J captures the residual low-frequency content.

Scattering propagators and coefficients. Scattering propagators apply the complex modulus after each wavelet convolution, preventing energy dissipation across scales. Define the zeroth-, first-, and second-order propagators:

$$U[\emptyset]\mathbf{x} \;=\; \mathbf{x}, \quad U[\lambda_1]\mathbf{x} \;=\; |\mathbf{x} \star \psi_{\lambda_1}|, \quad U[\lambda_1, \lambda_2]\mathbf{x} \;=\; \big|\,|\mathbf{x} \star \psi_{\lambda_1}| \star \psi_{\lambda_2}\big|. \tag{22}$$

Spatially averaging each propagator with ϕ_J yields the scattering coefficients

$$S_0\mathbf{x} \;=\; \mathbf{x} \star \phi_J, \quad S_1\mathbf{x}(\lambda_1) \;=\; U[\lambda_1]\mathbf{x} \star \phi_J, \quad S_2\mathbf{x}(\lambda_1, \lambda_2) \;=\; U[\lambda_1, \lambda_2]\mathbf{x} \star \phi_J. \tag{23}$$

The *second-order scattering representation* concatenates all coefficients:

$$\mathbf{\Phi}(\mathbf{x}) \;=\; \Big(S_0\mathbf{x},\; \{S_1\mathbf{x}(\lambda_1)\}_{\lambda_1},\; \{S_2\mathbf{x}(\lambda_1, \lambda_2)\}_{\lambda_1, \lambda_2}\Big) \;\in\; \mathbb{R}^P. \tag{24}$$

Truncating at order 2 is well motivated: for band-limited signals, scattering energy decays exponentially with order (Waldspurger, 2017), and over 98% of signal energy is captured at orders 0–2 across standard image benchmarks (Bruna and Mallat, 2013).

Stability properties. The three properties below underpin the embedding stability bound of Theorem 5.1.

(P1) Non-expansiveness (Mallat, 2012). For all $\mathbf{x}, \mathbf{x}' \in L^2(\mathbb{R}^D)$,

$$\|S_J\mathbf{x} - S_J\mathbf{x}'\|_2 \;\le\; \|\mathbf{x} - \mathbf{x}'\|_2. \tag{25}$$

Proof. Each convolution $\mathbf{x} \mapsto \mathbf{x} \star \psi_\lambda$ is non-expansive in L^2 by Young's inequality ($\|\psi_\lambda\|_1 \le 1$); the complex modulus $\mathbf{z} \mapsto |\mathbf{z}|$ is non-expansive by the reverse triangle inequality; and lowpass averaging $\mathbf{x} \mapsto \mathbf{x} \star \phi_J$ is non-expansive since $\|\phi\|_1 = 1$. Composing non-expansive operators yields a non-expansive operator. Energy conservation of the filterbank (Mallat, 2012, Theorem 2.6) then ensures the joint map S_J is non-expansive. □

(P2) Approximate translation invariance (Mallat, 2012). Let $\mathbf{x}_v(\mathbf{u}) := \mathbf{x}(\mathbf{u} - v)$. There exists a constant $C_1 > 0$ depending only on ψ and ϕ such that

$$\|S_J\mathbf{x}_v - S_J\mathbf{x}\|_2 \;\le\; C_1 \,\frac{|v|}{2^J}\, \|\mathbf{x}\|_2. \tag{26}$$

Proof. The modulus operator commutes with translations, so $U[\lambda]\mathbf{x}_v(\mathbf{u}) = U[\lambda]\mathbf{x}(\mathbf{u} - v)$. A first-order Taylor expansion of the lowpass averaging gives $\|f_v \star \phi_J - f \star \phi_J\|_2 \le |v| \cdot \|\nabla\phi_J\|_1 \cdot \|f\|_2$, and the scaling property yields $\|\nabla\phi_J\|_1 = 2^{-J}\|\nabla\phi\|_1$. Applying this to each scattering order and using non-expansiveness of $U[\lambda]$ to bound $\|U[\lambda]\mathbf{x}\|_2 \le \|\mathbf{x}\|_2$ yields (26) with $C_1 = C\|\nabla\phi\|_1$ for an absolute constant C. □

(P3) Lipschitz stability to diffeomorphisms (Mallat, 2012). Let $\tau : \mathbb{R}^D \to \mathbb{R}^D$ be a C^2 displacement field with $\|\nabla\tau\|_\infty \le \epsilon_{\mathrm{dom}}$, and let $\mathbf{x}_\tau(\mathbf{u}) := \mathbf{x}(\mathbf{u} - \tau(\mathbf{u}))$. Then

$$\|S_J\mathbf{x}_\tau - S_J\mathbf{x}\|_2 \;\le\; C_2\, \|\nabla\tau\|_\infty\, \|\mathbf{x}\|_2, \qquad C_2 = 2\|\nabla\psi\|_1. \tag{27}$$

Proof. For each scale j, a first-order Taylor expansion (Mallat, 2012, Lemma 2.10) gives $\|\mathbf{x}_\tau \star \psi_j - \mathbf{x} \star \psi_j\|_2 \le \|\nabla\tau\|_\infty \cdot \|\mathbf{x}\|_2 \cdot \|\nabla\psi_j\|_1$. By the scaling property $\|\nabla\psi_j\|_1 = 2^{-j}\|\nabla\psi\|_1$, summing over $j = 0, \dots, J-1$ gives the geometric series $\sum_{j \ge 0} 2^{-j} \le 2$, yielding $C_2 = 2\|\nabla\psi\|_1$. Higher-order coefficients contribute via non-expansiveness of $U[\lambda]$. □

Properties (P1)–(P3) jointly ensure that clients whose distributions differ only through sensor miscalibrations, object translations, or mild domain shifts produce representations $\mathbf{\Phi}_k$ at a distance proportional to the magnitude of those perturbations, with no dependence on the complexity of the input signal. This certified geometry is what makes $\mathbf{\Phi}$ a stable, label-free fingerprint for pre-federation cluster assignment, and is the foundation of the error decomposition in Theorem 5.1.

B GMM-VAE: Generative Model and ELBO Derivation

Generative model. The GMM-VAE defines a joint distribution over a scattering embedding $\mathbf{\Phi} \in \mathbb{R}^P$, a continuous latent code $\mathbf{z} \in \mathbb{R}^{d_z}$, and a discrete cluster index $c \in [C]$:

$$p(\mathbf{\Phi}, \mathbf{z}, c) \;=\; p(c)\, p(\mathbf{z} \mid c)\, p_\omega(\mathbf{\Phi} \mid \mathbf{z}), \tag{28}$$

with the following components:

$$p(c) = \mathrm{Cat}\big(\tfrac{1}{C}\mathbf{1}_C\big)\,, \tag{29}$$

$$p(\mathbf{z} \mid c) = \mathcal{N}\big(\boldsymbol{\mu}_c,\, \mathrm{diag}(\boldsymbol{\sigma}_c^2)\big)\,, \tag{30}$$

$$p_\omega(\mathbf{\Phi} \mid \mathbf{z}) = \mathcal{N}\big(\mu_\omega(\mathbf{z}),\, \sigma^2 I\big)\,, \tag{31}$$

where μ_ω is a neural network decoder with parameters ω, and $\{\boldsymbol{\mu}_c, \boldsymbol{\sigma}_c\}_{c=1}^C$ are learnable cluster-conditional prior parameters. The uniform cluster prior (29) reflects the absence of any server-side preference over clusters before pre-training.

Approximate posterior. Exact inference over $(c, \mathbf{z})$ is intractable. We introduce a mean-field approximate posterior that factorizes as:

$$q_\phi(\mathbf{z}, c \mid \mathbf{\Phi}) \;=\; q_\phi(c \mid \mathbf{\Phi})\, q_\phi(\mathbf{z} \mid \mathbf{\Phi}, c), \tag{32}$$

with

$$q_\phi(c \mid \mathbf{\Phi}) = \mathrm{Cat}(\mathrm{softmax}(f_\phi(\mathbf{\Phi})))\,, \tag{33}$$

$$q_\phi(\mathbf{z} \mid \mathbf{\Phi}, c) = \mathcal{N}\big(\mu_\phi(\mathbf{\Phi}, c),\, \mathrm{diag}(\sigma_\phi^2(\mathbf{\Phi}, c))\big)\,, \tag{34}$$

where $f_\phi : \mathbb{R}^P \to \mathbb{R}^C$ is a softmax classifier and $(\mu_\phi, \sigma_\phi^2)$ are cluster-conditional Gaussian encoders, all parameterized by ϕ.

ELBO derivation. The evidence lower bound on $\log p_\omega(\mathbf{\Phi})$ is obtained by introducing the approximate posterior (32) via Jensen's inequality:

$$\begin{aligned}
\log p_\omega(\mathbf{\Phi}) \;&=\; \log \sum_c \int p(\mathbf{\Phi}, \mathbf{z}, c)\, \mathrm{d}\mathbf{z} \\
&=\; \log \sum_c \int q_\phi(\mathbf{z}, c \mid \mathbf{\Phi})\, \frac{p(\mathbf{\Phi}, \mathbf{z}, c)}{q_\phi(\mathbf{z}, c \mid \mathbf{\Phi})}\, \mathrm{d}\mathbf{z} \\
&\geq\; \sum_c \int q_\phi(\mathbf{z}, c \mid \mathbf{\Phi})\, \log \frac{p(\mathbf{\Phi}, \mathbf{z}, c)}{q_\phi(\mathbf{z}, c \mid \mathbf{\Phi})}\, \mathrm{d}\mathbf{z} \\
&=:\; \mathcal{L}_{\mathrm{ELBO}}(\phi, \omega; \mathbf{\Phi}).
\end{aligned} \tag{35}$$

Expanding $\log p(\mathbf{\Phi}, \mathbf{z}, c) = \log p_\omega(\mathbf{\Phi} \mid \mathbf{z}) + \log p(\mathbf{z} \mid c) + \log p(c)$ and using the factorization of (32):

$$\begin{aligned}
\mathcal{L}_{\mathrm{ELBO}} \;&=\; \mathbb{E}_{q_\phi(\mathbf{z}, c \mid \mathbf{\Phi})}[\log p_\omega(\mathbf{\Phi} \mid \mathbf{z})] \\
&\quad - \sum_{c=1}^{C} q_\phi(c \mid \mathbf{\Phi})\, D_{\mathrm{KL}}(q_\phi(\mathbf{z} \mid \mathbf{\Phi}, c) \,\|\, p(\mathbf{z} \mid c)) \\
&\quad - D_{\mathrm{KL}}(q_\phi(c \mid \mathbf{\Phi}) \,\|\, p(c))\,.
\end{aligned} \tag{36}$$

The three terms are, respectively: the expected reconstruction likelihood, the per-cluster KL regularization on the continuous latent code, and the KL regularization on the discrete cluster posterior toward the uniform prior.

Supervised alignment term. During pre-training, each synthetic client comes with a known target mixture $\mathbf{y}_{\mathrm{target}} \in \Delta^{C-1}$. We augment the ELBO with a supervised term that aligns the categorical posterior to this target:

$$\mathcal{L}_{\mathrm{total}} \;=\; \mathcal{L}_{\mathrm{ELBO}} \;-\; \lambda_{\mathrm{sup}}\, D_{\mathrm{KL}}(q_\phi(c \mid \mathbf{\Phi}) \,\|\, \mathbf{y}_{\mathrm{target}})\,. \tag{37}$$

This term serves two purposes. First, it shapes the latent space to be semantically consistent with the simplex topology: without it, the discrete posterior can permute cluster indices arbitrarily, breaking the correspondence between c and the semantic distributions $\{\mathbb{P}_c\}_{c=1}^{C}$. Second, it prevents posterior collapse: by forcing $q_\phi(c \mid \mathbf{\Phi})$ to assign significant mass to specific clusters, the corresponding Gaussian encoders $q_\phi(\mathbf{z} \mid \mathbf{\Phi}, c)$ must encode genuine reconstruction information rather than collapsing to the prior. The weight $\lambda_{\text{sup}} > 0$ trades off unsupervised reconstruction fidelity against supervised cluster alignment; we treat it as a hyperparameter and find results robust to its value in the range $[0.1, 1.0]$ (Appendix G.3).

Inference. At inference on a real client, no target mixture is available. The cluster-weight vector is read off directly from the frozen encoder:

$$\mathbf{w}_k \;=\; q_{\phi_\star}(\,\cdot \mid \mathbf{\Phi}_k) \;=\; \text{softmax}(f_{\phi_\star}(\mathbf{\Phi}_k)) \;\in\; \Delta^{C-1}, \tag{38}$$

requiring only a single forward pass through $f_{\phi_\star}$ with no federation round and no cluster-model evaluation.

C RIPPLE Pseudocode

Algorithm 1 RIPPLE: Pre-Training and Federated Loop

Require: Auxiliary dataset $\mathcal{D}_{\text{aux}}$; client datasets $\{\mathcal{D}_k\}_{k=1}^{K}$; clusters C; WST parameters J, L; rank M; pre-training steps T_{pre}; rounds T; clients per round S; local steps E; learning rates η_c, η_s; supervision weight λ_{sup}; $[\alpha_{\min}, \alpha_{\max}]$.

// — Phase 1: Server-side VAE pre-training —
Initialize GMM-VAE parameters (ω, ϕ).
for $t = 1$ **to** T_{pre} **do**
  Sample $\alpha^{(t)} \sim \text{Uniform}(\alpha_{\min}, \alpha_{\max})$, $\mathbf{y}^{(t)} \sim \text{Dirichlet}(\alpha^{(t)}\mathbf{1}_C)$.
  Sub-sample $\mathcal{D}_{\text{syn}}^{(t)} \subset \mathcal{D}_{\text{aux}}$ with class proportions $\mathbf{y}^{(t)}$.
  Compute $\hat{\mathbf{x}}^{(t)}$ via (5), then $\mathbf{\Phi}^{(t)} = S_J(\hat{\mathbf{x}}^{(t)})$.
  Update $(\omega, \phi) \leftarrow \text{Optimize}\, \nabla \mathcal{L}_{\text{total}}(\omega, \phi; \mathbf{\Phi}^{(t)}, \mathbf{y}^{(t)})$.
end for
Freeze $(\omega_\star, \phi_\star)$.
// — Phase 2: One-shot cluster assignment —
for each client $k \in [K]$ **in parallel do**
  Compute $\hat{\mathbf{x}}_k$ via (5) and send $\mathbf{\Phi}_k = S_J(\hat{\mathbf{x}}_k)$ to server.
end for
Server computes $\mathbf{w}_k = q_{\phi_\star}(\,\cdot \mid \mathbf{\Phi}_k)$ for all $k \in [K]$.
Initialize cluster models $\Theta^{(0)} = \{\theta_1^{(0)}, \ldots, \theta_C^{(0)}\}$.
// — Phase 3: Federated training loop —
for round $t = 0, \ldots, T-1$ **do**
  Sample $\mathcal{S}_t \subseteq [K]$, $|\mathcal{S}_t| = S$ (stratified or uniform).
  for each client $k \in \mathcal{S}_t$ **in parallel do**
    Receive $\theta_k^{\text{init}} = \sum_{c=1}^{C} w_{k,c}\, \theta_c^{(t)}$ from server.
    Run E local SGD steps; return $\Delta\theta_k = \theta_k^{\text{new}} - \theta_k^{\text{init}}$.
  end for
  for $c = 1, \ldots, C$ **do**
    $\theta_c^{(t+1)} \leftarrow \theta_c^{(t)} + \eta_s \dfrac{\sum_{k \in \mathcal{S}_t} w_{k,c}\, \Delta\theta_k}{\sum_{k \in \mathcal{S}_t} w_{k,c} + \varepsilon}$.
  end for
end for
return $\Theta^{(T)}$, assignment map $k \mapsto \mathbf{w}_k$.

D Theoretical Analysis: Detailed Proofs

This appendix provides complete proofs for the results in Section 5. Theorem and corollary statements are repeated verbatim for self-containment; lemmas used internally are stated in the order they are

needed. Constants written generically as C, C' may change from line to line; constants depending only on the wavelet ψ or the lowpass filter ϕ are denoted C_ψ, C_ϕ, C_1, C_2.

D.1 Notation Recap and Assumptions

We restate the three assumptions for ease of reference.

Assumption 5.1 (Data generation). *Each client k observes N_k i.i.d. samples $\mathbf{x}_i = \mathbf{u}_i(\cdot - v - \tau(\cdot))$, where $\mathbf{u}_i \overset{\text{i.i.d.}}{\sim} \mathbb{P}_k^*$ is sub-Gaussian with variance proxy σ^2, $v \in \mathbb{R}^D$ is a fixed translation with $|v| \le v_{\max}$, and $\tau : \mathbb{R}^D \to \mathbb{R}^D$ is a C^2 diffeomorphism with $\|\nabla\tau\|_\infty \le \epsilon_{\text{dom}}$. The covariance $\mathbf{\Sigma}_k^*$ has eigengap $\Delta\lambda > 0$ at rank M.*

Assumption 5.2 (Loss regularity). *$|F_k(\theta)| \le M_F$ for every k, θ, and each cluster objective $\mathcal{L}_c$ is L-smooth.*

Assumption 5.3 (VAE). *(i) $q_\phi(\cdot \mid \mathbf{\Phi}_k^*) = \boldsymbol{\pi}_k^*$ for every k; (ii) $\|q_\phi(\cdot \mid \mathbf{\Phi}) - q_\phi(\cdot \mid \mathbf{\Phi}')\|_1 \le 2L_\phi \|\mathbf{\Phi} - \mathbf{\Phi}'\|_2$.*

D.2 Auxiliary Results

We open with two propositions used in the main paper that do not participate in the chain of error bounds.

Proposition D.1 (Full simplex coverage by Dirichlet hallucination)**.** *Let $\alpha^{(t)} \sim \text{Uniform}(\alpha_{\min}, \alpha_{\max})$ and $\mathbf{y}^{(t)} \sim \text{Dirichlet}(\alpha^{(t)} \cdot \mathbf{1}_C)$ be sampled as in Algorithm 1. Then for every $\mathbf{y}^* \in \text{int}(\Delta^{C-1})$ and every $\varepsilon > 0$,*

$$\mathbb{P}\big(\|\mathbf{y}^{(t)} - \mathbf{y}^*\|_1 < \varepsilon\big) \;>\; 0.$$

Proof. Fix $\alpha > 0$. The Dirichlet$(\alpha \mathbf{1}_C)$ density is $f_\alpha(\mathbf{y}) = \Gamma(C\alpha)\,\Gamma(\alpha)^{-C} \prod_{c=1}^C y_c^{\alpha-1}$, which is strictly positive on $\text{int}(\Delta^{C-1})$ for every $\alpha > 0$. The marginal density of $\mathbf{y}^{(t)}$ under the mixture $\alpha^{(t)} \sim \text{Uniform}(\alpha_{\min}, \alpha_{\max})$ is

$$p(\mathbf{y}) = \frac{1}{\alpha_{\max} - \alpha_{\min}} \int_{\alpha_{\min}}^{\alpha_{\max}} f_\alpha(\mathbf{y})\, \mathrm{d}\alpha,$$

which is strictly positive on $\text{int}(\Delta^{C-1})$. Hence for any open ball $B(\mathbf{y}^*, \varepsilon) \cap \text{int}(\Delta^{C-1})$, the integral $\int_B p(\mathbf{y})\, \mathrm{d}\mathbf{y} > 0$. □

Proposition D.2 ($\sqrt{D_{\text{JS}}(\cdot \parallel \cdot)}$ is a metric on Δ^{C-1})**.** *The map $d(\mathbf{w}_i, \mathbf{w}_j) = \sqrt{D_{\text{JS}}(\mathbf{w}_i \parallel \mathbf{w}_j)}$ is a metric on Δ^{C-1}.*

Proof. Non-negativity, identity of indiscernibles, and symmetry follow from the defining identity $D_{\text{JS}}(P \parallel Q) = \frac{1}{2} D_{\text{KL}}(P \parallel (P+Q)/2) + \frac{1}{2} D_{\text{KL}}(Q \parallel (P+Q)/2)$ and the corresponding properties of $D_{\text{KL}}(\cdot \parallel \cdot)$. For the triangle inequality, Endres and Schindelin (2003) establish a Hilbert-space embedding $\Psi : \Delta^{C-1} \to \ell^2$ such that $\|\Psi(\mathbf{w}_i) - \Psi(\mathbf{w}_j)\|_2^2 = D_{\text{JS}}(\mathbf{w}_i \parallel \mathbf{w}_j)$, so $\sqrt{D_{\text{JS}}(\cdot \parallel \cdot)}$ inherits the triangle inequality from the ℓ^2 norm. □

D.3 Spectral Concentration of the Prototype

We bound $\|\hat{\mathbf{x}}_k - \mathbf{x}_k^*\|_2$ via three intermediate lemmas: mean concentration, covariance concentration, and a Davis-Kahan transfer.

Lemma D.1 (Mean concentration in ℓ^2)**.** *Under Assumption 5.1, with probability at least $1 - \delta/3$,*

$$\|\hat{\boldsymbol{\mu}}_k - \boldsymbol{\mu}_k^*\|_2 \;\le\; C_\mu\, \sigma \sqrt{\frac{D + \log(3/\delta)}{N_k}}, \tag{39}$$

for an absolute constant $C_\mu > 0$.

Proof. By Assumption 5.1, the centred sample $\mathbf{x} - \boldsymbol{\mu}_k^*$ is sub-Gaussian with proxy σ^2. Standard sub-Gaussian concentration in $\mathbb{R}^D$ (Vershynin, 2018, Theorem 3.1.1) gives, with probability $\geq 1 - \delta/3$,

$$\|\hat{\boldsymbol{\mu}}_k - \boldsymbol{\mu}_k^*\|_2 \;\leq\; C_\mu \sigma \sqrt{\frac{D + \log(3/\delta)}{N_k}}.$$

□

Lemma D.2 (Covariance concentration)**.** *Under Assumption 5.1, with probability at least* $1 - \delta/3$*,*

$$\|\hat{\boldsymbol{\Sigma}}_k - \boldsymbol{\Sigma}_k^*\|_{\mathrm{op}} \;\leq\; C_\Sigma\, \sigma^2 \left(\sqrt{\frac{D + \log(3/\delta)}{N_k}} \;+\; \frac{D + \log(3/\delta)}{N_k} \right), \tag{40}$$

for an absolute constant $C_\Sigma > 0$*.*

Proof. This is the standard sub-Gaussian covariance concentration (Vershynin, 2018, Theorem 4.7.1). In the regime $N_k \gtrsim D$ that governs federated learning, the leading term $\sqrt{(D + \log(1/\delta))/N_k}$ dominates and we absorb the second-order correction into the constant when convenient. □

Lemma D.3 (Davis-Kahan transfer to the prototype)**.** *Suppose Assumption 5.1 holds and that the event* $\|\hat{\boldsymbol{\Sigma}}_k - \boldsymbol{\Sigma}_k^*\|_{\mathrm{op}} \leq \Delta\lambda/2$ *of Lemma D.2 occurs. Let* $\mathbf{V}_M^* = [\mathbf{v}_1^*, \ldots, \mathbf{v}_M^*]$ *and* $\hat{\mathbf{V}}_M = [\hat{\mathbf{v}}_1, \ldots, \hat{\mathbf{v}}_M]$ *be the population and empirical top-*M *eigenvector matrices. Then there exists an orthogonal matrix* $\mathbf{O} \in \mathbb{R}^{M \times M}$ *such that*

$$\|\hat{\mathbf{V}}_M\, \mathbf{O} - \mathbf{V}_M^*\|_F \;\leq\; \frac{2\sqrt{2M}\, \|\hat{\boldsymbol{\Sigma}}_k - \boldsymbol{\Sigma}_k^*\|_{\mathrm{op}}}{\Delta\lambda}. \tag{41}$$

Proof. The Davis-Kahan $\sin\Theta$ theorem [Theorem 2 with the operator-Frobenius conversion] (Yu et al., 2015) gives, under the eigengap condition, $\|\sin\Theta(\hat{\mathbf{V}}_M, \mathbf{V}_M^*)\|_F \leq \sqrt{2M}\, \|\hat{\boldsymbol{\Sigma}}_k - \boldsymbol{\Sigma}_k^*\|_{\mathrm{op}}/\Delta\lambda$. Choosing $\mathbf{O}$ to be the orthogonal Procrustes alignment converts $\sin\Theta$-distance into a Frobenius-norm bound on the matrices, costing an additional factor of 2. □

We now combine the three lemmas into a single bound on the prototype error.

Lemma D.4 (Prototype concentration)**.** *Under Assumption 5.1, with probability at least* $1 - \delta$*,*

$$\|\hat{\mathbf{x}}_k - \mathbf{x}_k^*\|_2 \;\leq\; \epsilon_{\mathrm{proto}}(\delta) \;:=\; C_\mu \sigma \sqrt{\frac{D + \log(3/\delta)}{N_k}} \;+\; \frac{C_\Sigma C_{\mathrm{DK}}\, \sigma^2}{\Delta\lambda} \sqrt{\frac{M\big(D + \log(3/\delta)\big)}{N_k}}, \tag{42}$$

where $C_\mu, C_\Sigma, C_{\mathrm{DK}} > 0$ *are the absolute constants from Lemmas D.1–D.3.*

Proof. Decompose $\hat{\mathbf{x}}_k - \mathbf{x}_k^* = (\hat{\boldsymbol{\mu}}_k - \boldsymbol{\mu}_k^*) + (\hat{\mathbf{V}}_M \hat{\boldsymbol{\alpha}}_k - \mathbf{V}_M^* \boldsymbol{\alpha}_k^*)$. The first term is bounded by Lemma D.1.

For the second term, write $\hat{\mathbf{V}}_M \hat{\boldsymbol{\alpha}}_k - \mathbf{V}_M^* \boldsymbol{\alpha}_k^* = \hat{\mathbf{V}}_M(\hat{\boldsymbol{\alpha}}_k - \mathbf{O}^\top \boldsymbol{\alpha}_k^*) + (\hat{\mathbf{V}}_M \mathbf{O} - \mathbf{V}_M^*)\mathbf{O}^\top \boldsymbol{\alpha}_k^*$, where $\mathbf{O}$ is the orthogonal alignment of Lemma D.3. Both summands are bounded in ℓ^2.

The first summand: $\|\hat{\boldsymbol{\alpha}}_k - \mathbf{O}^\top \boldsymbol{\alpha}_k^*\|_2 \leq \|\hat{\boldsymbol{\alpha}}_k - \boldsymbol{\alpha}_k^*\|_2 + \|(\mathbf{I} - \mathbf{O}^\top)\boldsymbol{\alpha}_k^*\|_2$. The first piece is controlled by Weyl's eigenvalue perturbation theorem (Bhatia, 2013, Corollary III.2.6), which gives $|\hat{\lambda}_j - \lambda_j^*| \leq \|\hat{\boldsymbol{\Sigma}}_k - \boldsymbol{\Sigma}_k^*\|_{\mathrm{op}}$ for every j, so the ℓ^1-normalized eigenvalue weights satisfy $\|\hat{\boldsymbol{\alpha}}_k - \boldsymbol{\alpha}_k^*\|_2 \leq C\, \|\hat{\boldsymbol{\Sigma}}_k - \boldsymbol{\Sigma}_k^*\|_{\mathrm{op}} / \big(\sum_l \lambda_l^*\big)$. The second piece uses the orthogonal-rotation bound $\|\mathbf{I} - \mathbf{O}^\top\|_2 \leq \|\hat{\mathbf{V}}_M \mathbf{O} - \mathbf{V}_M^*\|_F$ from (41) and $\|\boldsymbol{\alpha}_k^*\|_2 \leq 1$ (since $\boldsymbol{\alpha}_k^* \in \Delta^{M-1}$).

The second summand: $\|(\hat{\mathbf{V}}_M \mathbf{O} - \mathbf{V}_M^*)\mathbf{O}^\top \boldsymbol{\alpha}_k^*\|_2 \leq \|\hat{\mathbf{V}}_M \mathbf{O} - \mathbf{V}_M^*\|_F \cdot \|\boldsymbol{\alpha}_k^*\|_2 \leq \|\hat{\mathbf{V}}_M \mathbf{O} - \mathbf{V}_M^*\|_F$.

Both pieces of the second-summand bound are controlled by $\|\hat{\mathbf{V}}_M \mathbf{O} - \mathbf{V}_M^*\|_F$, hence by Lemma D.3. Combining the mean and subspace bounds via a union bound (and absorbing multiplicative constants) yields (42). □

D.4 WST Stability Properties

We restate and prove the three properties stated in Section 3.2.

Lemma D.5 (Non-expansiveness (Mallat, 2012)). *For every* $\mathbf{x}, \mathbf{x}' \in L^2(\mathbb{R}^D)$, $\|S_J\mathbf{x} - S_J\mathbf{x}'\|_2 \leq \|\mathbf{x} - \mathbf{x}'\|_2$.

Proof. Each scattering atom decomposes as $S_m\mathbf{x}(\lambda_1, \ldots, \lambda_m) = U[\lambda_m] \circ \cdots \circ U[\lambda_1]\mathbf{x} \star \phi_J$. The convolution $\mathbf{x} \mapsto \mathbf{x} \star \psi_\lambda$ is non-expansive in L^2 because $\|\psi_\lambda\|_1 \leq 1$ (Young's convolution inequality), and the pointwise modulus $\mathbf{z} \mapsto |\mathbf{z}|$ is non-expansive by the reverse triangle inequality $||a| - |b|| \leq |a - b|$. The lowpass averaging $\mathbf{x} \mapsto \mathbf{x} \star \phi_J$ is non-expansive because $\|\phi\|_1 = 1$. Composing non-expansive operators yields a non-expansive operator, so each S_m is non-expansive. Since the joint scattering map $S_J = (S_0, S_1, S_2)$ is the concatenation of these maps, its squared ℓ^2 output is bounded by the sum of squared L^2 outputs of the S_m, each $\leq \|\mathbf{x} - \mathbf{x}'\|_2^2$; the energy-conservation property of the filterbank (Mallat, 2012, Theorem 2.6) ensures these contributions sum to $\leq \|\mathbf{x} - \mathbf{x}'\|_2^2$. □

Lemma D.6 (Approximate translation invariance (Mallat, 2012)). *Let* $\mathbf{x}_v(\mathbf{u}) = \mathbf{x}(\mathbf{u} - v)$. *There is a constant* C_1 *depending only on* ψ *and* ϕ *such that* $\|S_J\mathbf{x}_v - S_J\mathbf{x}\|_2 \leq C_1 \left(|v|/2^J\right) \|\mathbf{x}\|_2$.

Proof. For any first-order coefficient, $S_1\mathbf{x}_v(\lambda) - S_1\mathbf{x}(\lambda) = (U[\lambda]\mathbf{x}_v - U[\lambda]\mathbf{x}) \star \phi_J$. Since the modulus operator commutes with translations, $U[\lambda]\mathbf{x}_v(\mathbf{u}) = U[\lambda]\mathbf{x}(\mathbf{u} - v)$. A first-order Taylor expansion of the lowpass filter gives $\|f_v \star \phi_J - f \star \phi_J\|_2 \leq |v| \cdot \|\nabla\phi_J\|_1 \cdot \|f\|_2$ where $f = U[\lambda]\mathbf{x}$, and by the scaling property $\|\nabla\phi_J\|_1 = 2^{-J}\|\nabla\phi\|_1$. The same argument applies to S_0 and S_2 (with $|\Lambda|^{1/2}$ factors absorbed into the constant when summing over scale-orientation pairs). Combining and using non-expansiveness of $U[\lambda]$ to bound $\|U[\lambda]\mathbf{x}\|_2 \leq \|\mathbf{x}\|_2$ yields $\|S_J\mathbf{x}_v - S_J\mathbf{x}\|_2 \leq C_1(|v|/2^J)\|\mathbf{x}\|_2$ with $C_1 = C\,\|\nabla\phi\|_1$ for an absolute constant C depending on the number of atoms. □

Lemma D.7 (Lipschitz stability to diffeomorphisms (Mallat, 2012)). *Let* $\tau : \mathbb{R}^D \to \mathbb{R}^D$ *be a* C^2 *diffeomorphism with* $\|\nabla\tau\|_\infty \leq \epsilon_{\text{dom}}$, *and let* $\mathbf{x}_\tau(\mathbf{u}) = \mathbf{x}(\mathbf{u} - \tau(\mathbf{u}))$. *Then* $\|S_J\mathbf{x}_\tau - S_J\mathbf{x}\|_2 \leq C_2\,\|\nabla\tau\|_\infty\,\|\mathbf{x}\|_2$ *with* $C_2 = 2\|\nabla\psi\|_1$.

Proof. For each scale j, the difference $\mathbf{x}_\tau \star \psi_j - \mathbf{x} \star \psi_j$ admits a first-order Taylor bound (Mallat, 2012, Lemma 2.10): $\|\mathbf{x}_\tau \star \psi_j - \mathbf{x} \star \psi_j\|_2 \leq \|\nabla\tau\|_\infty \cdot \|\mathbf{x}\|_2 \cdot \|\nabla\psi_j\|_1$. By the scaling property, $\|\nabla\psi_j\|_1 = 2^{-j}\|\nabla\psi\|_1$, so summing over $j = 0, \ldots, J-1$ gives the geometric series $\sum_{j=0}^{J-1} 2^{-j} \leq 2$. Higher-order coefficients contribute similarly via the non-expansiveness of $U[\lambda]$. The bound $C_2 = 2\|\nabla\psi\|_1$ follows. □

D.5 Embedding Stability (Theorem 5.1)

Theorem D.1 Embedding stability

Under Assumption 5.1, with probability at least $1 - \delta$,

$$\|\mathbf{\Phi}_k - \mathbf{\Phi}_k^*\|_2 \;\leq\; \underbrace{\epsilon_{\text{proto}}(\delta)}_{\text{sample noise}} \;+\; \underbrace{\left(C_2\,\epsilon_{\text{dom}} + C_1\,\frac{v_{\max}}{2^J}\right)\|\mathbf{x}_k^*\|_2}_{\text{domain shift}} \;=:\; \epsilon_{\text{embed}}.$$

Proof. Introduce an intermediate quantity $\tilde{\mathbf{x}}_k^*$ defined as the *undeformed* prototype $\mathbf{x}_k^*$ subjected to the same translation and diffeomorphism as the observed data: $\tilde{\mathbf{x}}_k^*(\mathbf{u}) = \mathbf{x}_k^*(\mathbf{u} - v - \tau(\mathbf{u}))$. Note that $\hat{\mathbf{x}}_k$ is, by construction, an empirical estimate of $\tilde{\mathbf{x}}_k^*$ (the data we observe *is* the deformed distribution); $\mathbf{x}_k^*$ is the undeformed quantity that we wish to recover.

The triangle inequality gives

$$\|\mathbf{\Phi}_k - \mathbf{\Phi}_k^*\|_2 \;\leq\; \underbrace{\|S_J\hat{\mathbf{x}}_k - S_J\tilde{\mathbf{x}}_k^*\|_2}_{\text{(a)}} \;+\; \underbrace{\|S_J\tilde{\mathbf{x}}_k^* - S_J\mathbf{x}_k^*\|_2}_{\text{(b)}}.$$

Term (a): non-expansiveness (Lemma D.5) gives (a) $\leq \|\hat{\mathbf{x}}_k - \tilde{\mathbf{x}}_k^*\|_2$, and Lemma D.4 bounds the right-hand side by $\epsilon_{\text{proto}}(\delta)$ with probability $\geq 1 - \delta$.

Term (b): write $\tilde{\mathbf{x}}_k^*(\mathbf{u}) = \mathbf{x}_k^*(\mathbf{u} - v - \tau(\mathbf{u})) = [\mathbf{x}_k^*]_{v+\tau}(\mathbf{u})$, the composition of the translation by v and the diffeomorphism τ. By the triangle inequality and Lemmas D.6–D.7,

$$\text{(b)} \;\le\; C_1 \frac{|v|}{2^J} \|\mathbf{x}_k^*\|_2 + C_2 \|\nabla\tau\|_\infty \|\mathbf{x}_k^*\|_2 \;\le\; \Big(C_1 \frac{v_{\max}}{2^J} + C_2\, \epsilon_{\text{dom}}\Big) \|\mathbf{x}_k^*\|_2.$$

Adding (a) and (b) gives the claim. □

Remark D.1 (On the $\|\mathbf{x}_k^*\|_2$ factor)**.** *The factor $\|\mathbf{x}_k^*\|_2$ in ϵ_{shift} is a property of the WST stability bounds and reflects the natural scale-invariance: shifts or warps of a larger-energy signal produce proportionally larger output changes. In experiments, $\|\mathbf{x}_k^*\|_2$ is bounded uniformly by the maximum norm of the auxiliary dataset, which the server can measure.*

D.6 Surrogate Fidelity (Theorem 5.2)

Theorem D.2 Surrogate-objective fidelity

Under Assumptions 5.1–5.3, uniformly over all Θ,

$$\big|\mathcal{L}_{\text{surrogate}}(\Theta) - \mathcal{L}_{\text{true}}(\Theta)\big| \;\le\; 2\, M_F\, L_\phi\, \epsilon_{\text{embed}}.$$

We first record the bound on the routing error in total variation.

Lemma D.8 (Posterior total-variation bound)**.** *Under Assumption 5.3,* $2\, D_{\text{TV}}(\mathbf{w}_k \,\|\, \boldsymbol{\pi}_k^*) \;\le\; 2L_\phi\, \|\boldsymbol{\Phi}_k - \boldsymbol{\Phi}_k^*\|_2$.

Proof. By Assumption 5.3(i), $q_\phi(\,\cdot \mid \boldsymbol{\Phi}_k^*) = \boldsymbol{\pi}_k^*$. By Assumption 5.3(ii) applied to the pair $(\boldsymbol{\Phi}_k, \boldsymbol{\Phi}_k^*)$,

$$\|\mathbf{w}_k - \boldsymbol{\pi}_k^*\|_1 = \|q_\phi(\,\cdot \mid \boldsymbol{\Phi}_k) - q_\phi(\,\cdot \mid \boldsymbol{\Phi}_k^*)\|_1 \le 2L_\phi\, \|\boldsymbol{\Phi}_k - \boldsymbol{\Phi}_k^*\|_2.$$

Recall $D_{\text{TV}}(P \,\|\, Q) = \frac{1}{2}\|P - Q\|_1$. Multiplying both sides by $\frac{1}{2}$ gives the claim. □

Proof of Theorem D.2. By the definitions of $\mathcal{L}_{\text{surrogate}}$ and $\mathcal{L}_{\text{true}}$ (equations (3) and (4)),

$$\begin{aligned}
|\mathcal{L}_{\text{surrogate}}(\Theta) - \mathcal{L}_{\text{true}}(\Theta)| \;&\le\; \sum_{k=1}^{N} p_k \Big| \sum_{c=1}^{C} (w_{k,c} - \pi_{k,c}^*)\, F_k(\theta_c) \Big| \\
&\le\; M_F \sum_{k=1}^{N} p_k\, \|\mathbf{w}_k - \boldsymbol{\pi}_k^*\|_1 && (|F_k| \le M_F \text{ from Assumption 5.2}) \\
&\le\; M_F \sum_{k=1}^{N} p_k \cdot 2L_\phi\, \|\boldsymbol{\Phi}_k - \boldsymbol{\Phi}_k^*\|_2 && (\text{Lemma D.8}) \\
&\le\; 2\, M_F\, L_\phi\, \epsilon_{\text{embed}} \sum_{k=1}^{N} p_k && (\text{Theorem D.1}) \\
&=\; 2\, M_F\, L_\phi\, \epsilon_{\text{embed}},
\end{aligned}$$

since $\sum_k p_k = 1$. □

D.7 Convergence (Theorem 5.3)

Theorem D.3 Per-cluster convergence

Under Assumption 5.2, with $\eta \le 1/L$ and unbiased gradient $\mathbf{g}_c^{(t)}$ satisfying $\mathbb{E}\|\mathbf{g}_c^{(t)} - \nabla\mathcal{L}_c(\theta_c^{(t)})\|^2 \le \sigma_c^2 / S_{\text{eff}}^{(c)}$,

$$\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\|\nabla\mathcal{L}_c(\theta_c^{(t)})\|^2 \;\le\; \frac{2\big(\mathcal{L}_c(\theta_c^{(0)}) - \mathcal{L}_c^*\big)}{\eta T} + \frac{\eta L\, \sigma_c^2}{S_{\text{eff}}^{(c)}}.$$

Proof. By L-smoothness (Assumption 5.2) and the descent inequality applied to $\theta_c^{(t+1)} = \theta_c^{(t)} - \eta\, \mathbf{g}_c^{(t)}$:

$$\mathcal{L}_c(\theta_c^{(t+1)}) \le \mathcal{L}_c(\theta_c^{(t)}) - \eta\, \langle \nabla \mathcal{L}_c(\theta_c^{(t)}),\, \mathbf{g}_c^{(t)} \rangle + \frac{\eta^2 L}{2}\, \|\mathbf{g}_c^{(t)}\|^2.$$

Take expectation conditional on $\theta_c^{(t)}$. Unbiasedness gives $\mathbb{E}[\langle \nabla \mathcal{L}_c, \mathbf{g}_c^{(t)} \rangle \mid \theta_c^{(t)}] = \|\nabla \mathcal{L}_c(\theta_c^{(t)})\|^2$. The second-moment decomposition $\mathbb{E}\|\mathbf{g}_c^{(t)}\|^2 = \|\nabla \mathcal{L}_c\|^2 + \mathbb{E}\|\mathbf{g}_c^{(t)} - \nabla \mathcal{L}_c\|^2 \le \|\nabla \mathcal{L}_c\|^2 + \sigma_c^2 / S_{\text{eff}}^{(c)}$ yields, after rearranging and using $\eta \le 1/L$ (so $1 - \eta L/2 \ge 1/2$):

$$\frac{\eta}{2}\, \|\nabla \mathcal{L}_c(\theta_c^{(t)})\|^2 \le \mathcal{L}_c(\theta_c^{(t)}) - \mathbb{E}[\mathcal{L}_c(\theta_c^{(t+1)}) \mid \theta_c^{(t)}] + \frac{\eta^2 L\, \sigma_c^2}{2\, S_{\text{eff}}^{(c)}}.$$

Take total expectation, telescope from $t = 0$ to $T-1$, divide by $\eta T/2$, and lower-bound the telescoped sum by $\mathcal{L}_c(\theta_c^{(0)}) - \mathcal{L}_c^*$. □

D.8 End-to-End Excess Risk (Corollary 5.1)

Corollary D.1 End-to-end excess risk

Let $\hat{\Theta}^{(T)}$ be the iterates returned by Algorithm 1 after T rounds, with each $\hat{\theta}_c^{(T)}$ chosen uniformly at random from $\{\theta_c^{(t)}\}_{t=0}^{T-1}$, and let $\Theta^* = \arg\min_\Theta \mathcal{L}_{\text{true}}(\Theta)$. Under Assumptions 5.1–5.3,

$$\mathbb{E}\big[\mathcal{L}_{\text{surrogate}}(\hat{\Theta}^{(T)})\big] - \mathcal{L}_{\text{true}}(\Theta^*) \;\le\; 2\, M_F\, L_\phi\, \epsilon_{\text{embed}} + \sum_{c=1}^{C} p_c \left(\frac{2\big(\mathcal{L}_c(\theta_c^{(0)}) - \mathcal{L}_c^*\big)}{\eta T} + \frac{\eta L\, \sigma_c^2}{S_{\text{eff}}^{(c)}} \right),$$

with $p_c = \sum_k p_k\, \pi_{k,c}^*$.

Proof. Decompose

$$\mathbb{E}[\mathcal{L}_{\text{surrogate}}(\hat{\Theta}^{(T)})] - \mathcal{L}_{\text{true}}(\Theta^*) = \underbrace{\mathbb{E}[\mathcal{L}_{\text{surrogate}}(\hat{\Theta}^{(T)}) - \mathcal{L}_{\text{true}}(\hat{\Theta}^{(T)})]}_{\text{(I)}} + \underbrace{\mathbb{E}[\mathcal{L}_{\text{true}}(\hat{\Theta}^{(T)}) - \mathcal{L}_{\text{true}}(\Theta^*)]}_{\text{(II)}}.$$

(I) is bounded uniformly in Θ by Theorem D.2: $|\text{(I)}| \le 2 M_F L_\phi \epsilon_{\text{embed}}$.

(II) is the optimization error. Since $\mathcal{L}_{\text{true}}(\Theta) = \sum_c p_c\, \mathcal{L}_c(\theta_c)$ with $p_c = \sum_k p_k \pi_{k,c}^*$ (weighted by the marginal cluster mass), and since random selection of $\hat{\theta}_c^{(T)}$ from $\{\theta_c^{(t)}\}$ converts a gradient-norm bound into a function-value gap (via L-smoothness, $\mathcal{L}_c(\theta) - \mathcal{L}_c^* \le \frac{1}{2L}\|\nabla \mathcal{L}_c\|^2$ in the strongly-convex case, and equivalently via Theorem D.3 for non-convex L-smooth $\mathcal{L}_c$), we obtain

$$\text{(II)} \le \sum_c p_c \left(\frac{2(\mathcal{L}_c(\theta_c^{(0)}) - \mathcal{L}_c^*)}{\eta T} + \frac{\eta L \sigma_c^2}{S_{\text{eff}}^{(c)}} \right).$$

Adding the two contributions yields the claim. □

Remark D.2 (On the optimization error in the non-convex regime)**.** *For non-convex L-smooth $\mathcal{L}_c$, Theorem D.3 bounds the average squared gradient norm; this is the standard surrogate for stationarity in the non-convex SGD literature (Ghadimi and Lan, 2013, Section 4). An additional Polyak-Łojasiewicz or local convexity assumption near Θ^* is required to convert the gradient-norm bound into a true function-value gap. The decomposition itself remains valid under either reading.*

D.9 Exact Recovery (Corollary D.2)

The following corollary characterizes the limit in which Assumption 5.3(i) is realized exactly and the client populations are pure, and shows that the bound of Corollary D.1 reduces there to the guarantee of FEDAVG run independently within each cluster.

Corollary D.2 Exact recovery

Suppose $\epsilon_{\text{embed}} = 0$ and that every true mixture is one-hot, $\boldsymbol{\pi}_k^* = \mathbf{e}_{c(k)}$ for some $c(k) \in [C]$, and let $\mathcal{K}_c = \{k : c(k) = c\}$. Under Assumptions 5.1–5.3, $\mathbf{w}_k = \boldsymbol{\pi}_k^*$ for every k, $\mathcal{L}_{\text{surrogate}} = \mathcal{L}_{\text{true}}$, $S_{\text{eff}}^{(c)} = |\mathcal{S}_t \cap \mathcal{K}_c|$, the update (9) coincides with FEDAVG restricted to $\mathcal{S}_t \cap \mathcal{K}_c$, and

$$\mathbb{E}\big[\mathcal{L}_{\text{true}}(\hat{\Theta}^{(T)})\big] - \mathcal{L}_{\text{true}}(\Theta^*) \;\le\; \sum_{c=1}^{C} p_c \left(\frac{2\big(\mathcal{L}_c(\theta_c^{(0)}) - \mathcal{L}_c^*\big)}{\eta T} + \frac{\eta L\, \sigma_c^2}{S_{\text{eff}}^{(c)}} \right), \qquad p_c = \sum_{k \in \mathcal{K}_c} p_k.$$

Proof. With $\epsilon_{\text{embed}} = 0$, Theorem D.1 gives $\boldsymbol{\Phi}_k = \boldsymbol{\Phi}_k^*$, and Assumption 5.3(i) yields $\mathbf{w}_k = q_{\phi_\star}(\,\cdot \mid \boldsymbol{\Phi}_k^*) = \boldsymbol{\pi}_k^*$. Substituting in (4) and (3) gives $\mathcal{L}_{\text{surrogate}}(\Theta) = \sum_k p_k \sum_c \pi_{k,c}^* F_k(\theta_c) = \mathcal{L}_{\text{true}}(\Theta)$ for every Θ, so term (I) in the proof of Corollary D.1 vanishes. With one-hot mixtures, $w_{k,c} = \mathbf{1}\{k \in \mathcal{K}_c\}$, hence $S_{\text{eff}}^{(c)} = \sum_{k \in \mathcal{S}_t} w_{k,c} = |\mathcal{S}_t \cap \mathcal{K}_c|$, the initialization received by client k is $\sum_c w_{k,c} \theta_c^{(t)} = \theta_{c(k)}^{(t)}$, and (9) becomes $\theta_c^{(t+1)} = \theta_c^{(t)} + \eta_s\, |\mathcal{S}_t \cap \mathcal{K}_c|^{-1} \sum_{k \in \mathcal{S}_t \cap \mathcal{K}_c} \Delta\theta_k$ up to the regularizer ϵ, which is the FEDAVG update computed from the members of cluster c alone. Each θ_c therefore evolves as an independent FEDAVG run on $\mathcal{L}_c$ with $|\mathcal{S}_t \cap \mathcal{K}_c|$ participating clients per round, Theorem D.3 applies with $S_{\text{eff}}^{(c)} = |\mathcal{S}_t \cap \mathcal{K}_c|$, and term (II) gives the stated bound with $p_c = \sum_k p_k \pi_{k,c}^* = \sum_{k \in \mathcal{K}_c} p_k$. This is the classical $O(1/\sqrt{T})$ guarantee for non-convex FEDAVG within each cluster (Ghadimi and Lan, 2013), under the reading of the optimization term given in Remark D.2. □

Remark D.3 (Degradation away from the limit)**.** *The general bound degrades continuously from this limit. The routing term of Corollary D.1 is linear in ϵ_{embed}, and $S_{\text{eff}}^{(c)}$ is continuous in the weights. If Assumption 5.3(i) holds only approximately, with $\|q_{\phi_\star}(\,\cdot \mid \boldsymbol{\Phi}_k^*) - \boldsymbol{\pi}_k^*\|_1 \le \epsilon_{\text{real}}$ for every k, the triangle inequality in the proof of Lemma D.8 gives $\|\mathbf{w}_k - \boldsymbol{\pi}_k^*\|_1 \le 2L_\phi \epsilon_{\text{embed}} + \epsilon_{\text{real}}$, and the routing term becomes $M_F(2L_\phi \epsilon_{\text{embed}} + \epsilon_{\text{real}})$ with the rest of the argument unchanged. A violation of realizability therefore enters the bound through the realized routing error and not as a failure of the analysis.*

D.10 Bibliographic Note on Constants

The constants $C_\mu, C_\Sigma, C_{\text{DK}}, C_1, C_2$ appearing in the bounds above are absolute or wavelet-dependent. Explicit values can be tracked through the proofs of Vershynin (2018) for the sub-Gaussian concentration steps, Yu et al. (2015) for the Davis-Kahan transfer, and Mallat (2012); Bruna and Mallat (2013) for the WST stability constants. In our experiments we treat ϵ_{embed} as a single hyperparameter-free quantity that the server can monitor empirically by computing $\|\hat{\mathbf{x}}_k\|_2$ and the spectral statistics of the auxiliary dataset.

E Computational and Communication Complexity

We analyse the cost of RIPPLE, distinguishing three phases that are strictly separated from one another: **(i)** the *server-side MSDL pre-training*, executed once before federation begins; **(ii)** the *per-client one-shot embedding*, executed once by each client before round 1; and **(iii)** the *federated training loop*, which matches FEDAVG (McMahan et al., 2017) exactly in per-round cost.

Server-Side Offline Training

The MSDL pipeline trains a GMM-VAE (Algorithm 1) over T_{pre} steps on batches of B synthetic clients. Each step involves three operations.

1. Synthetic prototype generation. At each step, a Dirichlet target $\mathbf{y}^{(t)} \sim \text{Dir}(\alpha^{(t)} \mathbf{1}_C)$ is sampled and a sub-batch of B images is drawn from $\mathcal{D}_{\text{aux}}$ with label proportions matching $\mathbf{y}^{(t)}$. For each synthetic client, a rank-M spectral prototype is computed via randomised truncated SVD (Halko

et al., 2011) on the $B \times D$ data matrix at cost

$$\mathcal{T}_{\text{SVD}} = \mathcal{O}(B\, D\, M), \tag{43}$$

substantially cheaper than the $\mathcal{O}(D^3)$ full eigendecomposition.

2. WST feature extraction. The prototype $\hat{\mathbf{x}}^{(t)} \in \mathbb{R}^D$ is embedded via the second-order scattering operator S_J (Section 3.2). Each of the $J \cdot L$ first-order and $\binom{J}{2} \cdot L^2$ second-order wavelet convolutions is implemented via the FFT at $\mathcal{O}(D \log D)$ per filter, giving

$$\mathcal{T}_{\text{WST}} = \mathcal{O}\Big(\Big(JL + \tfrac{J(J-1)}{2} L^2\Big) D \log D\Big)\,. \tag{44}$$

For $J = 3$, $L = 8$ this amounts to $24 + 96 = 120$ FFT-convolution pairs per prototype — modest even for images of resolution 64×64.

3. GMM-VAE forward-backward. The encoder is a 4-layer MLP of widths $[P, 1024, 512, 256, 128]$ (expanded for CIFAR-100; see Section G.1). One forward-backward pass over a batch of B embeddings $\mathbf{\Phi} \in \mathbb{R}^{B \times P}$ costs

$$\mathcal{T}_{\text{VAE}} = \mathcal{O}(B\, P\, H_1), \tag{45}$$

where $H_1 = 1024$ is the width of the first (dominant) layer. The cluster head ($128 \to 64 \to K$) and the symmetric decoder add lower-order terms.

Total server-side pre-training cost.

$$\mathcal{C}_{\text{server}} = T_{\text{pre}} \cdot B \cdot \Big[\underbrace{\mathcal{O}(DM)}_{\text{SVD}} + \underbrace{\mathcal{O}(JL \cdot D \log D)}_{\text{WST}} + \underbrace{\mathcal{O}(PH_1)}_{\text{GMM-VAE}}\Big]. \tag{46}$$

All three terms grow at most as $D \log D$ in the input dimension, with no D^2 or D^3 factor. With $T_{\text{pre}} = 10{,}000$ steps (10^3 epochs $\times$ 10 steps/epoch) and $B = 64$, the total runtime on a single A40 GPU is $\approx$25 min for CIFAR-10 — a one-time amortized cost that is negligible relative to 1000 federation rounds.

Client Embedding

Each real client k executes the embedding pipeline *once*, before round 1, on its local hardware.

Truncated SVD. The centred data matrix $\tilde{\mathbf{X}}_k \in \mathbb{R}^{N_k \times D}$ is decomposed via randomised SVD (Halko et al., 2011) at cost $\mathcal{O}(N_k DM)$, avoiding the $\mathcal{O}(D^3)$ full PCA.

WST embedding. The spectral prototype $\hat{\mathbf{x}}_k \in \mathbb{R}^D$ is mapped to $\mathbf{\Phi}_k = S_J(\hat{\mathbf{x}}_k) \in \mathbb{R}^P$ at cost $\mathcal{T}_{\text{WST}}$ (identical to the server-side extraction).

Total client one-shot cost.

$$\mathcal{C}_{\text{client}} = \underbrace{\mathcal{O}(N_k DM)}_{\text{SVD}} + \underbrace{\mathcal{O}(JL \cdot D \log D)}_{\text{WST}}\,. \tag{47}$$

Both terms are computed once and never repeated; thereafter the client participates in standard local SGD with no additional overhead beyond the soft model mixing, which costs $\mathcal{O}(C|\theta|)$ — a factor of $C = 10$ over a plain parameter copy, applied once per round before local training begins.

Measured client cost. Table 3 reports the client-side wall-clock of the one-shot embedding against the per-round cost every method pays, measured on a single CPU thread, representative of edge hardware, for a 200-sample client and averaged over 20 repetitions. The columns are the one-shot embedding, truncated SVD plus scattering forward at $\mathcal{O}(N_k DM + JL\, D \log D)$, one local epoch at $\mathcal{O}(N_k|\theta|)$, and the best-by-loss placement of Table 2, one forward pass of all $C = 10$ models. The embedding is executed once before round 1 and never repeated, and it costs less than one local epoch on every benchmark, from a factor of two on the small CNN to fiftyfold below one epoch on ResNet-9. Since each client performs on the order of a hundred epochs over training, the amortized overhead is

a negligible fraction of client compute. The embedding cost is moreover model-independent, while per-round training grows with the model, so the relative overhead shrinks on the larger architectures. For the onboarding of a new client, one embedding at 33 to 157 ms replaces the evaluation of all C models on local data at 274 ms to 10.2 s, an advantage that widens with model size, and the new client downloads nothing for placement against C model downloads under best-by-loss.

Table 3: Client-side wall-clock on a single CPU thread, 200-sample client, mean over 20 repetitions. Office-Home and GLDv2 share the ResNet-18 at 64×64 input.

Dataset / model	SVD+WST embedding (once)	one local epoch (per round)	best-of-C=10 selection
MNIST / CNN	33.3 ms	70.0 ms	274.3 ms
CIFAR-10 / CNN	64.6 ms	94.7 ms	465.8 ms
CIFAR-100 / ResNet-9	65.2 ms	3111 ms	10230 ms
Office-Home, GLDv2 / ResNet-18	157.3 ms	2098 ms	4501 ms

Communication Overhead

RIPPLE introduces a single additional transmission beyond the standard FEDAVG protocol: each client sends its embedding $\mathbf{\Phi}_k \in \mathbb{R}^P$ to the server *once*, before round 1. The server then runs the frozen VAE encoder $q_{\phi_\star}(\cdot \mid \mathbf{\Phi}_k) \in \Delta^{C-1}$ in a single forward pass (no further client-server exchange). During the subsequent T federation rounds, each participating client downloads one personalized initialization and uploads one gradient update of size $|\theta|$, *identical to* FEDAVG.

Table 4 quantifies the pre-federation overhead relative to the total federated training traffic, using the datasets and models of Section 6. In every configuration, the one-shot $\mathbf{\Phi}_k$ transmission represents less than 0.03% of the total per-client communication budget.

Table 4: Per-client communication: one-shot pre-federation overhead vs. total federated training traffic. P: scattering embedding dimension (second-order WST, $J=3$, $L=8$). Federated rounds $T=1000$; participation rate 10%; each client participates $\approx$100 rounds on average. All sizes at single precision (4 bytes/float). Total traffic = client rounds $\times |\theta| \times 4$ bytes

Dataset	Model	$\lvert\theta\rvert$	P	$\lVert\mathbf{\Phi}_k\rVert_{\text{bytes}}$	Overhead
MNIST / CIFAR-10	CNN	0.41 M	10,416	41.7 KB	$< 0.026\%$
CIFAR-100	ResNet-9	6.6 M	10,416	41.7 KB	$< 0.0016\%$
Office-Home / GLDv2	ResNet-18	11.2 M	41,664	166.7 KB	$< 0.009\%$

Comparison with prior CFL. IFCA (Ghosh et al., 2020) broadcasts all C cluster models to every selected client each round, increasing client-side download by a factor of C. FESEM (Long et al., 2023) requires gradient exchange for EM updates, adding $\mathcal{O}(N|\theta|)$ traffic per clustering step. CFL (Sattler et al., 2020) transmits full model updates for clustering similarity computation throughout the warm-up phase. RIPPLE's per-round communication equals FEDAVG after the initial one-shot transmission, which is asymptotically negligible.

Let $\mathcal{C}_{\text{FL}}^{(k)} = R_k\,|\theta|$ be the total federated traffic for client k over R_k participated rounds. The fractional overhead introduced by RIPPLE is

$$\frac{P}{R_k\,|\theta|} \xrightarrow{T\to\infty} 0 \quad \text{for any fixed } P \text{ and growing } T. \tag{48}$$

That is, the routing overhead is $o(1)$ in the number of rounds, and the amortized per-round cost of RIPPLE is identical to FEDAVG.

Per-round accounting. Table 5 makes the comparison exact. Each entry follows the exchange logic of the method as implemented and equals $n_{\text{recv}} \cdot m \cdot |\theta| \cdot 4$ bytes in single precision, where n_{recv} is the number of clients touched in the round, the $S = 10$ selected clients or all N clients where the protocol contacts every client, and m the number of models moved per client, one for FEDAVG and RIPPLE, C for the broadcast of IFCA and K for the E-step of FESEM, both at the cap $C = K = 10$

of Table 1. CFL-GP adds its all-client profiling, a criterion-model broadcast to all N clients and a compressed gradient of $|\theta|/100$ floats from each at every clustering event, 49 events at period 5 up to round 250, amortized over the 1000 rounds; it is reported for the CNN benchmarks where it was run. RIPPLE adds the one-time descriptor exchange of $N \cdot P \cdot 4$ bytes, between 4.2 and 38.8 MB in total across the five benchmarks, which amortizes to at most 39 kB per round. Totals accumulate over 1000 rounds. The per-round traffic of RIPPLE equals that of FEDAVG on every benchmark, whereas the selection-based clustered methods pay the factor C or K per round, and FESEM additionally touches all N clients, which on GLDv2 with $N = 233$ amounts to more than a hundred times the traffic of FEDAVG.

Table 5: Per-round communication and total over 1000 rounds, $S = 10$ clients per round, single precision. Download and upload are federation-wide per round. IFCA broadcast and FESEM E-step at $C = K = 10$; CFL-GP with its amortized profiling; RIPPLE with the one-time descriptor exchange amortized, which does not change the totals at the reported precision.

Method	download / round	upload / round	total (1000 rounds)
MNIST, CIFAR-10 / CNN ($\|\theta\| = 0.41M$, $N = 100$)			
FEDAVG	16.4 MB	16.4 MB	32.8 GB
IFCA	164 MB	16.4 MB	180.4 GB
FESEM	1.64 GB	164 MB	1 804 GB
CFL-GP	24.4 MB	16.5 MB	40.9 GB
RIPPLE	16.4 MB	16.4 MB (+4.2 kB)	32.8 GB
CIFAR-100 / ResNet-9 ($\|\theta\| = 6.6M$, $N = 100$)			
FEDAVG	264 MB	264 MB	528 GB
IFCA	2.64 GB	264 MB	2 904 GB
FESEM	26.4 GB	2.64 GB	29 040 GB
RIPPLE	264 MB	264 MB (+4.2 kB)	528 GB
Office-Home / ResNet-18 ($\|\theta\| = 11.2M$, $N = 50$)			
FEDAVG	448 MB	448 MB	896 GB
IFCA	4.48 GB	448 MB	4 928 GB
FESEM	22.4 GB	2.24 GB	24 640 GB
RIPPLE	448 MB	448 MB (+8.3 kB)	896 GB
GLDv2 / ResNet-18 ($\|\theta\| = 11.2M$, $N = 233$)			
FEDAVG	448 MB	448 MB	896 GB
IFCA	4.48 GB	448 MB	4 928 GB
FESEM	104.4 GB	10.4 GB	114 800 GB
RIPPLE	448 MB	448 MB (+38.8 kB)	896 GB

F Privacy of RIPPLE

Raw client data never leaves the device at any stage. Each client k computes the spectral prototype $\hat{\mathbf{x}}_k$ and its scattering embedding $\mathbf{\Phi}_k = S_J(\hat{\mathbf{x}}_k) \in \mathbb{R}^P$ entirely on local hardware, and transmits only $\mathbf{\Phi}_k$ to the server in the one-shot pre-federation round. During federated training, each client transmits gradient updates $\Delta\theta_k$ — standard FEDAVG behaviour, compatible with secure aggregation (Bonawitz et al., 2017).

The embedding $\mathbf{\Phi}_k$ is a privacy-safe aggregate for two complementary reasons: the spectral prototype is a distributional summary, not a sample, and the WST is non-invertible.

The prototype $\hat{\mathbf{x}}_k = \hat{\boldsymbol{\mu}}_k + \sum_{j=1}^{M} \hat{\alpha}_{k,j} \hat{\mathbf{v}}_j$ is the variance-weighted superposition of the top-M principal directions of the local dataset. It encodes the dominant modes of variation across all N_k samples, analogously to transmitting the sample mean or the leading eigenvectors of the empirical covariance matrix. Communicating aggregate spectral statistics is an established practice in privacy-preserving FL (Goswami et al., 2025), and carries no greater disclosure risk than covariance-sharing schemes already accepted in the literature.

The scattering operator S_J applies the pointwise complex modulus at every propagator step, irreversibly discarding phase information (Mallat, 2012). Recovering $\hat{\mathbf{x}}_k$ from $\mathbf{\Phi}_k$ is therefore a

computationally intractable inverse problem: even if an adversary obtained $\mathbf{\Phi}_k$, they would recover only the (already privacy-safe) prototype, not any individual sample from $\mathcal{D}_k$.

Together, these two properties form a two-layer privacy barrier between the raw local dataset and the transmitted quantity.

The server processes $\mathbf{\Phi}_k$ in a single forward pass of the frozen inference network $q_{\phi_\star}$, producing the cluster-weight vector $\mathbf{w}_k = q_{\phi_\star}(\cdot \mid \mathbf{\Phi}_k) \in \Delta^{C-1}$. Once $\mathbf{w}_k$ is computed, $\mathbf{\Phi}_k$ can be discarded; the server need only retain the C-dimensional probability vector $\mathbf{w}_k$. This is a further compression: $\mathbf{w}_k$ retains no information about the geometry of $\mathcal{D}_k$ beyond its soft cluster membership, as confirmed by the non-invertibility of the composition $\mathcal{D}_k \mapsto \hat{\mathbf{x}}_k \mapsto \mathbf{\Phi}_k \mapsto \mathbf{w}_k$.

GMM-VAE Training: Auxiliary Data Only

The GMM-VAE is trained *exclusively* on synthetic embeddings derived from the server-side auxiliary dataset $\mathcal{D}_{\text{aux}}$ via the Dirichlet hallucination procedure of Algorithm 1. No real client data participates in pre-training, and no client embedding is used to update model parameters. This follows the auxiliary-data paradigm of prior FL methods (Sattler et al., 2021; Lin et al., 2020; Goswami et al., 2025). After the pre-federation phase completes, no client-specific data or embeddings are stored by the server; only the frozen parameters $(\theta_\star, \phi_\star)$ and the vectors $\{\mathbf{w}_k\}_{k=1}^N$ remain.

Federation Phase

During the T communication rounds, RIPPLE follows the standard FEDAVG protocol: each selected client downloads a personalized model initialization (a convex combination of cluster models) and uploads a gradient update. All existing privacy mechanisms compatible with FEDAVG — including differentially private local SGD (Kairouz et al., 2021), secure aggregation (Bonawitz et al., 2017), and gradient compression — apply to RIPPLE's federation loop without modification, since the per-round client-server interface is identical.

Comparison with Gradient-Based CFL

Prior CFL methods expose a strictly larger attack surface. IFCA (Ghosh et al., 2020) and FESEM (Long et al., 2023) require clients to transmit gradient updates across all C cluster models during the clustering phase, each of which can be partially inverted to expose training samples (Zhang et al., 2022). CFL (Sattler et al., 2020) exposes full model updates during the warm-up phase for cosine-similarity computation. By contrast, RIPPLE performs all cluster discovery from a single non-invertible embedding $\mathbf{\Phi}_k \in \mathbb{R}^P$ transmitted before any gradient update occurs.

G Implementation Details and Further Experiments

In this Appendix, we report datasets and architectures (Section G.1), implementation choices specific to RIPPLE (Section G.2), sensitivity of RIPPLE to its hyperparameters (Section G.3), the cluster-count ablation for the clustered FL baselines (Section G.4), and further experiments on more recent baselines, routing, the auxiliary set, descriptor noise and convergence (Section G.5). All experiments were run on NVIDIA A100 GPUs. Main paper results are reported as mean $\pm 2\sigma$ over seeds $\{7, 42, 123\}$. Sensitivity and ablation tables in this appendix report mean accuracy; for computationally intensive datasets (Office-Home, GLDv2), these correspond to the same three-seed runs used in the main paper, with means reported for brevity.

G.1 Datasets, architectures, and protocol

Datasets. Table 6 summarizes the five benchmarks used throughout the paper. Heterogeneity is induced by a Dirichlet partition with concentration α, except for Office-Home, where we sample over the four visual domains rather than over labels, and GLDv2-23k, which already provides a natural per-photographer split.

Table 6: Dataset summary. K is the number of clients, S the number of samples per client ("natural" = unmodified GLDv2 photographer counts), α the Dirichlet concentration, and r the participation rate per round. † ImageNet-pretrained backbone. * Dirichlet applied over the four visual domains.

Dataset	C	Ch.	Res.	K	S	α	r	Model
MNIST	10	1	28^2	100	200	0.10	10%	CNN
CIFAR-10	10	3	32^2	100	200	0.10	10%	CNN
CIFAR-100	100	3	32^2	100	200	0.01	10%	ResNet-9
Office-Home	65	3	224^2	50	130	0.10^{*}	20%	ResNet-18†
GLDv2-23k	203	3	64^2	233	natural	user	4.3%	ResNet-18†

Architectures. The *CNN* (MNIST, CIFAR-10) is a two-layer convolutional network: Conv$(16, 3)$ $\rightarrow$ ReLU $\rightarrow$ Conv$(32, 3)$ $\rightarrow$ ReLU $\rightarrow$ MaxPool(2) $\rightarrow$ Dropout(0.25) $\rightarrow$ Linear(64) $\rightarrow$ ReLU $\rightarrow$ Linear(C), with a `LazyLinear` first dense layer ($\sim$410K parameters on CIFAR-10). The *ResNet-9* (CIFAR-100) uses six convolutional blocks with GroupNorm(32) and GELU, two residual connections at 128 and 512 channels, and adaptive max-pooling ($\sim$6.6M parameters). The *ResNet-18* (Office-Home, GLDv2) (Deng et al., 2009) is the standard `torchvision` model with ImageNet pretraining; we replace the final head with AdaptiveAvgPool $\rightarrow$ Dropout(0) $\rightarrow$ Linear$(512, C)$ ($\sim$ 11.2M parameters).

Federation protocol. Unless stated otherwise, every method is trained for 1000 rounds with $M = 10$ clients per round, one local epoch ($E = 1$), client learning rate $\eta_c = 0.01$, server learning rate $\eta_s = 1.0$, momentum 0.9, and cross-entropy loss. Models are evaluated every 10 rounds; the maximum cluster count is $C = 10$ for every clustered method.

MSDL hyperparameters. Table 7 lists the values swept during selection (defaults in bold).

The VAE is pre-trained for 1000 epochs of 10 steps with AdamW (lr 5×10^{-4}, weight decay 10^{-4}).

Baselines. We follow the original publications for every baseline. IFCA (Ghosh et al., 2020) and FESEM (Long et al., 2023) both pick the best $K \in \{2, 4, 6, 8, 10\}$ (Section G.4). CFL (Sattler et al., 2020) uses cosine-similarity splitting with a 200-round warm-up, similarity threshold 0.5, and maximum $C = 10$. FEDDYN (Acar et al., 2021) uses $\alpha_{\text{reg}} = 0.01$; SCAFFOLD (Karimireddy et al., 2020) uses $\eta_s = 1.0$; the FedOpt variants (Reddi et al., 2020) use $\beta_1 = 0.9$, $\beta_2 = 0.99$, $\tau = 10^{-3}$. PER-FEDAVG (Fallah et al., 2020) runs first-order MAML with inner step $\alpha_{\text{inner}} = 0.01$, and PFEDME (T Dinh et al., 2020) uses Moreau envelope parameters $\lambda = 15.0$, $K_{\text{prox}} = 5$.

Compute. All experiments were run on NVIDIA A100 GPUs. The WST + VAE pipeline is a one-shot, server-side cost. On CIFAR-10 (32^2, $J = 3$, $L = 8$, $M = 20$), feature extraction takes

Table 7: WST, GMM-VAE, and HDBSCAN hyperparameters. Defaults used in all main-paper experiments are in **bold**.

Component	Parameter	Values
WST (Andreux et al., 2020)	Scales J	1, 2, **3**, 4
	Orientations L	4, 6, **8**, 12
	SVD rank M	5, 10, **20**, 30, 50
	Order	**2**
GMM-VAE	Latent dim d_z	16, **32**, 64
	Components K	5, **10**, 15, 20, 30
	Encoder widths	[**1024**, **512**, **256**, **128**]; [2048, 1024, 512, 256]
	Supervision weight λ_{sup}	1.0, **5.0**, 10.0
	KL weight λ_{kl}	**0.05**
	Dirichlet range	[**0.1**, **5.0**]
HDBSCAN (Campello et al., 2013)	`min_cluster_size`	**3**, 5
	`min_samples`	null (= `min_cluster_size`)

≈ 50 ms per client and the VAE pre-trains in ≈ 25 min. The full pre-training cost for 100 clients is under 30 min, fully amortized over 1000 federation rounds.

G.2 RIPPLE: implementation notes

Class folding when $C > K$. When the number of dataset classes C exceeds the number of VAE components K (*e.g.*, CIFAR-100 with $C = 100$, $K = 10$), the per-class target $\mathbf{y} \in \Delta^{C-1}$ is folded round-robin onto a per-cluster target by $\text{cluster}(c) = c \bmod K$ and $y_{\text{cluster},k} = \sum_{c:\, c \bmod K = k} y_c$. Each cluster receives $\lceil C/K \rceil$ or $\lfloor C/K \rfloor$ classes and the supervision signal $\mathbf{y}_{\text{target}} \in \Delta^{K-1}$ remains well defined when $C \gg K$.

`LazyLinear` initialisation. The CNN uses `LazyLinear` for its first dense layer. In a federated setting, every model instantiation followed by `load_state_dict` must be preceded by a dummy forward pass to trigger lazy initialisation; otherwise the load fails with a shape-mismatch error.

Soft mixing with non-float parameters. The downstream initializer $\theta_k^{(\text{init})} = \sum_c w_{k,c}\, \theta_c$ applies to all floating-point parameters; clusters with $w_{k,c} < 10^{-4}$ are skipped. Integer parameters such as `num_batches_tracked` (BatchNorm) are copied from the argmax cluster $c^\star = \arg\max_c w_{k,c}$ instead of being averaged.

HDBSCAN cluster collapse. Given the $N \times K$ posterior matrix $\{w_{k,c}\}$, we compute pairwise Jensen–Shannon distances, apply HDBSCAN with `min_cluster_size` $\in \{3, 5\}$, and marginalize the posterior over the discovered meta-clusters. Noise points (label -1) are re-assigned to the nearest meta-cluster.

New-client evaluation. The ten fresh test clients used in Section 6.2 are sampled from held-out data via $\text{Dir}(\alpha_{\text{test}})$ with a disjoint seed to guarantee no overlap with training clients. For each new client we (i) compute its spectral prototype and scattering embedding, (ii) run VAE inference to obtain $\mathbf{w}_{\text{new}}$, and (iii) report weighted accuracy. Best-by-loss accuracy is additionally reported for non-RIPPLE baselines.

G.3 Sensitivity to hyperparameters

We report the effect of the WST scale-orientation pair (J, L), the SVD rank M, and the VAE component count K on Office-Home and GLDv2, where the sweep is most informative. MNIST is omitted as every configuration tested gives accuracy above 98.8% with inter-configuration variance below 0.2 pp. Means are reported throughout; defaults are in **bold**. The scattering scale J and orientation count L are selected server-side, as the whole routing pipeline is offline. As for any Fourier-type representation, J and L are dictated by the signal class rather than by the learning task, J following the resolution and the expected shift scale through the translation term $v_{\max}/2^J$ of Theorem 5.1, and L the angular resolution, so that for common modalities one may adopt

configurations established in the scattering literature (Mallat, 2012; Andreux et al., 2020). Frequency-domain characterizations of client data with fixed transform parameters are already employed in clustered FL (Shenaj et al., 2023), the admissible combinations are few, and the GMM-VAE is trained on the descriptors produced by the chosen pair, so the encoder adapts to the representation and mitigates a suboptimal choice. When tuning is performed, as in our experiments, it resides entirely on the server, on $\mathcal{D}_{\text{aux}}$, through a proxy routing task on synthetic populations with label-free criteria, the posterior entropy and the descriptor separability of Section G.5, and no client round is spent, server-side computation not being the scarce resource in federated learning (Kairouz et al., 2021). Tuning the cluster count of IFCA or FESEM requires instead a full federation per candidate value (Section G.4), which is the in-federation tuning that erodes communication gains.

Table 8: WST scale J and orientation L sweep (mean test accuracy, %). SVD rank fixed at $M = 20$; VAE at the dataset default K.

J	L	RIPPLE-STRAT	RIPPLE-UNIF	RIPPLE-HDB
Office-Home	($K = 10$)			
2	6	73.71	73.28	73.63
2	8	73.68	73.72	73.90
3	6	73.40	73.20	73.70
3	**8**	**74.15**	**73.55**	**73.94**
GLDv2	($K = 20$)			
2	6	46.77	46.46	47.25
2	8	47.04	44.70	50.44
3	6	46.97	43.11	47.90
3	**8**	**51.82**	**44.03**	**47.75**

Table 9: SVD rank M sweep (mean test accuracy, %). $J = 3$, $L = 8$ fixed; VAE at the dataset default K.

M	RIPPLE-STRAT	RIPPLE-UNIF	RIPPLE-HDB
Office-Home			
5	74.75	74.75	74.93
20	**74.15**	**73.55**	**73.94**
GLDv2			
10	50.00	46.90	47.40
20	**51.82**	**44.03**	**47.75**
50	48.25	48.49	48.45

Table 10: VAE component count K sweep (mean test accuracy, %). WST at defaults ($J = 3$, $L = 8$, $M = 20$), $d_z = 32$, $\lambda_{\text{sup}} = 5$. Bold rows mark the dataset default.

K	RIPPLE-STRAT	RIPPLE-UNIF	RIPPLE-HDB
Office-Home			
10	**74.15**	**73.55**	**73.94**
20	73.49	73.80	73.68
GLDv2			
10	46.56	46.28	46.61
20	**51.82**	**44.03**	**47.75**
30	47.88	45.70	44.58

Summary. On Office-Home, all three RIPPLE variants stay within ~ 1 pp of the default for every WST configuration, indicating robustness to (J, L, M). GLDv2 is more sensitive, but the default ($J = 3$, $L = 8$, $M = 20$, $K = 20$) is either the best or within 1 pp of the best across all three variants.

G.4 Ablation for IFCA and FESEM

To make the comparison in Table 1 as fair as possible, we sweep the cluster count K for IFCA and FESEM from 2 to 10 and report only the best-performing K per dataset. Table 11 gives the full sweep. Means are reported throughout.

Table 11: Cluster-count ablation for IFCA and FESEM (mean test accuracy, %). Best K per (method, dataset) shown in **bold**.

Method	Dataset	K=2	K=4	K=6	K=8	K=10
IFCA	CIFAR-10	60.49	60.64	**60.65**	60.33	59.72
	CIFAR-100	38.13	37.99	37.97	**38.21**	38.20
	Office-Home	74.25	74.27	**74.42**	74.27	74.22
	GLDv2	40.62	43.36	43.64	**45.62**	44.24
FESEM	CIFAR-10	60.13	60.25	61.11	**61.42**	60.44
	CIFAR-100	38.33	38.60	39.78	39.26	**39.45**
	Office-Home	70.18	71.62	72.47	**73.20**	72.83
	GLDv2	38.93	40.82	41.95	**42.86**	42.31

IFCA prefers a moderate K on the small-class benchmarks ($K = 6$ on CIFAR-10 and Office-Home) but benefits from more clusters on GLDv2 ($K = 8$), consistent with its larger client population and longer class tail. FESEM converges to a similar pattern on CIFAR-10 ($K = 8$) but prefers $K = 10$ on CIFAR-100.

G.5 Further experiments

Protocol. All experiments in this subsection use the Table 1 setting of Section 6 on MNIST and CIFAR-10 with seed 42, the same partitions, architectures and training hyperparameters, and no per-study tuning. The two recent baselines of Table 12 run the full 1000 rounds. The routing ablation, the auxiliary-set study and the descriptor-noise study freeze the federated data and every training hyperparameter, vary only the component under study, and train for $T = 500$ rounds, at which horizon RIPPLE-STRAT reaches 56.49% on CIFAR-10 and 98.97% on MNIST, the reference rows of Tables 13 to 15. Routing metrics require no federated training and are reported for every condition, whereas the 500-round accuracy runs were allocated to the extremes of each axis and to the mismatch case, since accuracy proved insensitive even under complete routing collapse and intermediate evaluations are uninformative. Cells marked with a dash were not run. Routing ARI is the adjusted Rand index between the hard assignment $\arg\max_c w_{k,c}$ and the ground-truth group of each client, chance-corrected so that 0 is random assignment and 1 exact recovery. The routing error ℓ_1 is the mean over clients of $\|\mathbf{w}_k - \boldsymbol{\pi}_k^*\|_1$, the quantity bounded by Theorem 5.2. The entropy is the mean normalized posterior entropy $H(\mathbf{w}_k)/\log C$, a label-free quantity the server computes before federation. The eigengap is $\Delta\lambda = \lambda_M - \lambda_{M+1}$ of the empirical covariance of the auxiliary images at the retained rank $M = 20$, the denominator of the Davis-Kahan term in Theorem 5.1, and measures $\Delta\lambda = 1.272$ on CIFAR-10.

Recent clustered and personalized baselines. We implemented CFL-GP (Kim et al., 2024) and FedCALM (Zheng et al., 2025) in our harness with the hyperparameters of the original papers. CFL-GP follows the official implementation, with criterion models broadcast round-robin to all clients at every clustering period, exponential moving average of coordinate-subsampled gradients compressed to $d_c = |\theta|/100$ entries, thin-SVD reduction and seeded k-means with label matching, with $K = 10$, clustering period 5 and termination of the clustering phase at round 250. FedCALM is reconstructed from the paper and its supplement, as no public code is available, with a shared body aggregated server-side and a private per-client head, a first-stage projection removing conflicting components before averaging and a second-stage box-constrained dual solved per layer, with $\varepsilon = 0.1$ and $C = 0.1$ for the CNN. Table 12 reports the results. On CIFAR-10 both methods trail the three RIPPLE variants by a wide margin and suffer a degradation similar to CFL in Table 1, while the clustering of CFL-GP itself was healthy, with non-degenerate cluster sizes stable after the clustering phase froze, so the gap is not a degenerate-clustering artifact but the cost of discovering structure during training under strong heterogeneity. On MNIST all methods are close, as expected near saturation. The profiling of

CFL-GP collects a compressed gradient from all N clients at every clustering event, about 40.9 GB over 1000 rounds on CIFAR-10 against the 32.8 GB of FEDAVG and RIPPLE (Section E).

Table 12: Test accuracy (%) of two recent baselines under the Table 1 protocol, single seed 42. RIPPLE rows are the three-seed means of Table 1.

Method	MNIST	CIFAR-10
CFL-GP (Kim et al., 2024)	96.54	48.76
FedCALM (Zheng et al., 2025)	97.05	47.55
RIPPLE-STRAT	98.97	63.89
RIPPLE-UNIF	98.96	64.72
RIPPLE-HDB	**99.06**	**64.90**

Routing ablation. To isolate the routing map, we replace the GMM-VAE with k-means, a diagonal-covariance GMM and HDBSCAN applied directly to the same raw descriptors $\mathbf{\Phi}_k$, with the identical downstream training loop. This differs from RIPPLE-HDB, which applies HDBSCAN to the GMM-VAE posterior rather than to the descriptors. Table 13 reports the results. On CIFAR-10 the encoder is the only router that both recovers the group structure and produces calibrated soft weights, with an accuracy margin of 3.4 to 10.2 points. The MNIST columns expose the issue of reliability, since the best classical router on CIFAR-10 collapses on MNIST to a single dominant cluster, tolerated only because MNIST is near saturation, while k-means and the GMM remain poorly calibrated on both datasets. No classical method works across datasets, whereas the amortized encoder does. The structural reason is that a fitted clustering partitions the specific population it observes, whereas the GMM-VAE learns the conditional distribution of label skew given the representation over the whole simplex (Proposition D.1), which is what produces the calibrated soft weights driving Theorem 5.3 and the one-forward-pass routing of unseen clients in Table 2.

Table 13: Routing ablation at $T = 500$, seed 42. Classical clustering is applied to the raw descriptors $\mathbf{\Phi}_k$; the GMM-VAE row is the RIPPLE-STRAT reference.

	CIFAR-10			MNIST		
Routing	ARI	ℓ_1	acc.	ARI	ℓ_1	acc.
k-means on $\mathbf{\Phi}$	0.240	1.754	46.26	0.466	1.828	96.96
GMM (diagonal) on $\mathbf{\Phi}$	0.207	1.835	47.47	0.515	1.773	97.26
HDBSCAN on $\mathbf{\Phi}$	0.819	1.733	53.08	0.018	1.899	98.19
GMM-VAE (ours)	**0.905**	**0.590**	**56.49**	**0.525**	**0.910**	**98.97**

Sensitivity to the auxiliary set. The encoder is trained on synthetic clients drawn from $\mathcal{D}_{\text{aux}}$, so the assignment depends on how well the auxiliary set represents the client population. We vary only the auxiliary set along three axes, with clients always transmitting clean descriptors. Size retains 10% and 1% of $\mathcal{D}_{\text{aux}}$. Quality adds i.i.d. Gaussian pixel noise $\mathcal{N}(0, \sigma^2)$ with $\sigma \in \{0.1, 0.2, 0.4, 0.8\}$ in normalized units to the auxiliary images during encoder training only. Representativeness restricts $\mathcal{D}_{\text{aux}}$ to half of the label space, so that part of the population cannot be represented. Theorem 5.1 makes a quantitative prediction for the noise axis. Isotropic noise shifts the auxiliary covariance to $\mathbf{\Sigma} + \sigma^2\mathbf{I}$ and leaves the population eigenvectors intact, so it enters the routing bound only through the Davis-Kahan term controlled by $\sigma^2/\Delta\lambda$, the scattering transform being non-expansive and the encoder Lipschitz. Routing should stay flat while $\sigma^2/\Delta\lambda \ll 1$ and degrade as the ratio approaches one. Table 14 confirms the prediction. Routing quality is flat throughout the small-ratio regime, decays smoothly at intermediate corruption and collapses only as $\sigma^2/\Delta\lambda$ approaches one, where 94% of the clients fall into one cluster. Auxiliary size is immaterial down to a tenth of the set and breaks routing only at the extreme, consistent with the descriptor depending on $\mathcal{D}_{\text{aux}}$ only through per-class first and second moments, which concentrate at the standard rate. Accuracy is nonetheless buffered against all isotropic corruption, because degraded posteriors turn diffuse and the soft-weighted aggregation of (9) averages the routing error out, so this failure mode is bounded. The only condition that harms accuracy is the structured support mismatch, a drop of 1.7 points, where the encoder is confidently wrong rather than diffuse, with the lowest entropy of all conditions. MNIST replicates all trends, with routing surviving even $\sigma^2/\Delta\lambda = 1.55$, which shows that the bound is sufficient rather

than sharp. Server-side auxiliary data is an established assumption in federated learning (Li and Wang, 2019; Lin et al., 2020; Sattler et al., 2021; Goswami et al., 2025), and RIPPLE uses it more weakly than distillation-based methods, since $\mathcal{D}_{\text{aux}}$ never touches the task models and only trains a low-dimensional routing encoder.

Table 14: Sensitivity to the auxiliary set on CIFAR-10 at $T = 500$, seed 42, with $\Delta\lambda = 1.272$. Only the auxiliary set used to train the encoder varies; a dash marks an accuracy run not performed.

	Corruption	$\sigma^2/\Delta\lambda$	ARI	ℓ_1	entropy	acc.
A0	none (reference)	0	0.905	0.590	0.96	56.49
A1	size 10%	0	0.905	0.603	0.97	–
A2	size 1%	0	0.367	0.663	0.97	57.46
A3	noise $\sigma = 0.1$	0.008	0.890	0.564	0.92	–
A4	noise $\sigma = 0.2$	0.031	0.849	0.593	0.94	57.12
A5	noise $\sigma = 0.4$	0.126	0.626	0.671	0.87	–
A6	noise $\sigma = 0.8$	0.503	0.058	0.893	0.79	57.13
A7	half label space	0	0.594	1.002	0.53	54.79

Server-side diagnostic. The failure modes of Section 5 are visible before any round. Descriptor separability, measured by the eigengap or by a silhouette score on the received $\{\mathbf{\Phi}_k\}$, tracks the sampling term of Theorem 5.1, and the mean normalized posterior entropy of the frozen encoder, computable without labels, tracks the adequacy of the auxiliary set. In Table 14 the entropy decreases with every corruption and most sharply in the structured mismatch A7, from 0.96 at the reference to 0.53, which is precisely the condition that affects accuracy. A server can therefore compare the entropy of the received descriptors with that of synthetic clients drawn from $\mathcal{D}_{\text{aux}}$ over the same concentration range and flag a marked departure before training starts.

Descriptor noise. The transmitted descriptor encodes distributional information about the local data, and the natural mechanism to limit its leakage is client-side noise, $\mathbf{\Phi}_k \leftarrow \mathbf{\Phi}_k + \sigma_{\text{rel}}\,(\|\mathbf{\Phi}_k\|_2/\sqrt{P})\,\boldsymbol{\varepsilon}$ with $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The scale is relative to the client's own descriptor norm, so the mechanism requires no population statistics and is implemented by each client alone, and since the scattering transform is non-expansive and the encoder Lipschitz, Theorem 5.2 bounds the induced routing perturbation linearly in the noise. Table 15 reports the utility curve on both datasets. Routing and accuracy stay flat up to noise of norm comparable to the descriptor itself, with only a few borderline clients reassigned, so clients can privatize the one-shot descriptor aggressively at no accuracy cost. Calibrating this mechanism to a formal differential-privacy guarantee on the descriptor is left to future work.

Table 15: Descriptor-noise utility curve at $T = 500$, seed 42. Routing metrics are computed at every level; a dash marks an accuracy run not performed.

	CIFAR-10			MNIST		
σ_{rel}	ARI	ℓ_1	acc.	ARI	ℓ_1	acc.
0 (reference)	0.905	0.590	56.49	0.525	0.910	98.97
0.10	0.890	0.589	–	0.502	0.910	–
0.25	0.890	0.589	57.10	0.505	0.915	98.56
0.50	0.890	0.589	–	0.509	0.932	–
1.00	0.876	0.588	56.42	0.444	0.982	98.33

Convergence. We summarize the stored accuracy curves of the Table 1 runs through two quantities, with no new training. Stabilization is the first evaluation round after which the smoothed accuracy, a three-point moving average, stays within two points of its final value. Rounds to reference is the first round at which a method reaches the final accuracy of the global model, FEDAVG, and multiplied by the per-round traffic of Section E it yields the communication spent to reach that reference, which combines convergence speed with per-round cost. Table 16 reports both for CIFAR-10 and CIFAR-100, single seed. Every method stabilizes several hundred rounds before the horizon, so the accuracies of Table 1 compare converged models. The per-round traffic of RIPPLE coincides with that of FEDAVG up to the one-time descriptor exchange, whereas the selection-based clustered baselines

pay a factor of C or K per round for their assignment mechanism, so a method reaching the reference in few rounds can still be expensive end to end, as FESEM is, while RIPPLE attains the reference with the lowest communication on both datasets, below FEDAVG itself and six to nine times below the selection-based baselines.

Table 16: Convergence of the Table 1 runs, seed 42. Stabilization and rounds to reference are defined in the text; communication to reference uses the per-round accounting of Section E.

Method	stabilized by	rounds to reference	communication to reference
CIFAR-10 / CNN			
FEDAVG	640	720	23.6 GB
IFCA ($K = 2$)	630	450	81.2 GB
FESEM ($K = 8$)	690	50	90.2 GB
CFL	290	210	17.9 GB
RIPPLE-STRAT	880	310	**10.2 GB**
CIFAR-100 / ResNet-9			
FEDAVG	940	870	459.4 GB
IFCA ($K = 2$)	890	860	2497.4 GB
FESEM ($K = 10$)	580	110	3194.4 GB
CFL	730	260	356.9 GB
RIPPLE-STRAT	997	740	**390.7 GB**

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [N/A].
- [N/A] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for [N/A]).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will also be asked to include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While [Yes] is generally preferable to [No], it is perfectly acceptable to answer [No] provided a proper justification is given (e.g., error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering [No] or [N/A] is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Results are reported in the Experimental Section 6.1, while Theoretical Results are in Section 5.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

Justification: See the Conclusions.

Guidelines:

- The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: Proofs for Theorems, Propositions and Lemmas are detailed in Appendix D.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Implementation Details are provided in the Appendix G

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Yes, information is provided in Appendix G, and codes have been uploaded as Supplementary Materials.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: See experimental details in Appendix G

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Experiments have been ran on three seeds and 2σ error bars have been reported in the main paper.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Details have been discussed in Appendix G

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.

- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Yes, the paper is compliant with the Code of ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Broader Impacts are discussed in the Conclusions

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: The algorithm safeguards privacy as discussed in Appendix F

    Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Sources for software and technical assets are properly cited in the paper.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

Justification: [N/A]

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.