SMOOTH CLIENT–INSTANCE ADAPTER FIELDS FOR PERSONALIZED FEDERATED LEARNING UNDER COMPOUND HETEROGENEITY

Anonymous authors
Paper under double-blind review

ABSTRACT

Personalized federated learning (PFL) commonly binds one model to each client or selects among finite experts, creating artificial boundaries between nearby distributions and reducing a multimodal client to one identity. We introduce FEDSPLINE (Federated Smooth Personalization via Layerwise Implicit Neural Experts), a tensor-product cubic Bernstein field of low-rank adapters over learned client–instance coordinates. A distribution signature anchors each client, a bounded router refines instance queries, and sixteen nonnegative analytic weights blend shared controls while offsets and heads remain private. This explicit convex geometry gives partition-of-unity and Lipschitz properties, a cold-start decomposition, and supporting conditional shared-parameter analysis. Under source-identity-disjoint evaluation on three compound-shift benchmarks and twelve controlled baselines, FEDSPLINE has the highest observed mean accuracy (83.11%) and label-free cold-start mean (63.23%). Its 0.14-point three-seed and 0.19-point five-seed HyperAdapter gaps are descriptively unresolved ($p = 0.658$ and 0.390), while it remains Pareto-relevant through explicit geometry and slightly lower communication than the capacity-matched generator. On unseen interior coordinates, its 0.49-point advantage over that generator is nonsignificant ($p = 0.150$), and extrapolation is weaker. The evidence supports an inspectable geometry–utility trade-off, not a universally dominant PFL rule.

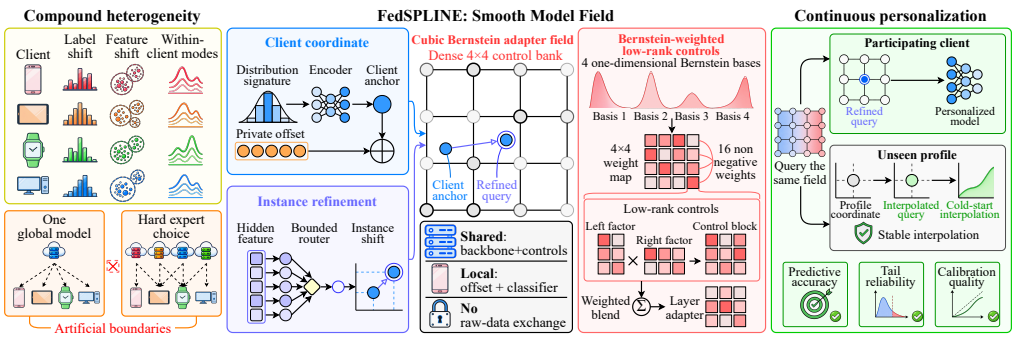

Figure 1: FEDSPLINE replaces brittle global or discrete expert selection with a smooth model family. Client distribution coordinates and bounded instance displacements query a cubic Bernstein low-rank adapter field; only shared controls are aggregated, while offsets and classifier heads remain local.

1 INTRODUCTION

Federated learning (FL) enables many clients to train collaboratively without centralizing their raw data (McMahan et al., 2017; Tan et al., 2023). Its central statistical obstacle is heterogeneity: client distributions differ in labels, covariates, conditional mechanisms, and sample size. A single global model consequently experiences conflicting local updates and uneven client utility (Li et al., 2020b; Karimireddy et al., 2020). Personalized FL addresses the latter failure by learning a client-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

specific predictor, but the representation used for "client-specific" variation determines what forms of heterogeneity the method can express.

Most PFL methods choose among three representations. Parameter separation shares a representation while keeping a local head or normalization state (Collins et al., 2021; Li et al., 2021b; Oh et al., 2022; Chen & Chao, 2022); local aggregation interpolates a global and local model (Zhang et al., 2023a); and clustered, conditional, or simplex methods select from a finite set of parameter configurations (Sattler et al., 2021; Zhang et al., 2023b; Grinwald et al., 2024). These representations are effective, yet they share a structural discontinuity: two nearly identical distributions can be assigned different models, whereas one multimodal client is compressed into one identity. Recent profile mapping (Li et al., 2026), manifold calibration (Ma et al., 2026), and unseen-client hypernetworks (Zhou et al., 2026) make distribution structure explicit, but none instantiates a shared client–instance parameter field with explicit interpolation geometry. The unresolved question is therefore not only *which model belongs to a client*, but *how a coherent family of models should vary across client and instance context*.

Our key proposal is to represent personalization as a smooth parameter field rather than a model index. As Figure 1 illustrates, FEDSPLINE maps a compact distribution signature to a two-dimensional client coordinate, adds a private bounded offset, and refines it using an instance router. Each adapted layer queries a dense tensor-product cubic Bernstein field whose 4×4 control bank stores low-rank adapters. All sixteen analytic weights are nonnegative and sum to one, so neighboring contexts change their control mixture continuously. Bending energy limits oscillatory control configurations, and a tangent-orthogonality penalty discourages personalized directions from duplicating the shared base map.

This design has three consequences. First, the Bernstein basis forms a partition of unity, so the generated adapter is a stable convex blend of finitely many communicated controls. Second, bounded derivatives translate coordinate error into bounded parameter and prediction error, yielding a cold-start stability and approximation decomposition for unseen profiles. Third, only the base network, coordinate encoder, router, and Bernstein controls are federated; client offsets and classifier heads remain local. FEDSPLINE is thus a software-only learning architecture and requires neither specialized devices nor public server data.

We make the following contributions:

- We formulate PFL as a *client–instance adapter field*: unlike a generic conditional generator, our layerwise cubic Bernstein architecture maps every query to an explicit convex combination of named controls with analytically bounded sensitivity.
- We combine compact distribution coordinates, bounded instance refinement, a dense 4×4 bank of low-rank Bernstein controls, and geometric regularization in an end-to-end federated objective.
- We establish partition-of-unity and Lipschitz guarantees, a cold-start stability/approximation bound, and a supporting nonconvex local-SGD analysis of the implemented-query shared objective, conditional on fixed private states.
- We compare against twelve controlled baselines, flag and audit recent mechanism-matched comparators, and evaluate interpolation, unseen clients, stress, communication, and a 32-client raw-image ResNet-18 protocol.

2 RELATED WORK

Personalized representations and aggregation. Optimization-based personalization regularizes local models around a global solution, as in FedProx, Ditto, and pFedMe (Li et al., 2020a; 2021a; Dinh et al., 2020), while representation methods factor the predictor into shared and private components (Collins et al., 2021; Li et al., 2021b). FedBABU and FedRoD further expose the distinct roles of body and head parameters (Oh et al., 2022; Chen & Chao, 2022), and adaptive local aggregation learns how much downloaded information each parameter should retain (Zhang et al., 2023a). These approaches provide strong client models, but their personalized state is still attached to a discrete client identifier. FEDSPLINE retains a local head while placing representation adaptation on a shared continuous field.

Conditional personalization and connected model families. Hypernetworks generate client parameters from learned embeddings (Shamsian et al., 2021), and FedL2P predicts client-specific per-

sonalization hyperparameters from local statistics (Lee et al., 2023). FedCP uses sample-conditioned feature separation (Zhang et al., 2023b); FLoCo learns a low-loss solution simplex (Grinwald et al., 2024); and pFedClub searches and reassembles functional blocks (Wang et al., 2024b). Outside FL, Bézier mode connectivity constructs tensor-product Bernstein parameter surfaces connecting finitely many independently trained modes (Ren et al., 2025). FEDSPLINE does not claim the Bernstein primitive itself: it jointly learns a communicated low-rank adapter field inside federated optimization, indexes it by client–instance queries, separates private from shared state, and analyzes zero-step cold-start stability. Thus every personalized query is an inspectable convex combination with bounded derivatives rather than an unconstrained generator output or a fixed mode-connecting surface.

Distribution structure under federated shift. Feature-distribution adaptation (McLaughlin & Su, 2024), cross-round aggregation (Wang et al., 2024a), and layer-wise shrinking (Shi et al., 2025) mitigate heterogeneous drift. Semi-supervised representation augmentation can additionally transfer cross-institutional knowledge from aligned to non-aligned samples in vertical healthcare FL (Yin et al., 2024); this addresses missing supervision and sample alignment rather than continuous horizontal personalization. Closely related recent directions include profile-based model assignment in Feroma (Li et al., 2026), on-manifold sample calibration in FedMC (Ma et al., 2026), and hypernetwork deployment to non-participating clients (Zhou et al., 2026). Our architectural distinction is that distribution structure indexes a smooth *parameter field*, while instance features refine that index inside each client. This targets compound inter- and intra-client variation without public data or hardware assumptions.

3 PROBLEM FORMULATION

Let N clients hold private distributions P_i and datasets $\mathcal{D}_i = \{(x, y)\}$. The goal is to learn personalized predictors f_i that minimize

$$\min_{\Theta,\{\psi_i\}} F(\Theta, \psi) = \sum_{i=1}^{N} p_i \, \mathbb{E}_{(x,y)\sim P_i} \ell(f(x; \Theta, \psi_i), y) \,, \tag{1}$$

where Θ is communicated and ψ_i remains local. We study *compound heterogeneity*: both $P_i(y)$ and $P_i(x \mid y)$ vary across clients, and P_i may itself contain multiple modes. In the standard PFL protocol, each participating client computes a fixed-dimensional signature $s_i = [\widehat{P}_i(y), \mu_x, \sigma_x, q_{.1}, q_{.9}, \sigma_{\mu(x)}, \mathrm{MAD}(x)]$. The held-out protocol in Section 5.3 instead retrains every signature-dependent method using a 16-dimensional input-only summary for *both* participating and unseen clients; no component is computed from y. The server receives model updates but never examples. We do not claim cryptographic privacy for s_i; signatures can be combined with secure aggregation or differential privacy, which is orthogonal to this work.

4 FEDSPLINE

4.1 ARCHITECTURE OVERVIEW

Figure 2 maps the forward and federated paths. Four modules answer four design questions: the coordinate encoder locates a distribution; the instance router represents within-client context; the dense Bernstein field generates a smooth low-rank adapter; and geometric regularization controls field variation. The private classifier absorbs label-prior differences, while the field adapts representations to covariate and conditional shifts.

4.2 CLIENT–INSTANCE COORDINATES

One-hot client IDs memorize participants and cannot interpolate to a new client. We instead encode observable distribution statistics:

$$z_i = \Pi_{[0,1]^2}(q_\phi(s_i) + \rho \tanh \delta_i) \,, \tag{2}$$

where $q_\phi = \mathrm{sigmoid} \circ g_\phi \in (0, 1)^2$ is the client anchor produced by a two-layer MLP, $\delta_i \in \mathbb{R}^2$ is private, ρ is fixed, and Π clips to the unit square. The signature gives a cold-start coordinate; the offset corrects finite-sample signature error for participating clients.

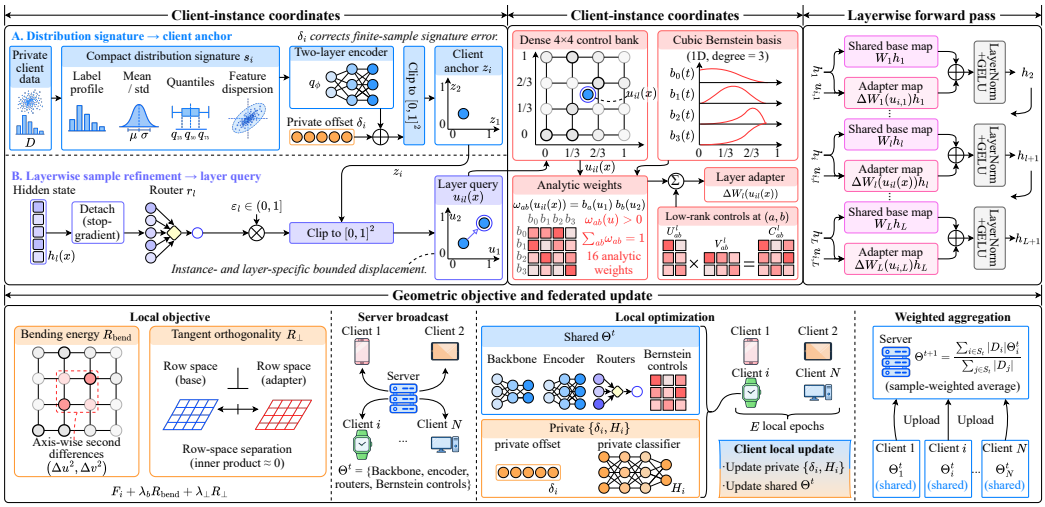

Figure 2: The coupled components of FEDSPLINE. Continuous client–instance coordinates query a tensor-product cubic field of low-rank controls, producing adapters that act as layerwise implicit experts. The backbone, encoder, routers, and controls are federated; client offsets and classifier heads remain local.

A single coordinate is still insufficient when a client mixes contexts. At adapted layer l, a lightweight router produces

$$u_{il}(x) = \Pi_{[0,1]^2}\left(z_i + \epsilon_l \tanh r_l(h_l(x))\right). \tag{3}$$

The learned scalar $\epsilon_l \in (0,1)$ bounds the displacement. Detaching the router input when computing the displacement prevents a shortcut in which the representation changes solely to move coordinates. This module provides sample specificity without selecting a hard expert.

4.3 TENSOR-PRODUCT CUBIC BERNSTEIN ADAPTER FIELD

Let $b_0(t) = (1-t)^3$, $b_1(t) = 3t(1-t)^2$, $b_2(t) = 3t^2(1-t)$, and $b_3(t) = t^3$. For a query $u = (u_1, u_2)$, control weight $\omega_{ab}(u) = b_a(u_1)b_b(u_2)$ is nonnegative and the 16 weights sum to one. Each control stores low-rank factors $U^l_{ab} \in \mathbb{R}^{d_{l+1}\times r}$ and $V^l_{ab} \in \mathbb{R}^{r\times d_l}$, yielding

$$\Delta W_l(u) = \frac{1}{\sqrt{r}} \sum_{a,b=0}^{3} \omega_{ab}(u) U^l_{ab} V^l_{ab}, \qquad h_{l+1} = \mathrm{GELU}(\mathrm{LN}\left[(W_l + \Delta W_l(u))h_l\right]). \tag{4}$$

The base W_l learns invariant structure. In contrast to an unconstrained generator, Equation 4 is geometrically interpretable and cannot jump between neighboring coordinates. Low rank keeps field capacity and communication bounded.

4.4 GEOMETRIC OBJECTIVE AND FEDERATED UPDATES

For either factor lattice $X \in \{U^l, V^l\}$, define its factor-level bending energy

$$\begin{aligned} \mathcal{B}(X) &= \sum_{a=1}^{2}\sum_{b=0}^{3} \|X_{a+1,b} - 2X_{a,b} + X_{a-1,b}\|_F^2 \\ &\quad + \sum_{a=0}^{3}\sum_{b=1}^{2} \|X_{a,b+1} - 2X_{a,b} + X_{a,b-1}\|_F^2, \\ R_{\text{bend}} &= \sum_l \left[\mathcal{B}(U^l) + \mathcal{B}(V^l)\right]. \end{aligned} \tag{5}$$

Thus the objective exactly matches the implementation, which regularizes the two factors separately. We reserve $C^l_{ab} = U^l_{ab}V^l_{ab}/\sqrt{r}$ for the product control matrix used in the analysis. We also penalize

Algorithm 1 Federated training of FEDSPLINE

```
Initialize shared base, encoder, routers, and Bernstein controls Θ^0
Initialize private (δ_i, H_i) for each participating client
for t = 0, ..., T − 1 do
    Sample clients S_t and broadcast Θ^t
    for all i ∈ S_t in parallel do
        Compute s_i; form z_i by Eq. 2
        For each batch, query layer fields by Eqs. (3)–(4)
        Update shared Θ_i and private (δ_i, H_i) for E local epochs
        Upload only Θ_i
    end for
    Θ^{t+1} ← Σ_{i∈S_t} |D_i| Θ_i / Σ_{j∈S_t} |D_j|
end for
```

squared cosine overlap between normalized rows of W_l and V_{ab}^l, denoted $R_\perp$. Client i minimizes $F_i + \lambda_b R_{\text{bend}} + \lambda_\perp R_\perp$. At each round, participating clients receive Θ, optimize locally, and return shared parameters. The server performs sample-weighted averaging. Heads and offsets never leave clients. Algorithm 1 gives the complete procedure.

Guarantees. The analytic weights form a partition of unity, making the field a convex blend that is Lipschitz in its coordinate when adjacent controls have bounded variation. For cold start, we decompose error into coordinate estimation, best approximation within the fixed Bernstein family, and optimization/estimation terms; no mesh-refinement rate is assumed. A supporting local-SGD analysis covers the implemented-query shared objective conditional on fixed private states; approximation to an ideal latent-query objective is handled separately by the cold-start result. Formal statements and derivations are deferred to Appendix A.

5 EXPERIMENTS

5.1 PROTOCOL

Our scope is controlled compound-heterogeneity PFL: three image-classification simulations with label and feature shift, complemented by source-disjoint raw-image validation in Section 5.6. **MNIST-C (constructed)** is our compound-shift simulation built from the official MNIST source pools, not a pre-existing corruption benchmark. We preserve the official 60,000/10,000 train/test identity split, sample 12 client label distributions from Dirichlet(0.3), and then apply client-specific rotation, blur, contrast, brightness, and noise. **CIFAR-10-C** uses six corruption episodes represented by fixed ResNet-18 features (Hendrycks & Dietterich, 2019); **Office-Home-10** uses ten classes from four natural domains with fixed ResNet-50 features (Venkateswara et al., 2017). Both add label skew. For every cached-feature benchmark, source identities are partitioned into disjoint training and test pools before client resampling; replacement sampling occurs only within the corresponding pool, and audit records verify zero cross-side identity overlap. Each client has at most 420 training and 180 test examples. Unless stated otherwise, we use 12 clients, 75% participation, 24 rounds, one local epoch, and seeds 1, 7, and 19.

We include twelve baselines: FedAvg and FedProx (McMahan et al., 2017; Li et al., 2020a); Ditto and pFedMe (Li et al., 2021a; Dinh et al., 2020); FedRep and FedRoD (Collins et al., 2021; Chen & Chao, 2022); FedALA, FedCP, and FLoCo (Zhang et al., 2023a;b; Grinwald et al., 2024); and **FedLWS (matched)** and **Feroma (matched)** (Shi et al., 2025; Li et al., 2026). The latter two are mechanism-matched common-harness comparators, not official system reproductions; hereafter the shorter figure and table labels "FedLWS" and "Feroma" always denote these matched implementations. Appendix D audits provenance. **HyperAdapter**, motivated by pFedHN (Shamsian et al., 2021), replaces analytic weights with a capacity-matched MLP while retaining coordinates and private state. All share data order, steps, and backbones. FedMC is omitted because its calibrator lacks a faithful prompt-free operator.

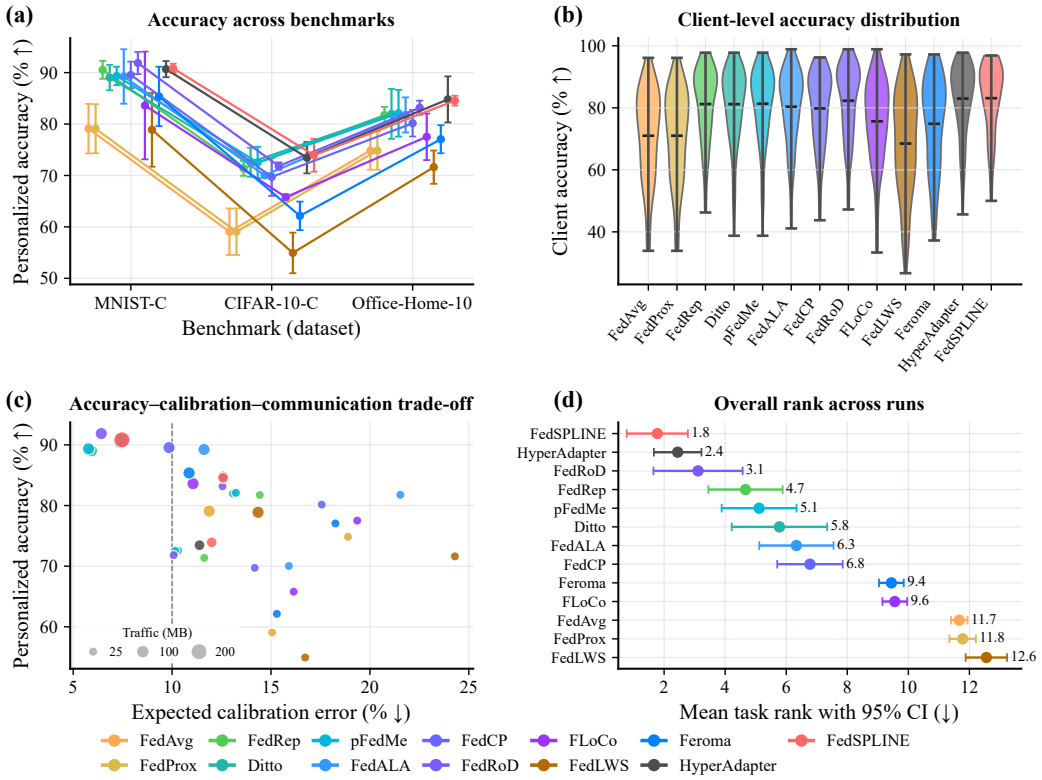

Figure 3: Main evaluation over twelve baselines. FedLWS and Feroma denote matched-harness reimplementations, not official-system reproductions. (a) Accuracy with two-sided 95% Student-t intervals over three seeds; (b) client distributions; (c) accuracy–calibration trade-off with marker area encoding cumulative communication (MB); and (d) mean dataset–seed rank with a descriptive pooled Student-t interval.

5.2 Main Results

Figure 3 and Table 1 compare all thirteen methods. FedSpline has the highest controlled three-task mean (83.11%), followed by HyperAdapter (82.98%) and FedRoD (82.29%). Its nine dataset–seed differences from FedRoD average $+0.82$ points with 6/9 wins; the matched-generator gap is only 0.14 points ($p = 0.658$). Because three seeds are nested within three tasks, pooled inference is descriptive. The matched Feroma and FedLWS comparators reach 74.85% and 68.48%. FedSpline leads on CIFAR-10-C (73.94%), HyperAdapter leads on Office-Home-10 (84.79% versus 84.55%), and FedRoD leads on MNIST-C (91.87% versus 90.85%). A five-seed matched comparison likewise gives a nonsignificant $+0.19 \pm 0.46$ points over HyperAdapter ($p = 0.390$; 10/15 wins). The primary claim is therefore a competitive, auditable geometry–utility trade-off, not generator superiority.

Across the main tasks, FedSpline averages 65.09% worst-client accuracy and 10.68% ECE, versus 65.92% and 10.46% for HyperAdapter and 65.85% and 9.68% for FedRoD. Communicating sixteen controls raises mean traffic from 43.95–45.02 MB for compact baselines to 102.88 MB, slightly below HyperAdapter's 103.99 MB. Thus FedSpline is Pareto-relevant through the best mean accuracy and cold-start accuracy plus explicit geometry, but it is not best on calibration, worst-client accuracy, or communication in isolation.

5.3 Held-out-client cold start

We reserve four of sixteen clients from every dataset–seed split. They never participate in optimization. To make the information boundary genuinely label-free, we discard $\widehat{P}_i(y)$ and retrain all profile-dependent methods using the same 16 input-only statistics for training and held-out clients: global

Table 1: Personalized accuracy (%, mean±SD over three seeds). Bold and underline denote best and second best. Averages are computed from unrounded dataset means; displayed dataset entries are rounded independently.

Method	MNIST-C	CIFAR-10-C	Office-Home-10	Average
FedAvg	79.10±1.94	59.06±1.82	74.84±1.50	71.00
FedProx	79.09±1.92	59.06±1.82	74.84±1.50	71.00
FedRep	90.54±0.70	71.37±0.58	81.74±0.65	81.22
Ditto	89.04±1.00	72.56±1.13	81.98±1.95	81.19
pFedMe	89.35±0.72	72.53±1.23	82.10±1.83	81.33
FedALA	89.24±2.13	70.03±0.16	81.77±1.38	80.35
FedCP	89.55±1.04	69.72±1.49	80.16±1.04	79.81
FedRoD	**91.87±0.87**	71.82±0.29	83.18±0.55	82.29
FLoCo	83.61±4.22	65.79±0.21	77.52±1.82	75.64
FedLWS (matched)	78.89±2.89	54.94±1.59	71.61±1.30	68.48
Feroma (matched)	85.37±2.33	62.15±1.12	77.05±1.10	74.85
HyperAdapter	90.68±0.63	<u>73.46±1.23</u>	**84.79±1.80**	<u>82.98</u>
FEDSPLINE	<u>90.85±0.36</u>	**73.94±1.29**	<u>84.55±0.38</u>	**83.11**

quantiles and moments, median absolute deviation, and across-sample mean, dispersion, and norm summaries. The signature routine does not accept labels. At test time no gradient step, labeled example, or held-out-tuned rule is allowed. We additionally include **UnseenHN (HFZ-inspired)**, a mechanism-matched comparator inspired by the client-level hypernetwork for non-participating clients (Zhou et al., 2026); it maps the input-only profile directly to low-rank adapter weights without a client ID, private offset, or instance query and is not an official-system reproduction. Appendix B specifies the deterministic fallback of every method.

Held-out mean accuracy is 63.23% for FEDSPLINE, 62.73% for HyperAdapter, 62.11% for the HFZ-inspired UnseenHN, and 62.11% for FedRoD. Descriptive pooled gaps over nine dataset–seed units are respectively 0.50 ± 1.67, 1.13 ± 4.45, and 1.12 ± 5.48 points; all overlap zero. This supports label-free operability, not universal dominance. UnseenHN has the best ECE among these four (16.03%); FEDSPLINE gives 19.26% and HyperAdapter 19.58%. Appendix C reports all methods and profile-distance sensitivity.

5.4 SCALE, COMMUNICATION, AND FAILURE BOUNDARY

Doubling the Office-Home domain-derived simulation to 24 clients and using 50% participation gives 85.63±2.00% for FEDSPLINE, 86.05±2.05% for HyperAdapter, and 84.67±2.30% for FedRoD. The appendix reports this controlled simulation, communication trajectories, and the rank sweep: rank two reaches 73.40% while reducing 20-round traffic from 46.69 to 32.63 MB relative to rank four (73.94%).

5.5 STRESS ROBUSTNESS

Figure 5 covers all thirteen methods, freshly rerun on MNIST-C under the official source-disjoint train/test pools with zero audited identity overlap. FEDSPLINE has the best mean stress rank (2.22 versus 2.44 for FedRoD), ranks first or second in six of nine settings, and leads at medium/severe feature shift. Participation remains the boundary: at 40%, FedRoD and FedRep reach 83.06% and 82.31%, versus 81.75% for FEDSPLINE. Panel (f) reports the signed margin over the strongest alternative; the complete rank table is in the appendix.

5.6 TARGETED INTERPOLATION AND END-TO-END VALIDATION

Figure 6(a–b) evaluates 25 unseen coordinates inside the convex hull of eight sparse training coordinates and 18 outside it over five seeds. With oracle coordinates, FEDSPLINE obtains 87.52% interior accuracy versus 87.03% for HyperAdapter and 88.47% for Feroma; its matched-generator gap is nonsignificant ($p = 0.150$). Learned queries retain 87.51% interior accuracy. Outside the training

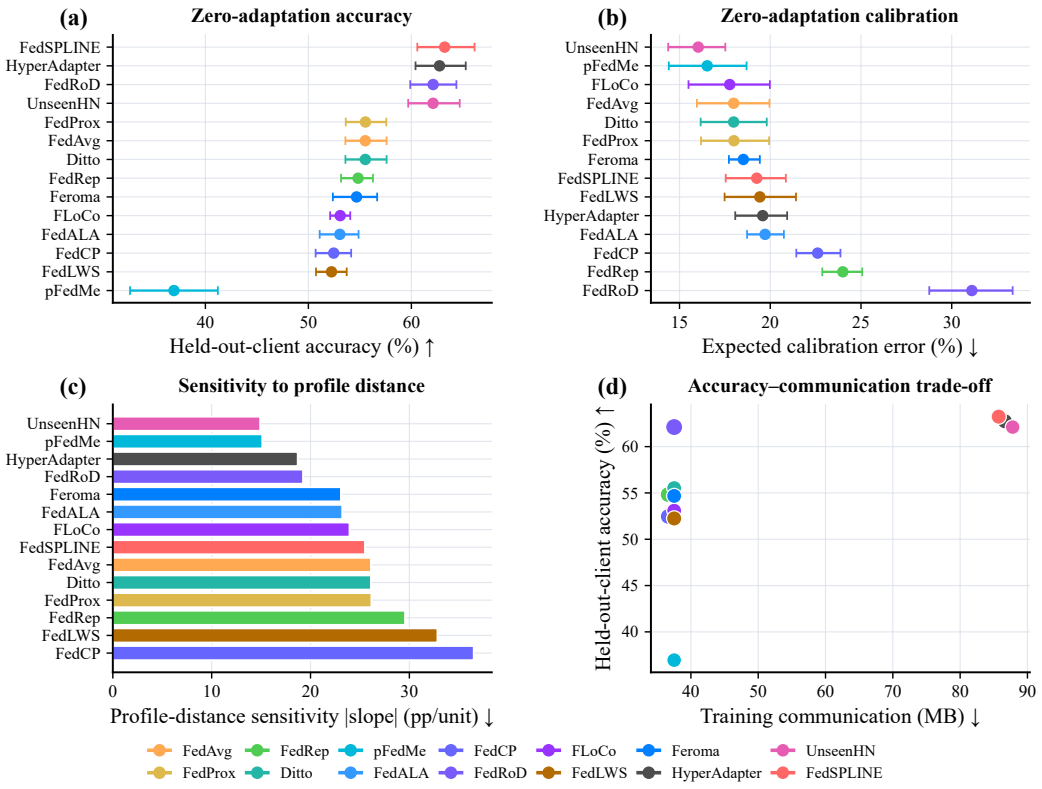

Figure 4: Label-free task-level cold start over 36 held-out clients per method. (a) Accuracy and (b) ECE with 95% dataset-stratified cluster-bootstrap intervals; the independent units are nine dataset–seed training runs, each averaging four held-out clients. (c) Run-clustered profile-distance sensitivity and (d) accuracy–communication trade-off. All panels include thirteen PFL methods plus the HFZ-inspired UnseenHN comparator.

convex hull, accuracy drops to 80.36% with oracle coordinates and 83.29% with learned queries, so smooth interpolation is not an out-of-support guarantee.

Figure 6(c–d) uses source-disjoint raw Office-Home-10, 32 clients, and a jointly trained ResNet-18: 1,944 training and 836 test identities have zero overlap. FEDSPLINE has the highest observed mean (50.03%), worst-client accuracy (23.44%), and lowest ECE (16.02%); its mean gaps to HyperAdapter (48.77%) and FedRoD (48.61%) are nonsignificant ($p = 0.441, 0.596$). This supports structured reuse under representation learning, not significant superiority.

6 DISCUSSION

Evidence, boundary, and deployment regime. FEDSPLINE contributes an explicit convex interpolation geometry with competitive prediction and auditable smoothness. Across mean and worst-client accuracy, calibration, cold start, natural-domain scaling, and communication, no method dominates: FEDSPLINE is Pareto-relevant among the reported point estimates and adds an explicit analytic certificate absent from the compared baselines, while FedRoD and HyperAdapter lead individual metrics. We therefore avoid a post-hoc scalar "overall" score; the evidence supports controlled interpolation, with unresolved generator gaps and weaker out-of-support behavior.

The field suits related distributions when smooth reuse outweighs traffic. Low participation or strict bandwidth favors simpler methods; matched baselines and unprotected signatures remain limitations. Sparse activation, adaptive resolution, naturally arriving clients, and protected profiles are key extensions.

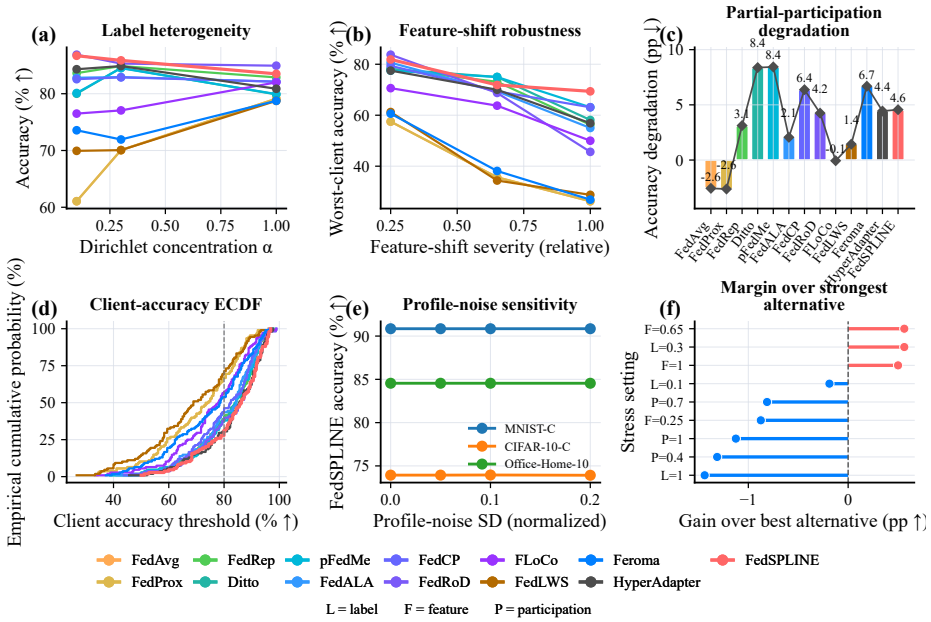

Figure 5: Robustness over twelve baselines. (a) Label heterogeneity; (b) feature-shift tail accuracy; (c) participation degradation; (d) client-accuracy ECDF; (e) profile-noise sensitivity; and (f) FedSPLINE's margin over the best alternative in each stress setting.

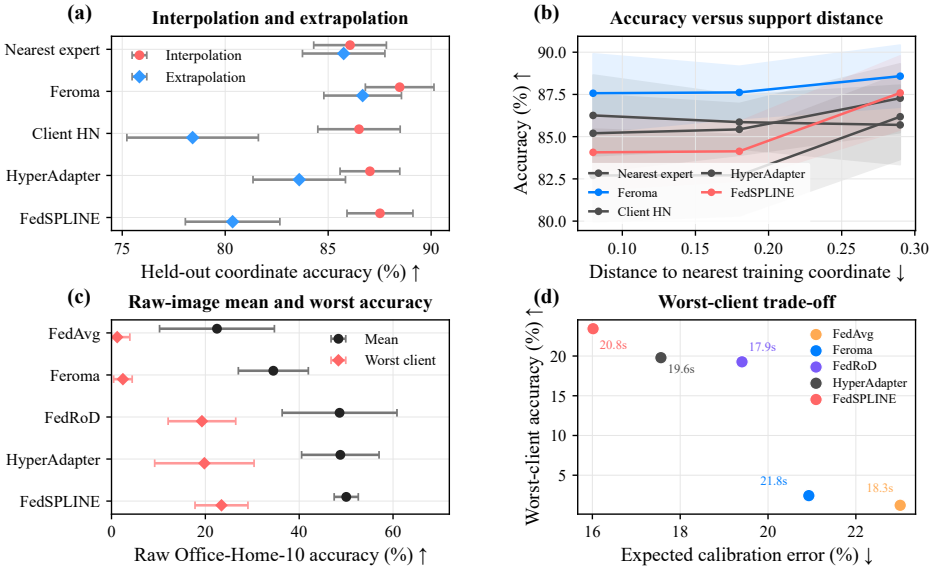

Figure 6: Known-coordinate and source-disjoint raw-image validation. (a) Oracle interpolation/extrapolation accuracy; (b) accuracy versus support distance; (c) raw Office-Home-10 mean/worst accuracy; and (d) worst accuracy–calibration trade-off (area: traffic; labels: runtime in seconds).

7 Conclusion and Limitations

FedSpline recasts PFL as a continuous client–instance parameter field with auditable interpolation, sensitivity, and curvature. It has the highest source-disjoint controlled and cold-start means, but matched gaps remain unresolved, HyperAdapter leads natural-domain scaling, and extrapolation is weaker. Thus its aggregate contribution is a competitive and auditable geometry–utility trade-off—*controllable reuse*, not every-metric superiority.

AI USE STATEMENT

Generative AI tools were used to assist with generating and refining portions of the experimental code and with checking and revising intermediate steps in the mathematical proofs. All AI-assisted code was manually reviewed and tested, and all mathematical derivations and proofs were independently verified by the authors. Generative AI was also used for language polishing and improving the clarity of the manuscript; the research problem, proposed method, experimental design, and interpretation of results were developed by the authors. The authors take full responsibility for the final content of the paper, including its code, theoretical claims, experimental results, and text.

ETHICS STATEMENT

This work uses public benchmark datasets and software-only federated simulations; it involves no human-subject study or release of personal data. Although raw samples, private offsets, and classifier heads remain local in the evaluated protocol, shared summaries and gradients do not constitute a formal privacy guarantee and may reveal attributes. Deployment with sensitive data therefore requires independent assessment and appropriate protections such as secure aggregation or differential privacy.

REPRODUCIBILITY STATEMENT

Sections 4 and 5 specify the method and evaluation protocol. Appendix A gives assumptions and complete proofs, while Appendices B–D provide extended results, sensitivity analyses, implementation details, baseline mechanics, and provenance checks. The anonymous code supplement contains the implementation, environment specification, data-acquisition instructions, invariant tests, experiment drivers, analysis scripts, and lightweight numerical audit tables; datasets, checkpoints, and figures are excluded from the archive.

REFERENCES

Hong-You Chen and Wei-Lun Chao. On bridging generic and personalized federated learning for image classification. In *ICLR*, 2022.

Liam Collins, Hamed Hassani, Aryan Mokhtari, and Sanjay Shakkottai. Exploiting shared representations for personalized federated learning. In *ICML*, pp. 2089–2099, 2021.

Carl de Boor. *A Practical Guide to Splines*, volume 27 of *Applied Mathematical Sciences*. Springer, 1978.

Canh T. Dinh, Nguyen H. Tran, and Tuan Dung Nguyen. Personalized federated learning with moreau envelopes. In *NeurIPS*, volume 33, pp. 21394–21405, 2020.

Dennis Grinwald, Philipp Wiesner, and Shinichi Nakajima. Federated learning over connected modes. In *NeurIPS*, volume 37, 2024. doi: 10.52202/079017-2705.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *ICLR*, 2019.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. SCAFFOLD: Stochastic controlled averaging for federated learning. In *ICML*, pp. 5132–5143, 2020.

Royson Lee, Minyoung Kim, Da Li, Xinchi Qiu, Timothy Hospedales, Ferenc Huszár, and Nicholas D. Lane. FedL2P: Federated learning to personalize. In *NeurIPS*, volume 36, 2023. doi: 10.52202/075280-0653.

Mohan Li, Dario Fenoglio, Martin Gjoreski, and Marc Langheinrich. Federated learning with profile mapping under distribution shifts and drifts. In *ICLR*, 2026.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. In *Proceedings of Machine Learning and Systems*, volume 2, pp. 429–450, 2020a.

Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Ditto: Fair and robust federated learning through personalization. In *International Conference on Machine Learning*, pp. 6357–6368, 2021a.

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

Xiaoxiao Li, Meirui Jiang, Xiaofei Zhang, Michael Kamp, and Qi Dou. FedBN: Federated learning on Non-IID features via local batch normalization. In *ICLR*, 2021b.

Yanbiao Ma, Wei Dai, Gaoyang Jiang, Wanyi Chen, Chenyue Zhou, Yiwei Zhang, Fei Luo, Junhao Wang, and Andi Zhang. FedMC: Federated manifold calibration. In *ICLR*, 2026.

Connor J. McLaughlin and Lili Su. Personalized federated learning via feature distribution adaptation. In *NeurIPS*, volume 37, 2024. doi: 10.52202/079017-2451.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *AISTATS*, pp. 1273–1282, 2017.

Jaehoon Oh, SangMook Kim, and Se-Young Yun. FedBABU: Toward enhanced representation for federated image classification. In *ICLR*, 2022.

Jie Ren, Pin-Yu Chen, and Ren Wang. Revisiting mode connectivity in neural networks with bezier surface. In *ICLR*, 2025.

Felix Sattler, Klaus-Robert Müller, and Wojciech Samek. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. *IEEE Transactions on Neural Networks and Learning Systems*, 32(8):3710–3722, 2021. doi: 10.1109/TNNLS.2020.3015958.

Aviv Shamsian, Aviv Navon, Ethan Fetaya, and Gal Chechik. Personalized federated learning using hypernetworks. In *ICML*, pp. 9489–9502, 2021.

Changlong Shi, Jinmeng Li, He Zhao, Dandan Guo, and Yi Chang. FedLWS: Federated learning with adaptive layer-wise weight shrinking. In *ICLR*, 2025.

Alysa Ziying Tan, Han Yu, Lizhen Cui, and Qiang Yang. Towards personalized federated learning. *IEEE Transactions on Neural Networks and Learning Systems*, 34(12):9587–9603, 2023. doi: 10.1109/TNNLS.2022.3160699.

Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *CVPR*, pp. 5018–5027, 2017.

Haozhao Wang, Haoran Xu, Yichen Li, Yuan Xu, Ruixuan Li, and Tianwei Zhang. FedCDA: Federated learning with cross-rounds divergence-aware aggregation. In *ICLR*, 2024a.

Jiaqi Wang, Qi Li, Lingjuan Lyu, and Fenglong Ma. pFedClub: Controllable heterogeneous model aggregation for personalized federated learning. In *NeurIPS*, volume 37, 2024b. doi: 10.52202/079017-1733.

Zilong Yin, Haoyu Wang, Bin Chen, Xin Zhang, Xiaogang Lin, Hangling Sun, Anji Li, and Chenyu Zhou. Federated semi-supervised representation augmentation with cross-institutional knowledge transfer for healthcare collaboration. *Knowledge-Based Systems*, 300:112208, 2024. doi: 10.1016/j.knosys.2024.112208.

Jianqing Zhang, Yang Hua, Hao Wang, Tao Song, Zhengui Xue, Ruhui Ma, and Haibing Guan. FedALA: Adaptive local aggregation for personalized federated learning. In *AAAI*, pp. 11237–11244, 2023a. doi: 10.1609/AAAI.V37I9.26330.

Jianqing Zhang, Yang Hua, Hao Wang, Tao Song, Zhengui Xue, Ruhui Ma, and Haibing Guan. FedCP: Separating feature information for personalized federated learning via conditional policy. In *KDD*, pp. 3249–3261, 2023b. doi: 10.1145/3580305.3599345.

Yuhao Zhou, Jindi Lv, Yuxin Tian, Dan Si, Qing Ye, and Jiancheng Lv. Deploying models to non-participating clients in federated learning without fine-tuning: A hypernetwork-based approach. In *ICLR*, 2026.

A Theoretical Results and Detailed Proofs

A.1 Objects, assumptions, and scope

Write $C^l_{ab} = U^l_{ab} V^l_{ab}/\sqrt{r}$ and $A_l(u) = \sum_{a,b=0}^{3} \omega_{ab}(u) C^l_{ab}$. Let

$$\Gamma_l = \max_{a,b} \left\{ \left\| C^l_{a+1,b} - C^l_{a,b} \right\|_2, \left\| C^l_{a,b+1} - C^l_{a,b} \right\|_2 \right\}, \tag{6}$$

where invalid boundary differences are omitted. The theory concerns the implemented fixed degree-three, 4×4 family; it does not invoke grid refinement or claim a universal approximation rate. Throughout the stability results, $\|\cdot\|_2$ denotes the matrix spectral norm for adapter/control matrices and the Euclidean norm for vectors.

Scope of the optimization result. The forward-field and cold-start results allow the private offset through its effect on the query. Theorem 3 is narrower: for a fixed collection of client-private variables $\psi = \{\psi_i\}$, define $F_i^{\psi}(\Theta) := F_i(\Theta, \psi_i)$ and suppress the superscript inside the theorem. The result analyzes only this shared-parameter subproblem. It conditions on the private states and does not claim convergence of Algorithm 1's full alternating dynamics in $(\Theta, \{\psi_i\})$. All objectives and shared gradients in Assumption 1 and Section A.6 have this conditional meaning.

Assumption 1 (Optimization)**.** *Each local objective $F_i(\Theta)$, including the two smooth geometric penalties, is L-smooth and lower bounded by F^*. A stochastic gradient $g_i(\Theta; \xi)$ is conditionally unbiased, conditionally independent across sampled clients and local draws, and satisfies $\mathbb{E}\|g_i - \nabla F_i\|^2 \le \sigma^2$. Client gradient dissimilarity obeys $\sum_i p_i \|\nabla F_i - \nabla F\|^2 \le \zeta^2$.*

Assumption 2 (Reached-domain regularity)**.** *Inputs and hidden states are bounded by H_l. GELU and LayerNorm are Lipschitz on the compact set reached by training. Effective layer maps have finite constants κ_l, control products have finite norm M_l, and their adjacent variation Γ_l in Equation 6 is finite.*

Assumption 3 (Coordinate observation)**.** *For a target client–instance context, the ideal coordinate is u, the coordinate available to the learned field is $\widehat{u}$, and $\|\widehat{u} - u\|_2 \le \varepsilon_s$. This term may contain signature estimation, encoder estimation, and bounded router error; no independence between its coordinates is required.*

Assumption 3 is used only to compare an implemented query with an ideal latent context in the forward-stability and cold-start results. The optimization theorem instead defines its objective using the implemented encoder, router, and control field and therefore introduces no separate query- or field-estimation bias.

A.2 Bernstein interpolation and bounded queries

Lemma 1 (Basis identities)**.** *For $t \in [0,1]$, $b_a(t) \ge 0$, $\sum_{a=0}^{3} b_a(t) = 1$, and*

$$\frac{d}{dt} \sum_{a=0}^{3} b_a(t) C_a = 3 \sum_{a=0}^{2} b^{(2)}_a(t)(C_{a+1} - C_a), \tag{7}$$

where the degree-two bases $b^{(2)}_a$ are nonnegative and sum to one.

Proof. Nonnegativity follows from the monomial form. The binomial theorem gives $\sum_a b_a(t) = ((1-t)+t)^3 = 1$. Differentiating the four terms and collecting adjacent coefficients gives Equation 7; this is the standard Bernstein derivative identity (de Boor, 1978). □

Proposition 1 (Stable interpolation and capacity)**.** *For every $u \in [0,1]^2$, the sixteen weights form a convex partition of unity and*

$$\|A_l(u)\|_2 \le \max_{a,b} \left\|C^l_{ab}\right\|_2 \le M_l. \tag{8}$$

For L_a adapted layers, grid width G, and rank r, field storage is $G^2 r \sum_{l=1}^{L_a}(d_l + d_{l+1})$ scalars.

Proof. The tensor product of two nonnegative partitions is nonnegative and sums to $(\sum_a b_a(u_1))(\sum_b b_b(u_2)) = 1$. Convexity of the spectral norm yields Equation 8. Each of the G^2 controls stores an $r \times d_l$ down factor and a $d_{l+1} \times r$ up factor, giving the stated count. This is the quantity used by the communication calculation in Table 9. □

Proposition 2 (Query containment)**.** *The sigmoid output ensures $q_\phi(s_i) \in (0,1)^2$. Projection onto $[0,1]^2$ is nonexpansive, so*

$$\|z_i - q_\phi(s_i)\|_2 \le \sqrt{2}\rho, \qquad \|u_{il}(x) - z_i\|_2 \le \sqrt{2}\epsilon_l. \tag{9}$$

If q_ϕ is L_q-Lipschitz and $\|\widehat{s}_i - s_i\| \le \varepsilon_{\mathrm{sig}}$, then a held-out query with zero private offset differs from its population-anchor query by at most $L_q\varepsilon_{\mathrm{sig}}$ before router error is added.

Proof. Coordinatewise $|\tanh(\cdot)| \le 1$ gives an unprojected two-dimensional displacement of norm at most $\sqrt{2}\rho$ or $\sqrt{2}\epsilon_l$. Because $q_\phi(s_i)$ and z_i already lie in the projection set, nonexpansiveness gives $\|\Pi(q_\phi(s_i) + a) - \Pi(q_\phi(s_i))\|_2 \le \|a\|_2$ and the first bound; applying the same argument around z_i gives the second. The last statement follows directly from Lipschitzness of q_ϕ. This proposition formalizes the coordinate and bounded-query constructions in Equations 2–3. □

A.3 Adapter, prediction, and loss stability

Theorem 1 (Lipschitz personalization)**.** *Under Assumption 2, for any $u, v \in [0,1]^2$,*

$$\|A_l(u) - A_l(v)\|_2 \le 3\sqrt{2}\Gamma_l \|u - v\|_2 . \tag{10}$$

Moreover, a network with a common query at its adapted layers satisfies

$$\|f(x;u) - f(x;v)\|_2 \le L_f(x)\|u - v\|_2 , \quad L_f(x) = \sum_{l=1}^{L_a} 3\sqrt{2}\Gamma_l H_l \prod_{j=l+1}^{L_a} \kappa_j. \tag{11}$$

The same result holds for layer-specific queries after summing $L_{f,l}\|u_l - v_l\|$ over adapted layers.

Proof. Hold u_2 fixed. Lemma 1 and convexity give $\|\partial A_l/\partial u_1\|_2 \le 3\Gamma_l$; the vertical derivative has the same bound. Cauchy–Schwarz therefore bounds the Jacobian norm by $3\sqrt{2}\Gamma_l$, and integration along the segment between u and v proves Equation 10. For Equation 11, insert intermediate networks in which one adapted layer at a time changes from query u to v. At layer l, Equation 10 multiplies a hidden state of norm at most H_l; all later maps contribute $\prod_{j>l}\kappa_j$. Summing the telescoping differences proves the claim. □

Proposition 3 (Prediction and risk stability)**.** *If the per-example loss is G_ℓ-Lipschitz in logits on the reached domain, then*

$$|\ell(f(x;u),y) - \ell(f(x;v),y)| \le G_\ell L_f(x)\|u - v\|_2 . \tag{12}$$

If logits are bounded, softmax probabilities are bounded away from zero and the Jensen–Shannon response is upper bounded by a finite constant times $L_f(x)^2\|u - v\|_2^2$.

Proof. The first statement composes the loss Lipschitz condition with Theorem 1. Softmax has a bounded Jacobian; on a compact logit set the logarithms in Jensen–Shannon divergence also have bounded derivatives. A second-order expansion around equal predictions then yields the quadratic bound. Figure 7(d) measures this predicted response rather than assuming that a nonconstant adapter necessarily changes predictions. □

Figure 7(b) evaluates Equation 10 without fitting a slope after seeing test accuracy. For each trained checkpoint, it computes the largest observed spectral-norm adapter-distance/coordinate-distance ratio over 64 deterministic coordinate-pair probes and compares it with $3\sqrt{2}\Gamma_l$, where adjacent-control variation uses the same matrix spectral norm as Equation 6. All nine points fall above the equality boundary when plotted as envelope versus observation, providing a norm-consistent numerical certificate of the analytic inequality.

A.4 What the bending penalty controls

Define the horizontal second difference $D_{11}C_{ab} = C_{a+2,b} - 2C_{a+1,b} + C_{a,b}$ and analogously D_{22}. Differentiating Equation 7 again gives

$$\|\partial_{11}A_l(u)\|_F \le 6 \max_{a,b} \left\|D_{11}C^l_{ab}\right\|_F, \quad \|\partial_{22}A_l(u)\|_F \le 6 \max_{a,b} \left\|D_{22}C^l_{ab}\right\|_F. \tag{13}$$

Thus discrete control curvature bounds the continuous field Hessian. The implementation applies second differences to both factors. To connect factor and product curvature, suppress indices and write $C_k = U_k V_k/\sqrt{r}$; then

$$D_{11}C_k = \frac{1}{\sqrt{r}}\left[(D_{11}U_k)V_{k+1} + U_{k+2}(V_{k+2} - V_{k+1}) - U_k(V_{k+1} - V_k)\right]. \tag{14}$$

Under bounded factor norms and bounded first differences, Equation 14 shows that the implemented factor-level bending penalty provides conditional control of product curvature; second-difference control alone is insufficient. Thus R_{bend} is a geometric regularizer, not a guarantee of a point-accuracy gain. Table 7 measures both accuracy and the factor-level energy targeted by the implementation.

A.5 Cold-start decomposition at parameter and risk levels

Theorem 2 (Cold-start stability and fixed-family approximation)**.** *Let A^*_l be a target adapter field and $A^*_{l,B}$ its best approximation in the implemented degree-three tensor-product Bernstein family. Suppose $\|\widehat{u} - u\|_2 \le \varepsilon_s$, $\sup_v \left\|\widehat{A}_l(v) - A^*_{l,B}(v)\right\|_2 \le \varepsilon_{\mathrm{opt},l}$, and define $\varepsilon_{\mathrm{app},l}(u) = \left\|A^*_{l,B}(u) - A^*_l(u)\right\|_2$. Then*

$$\left\|\widehat{A}_l(\widehat{u}) - A^*_l(u)\right\|_2 \le 3\sqrt{2}\widehat{\Gamma}_l\varepsilon_s + \varepsilon_{\mathrm{app},l}(u) + \varepsilon_{\mathrm{opt},l}. \tag{15}$$

If the network and loss satisfy Proposition 3, the excess query-induced risk is bounded by the corresponding layerwise propagated sum of the three terms in Equation 15.

Proof. Insert both the learned field at the ideal coordinate and the best fixed-family field:

$$\left\|\widehat{A}_l(\widehat{u}) - A^*_l(u)\right\|_2 \le \left\|\widehat{A}_l(\widehat{u}) - \widehat{A}_l(u)\right\|_2 \tag{16}$$

$$+ \left\|\widehat{A}_l(u) - A^*_{l,B}(u)\right\|_2 + \left\|A^*_{l,B}(u) - A^*_l(u)\right\|_2. \tag{17}$$

Theorem 1 bounds the first term, while the last two are the optimization/estimation and approximation definitions. Applying the layerwise telescoping argument from Equation 11, followed by Equation 12, proves the risk statement. The bound predicts that cold start can fail through an inaccurate profile coordinate, a target field outside the fixed cubic family, or insufficient field estimation. Figure 4 tests the combined outcome under a label-free information boundary; its pooled intervals against HyperAdapter and UnseenHN overlap zero, consistent with these irreducible terms. □

A.6 Nonconvex optimization of the implemented-query objective

Theorem 3 is a supporting shared-parameter result conditional on the fixed private states defined above. In particular, $F_i(\Theta)$ is the objective actually differentiated by Algorithm 1: its query is produced by the implemented encoder and router, and its Bernstein controls are components of Θ. The theorem neither substitutes an unobserved ideal query nor compares against a population control field. Changing private offsets or heads would introduce a tracking/coupling error not bounded here, so the result does not establish convergence of the full alternating dynamics. We consider equal-budget clients, matching the controlled simulations after per-client resampling. Then

$p_i = 1/N$ and the shared update reduces to uniform averaging. Let $K = |S_t|$ clients be sampled uniformly without replacement, let $\Theta^i_{t,e}$ be client i's shared state after local step e, and define $G_t = K^{-1} \sum_{i \in S_t} \sum_{e=0}^{E-1} g_i(\Theta^i_{t,e}; \xi^i_{t,e})$ and $\chi_{N,K} = (N-K)/(K(N-1))$.

Theorem 3 (Supporting local-SGD bound)**.** *Conditional on fixed private states, under Assumption 1, equal client budgets, uniform sampling without replacement, $N > 1$, and $\eta LE \leq 1/4$, there are constants $c_1, c_{\mathrm{sam}}, c_2$ independent of T, K, E such that*

$$\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \left\| \nabla F(\Theta_t) \right\|_2^2 \leq \frac{2(F(\Theta_0) - F^*)}{\eta ET} + c_1 \frac{\eta \sigma^2}{K} + c_{\mathrm{sam}} \eta E \chi_{N,K} \zeta^2 \tag{18}$$

$$+ c_2 \eta^2 E^2 (\zeta^2 + \sigma^2), \tag{19}$$

The client-subsampling term is solely due to partial participation and vanishes at $K = N$. With $\eta = \Theta((ET)^{-1/2})$ and bounded E, every displayed residual vanishes. This is a stationarity result for the implemented-query objective; ideal-query approximation is intentionally handled by Theorem 2, not by adding a bias floor here.

Proof. L-smoothness and the server update $\Theta_{t+1} = \Theta_t - \eta G_t$ give

$$\mathbb{E}_t F(\Theta_{t+1}) \leq F(\Theta_t) - \eta \langle \nabla F(\Theta_t), \mathbb{E}_t G_t \rangle + \frac{L\eta^2}{2} \mathbb{E}_t \left\| G_t \right\|_2^2 . \tag{20}$$

Let $a_t = \nabla F(\Theta_t)$ and isolate the root-gradient client-sampling error

$$s_t = E \left[\frac{1}{K} \sum_{i \in S_t} \nabla F_i(\Theta_t) - a_t \right]. \tag{21}$$

Uniform sampling without replacement gives the finite-population identity

$$\mathbb{E}_t s_t = 0, \qquad \mathbb{E}_t \left\| s_t \right\|_2^2 = E^2 \chi_{N,K} \frac{1}{N} \sum_{i=1}^{N} \left\| \nabla F_i(\Theta_t) - a_t \right\|_2^2 \leq E^2 \chi_{N,K} \zeta^2. \tag{22}$$

Write $G_t = E a_t + s_t + d_t + \xi_t$, where ξ_t is within-client stochastic-gradient noise and d_t is solely the drift from evaluating later local gradients at $\Theta^i_{t,e}$ rather than Θ_t. Because the implemented query computation is part of F_i, it creates no additional mismatch term. Conditional independence and Assumption 1 give $\mathbb{E}_t \xi_t = 0$ and $\mathbb{E}_t \left\| \xi_t \right\|_2^2 \leq E\sigma^2/K$. Because s_t and ξ_t are zero mean, they do not enter the first-order bias against a_t. Young's inequality is needed only for d_t:

$$-\eta \langle a_t, E a_t + d_t \rangle \leq -\frac{\eta E}{2} \left\| a_t \right\|^2 + \frac{\eta}{2E} \left\| d_t \right\|^2 . \tag{23}$$

The smoothness term in Equation 20, together with Equation 22, contributes a constant times $L\eta^2 E^2 \chi_{N,K} \zeta^2$ per round; normalization by the descent coefficient ηE yields $c_{\mathrm{sam}} \eta E \chi_{N,K} \zeta^2$. The same calculation for ξ_t yields $c_1 \eta \sigma^2 / K$. Repeated smoothness over the E local steps gives $\mathbb{E}_t \|d_t\|^2 \leq c\eta^2 E^3 (\zeta^2 + \sigma^2)$ after the usual gradient-norm term is absorbed under $\eta LE \leq 1/4$. Substituting these bounds into Equation 20, summing over rounds, and telescoping $F(\Theta_t)$ proves Equation 19. The three residuals therefore correspond exactly to stochastic gradients, random client subsampling, and local-iterate drift. □

This theorem is supporting rather than a claim that FEDSPLINE changes the asymptotic class of local SGD or proves convergence of changing private states. Partial participation enters the conditional shared subproblem only through random client subsampling and the resulting local trajectory. Coordinate observation and out-of-support behavior are separate approximation questions tested in Figures 4 and 6(a–b).

A.7 THEORY–ALGORITHM–EXPERIMENT CORRESPONDENCE

Theory–algorithm–experiment map. Each theoretical object is tied to an implemented operation and a non-duplicated empirical check.

Theoretical object	Algorithmic realization	Empirical check
Convex partition, Eq. 8	Sixteen analytic Bernstein weights in Eq. 4	Basis invariant tests and bounded field probes
Query containment, Eq. 9	Private offset and bounded instance router	Table 7: no offset / no router
Adapter Lipschitz envelope, Eq. 10	Adjacent low-rank controls	Figure 7(b): observed slope versus certificate
Field Hessian, Eq. 13	Factor bending penalty	Table 7: accuracy and trained bending energy
Cold-start decomposition, Eq. 15	Input-only coordinate, zero offset, no adaptation	Figure 4 and Table 11
Conditional shared-parameter bound, Eq. 19	Fixed private state; implemented queries; client sampling and local drift	Figure 5(c): participation boundary
Capacity formula, Proposition 1	Rank-r, $G \times G$ control bank	Table 9 and rank–communication sweep

B ADDITIONAL EXPERIMENTAL DETAIL

B.1 DATA CONSTRUCTION

MNIST-C is generated by our pipeline from the official MNIST arrays rather than loaded as a pre-existing corruption benchmark. We preserve the official 60,000/10,000 source pools before client resampling. After client-specific Dirichlet label sampling, four domains use rotation centers $(-32^\circ, -11^\circ, 13^\circ, 34^\circ)$ multiplied by shift severity, Gaussian blur up to 0.43 pixels, client-dependent contrast and brightness, and Gaussian noise up to standard deviation 0.052. CIFAR-10-C uses six cached corruption episodes with 2,000 ResNet-18 feature vectors each; before client construction, base-image feature identities are stratified 70/30 and the same identity partition is shared across corruptions. Office-Home-10 contains 361 Art, 800 Clipart, 854 Product, and 703 Real-World ResNet-50 features; feature identities are stratified 70/30 within each domain/class. Projection and standardization statistics are fitted on the training pool only. Sampling with replacement occurs only within the assigned side, equalizing client budgets without crossing source identities; serialized audits record zero train/test overlap for every seed. The end-to-end protocol reads 2,780 unique raw images from ten alphabetically fixed Office-Home classes and likewise splits unique paths 70/30 within each domain/class before resampling. Images are resized to 32×32, 32 clients cycle over four natural domains, and client labels follow Dirichlet(0.3); no pretrained feature is used.

B.2 OPTIMIZATION AND ARCHITECTURE

All compact models use two hidden layers of width 64, LayerNorm, GELU, and a ten-class head. FEDSPLINE uses rank $r = 4$, a dense 4×4 Bernstein control bank in each adapted layer, private two-dimensional offsets, $\rho = 0.3$, learned bounded router radii, bending weight 0.008 for feature benchmarks and orthogonality weight 2×10^{-4}. The raw-image experiment trains a CIFAR-stem ResNet-18 jointly with width-128 personalization layers for 12 rounds, 32 clients, 37.5% participation, 384/192 train/test images per client, AdamW learning rate 0.0025, and bending weight 2×10^{-4}.

FedLWS and Feroma receive the same backbone, rounds, participation, data order, and optimizer budget as the compact baselines; no coefficient is selected on test or held-out clients. Their mechanism-specific scales are data-adaptive rather than dataset-tuned: FedLWS sets each layer contraction from the aggregated-update norm and cross-client dispersion, while Feroma standardizes profile distances and sets its temperature to the median observed squared distance. As a sanity check on the 24-client simulation over three seeds, FedLWS produces nontrivial contraction factors with mean 0.605 and observed range $[0.288, 0.876]$. Feroma's weights are always nonnegative and sum to one; its mean

Table 2: Method rank across the nine stress settings; lower is better.

Method	Mean rank↓	Wins↑
FedAvg	12.00	0
FedProx	11.89	0
FedRep	4.00	0
Ditto	5.44	0
pFedMe	5.22	1
FedALA	6.17	0
FedCP	6.83	0
FedRoD	<u>2.44</u>	**4**
FLoCo	8.44	0
FedLWS	11.61	0
Feroma	10.39	0
HyperAdapter	4.33	0
FEDSPLINE	**2.22**	<u>3</u>

maximum assignment weight is 0.177 and mean normalized entropy is 0.953, confirming soft rather than collapsed profile assignment. These checks validate the intended qualitative mechanisms but do not claim reproduction of the original systems' published numbers.

B.3 METRICS AND STATISTICS

Accuracy and calibration are computed per client and then macro-averaged, so large clients cannot dominate. ECE uses 15 equal-width confidence bins, and the worst-client statistic is the minimum client accuracy. Figures 3 and 6(c–d) use two-sided Student-t intervals with three independent seeds; Figure 6(a–b) first averages held-out coordinates within each of five independent worlds and treats the world seed as the inferential unit. Figure 4 uses 20,000 deterministic dataset-stratified cluster-bootstrap replicates. World-seed comparisons use independent worlds rather than coordinate points. Whenever dataset–seed differences are flattened across heterogeneous tasks—in Section 5.3, the matched control in Appendix C, or Table 8—their pooled intervals and p-values are descriptive rather than confirmatory because seeds are nested within tasks.

C ADDITIONAL EXPERIMENTS AND DESIGN ANALYSIS

C.1 STRESS TESTS AND FAILURE BOUNDARY

Figure 5 changes label concentration, feature-shift severity, and participation in a forced rerun whose records carry the `source-disjoint-v2` tag and zero-overlap audit. MNIST-C draws training clients only from the official 60,000-image training pool and test clients only from the official 10,000-image test pool. FEDSPLINE is first at feature severity 0.65 and 1.0 and at $\alpha = 0.3$, and second in three additional settings. At severity 1.0 it reaches 82.00% mean accuracy, compared with 81.50% for pFedMe and 81.13% for FedRep. At 40% participation, however, FedRoD obtains 83.06%, FedRep 82.31%, and FEDSPLINE 81.75%; at full participation, pFedMe and Ditto also exploit their cheaper per-client adaptation effectively. This failure boundary is consistent with sparse rounds updating too few regions of the dense control field.

C.2 HELD-OUT DETAILS AND DIRECT FIELD DIAGNOSTICS

Table 3 gives the all-method cold-start aggregates behind Figure 4. Distance sensitivity is the macro-average absolute slope of three dataset-level regressions and is descriptive rather than a hypothesis-test unit. The HFZ-inspired UnseenHN has the flattest slope and best ECE, whereas FEDSPLINE has the highest mean accuracy but a larger slope; cold-start level and sensitivity must therefore be interpreted jointly.

The input-only signature is computed from the same unlabeled local input pool on both sides of the split. Its sixteen entries are global mean, standard deviation, 10/25/50/75/90 percentiles, median absolute deviation, mean absolute value, root-mean-square value, and the mean and standard deviation

Table 3: Label-free zero-adaptation results over 36 held-out clients per method.

Method	Accuracy (%)↑	ECE (%)↓	|Distance slope| ↓
FedAvg	55.53	17.98	26.11
FedProx	55.54	18.00	26.16
FedRep	54.83	24.00	29.56
Ditto	55.53	17.98	26.11
pFedMe	36.95	<u>16.53</u>	<u>15.12</u>
FedALA	53.06	19.72	23.21
FedCP	52.46	22.61	36.51
FedRoD	62.11	31.11	19.24
FLoCo	53.09	17.77	23.94
FedLWS	52.25	19.43	32.84
Feroma	54.68	18.52	23.08
HyperAdapter	<u>62.73</u>	19.58	18.70
UnseenHN (HFZ-inspired) (Zhou et al., 2026)	62.11	**16.03**	**14.90**
FEDSPLINE	**63.23**	19.26	25.51

of per-sample means, dispersions, and norms. Table 11 makes the zero-adaptation fallback explicit; no rule or hyperparameter is selected using held-out performance.

A five-seed matched-generator analysis uses 15 paired dataset–seed trials (five seeds nested within each of three tasks). Across these trials, FEDSPLINE improves by 0.19 ± 0.46 points with 10 wins; the flat pooled $p = 0.390$ is descriptive rather than confirmatory. Figure 7 is a mechanism audit, not a leaderboard. Panel (b) tests Theorem 1 directly: in all nine checkpoints, the observed maximum spectral-norm adapter slope lies below the analytic $3\sqrt{2}\Gamma_l$ envelope computed in the same norm. Conventional PFL baselines do not define coordinate-indexed adapter paths, so only panels (c)–(d) use the capacity-matched HyperAdapter as a valid counterfactual. Across 64 chart points in nine FEDSPLINE checkpoints, coordinate and adapter distances have Spearman $\rho = 0.81 \pm 0.09$. HyperAdapter has $52\times$ lower normalized path variation (0.071 versus 3.657) and nearly zero endpoint prediction response on these probes; this observation is scoped to the controlled runs.

C.3 KNOWN-COORDINATE GEOMETRY BENCHMARK

We sample six-class, 32-dimensional observations from fixed class prototypes. Two known coordinates continuously rotate disjoint feature subspaces, rescale feature groups, and add a smooth cross-factor rotation; the generator itself contains no Bernstein polynomial. The training set is $\mathcal{U}_{\text{tr}} = \{0.18, 0.50, 0.82\}^2 \setminus \{(0.50, 0.50)\}$, with 720 samples at each of its eight coordinates. We define interpolation by $u \in \text{conv}(\mathcal{U}_{\text{tr}}) = [0.18, 0.82]^2$ and evaluate the 25-point grid $\{0.30, 0.40, 0.50, 0.60, 0.70\}^2$. Extrapolation means $u \notin \text{conv}(\mathcal{U}_{\text{tr}})$; its 18 coordinates are $\{0.02, 0.10, 0.90, 0.98\} \times \{0.10, 0.50, 0.90\}$ together with $\{0.30, 0.50, 0.70\} \times \{0.02, 0.98\}$. Each test coordinate supplies 420 samples. The oracle protocol gives every method the exact coordinate; a separate 15-statistic input-only encoder tests coordinate estimation. Feroma soft-mixes coordinate experts, Nearest Expert selects one, ClientHN directly generates a low-rank classifier, and HyperAdapter replaces analytic weights by a parameter-matched MLP.

Table 4: Known-coordinate results over five world seeds. NLL and ECE refer to interior coordinates. The oracle row isolates field geometry; the learned row changes only the query source.

Method	Interior acc.↑	Extrapolation acc.↑	NLL↓	ECE↓
Nearest expert	86.06	<u>85.75</u>	0.396	**0.046**
Feroma	**88.47**	**86.67**	**0.340**	0.058
ClientHN	86.50	78.42	0.438	0.068
HyperAdapter (oracle)	87.03	83.60	0.382	<u>0.048</u>
FEDSPLINE (oracle)	<u>87.52</u>	80.36	<u>0.375</u>	0.051
FEDSPLINE (learned)	87.51	83.29	0.378	0.052

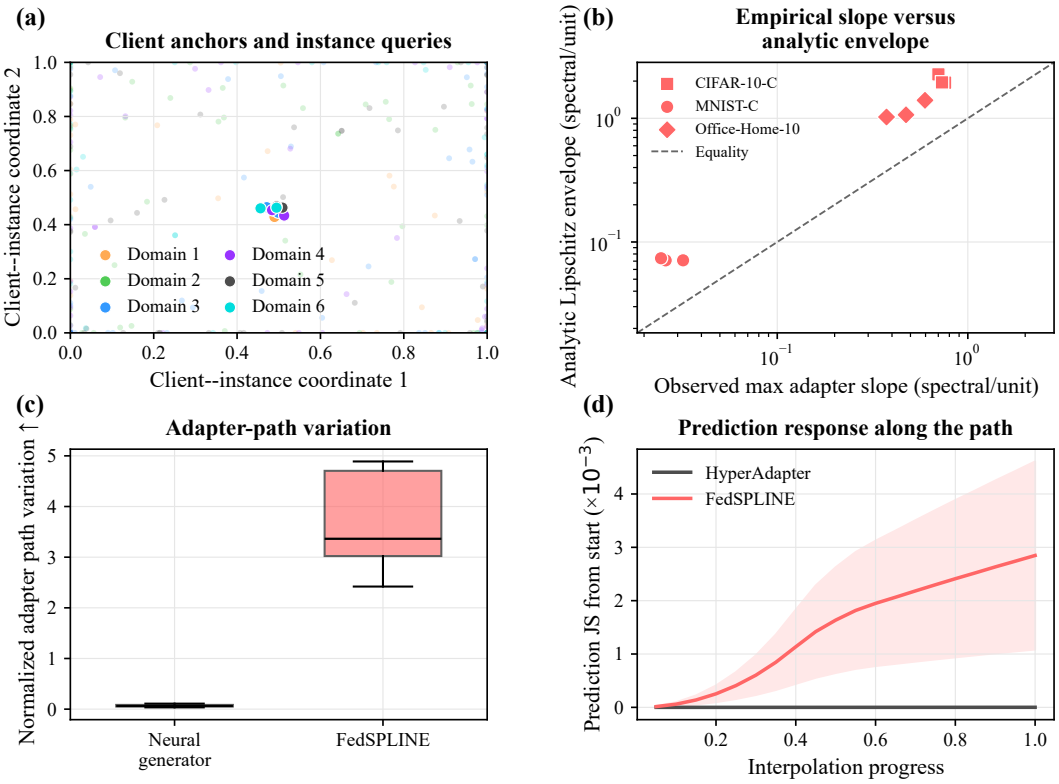

Figure 7: Mechanism-specific field audit. (a) FedSPLINE anchors and instance queries; (b) observed maximum spectral-norm adapter slope against the same-norm analytic Lipschitz envelope (the dashed equality line is the required boundary); (c) run-level normalized path variation; and (d) prediction Jensen–Shannon response with descriptive Student-t bands over nine dataset–seed runs. HyperAdapter appears only in (c)–(d), where a coordinate-indexed path is well defined and capacity matched.

The oracle interior gain over HyperAdapter is 0.49 points ($p = 0.150$ over five seeds); it is directional but not significant. The learned coordinate changes interior accuracy by only -0.004 points, so encoder quality is not hiding the interpolation result. In extrapolation, learned queries shrink toward observed support and improve FEDSPLINE from 80.36% to 83.29%, but Feroma and nearest experts remain stronger. We therefore do not interpret analytic smoothness as extrapolation accuracy.

C.4 RAW-IMAGE SCALE AND DESIGN SENSITIVITY

Table 5 gives the leakage-safe raw Office-Home result behind Figure 6(c–d). FEDSPLINE has the highest point estimate for mean accuracy, worst-client utility, and calibration, while its mean-accuracy gaps to HyperAdapter and FedRoD are nonsignificant. All five methods use the same source split, ResNet-18 initialization, client stream, optimizer family, and update budget.

Table 5: End-to-end raw Office-Home-10, 32 clients, ResNet-18, three seeds.

Method	Mean acc.↑	Worst acc.↑	ECE (%)↓
FedAvg	22.48	1.22	23.01
Feroma	34.49	2.43	20.93
FedRoD	48.61	19.27	19.41
HyperAdapter	<u>48.77</u>	<u>19.79</u>	<u>17.56</u>
FEDSPLINE	**50.03**	**23.44**	**16.02**

Table 6 is a separate three-seed Office-Home-10 sweep that fixes all budgets across its four rows (16 rounds, 360/160 train/test samples per client), whereas Table 1 uses 24 rounds and 420/180 samples; its default row is therefore not intended to reproduce the main checkpoint. The smaller 2-D quadratic field has the highest point accuracy (86.79%), while the 1-D cubic field is cheapest and second in accuracy. The prespecified 2-D cubic 4×4 field reaches 84.83% and is our geometry-rich instantiation, not an empirically universal optimum. Relative to degree two, degree three permits second directional curvature to vary along the queried coordinate, matching our curvature-control objective; nevertheless, the quadratic 3×3 result is both more accurate and cheaper here. Basis degree and grid resolution are therefore task-dependent deployment choices, motivating validation-based selection and adaptive sparse controls.

Table 6: Field-design sensitivity over three seeds. Degree equals controls per axis minus one.

Chart	Grid/degree	Controls	Acc. (%)↑	Traffic (MB)↓
1-D	$4/3$	4	<u>86.23</u>	**20.24**
2-D	$3/2$	9	**86.79**	<u>27.51</u>
2-D	$4/3$	16	84.83	37.35
2-D	$5/4$	25	84.91	50.01

C.5 Component ablations

Table 7 aligns every component with an observable consequence under exactly the Table 1 protocol. Full has the highest accuracy point estimate (73.935187%), effectively tied after rounding with removing the private offset (73.935186%). Removing instance routing loses 0.57 points, rank one loses 0.94, and a single shared adapter loses 0.32. Removing bending loses 0.11 points and increases trained factor-level bending energy from 0.082 to 0.138 (69%). No variant dominates across worst-client accuracy, calibration, noise accuracy, and bending energy. We therefore retain the prespecified Full configuration without test-set selection: routing and field capacity support prediction, while bending provides the intended conditional curvature control.

Table 7: Ablation on CIFAR-10-C using the same seeds (1, 7, 19), client splits, 24 rounds, and sample budgets as Table 1. Full denotes the prespecified geometry-controlled configuration; each ablation modifies one component while retaining all remaining settings. Bold and underline mark best and second best point estimates. Profile noise has SD 0.5; Bend E. is the implemented factor-level energy $\sum_l [\mathcal{B}(U^l) + \mathcal{B}(V^l)]$.

Variant	Acc. ↑	Worst ↑	ECE (%) ↓	Noise acc. ↑	Bend E.$\times 10^2$ ↓
Full FedSpline	**73.94±1.29**	54.07	12.00	<u>73.92</u>	8.16
w/o instance router	73.36±2.50	53.70	**11.77**	73.38	**2.67**
w/o private offset	<u>73.94±1.32</u>	<u>54.44</u>	<u>11.78</u>	**74.00**	8.13
w/o bending energy	73.83±1.64	53.33	11.86	73.81	13.80
w/o orthogonality	73.69±1.18	53.89	11.92	73.66	8.19
rank-one field	72.99±2.18	52.59	12.20	72.98	6.17
single shared adapter	73.61±2.43	**55.56**	11.88	73.61	<u>2.99</u>

C.6 Statistical reliability and computational cost

Table 8 gives a descriptive pooled analysis at the dataset–seed level rather than treating clients from one run as independent. Every pooled mean effect is positive, but the nine rows comprise three seeds nested within three heterogeneous tasks and are not naturally exchangeable replicates of one target population. We therefore use the table to report magnitudes and win counts, not confirmatory cross-task significance. FedRoD has a 0.82-point pooled mean gap with 6/9 wins; the matched HyperAdapter gap is only 0.14 points with 7/9 wins and a confidence interval spanning zero.

Table 8: Descriptive pooled paired analysis across nine dataset–seed runs (three seeds nested within each of three tasks). Intervals and p-values assume flat exchangeability and are not confirmatory cross-task inference.

Comparison	Gain (pp)↑	Pooled 95% CI	Pooled p	Wins↑
vs. FedAvg	12.11	±1.93	5.15×10^{-7}	**9/9**
vs. FedProx	<u>12.11</u>	±1.93	5.06×10^{-7}	**9/9**
vs. FedRep	1.89	±1.16	0.0056	8/9
vs. Ditto	1.92	±0.95	0.0017	**9/9**
vs. pFedMe	1.78	±0.86	0.0014	**9/9**
vs. FedALA	2.76	±1.32	0.0013	8/9
vs. FedCP	3.30	±1.53	0.0011	**9/9**
vs. FedRoD	0.82	±1.24	0.165	6/9
vs. FLoCo	7.47	±1.82	1.29×10^{-5}	**9/9**
vs. FedLWS	**14.63**	±2.87	2.53×10^{-6}	**9/9**
vs. Feroma	8.26	±2.37	4.20×10^{-5}	**9/9**
vs. HyperAdapter	0.14	±0.68	0.658	7/9

Table 9: Accuracy–cost trade-off averaged over the three main datasets. Communication is cumulative over 24 rounds.

Method	Params (M)↓	Comm. (MB)↓	Time (s)↓	Acc. (%)↑
FedAvg	**0.027**	45.02	<u>2.19</u>	71.00
FedProx	**0.027**	45.02	2.67	71.00
FedRep	**0.027**	**43.95**	2.20	81.22
Ditto	**0.027**	45.02	3.88	81.19
pFedMe	**0.027**	45.02	2.84	81.33
FedALA	**0.027**	45.02	2.95	80.35
FedCP	**0.027**	**43.95**	2.44	79.81
FedRoD	**0.027**	45.02	2.29	82.29
FLoCo	**0.027**	45.02	2.88	75.64
FedLWS	**0.027**	45.02	**2.17**	68.48
Feroma	**0.027**	45.02	2.51	74.85
HyperAdapter	0.064	103.99	4.61	<u>82.98</u>
FEDSPLINE	0.063	102.88	8.09	**83.11**

C.7 DOMAIN-DERIVED SCALING

Table 10 removes the synthetic residual shift from Office-Home and doubles the population to 24 clients with 50% participation. It reports all thirteen main-protocol methods. HyperAdapter has the highest mean accuracy and lowest ECE, FedRoD the strongest worst-client result, and FEDSPLINE ranks second in mean accuracy and calibration while using 2.45 MB less traffic than HyperAdapter. This independent protocol reinforces the trade-off claim rather than universal metric dominance.

C.8 WHY AN EXPLICIT FIELD?

A generic hypernetwork can generate client parameters, but its behavior between observations is difficult to inspect or constrain. The Bernstein tensor product is a partition of unity: every query is a convex mixture of sixteen named controls, derivative bounds follow directly, and low-rank factorization makes the construction practical. The near-constant paths observed for our matched HyperAdapter checkpoints show why a continuous input alone is insufficient evidence that continuous personalization is being used. A two-dimensional chart can still fold when heterogeneity is intrinsically high-dimensional; learned chart mixtures would be required there.

Table 10: Office-Home domain-derived simulation with 24 clients (mean±SD over three seeds).

Method	Acc. (%)↑	Worst (%)↑	ECE (%)↓	Comm. (MB)↓
FedAvg	75.60±3.40	39.79	19.42	29.27
FedProx	75.60±3.40	39.79	19.43	29.27
FedRep	81.24±0.94	62.08	15.59	**27.84**
Ditto	80.81±1.94	48.33	14.99	29.27
pFedMe	80.76±1.89	48.33	15.09	29.27
FedALA	82.22±1.43	59.38	25.86	29.27
FedCP	80.94±0.72	58.54	21.74	**27.84**
FedRoD	84.67±2.30	**63.54**	12.64	29.27
FLoCo	77.79±2.97	52.50	22.79	29.27
FedLWS	72.74±3.19	39.58	25.27	29.27
Feroma	77.29±3.23	40.63	19.24	29.27
HyperAdapter	**86.05±2.05**	56.04	**10.47**	77.15
FEDSPLINE	85.63±2.00	52.71	10.95	74.71

Table 11: Deterministic instantiation rules for a non-participating client. Every method receives only the same input-only profile and uses no held-out labels or updates.

Method	Training state retained	Held-out instantiation rule
FedAvg/FedProx	Global checkpoint	Use the global model directly.
FedRep	Shared body, local heads	Shared body plus the arithmetic mean of training-client heads.
Ditto/pFedMe	Shared and private checkpoints	Use the shared checkpoint; no unseen private state is synthesized.
FedALA	Global model and local caches	Use the global server model; the unseen local cache is absent.
FedCP	Shared body, local heads/gates	Shared body plus the mean training head; no private gate is applied.
FedRoD	Global head, private residual heads	Use the global generalist and set the unseen residual head to zero.
FLoCo	Aggregated connected solution	Use the server solution at the default shared coordinate.
FedLWS	Layer-shrunk global model	Use the final shrunk global checkpoint.
Feroma	Training-client model bank	Similarity-weighted mixture using the input-only profile.
HyperAdapter	Shared generator, local offsets/heads	Query with zero offset and the mean training head.
UnseenHN	Client-level hypernetwork	Generate adapter weights directly from the input-only profile.
FEDSPLINE	Bernstein field, local offsets/heads	Query the field with zero offset, retain instance refinement, and use the mean training head.

Capacity and identifiability. For L_a adapted layers, grid width G, rank r, and dimensions (d_l, d_{l+1}), field storage is

$$P_{\text{field}} = G^2 r \sum_{l=1}^{L_a} (d_l + d_{l+1}), \tag{24}$$

plus small coordinate and router networks. Communication scales linearly with this quantity. Our dense Bernstein implementation evaluates all sixteen controls; compactly supported bases or sparse control distillation can reduce larger-grid cost. The chart is not identifiable under smooth reparameterization; our claims concern prediction continuity and adapter geometry, not semantic meanings for coordinate axes.

Privacy boundary and intended regime. Raw samples, private heads, offsets, and instance queries remain local. Shared controls and compact distribution summaries participate in aggregation. This is not a formal privacy guarantee: summaries and gradients may leak attributes, so differential privacy or secure aggregation requires separate evaluation. The field is intended for recurring covariate modes,

sufficient participation to train the chart, and objectives valuing weak-client utility. Pure label-prior shift or extremely sparse participation favors cheaper private-head methods.

D Reproducibility Checklist

Baseline mechanics. The controlled harness changes only the personalization mechanism. FedAvg communicates the complete network; FedProx adds a proximal penalty. FedRep keeps its classifier local. Ditto maintains shared and proximal private models; pFedMe alternates private proximal optimization and aggregation. FedALA interpolates downloaded and local parameters; FedCP uses a global/private gate; FedRoD combines global and private heads; and FLoCo maintains a coordinate over connected solutions. FedLWS shrinks each aggregated layer update according to cross-client update dispersion. Feroma retains a profile-indexed model bank and assigns similarity-weighted model mixtures without client identity. HyperAdapter matches FedSpline in coordinate, private state, rank, and control capacity but learns mixture weights with an MLP. The cold-start-only UnseenHN comparator maps the label-free client profile directly to a low-rank adapter mixture and has no client-specific state. These compact implementations preserve operative distinctions under shared data order, backbone, steps, and participation; they are not bitwise official-repository reproductions.

Recent-method provenance. We audited FEROMA's official ICLR 2026 repository and configuration: its public system combines distribution-profile extraction, standardized profile similarity, profile-guided aggregation, and test-time assignment, with LeNet-5/ResNet-9 experiment drivers. Our Feroma comparator retains standardized profile similarity, adaptive soft assignment, identity-free model mapping, and zero-step assignment, but reimplements orchestration inside the common client stream and uses the shared MLP/ResNet-18 backbone; it does not import the repository's distributed process code. HyperFedZero's published design (Zhou et al., 2026) uses a noisy distribution embedding and chunk-wise hypernetwork. We retain its defining input-only client-level generation and absence of private client state in UnseenHN, but not its complete training system; no compatible official repository was available at artifact freeze. Consequently, neither row is presented as a reproduction of the paper's published number, and novelty conclusions rely additionally on the capacity-matched HyperAdapter control.

FedMC (Ma et al., 2026) is a complementary plug-in calibration framework rather than a standalone personalized predictor: its evaluated realization applies local kernel PCA to embeddings, aggregates a server-side geometry dictionary, and queries that dictionary to calibrate samples within federated prompt learning built on a pretrained vision–language encoder. Our controlled harness deliberately compares personalization rules under the same prompt-free MLP or ResNet backbone, initialization, and training objective; it has neither prompt tokens nor FedMC's pretrained embedding/calibration interface. Adding only a geometry dictionary would not preserve FedMC, whereas replacing the backbone and objective would confound the comparison. We therefore cite FedMC as a related geometric direction but do not make a performance claim about it or report an unfaithful leaderboard row.

Execution environment. Experiments ran with Python 3.9.16, PyTorch 2.0.0+cu118, NumPy 1.24.3, scikit-learn 1.3.0, and Matplotlib 3.9.4 on one NVIDIA RTX 4080. Main and end-to-end protocols use seeds 1, 7, and 19 (end-to-end: 2, 13, and 29); the controlled HyperAdapter comparison adds 31 and 43. Deterministic client construction, cached-feature provenance, and exact commands accompany the code. Wall-clock times cover federated optimization and evaluation after feature caches are available, not one-time feature extraction. No method component depends on specialized hardware.

Result provenance. Experimental records store aggregate and per-client metrics, communication costs, and runtime. Cold-start evaluations use label-free profiles without selecting fallback rules based on held-out performance. Source-identity audits verify zero train–test overlap. These records support the traceability of the reported results.

E BROADER IMPACT

Continuous personalization may improve utility for minority clients by reducing sharp model-assignment boundaries. The same capacity can also encode sensitive distribution attributes in coordinates or updates. Deployment should therefore combine signatures with minimization, secure aggregation, privacy auditing, and client consent. Our experiments are algorithmic simulations and do not justify claims about medical, mobile, or societal deployment.