# FROM TASK MIXTURES TO SPECIALIZED EXPERTS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

In collaborative foundation model fine-tuning, client data is rarely homogeneous. Instead, clients typically possess unknown mixtures of distinct data distributions, or *tasks*. Conventional federated learning primarily addresses heterogeneity across clients without explicitly resolving latent task mixtures within each client. We study this setting as *compound heterogeneity*, where data is heterogeneous both across and within clients. We study adaptation over a common frozen representation and show that, when tasks share the same feature geometry, the optimal model for a client's task mixture under squared loss is a convex combination of the optimal models for its underlying tasks. Thus, a single locally trained model represents the client's overall task mixture, while individual inputs may be drawn from different underlying task distributions. This motivates routing inputs to specialized experts, and we show that, when the task optima form a simplex, task-aligned routing achieves lower risk than any single adapted model for genuinely mixed clients. With access to a small set of task-labeled public samples, we derive a convex program to recover task experts and match them to their corresponding tasks. Our routing analysis shows that effective specialization requires input-dependent expert selection aligned with each client's task mixture. Motivated by this analysis, we propose FedSEE. Across our experiments, FedSEE avoids the negative transfer observed in the evaluated baselines and improves performance by 2.9 points overall and 3.7 points for the worst-served quartile.

## 1 INTRODUCTION

Modern learning systems often adapt models using data distributed across users, institutions, or devices. A common distributed-learning abstraction assigns each participant a single local distribution $\mathcal{P}_i$, so non-IID structure arises across participants ($\mathcal{P}_i \neq \mathcal{P}_j$) (McMahan et al., 2017; Li et al., 2020; Karimireddy et al., 2020). Yet a participant may itself come from multiple underlying data distributions, such as conversation, code, and prose for a personal assistant or records from different hospital departments. In this regime, heterogeneity exists both *across* and *within* participants, and each local dataset is a mixture whose composition varies across the population.

A latent-mixture view of client distributions has been studied in federated multi-task learning (Marfoq et al., 2021). We term the resulting combined within- and across-client variation *compound heterogeneity*, modeling each client $i$'s data distribution $\mathcal{D}_i$ as an unknown mixture $\mathcal{D}_i = \sum_{m=1}^{M} \alpha_{im}\mathcal{P}_m$ of $M$ shared task distributions, where the latent mixture proportions $\alpha_i \in \Delta^{M-1}$ vary across clients. Specifically, we focus on the regime in which clients are themselves mixed and their mixture proportions differ across the population.

For such mixed clients, under a shared feature representation and squared loss, we show that the client optimum is a weighted combination of the task-specific optima, with weights given by the client's mixture proportions, $\theta_i^* = \sum_{m=1}^{M} \alpha_{im}\theta_m^*$ (Theorem 1). The resulting client optimum therefore captures the client's overall task mixture rather than recovering the individual task optima. In contrast, if the underlying task were known for each input, an ideal router could select the corresponding task optimum. When the task optima form a non-degenerate simplex, this idealized routing achieves strictly lower risk for every genuinely mixed client, while the two coincide for a pure client (Remark 6). This shifts the learning objective from estimating one optimum per client to recovering the underlying task experts and learning to route among them. This naturally leads to a mixture-of-experts (MoE) formulation (Jacobs et al., 1991; Shazeer et al., 2017) in which the task-specific optima define the experts and input-dependent routing determines how they are combined.



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.

Recovering these task experts from mixed-client models is not straightforward. When the task optima form a simplex, the task experts correspond to its vertices and mixed-client optima lie within their convex hull. Averaging mixed-client optima therefore remains inside the simplex rather than recovering its vertices, while hard client clustering assigns each mixed client to a single cluster even when its optimum lies between task experts (Figure 1). The geometric structure that limits averaging and hard clustering also provides a direct geometric characterization of the task experts.

To recover the task experts, the server optimizes task-specific losses over convex combinations of the pooled client uploads. A small task-labeled public set provides these losses, allowing each extraction problem to target a specific task and yielding a convex recovery program (Theorem 2). The supervision also resolves the identity of the recovered experts. Without task-indexed information, the expert set may still be recovered, but one cannot determine which expert corresponds to which task (Theorem 3). As a result, experts learned on different clients cannot be reliably matched by task before aggregation.

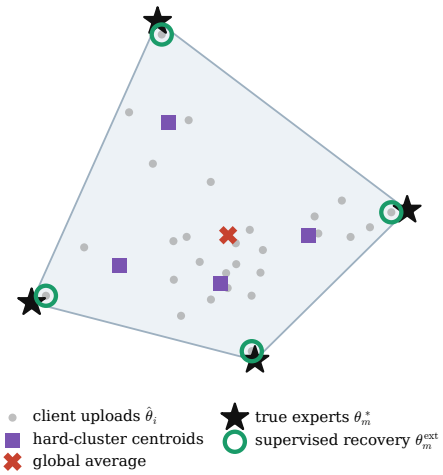



Figure 1: **Simplex geometry of compound heterogeneity.** Task experts are vertices and client optima are barycentric points. Averaging stays inside the simplex, hard clustering collapses mixed clients to single clusters, and supervised extraction targets the vertices.

Recovering the task experts is not sufficient for specialization because each input must also be routed to the expert corresponding to its underlying task. If inputs are routed to their corresponding task experts, the client's average expert usage naturally reflects its task mixture $\alpha_i$. Our analysis quantifies how both the client's average expert usage and the input-dependent routing decisions affect the resulting risk (Theorem 5). In particular, enforcing uniform average expert usage is incompatible with exact specialization when a client's task mixture is non-uniform (Proposition 1).

Motivated by this analysis, we propose **FedSEE** (Federated Supervised Expert Extraction), which combines task-specific LoRA experts (Hu et al., 2022) with supervised extraction and mixture-aware routing. FedSEE shares the LoRA down-projection across experts, isolating task-specific adaptation in the remaining trainable parameters and enabling supervised expert extraction from mixed client updates. The resulting extraction weights indicate how strongly each client contributes to each recovered task expert, providing a proxy for the client's latent task mixture that guides expert selection and routing calibration. FedSEE also supports client-dependent training capacity by allowing each client to update only a subset of its available experts. We summarize our main contributions as follows.

- **Geometry under task mixtures.** We characterize how within-client task mixtures are represented after local optimization and show that, under the stated squared-loss conditions, each client optimum is a mixture-weighted combination of the underlying task optima, inducing a simplex geometry.
- **Expert recovery and identifiability.** We derive a task-supervised convex program for recovering task experts from mixed client uploads, together with recovery guarantees. We further characterize when task identities and the expert set itself cannot be determined from the available information.
- **Mixture-aware routing.** We characterize two requirements for specialization. Average expert usage should reflect the client's task mixture, while routing must remain input dependent to specialize individual inputs.
- **FedSEE.** We instantiate these principles in a federated LoRA framework that extracts task-specific experts from mixed client updates and performs mixture-aware routing with client-specific calibration. Across our experiments, FedSEE improves over the strongest baseline by 2.9 points overall and 3.7 points on the worst-served quartile, and is the only evaluated method to avoid negative transfer across the five-point heterogeneity sweep.

**Closest work.** Related approaches span mixture modeling, federated expert construction, and centralized model composition. Mixture-based methods such as FedEM (Marfoq et al., 2021) and FedSoft (Ruan & Joe-Wong, 2022) allow a client's data to combine multiple latent distributions, but jointly learn the corresponding components during federated optimization rather than recovering task-indexed experts from already mixed client adaptations. Hard clustering methods such as IFCA (Ghosh et al., 2020) instead assign each client to a single component and therefore cannot directly represent clients spanning multiple tasks. Among federated LoRA and mixture-of-experts methods, FedLEASE (Wang et al., 2025) is closest to our setting, but constructs experts through client clustering and within-cluster aggregation rather than task-indexed recovery from mixed uploads. Centralized model and adapter composition methods (Ilharco et al., 2023; Yadav et al., 2023; Zhao et al., 2025; Huang et al., 2024) assume that task-specific models or adapters are already available and learn how to combine them. FedSEE addresses the inverse problem in which the observed client adaptations are mixtures of latent task experts that must first be recovered and identified. Appendix A provides a detailed comparison.

## 2 Problem Formulation

Consider $N$ clients and $M$ shared task distributions $\{\mathcal{P}_m\}_{m=1}^M$ on the sample space $\mathcal{X} \times \mathcal{Y}$. Under compound heterogeneity, the population distribution of client $i$ is

$$\mathcal{D}_i \;=\; \sum\nolimits_{m=1}^{M} \alpha_{im} \mathcal{P}_m, \qquad \alpha_i \in \Delta^{M-1}, \tag{1}$$

where $\alpha_i = (\alpha_{i1}, \ldots, \alpha_{iM})$ is the client's latent task-mixture vector, unobserved by both the client and the server, and $\Delta^{M-1}$ is the probability simplex in $\mathbb{R}^M$. Throughout, $[K] := \{1, \ldots, K\}$; $m, k \in [M]$ index tasks and $i, j \in [N]$ index clients. We study personalization through a parameter block $\theta \in \mathbb{R}^{l \times r}$ acting linearly on a shared frozen representation. Let $f_0 : \mathcal{X} \to \mathbb{R}^l$ denote the shared backbone and $\phi : \mathcal{X} \to \mathbb{R}^r$ the shared feature map. The personalized model is

$$f_\theta(x) \;=\; f_0(x) \;+\; \theta\, \phi(x). \tag{2}$$

For any non-negative measurable loss $\ell$, define the task and client population risks as

$$\begin{aligned} \mathcal{L}_m(\theta) &:= \mathbb{E}_{(x,y)\sim\mathcal{P}_m}[\ell(f_\theta(x), y)], \\ \mathcal{L}_i(\theta) &:= \mathbb{E}_{(x,y)\sim\mathcal{D}_i}[\ell(f_\theta(x), y)] = \sum\nolimits_{m=1}^{M} \alpha_{im} \mathcal{L}_m(\theta). \end{aligned} \tag{3}$$

The final equality follows from linearity under Eq. (1), since the same loss and predictor are evaluated under each component distribution. For the exact geometry developed in Section 3, we specialize to the squared loss $\ell(\hat{y}, y) = \|y - \hat{y}\|_2^2$. Appendix D.6 shows that for smooth general losses, including softmax cross-entropy, the optimum remains an approximate barycenter with deviation controlled by task-curvature dispersion. In this squared-loss setting, define the task and client population optima as

$$\theta_m^* \in \arg\min_{\theta \in \mathbb{R}^{l\times r}} \mathcal{L}_m(\theta), \qquad \theta_i^* \in \arg\min_{\theta \in \mathbb{R}^{l\times r}} \mathcal{L}_i(\theta). \tag{4}$$

Under the conditions introduced in Section 3, these minimizers exist and are unique.

In the finite-sample recovery abstraction, client $i$ holds $n_i$ i.i.d. samples from $\mathcal{D}_i$ and contributes a locally trained estimate $\hat{\theta}_i$ of $\theta_i^*$. The server observes the resulting pool $\{\hat{\theta}_i\}_{i=1}^N$ of parameter uploads together with the public reference sets. The learning objective is to recover the task experts $\{\theta_m^*\}_{m=1}^M$ and to route each input among the learned experts. Client $i$ uses a router $p_i : \mathcal{X} \to \Delta^{M-1}$, where $p_{im}(x)$ is the weight assigned to expert $m$, yielding,

$$f_i(x) = f_0(x) + \sum\nolimits_{m=1}^{M} p_{im}(x)\, \theta_m\, \phi(x), \tag{5}$$

where $\{\theta_m\}_{m=1}^M$ denote the learned expert parameters, with exact recovery corresponding to $\theta_m = \theta_m^*$. Equivalently, $f_i(x) = f_{\theta_i^{\mathrm{rt}}(x)}(x)$ with $\theta_i^{\mathrm{rt}}(x) := \sum_m p_{im}(x)\theta_m \in \mathrm{conv}\{\theta_m\}_{m=1}^M$, extending the fixed parameter in Eq. (2) to an input-dependent convex combination. We denote the resulting population risk by $\mathcal{R}_i := \mathbb{E}_{(x,y)\sim\mathcal{D}_i}[\ell(f_i(x), y)]$. Thus the learning problem has two coupled components: *expert recovery* and *input-dependent routing*. To make expert recovery task-indexed, we assume a small labeled reference set for each task.

**Assumption 1 (Task-Labeled Public Data)** *For each task $m \in [M]$, the server has access to a labeled reference set $\mathcal{D}_m^s$ containing $n_m$ i.i.d. samples from $\mathcal{P}_m$.*

In standard benchmark settings, small labeled reference splits are commonly available. We define the empirical task-$m$ public risk as $\widehat{\mathcal{L}}_m^s(\theta) := (1/n_m) \sum_{(x,y)\in\mathcal{D}_m^s} \ell(f_\theta(x), y)$. These task-specific losses provide the supervision used to associate recovered experts with their underlying tasks and to guide expert extraction.

## 3 A General Framework for Federated Mixture Personalization

We first characterize the geometry induced by the mixture model in Eq. (1) and the model class in Eq. (2). We then use this structure to study supervised expert recovery and its identifiability limits, and finally derive the routing requirements for task-aligned specialization. Complete proofs and supporting theoretical results appear in Appendix D.

### 3.1 Barycentric Geometry of Client Optima

We analyze the squared-loss model under the following conditions on the shared representation and task distributions.

**Assumption 2 (Shared-Feature Linear Regression)** (i) Task realizability. *For each task $m$, data drawn from $\mathcal{P}_m$ satisfy $y = f_0(x) + \theta_m^* \phi(x) + \zeta_m$, with $\mathbb{E}[\zeta_m \mid x] = 0$ and $\mathbb{E}[\zeta_m \zeta_m^\top \mid x] = \sigma_m^2 I_l$.* (ii) Shared feature second moment. $\Phi_m := \mathbb{E}_{\mathcal{P}_m}[\phi(x)\phi(x)^\top] = \Phi$ *for all $m$.* (iii) Feature non-degeneracy. $\Phi \succ 0$.

Part (i) makes each task regression well specified relative to the shared representation. Part (ii) gives the exact common quadratic geometry. When the $\Phi_m$ differ, exact barycentricity is replaced by a controlled perturbation governed by the relative covariance dispersion $\rho_\Phi$ (Remark 1 in Appendix D.1). We write $\|C\|_\Phi^2 := \mathrm{tr}(C\Phi C^\top)$. Part (iii) ensures unique population minimizers.

**Theorem 1 (Barycentric Characterization of Client Optima)** *Under Assumption 2, client $i$'s squared-loss population risk $\mathcal{L}_i$ has a unique minimizer given by*

$$\theta_i^* \;=\; \sum\nolimits_{m=1}^{M} \alpha_{im} \theta_m^*. \tag{6}$$

Theorem 1 shows that local optimization preserves the latent mixture weights in parameter space, as each client optimum is a barycentric combination of the task experts and therefore lies in their convex hull. To characterize this geometry, define

$$\mathrm{sep} \;:=\; \min_{m\neq m'} \|\theta_m^* - \theta_{m'}^*\|_{\mathrm{F}}, \qquad h \;:=\; \min_{m\in[M]} \mathrm{dist}_{\mathrm{F}}\big(\theta_m^*,\, \mathrm{aff}\{\theta_k^* : k \neq m\}\big). \tag{7}$$

Here $\mathrm{sep}$ measures pairwise expert separation, while $h$ measures the minimum affine height. When $h > 0$, the experts are affinely independent, their convex hull is a non-degenerate simplex, and $\alpha_i$ is the unique barycentric coordinate vector of $\theta_i^*$. Appendix D.1 details the resulting face and mixing-gap geometry.

Under $h > 0$, a mixed client's optimum is a non-vertex barycentric point. A task-oracle router that selects $\theta_m^*$ for each task-$m$ input incurs only the irreducible noise, with risk $\Lambda_i := \sum_m \alpha_{im}\sigma_m^2 l$. In contrast, the best fixed adapter $\theta_i^*$ incurs the mixing gap $G_i := \mathcal{L}_i(\theta_i^*) - \Lambda_i = \sum_m \alpha_{im}\|\theta_i^* - \theta_m^*\|_\Phi^2$, which is strictly positive for every mixed client and vanishes exactly for a pure client (Proposition 2, Appendix D.1). This oracle gap motivates *task-aligned specialization*. Approaching it with an input-dependent router additionally requires routing inputs toward their corresponding experts, as characterized in Section 3.3.

### 3.2 Supervised Expert Recovery and Identifiability

Theorem 1 places client optima in the convex hull of the task experts. The task-indexed public losses allow the server to extract each expert by minimizing its task risk over convex combinations of the uploaded models. Using the population loss for task $m$, define

$$\omega_m^* \in \operatorname*{arg\,min}_{\omega_m \in \Delta^{N-1}} \mathcal{L}_m\left(\sum\nolimits_{i=1}^{N} \omega_{m,i} \hat{\theta}_i\right), \qquad \theta_m^{\mathrm{ext}} = \sum\nolimits_{i=1}^{N} \omega_{m,i}^* \hat{\theta}_i. \tag{8}$$

Under the squared-loss model, the task population risk satisfies $\mathcal{L}_m(\theta) - \mathcal{L}_m(\theta_m^*) = \|\theta - \theta_m^*\|_\Phi^2$, so Eq. (8) computes the unique $\Phi$-projection of $\theta_m^*$ onto the convex hull of the uploads. The coefficients $\omega_m^*$ may be nonunique even though the recovered expert is unique. We next introduce a structural condition that enables a recovery guarantee.

**Assumption 3 (Near-Pure Client Anchors)** *For each task $m \in [M]$, there exists a client $i_m \in [N]$ satisfying $\alpha_{i_m,m} \geq 1 - \varepsilon$ for some $\varepsilon \in [0, 1/2)$.*

This is an approximate separability condition analogous to anchor assumptions in topic modeling and separable non-negative matrix factorization (Arora et al., 2012). It requires only one near-pure client per task. By Theorem 1, each such client optimum lies near the corresponding task expert and therefore provides a nearby feasible point for Eq. (8).

**Theorem 2 (Population Expert Recovery)** *Under Assumptions 2 and 3, with exact client uploads $\{\theta_i^*\}$ and population task losses, the extractor in Eq. (8) satisfies $\|\theta_m^{\mathrm{ext}} - \theta_m^*\|_{\mathrm{F}} \leq \kappa(\Phi)^{1/2} \varepsilon \operatorname{diam}_{\mathrm{F}}(\Theta)$ for every $m \in [M]$, where $\kappa(\Phi) = \sigma_{\max}(\Phi)/\sigma_{\min}(\Phi)$ and $\operatorname{diam}_{\mathrm{F}}(\Theta) = \max_{m,k} \|\theta_m^* - \theta_k^*\|_{\mathrm{F}}$ for $\Theta = \{\theta_m^*\}_{m=1}^M$.*

The recovery error decreases linearly with anchor impurity and vanishes for pure anchors. With estimated client uploads and finite public data, additional client-estimation and public-sampling errors arise and vanish as the corresponding sample sizes grow (Proposition 3, Appendix D.2).

**Two identifiability limits.** The supervised extractor is task-indexed because its $m$th output is defined using the corresponding task loss $\mathcal{L}_m$. To isolate the role of this supervision, call an estimator *label-blind* if it receives no task-indexed side information, and *pool-only* if it uses only client uploads. We first show that label-blind observations cannot identify which expert corresponds to which task, even when the expert set itself is recoverable.

**Theorem 3 (Task-Identity Indeterminacy)** *Let $M \geq 2$ and let $\mathcal{O}(\mathcal{I}) := \{\mathcal{I}^\pi : \pi \in S_M\}$ denote the permutation orbit of an instance $\mathcal{I}$. Every $\mathcal{I}^\pi \in \mathcal{O}(\mathcal{I})$ induces identical label-free observables. Consequently, for any label-blind estimator $\mathcal{A}$,*

$$\max_{\mathcal{J} \in \mathcal{O}(\mathcal{I})} \mathbb{E}_{\mathcal{A}}\left[\max_{m \in [M]} \|\hat{\theta}_m - \theta_m^*(\mathcal{J})\|_{\mathrm{F}}\right] \geq \frac{1}{2}\operatorname{diam}_{\mathrm{F}}(\Theta) \geq \frac{\mathrm{sep}}{2}. \tag{9}$$

*This holds for every $N$, every anchor quality $\varepsilon$, including $\varepsilon = 0$, and every amount of local data and computation.*

Thus even perfect recovery of the vertex set cannot determine its task correspondence without task-indexed supervision. We next ask whether the unordered expert set itself can be recovered from client uploads.

**Theorem 4 (Vertex-Set Non-Identifiability from Interior Mixtures)** *Suppose $h > 0$ and every client is $\gamma$-interior for some $\gamma \in (0, 1]$, so that $\alpha_{im} \geq \gamma/M$ for all $i \in [N]$ and $m \in [M]$. Let $t := \min\{\gamma, \mathrm{sep}/(2 \operatorname{diam}_{\mathrm{F}}(\Theta))\}$. Then there exists a distinct valid instance $\tilde{\mathcal{I}}$ with its own expert set and mixture weights that induces the same upload pool as $\mathcal{I}$. For any pool-only estimator $\mathcal{A}$, even under the most favorable relabeling,*

$$\max_{\mathcal{J} \in \{\mathcal{I}, \tilde{\mathcal{I}}\}} \mathbb{E}_{\mathcal{A}}\left[\min_{\pi \in S_M} \max_{m \in [M]} \|\hat{\theta}_{\pi(m)} - \theta_m^*(\mathcal{J})\|_{\mathrm{F}}\right] \geq \frac{M-1}{2M}\, t\, h. \tag{10}$$

The two results isolate distinct requirements. Theorem 3 shows that task-indexed side information is needed to identify the expert-to-task correspondence. Theorem 4 shows that interior upload pools can be consistent with different expert sets, so upload-only vertex recovery requires sufficient boundary information. Near-pure client anchors provide a sufficient form of such information.

### 3.3 Routing Requirements for Specialization

After recovering the task experts, specialization also requires routing each input toward the appropriate expert. A task-aligned router should therefore satisfy two complementary properties. Its mean expert usage should reflect the client's task mixture $\alpha_i$, while its routing decisions should remain input dependent so that different tasks can activate different experts. We next characterize how these two properties affect personalized risk.

**Theorem 5 (Mean Calibration and Input-Dependent Routing)** *Suppose Assumption 2 holds, $h > 0$, and $\|\phi(x)\|_2 \le Q$ almost surely. Let $D_\Theta := \mathrm{diam}_{\mathrm{F}}(\Theta)$ and let $p_i : \mathcal{X} \to \Delta^{M-1}$ be any input-dependent routing policy. Define its mean usage as $\bar{p}_i := \mathbb{E}_{x\sim\mathcal{D}_i}[p_i(x)]$ and its routing variation as $V_i(p_i) := \mathbb{E}_{x\sim\mathcal{D}_i}\|p_i(x) - \bar{p}_i\|_1$. Then the population risk $\mathcal{R}_i$ of routing over the true experts satisfies*

$$\sigma_{\min}(\Phi)h^2\|\bar{p}_i - \alpha_i\|_\infty^2 - Q^2 D_\Theta^2 V_i(p_i) \le \mathcal{R}_i - \mathcal{L}_i(\theta_i^*) \le \frac{\sigma_{\max}(\Phi)D_\Theta^2}{4}\|\bar{p}_i - \alpha_i\|_1^2 + Q^2 D_\Theta^2 V_i(p_i). \tag{11}$$

Theorem 5 separates the effects of mean calibration and input-dependent routing. If $p_i(x) \equiv \alpha_i$, the router recovers the optimal static adapter $\theta_i^*$ but cannot improve on it. Closing the mixing gap therefore requires input-dependent routing that is aligned with the underlying tasks. Corollary 3 shows that such task-aligned routing approaches the oracle risk as the routing and expert-recovery errors vanish.

**Proposition 1 (Uniform Mean Usage Limits Specialization)** *Suppose the $M$ experts admit a task matching $\pi$ such that, for every task $m$ present at client $i$, the router assigns its matched expert average probability at least $1-\beta$ on inputs from $\mathcal{P}_m$. Then $\|\bar{p}_i - \Pi_\pi \alpha_i\|_1 \le 2\beta$, where $\Pi_\pi$ is the permutation matrix for this matching. If additionally $\|\bar{p}_i - u\|_1 \le \delta_{\mathrm{bal}}$ for $u = \frac{1}{M}\mathbf{1}$, then $\beta \ge \frac{1}{2}(\|\alpha_i - u\|_1 - \delta_{\mathrm{bal}})_+$. Hence exact uniform mean usage is incompatible with exact specialization for any client with $\alpha_i \ne u$.*

Proposition 1 therefore identifies the client mixture $\alpha_i$, rather than uniform usage, as the appropriate mean-routing target for specialization. Since $\alpha_i$ is latent, FedSEE uses a contribution-based proxy $\tilde{c}_i$ for mixture calibration. Let $\hat{\alpha}_i$ denote the barycentric estimate of Proposition 5, satisfying $\|\hat{\alpha}_i - \alpha_i\|_1 \le \eta_C$, and define $\delta_i^{\mathrm{proxy}} := \|\tilde{c}_i - \hat{\alpha}_i\|_1$. Then $\|\tilde{c}_i - \alpha_i\|_1 \le \eta_C + \delta_i^{\mathrm{proxy}}$. Consequently, Pinsker's inequality gives $\|\bar{p}_i - \alpha_i\|_1 \le \sqrt{2\,\mathrm{KL}(\tilde{c}_i\|\bar{p}_i)} + \eta_C + \delta_i^{\mathrm{proxy}}$ (Corollary 7). Thus anchoring mean routing toward $\tilde{c}_i$ controls mixture calibration up to the certified mixture-estimation error and the additional proxy error. Input-level specialization remains the responsibility of the task loss, as characterized by Corollary 3. Section 4 instantiates these two complementary objectives.

## 4 FedSEE: The LoRA Instantiation

**FedSEE** (Federated Supervised Expert Extraction) instantiates the framework with LoRA, as summarized in Figure 2. At a single adapted projection with $A^{\mathrm{global}}$ fixed, the routed LoRA component matches Eq. (2) with $\phi(x) = sA^{\mathrm{global}}x$ and $\theta = B$. Under the conditions of Theorem 1, this gives $B_i^* = \sum_m \alpha_{im} B_m^*$ exactly at the projection level. For the composed transformer, Appendix D.8 bounds the departure from this geometry under a local first-order approximation.

FedSEE also maintains an always-on shared LoRA expert $(A^{\mathrm{s}}, B^{\mathrm{s}})$ to capture task-shared adaptation outside the routed expert pool. Its deployed forward pass is

$$h(x) = \underbrace{W_0 x + sB^{\mathrm{s}}A^{\mathrm{s}}x}_{\tilde{f}_0(x)} + s\sum\nolimits_{m\in\mathcal{U}_i} p_{im}(x) B_m A^{\mathrm{global}} x, \tag{12}$$

where $s = \alpha_{\mathrm{LoRA}}/r$ and $\mathcal{U}_i$ is client $i$'s inference set. The shared expert is always active and is not routed. For a single projection, Eq. (12) therefore matches the routed model in Eq. (5) with the shared term absorbed into $\tilde{f}_0$.

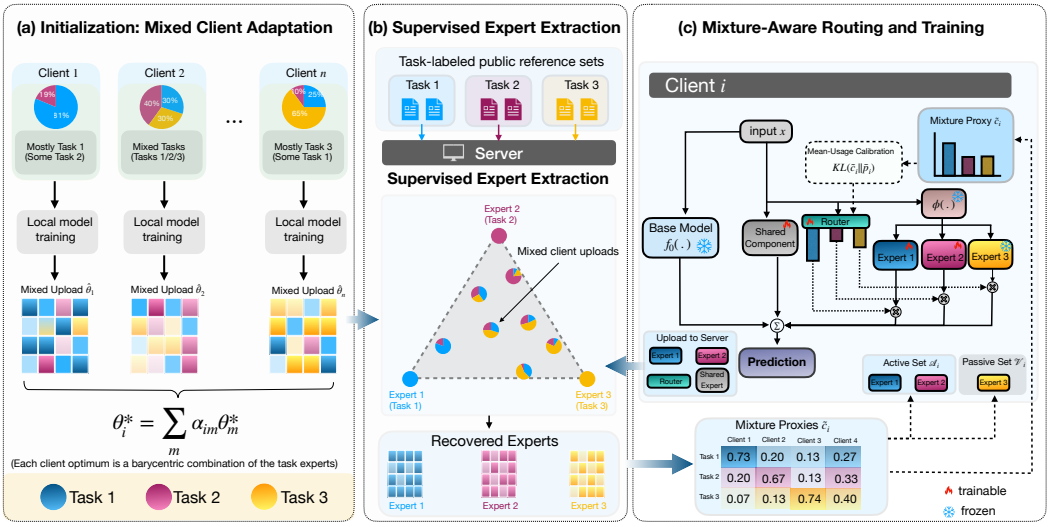



Figure 2: **FedSEE overview.** Supervised extraction recovers task experts from mixed client updates, followed by mixture-aware routing with client-specific calibration.

**Supervised expert extraction.** FedSEE begins with one initialization round in which each client trains a common LoRA pair $(A^{(0)}, B^{(0)})$ and returns $(A_i, B_i)$. The server sets $A^{\text{global}} = N^{-1}\sum_i A_i$ and keeps it fixed thereafter. Let $\mathcal{I}^{(t)} := \{(i, m') : B^i_{m'} \text{ is uploaded at round } t\}$ denote the current upload pool. Using the task-labeled public loss, expert $m$ is extracted as

$$\omega_m^* \in \underset{\omega_m \in \Delta^{|\mathcal{I}^{(t)}|-1}}{\arg\min}\ \widehat{\mathcal{L}}^s_m\left(\sum\nolimits_{(i,m')\in\mathcal{I}^{(t)}} \omega_{m,(i,m')} B^i_{m'}\right), \qquad B^{\text{expert}}_m = \sum\nolimits_{(i,m')\in\mathcal{I}^{(t)}} \omega^*_{m,(i,m')} B^i_{m'}. \tag{13}$$

FedSEE optimizes the $M$ taskwise programs jointly while constraining the raw client contributions $c_{mi} := \sum_{m':(i,m')\in\mathcal{I}^{(t)}} \omega_{m,(i,m')}$ to satisfy $\sum_i c_{mi} = 1$ and $\sum_m c_{mi} = M/N$. Balanced transport prevents repeated concentration on the same clients but can exclude independently optimal extraction weights. Theorem 2 applies to independent extraction, while Proposition 11 quantifies the coupling excess and gives exact recovery for transport-compatible pure anchors. Solver details appear in Appendix C.2. The same contribution matrix $C = [c_{mi}]$ also provides a client-specific relevance signal for capacity allocation and router calibration.

**Mixture-aware capacity and routing.** For client $i$, the contribution column $c_{\cdot i}$ ranks the recovered experts by their relevance to that client. FedSEE assigns the highest-ranked experts to an inference set $\mathcal{U}_i$ and selects an active subset $\mathcal{A}_i \subseteq \mathcal{U}_i$ for local updates. Experts in $\mathcal{U}_i \setminus \mathcal{A}_i$ remain available to the router but are detached during backpropagation, allowing inference and training capacity to differ. Normalizing the retained contributions over $\mathcal{U}_i$ gives the mixture proxy $\tilde{c}_i$ used to calibrate the router. Further details of the capacity assignment appear in Appendix C.3, while Appendix B.5 examines its empirical training-capacity tradeoff. Let $\psi_i = \{W_r^\ell, b_r^\ell\}_{\ell\in\mathcal{P}}$ denote the client $i$'s local copy of the per-layer router parameters. Local training uses

$$\min_{\{B_m\}_{m\in\mathcal{A}_i},(A_i^s,B_i^s),\psi_i} \mathcal{L}_{\text{task}}(\{B_m\}_{m\in\mathcal{U}_i}, \psi_i; \mathcal{D}_i) + \frac{\beta_{\text{KL}}}{|\mathcal{P}|}\sum_{\ell\in\mathcal{P}} \text{KL}\big(\tilde{c}_i \| \bar{p}_i^\ell\big), \quad \bar{p}_i^\ell := \mathbb{E}_{x\sim\mathcal{D}_i}[p_i^\ell(x)]. \tag{14}$$

The shared pair $(A_i^s, B_i^s)$ is updated on the same objective and aggregated by FedAvg. The task loss learns input-dependent expert assignments, while the KL term calibrates mean usage toward the client-specific proxy $\tilde{c}_i$, as motivated by Section 3.3. No uniform mean-usage penalty is used. After each round the routers are aggregated across clients, with the corresponding justification deferred to Appendix C.4. For evaluation, client $i$ uses its personalized model Eq. (12), comprising the locally updated active experts, shared expert, and router together with the recovered passive experts in $\mathcal{U}_i$. The complete procedure appears in Algorithm 1 in Appendix C.

Table 1: Personalized accuracy (%) under compound heterogeneity ($\alpha_{\mathrm{Dir}} = 0.1$) with RoBERTa-large. Results are reported as mean ± standard deviation over three seeds.

| Method | Overall | Worst-50% | Worst-25% |
|---|---|---|---|
| FedIT | $78.06 \pm 2.04$ | $69.35 \pm 5.17$ | $64.44 \pm 6.09$ |
| FFA-LoRA | $73.43 \pm 1.09$ | $64.64 \pm 2.27$ | $62.21 \pm 2.49$ |
| IFCA+LoRA | $79.42 \pm 0.99$ | $70.95 \pm 2.42$ | $67.37 \pm 1.49$ |
| FedSA-LoRA | $\underline{79.75 \pm 1.37}$ | $71.89 \pm 2.60$ | $66.78 \pm 2.83$ |
| FedDPA-T | $79.20 \pm 1.00$ | $71.40 \pm 3.74$ | $67.34 \pm 4.20$ |
| FedLEASE | $79.52 \pm 1.80$ | $\underline{72.46 \pm 3.99}$ | $\underline{69.10 \pm 3.77}$ |
| **FedSEE (L1, Ours)** | $\mathbf{82.65 \pm 0.35}$ | $\mathbf{76.54 \pm 1.65}$ | $\mathbf{72.75 \pm 2.21}$ |
| **Gain over best baseline** | $\mathbf{+2.90}$ | $\mathbf{+4.08}$ | $\mathbf{+3.65}$ |

# 5 EXPERIMENTS

**Synthetic validation.** Under the assumptions of Section 3, controlled experiments instantiate the simplex geometry illustrated in Figure 1. They confirm the main theoretical predictions. Expert-recovery error scales linearly with anchor impurity, task supervision resolves the identity ambiguity of label-free recovery, sufficiently interior upload pools exhibit the non-identifiability of Theorem 4, and mixture-aware routing avoids the cost of uniform calibration. Appendix B.1 provides the full construction and additional results.

## 5.1 NATURAL LANGUAGE UNDERSTANDING BENCHMARK

We evaluate FedSEE on six GLUE/SuperGLUE tasks (Wang et al., 2018; 2019): SST-2, QNLI, CoLA, QQP, WiC, and BoolQ, forming $M = 6$ tasks. Client task proportions follow a symmetric Dirichlet distribution with concentration $\alpha_{\mathrm{Dir}}$, where smaller values produce more concentrated mixtures and larger values approach uniform composition. Each client has disjoint 200-example validation and test sets matched to its realized task mixture. The validation set is used only for model selection, while the test set is held out for final personalized evaluation.

FedSEE assumes access to a small task-labeled public reference set, used only for server-side expert extraction in Eq. (13). We use 1,000 public examples per task by default, all disjoint from client training, validation, and test data. Appendix B.6 studies sensitivity to the amount of public supervision and shows that the advantage is retained with only 100 labeled examples per task. By default, all six recovered experts are available at inference while each client updates one active expert, denoted **FedSEE (L1)**. We compare against averaging methods (**FedIT** (Zhang et al., 2024), **FFA-LoRA** (Sun et al., 2024)), personalized LoRA methods (**FedSA-LoRA** (Guo et al., 2025), **FedDPA-T** (Yang et al., 2024)), and clustering methods (**IFCA+LoRA** (Ghosh et al., 2020), **FedLEASE** (Wang et al., 2025)). Implementation details appear in Appendix B.3, with conceptual comparisons in Appendix A.

## 5.2 COMPOUND HETEROGENEITY AT $\alpha_{\mathrm{Dir}} = 0.1$

**Protocol.** We use RoBERTa-large (Liu et al., 2019) with $N = 10$ fully participating clients, $T = 10$ communication rounds, and $E = 5$ local epochs. Each method selects its learning rate from $\{5 \times 10^{-4}, 10^{-3}, 5 \times 10^{-3}\}$ and is evaluated over three random seeds. Complete optimization and model-selection details appear in Appendix B.3. We report overall personalized accuracy together with *Worst-50%* and *Worst-25%*, averaging the lowest $\lceil N/2 \rceil$ and $\lceil N/4 \rceil$ client accuracies, respectively. Per-task results appear in Appendix B.4.

**Results.** FedSEE achieves the highest overall personalized accuracy (Table 1), improving over the strongest baseline from 79.75% to 82.65%, a gain of 2.90 points. FedSEE improves Worst-50% accuracy from 72.46% to 76.54% and Worst-25% accuracy from 69.10% to 72.75%, corresponding to gains of 4.08 and 3.65 points, respectively. The strongest baseline varies by metric: FedSA-LoRA overall and FedLEASE on both client-level tail metrics. Component ablations appear in Appendix B.6.

Table 2: Personalized accuracy (%) across the heterogeneity sweep. Differences are method minus FedSEE, reported as mean ± standard deviation over three matched seeds.

| **Method** | $\alpha_{\text{Dir}}=0$ | 0.1 | 0.5 | 1.0 | **uniform** |
|---|---|---|---|---|---|
| **FedSEE (L1, Ours)** | **76.46** | **74.95** | **71.28** | **72.10** | 69.97 |
| *paired difference from FedSEE (L1)* | | | | | |
| FedIT | $-15.47 \pm 0.46$ | $-6.40 \pm 1.07$ | $-2.18 \pm 0.42$ | $\underline{-0.60 \pm 1.02}$ | $+0.10 \pm 0.22$ |
| FFA-LoRA | $-16.30 \pm 0.62$ | $-13.43 \pm 1.09$ | $-9.16 \pm 2.15$ | $-10.14 \pm 2.20$ | $-7.86 \pm 1.33$ |
| FedSA-LoRA | $-0.83 \pm 0.42$ | $\underline{-1.59 \pm 0.65}$ | $-2.37 \pm 0.46$ | $-3.07 \pm 0.95$ | $-3.26 \pm 0.92$ |
| FedDPA-T | $-5.44 \pm 0.51$ | $-3.08 \pm 1.57$ | $\underline{-1.36 \pm 0.62}$ | $-1.72 \pm 0.83$ | $-1.18 \pm 0.31$ |
| IFCA+LoRA | $-1.46 \pm 0.39$ | $-1.89 \pm 0.87$ | $-2.46 \pm 0.44$ | $-1.31 \pm 0.43$ | $-0.19 \pm 2.09$ |
| FedLEASE | $\underline{-0.57 \pm 0.56}$ | $-3.29 \pm 1.41$ | $-1.44 \pm 0.33$ | $-0.74 \pm 0.47$ | $\underline{+0.38 \pm 1.07}$ |
| *local-only* | $-0.65 \pm 0.20$ | $-1.15 \pm 1.00$ | $-1.96 \pm 0.68$ | $-3.52 \pm 0.73$ | $-5.33 \pm 0.90$ |
| *centralized* | $-0.67 \pm 0.27$ | $+0.21 \pm 0.29$ | $+3.76 \pm 1.18$ | $+3.52 \pm 0.50$ | $+6.15 \pm 0.74$ |

### 5.3 Where the Advantage Holds: A Heterogeneity Sweep

We next evaluate robustness from pure single-task clients to near-uniform mixtures. This sweep tests whether FedSEE's task-expert representation remains effective across regimes that increasingly favor client partitioning at one extreme and global aggregation at the other. We use `distilroberta-base` with $N = 12$ clients and evaluate $\alpha_{\text{Dir}} \in \{0, 0.1, 0.5, 1.0, 50\}$, denoting $\alpha_{\text{Dir}} = 50$ as *uniform*. All methods use matched client partitions and random seeds. We also report *local-only* training and a centralized task-specific reference trained on pooled task data. Table 2 reports FedSEE's absolute accuracy and paired differences from competing methods. Full construction appears in Appendix B.3.

**Robustness across mixture regimes.** FedSEE achieves the highest federated accuracy in four of five regimes and is within 0.38 points of the strongest baseline at the near-uniform endpoint. The closest baseline changes across the sweep, while FedSEE uses the same task-expert representation and client-dependent routing throughout.

**Positive transfer across heterogeneity.** Figure 3 shows that FedSEE improves over local-only training in every regime, with gains increasing from 0.65 points for pure clients to 5.33 points near uniform. In contrast, every competing federated method falls below local-only training in at least one regime. The training-capacity tradeoff across the heterogeneity sweep is analyzed in Appendix B.5.

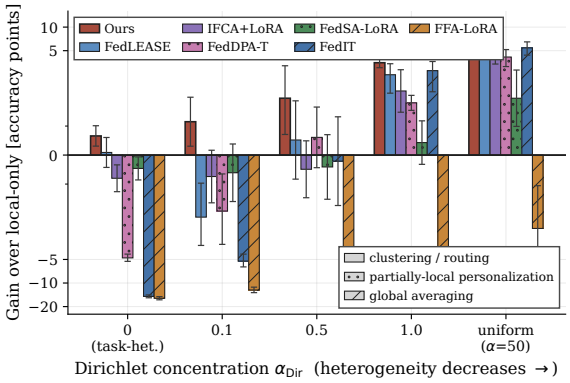



Figure 3: Gain over local-only training across heterogeneity ($y$-axis uses symlog scaling).

## 6 Conclusion

We studied personalization under compound heterogeneity, where clients may contain mixtures of latent tasks. Under our shared-feature quadratic model, client optima are mixture-weighted barycenters of task optima, with affine independence inducing a simplex geometry. This exposes limitations of single-adapter aggregation and label-free expert construction, motivating task-supervised expert recovery and mixture-aware, input-dependent routing. FedSEE instantiates these principles with LoRA by extracting task-aligned experts from mixed client updates and routing them using client-specific mixture calibration. On natural language understanding tasks, FedSEE improves average and tail personalization and remains competitive from pure to near-uniform client populations. These results support explicit modeling of within-client mixture structure across heterogeneity regimes.

### AI USE STATEMENT

The authors used Claude and GPT-5 as supporting tools during the preparation of this work. These tools were used to suggest experimental directions and ablations, assist with literature discovery, improve the clarity and presentation of existing manuscript text, and occasionally assist with code and LaTeX debugging or modification. AI tools were not used to develop the core research idea, formulate the theoretical results, derive proofs, design the proposed methodology, generate experimental data, or interpret experimental results. Literature identified with AI assistance was independently verified by reading the original sources, and any AI-assisted code modifications were validated through testing, logging, and debugging. All scientific decisions, claims, experimental conclusions, and final manuscript content were reviewed and approved by the authors.

### ETHICS STATEMENT

This work uses publicly available benchmark datasets and simulated clients and does not involve human-subject experiments, participant recruitment, private user data, personally identifiable information, or newly collected human annotations. The datasets and pretrained models used in the experiments are used in accordance with their publicly available licenses and terms. Although the proposed setting is motivated by distributed learning, the method does not provide a formal privacy guarantee against information leakage from shared model updates. Such privacy protection is outside the scope of this work and may be incorporated through complementary privacy-preserving mechanisms when required.

### REPRODUCIBILITY STATEMENT

We provide the assumptions and complete proofs for the main theoretical results in the appendix. The paper and appendix describe the dataset construction, client-mixture generation procedure, model and training configurations, baselines, hyperparameters, routing and extraction procedures, ablations, evaluation protocol, and computational setup.

## REFERENCES

Sanjeev Arora, Rong Ge, and Ankur Moitra. Learning topic models – going beyond SVD. In *2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, pp. 1–10, 2012. doi: 10.1109/FOCS.2012.49.

Jiamu Bai, Daoyuan Chen, Bingchen Qian, Liuyi Yao, and Yaliang Li. Federated fine-tuning of large language models under heterogeneous tasks and client resources. *Advances in Neural Information Processing Systems*, 37:14457–14483, 2024.

Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning: A meta-learning approach. *arXiv preprint arXiv:2002.07948*, 2020.

Avishek Ghosh, Jichan Chung, Dong Yin, and Kannan Ramchandran. An efficient framework for clustered federated learning. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 19586–19597. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/e32cc80bf07915058ce90722ee17bb71-Paper.pdf.

Pengxin Guo, Shuang Zeng, Yanran Wang, Huijie Fan, Feifei Wang, and Liangqiong Qu. Selective aggregation for low-rank adaptation in federated learning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=iX3uESGdsO.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Chengsong Huang, Qian Liu, Bill Yuchen Lin, Tianyu Pang, Chao Du, and Min Lin. LoraHub: Efficient cross-task generalization via dynamic LoRA composition. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=TrloAXEJ2B.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=6t0Kwf8-jrj.

Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. SCAFFOLD: Stochastic controlled averaging for federated learning. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5132–5143. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/karimireddy20a.html.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2:429–450, 2020.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Othmane Marfoq, Giovanni Neglia, Aurélien Bellet, Laetitia Kameni, and Richard Vidal. Federated multi-task learning under a mixture of distributions. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 15434–15447. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/82599a4ec94aca066873c99b4c741ed8-Paper.pdf.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-Efficient Learning of Deep Networks from Decentralized Data. In Aarti Singh and Jerry Zhu (eds.), *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pp. 1273–1282. PMLR, 20–22 Apr 2017. URL https://proceedings.mlr.press/v54/mcmahan17a.html.

Le-Tuan Nguyen, Minh-Duong Nguyen, Seon-Geun Jeong, Dung D Le, and Quoc-Viet Pham. Communication-efficient and accurate approach for aggregation in federated low-rank adaptation. *arXiv preprint arXiv:2509.26399*, 2025.

Yichen Ruan and Carlee Joe-Wong. FedSoft: Soft clustered federated learning with proximal local updating. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 8124–8131, 2022.

Viraj Shah, Nataniel Ruiz, Forrester Cole, Erika Lu, Svetlana Lazebnik, Yuanzhen Li, and Varun Jampani. ZipLoRA: Any subject in any style by effectively merging LoRAs. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024*, pp. 422–438, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-73232-4.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=B1ckMDqlg.

Raghav Singhal, Kaustubh Ponkshe, and Praneeth Vepakomma. FedEx-LoRA: Exact aggregation for federated parameter-efficient fine-tuning of foundation models. In *NeurIPS 2024 Workshop on Fine-Tuning in Modern Machine Learning: Principles and Scalability*, 2024. URL https://openreview.net/forum?id=vOtEFcKIi5.

Youbang Sun, Zitao Li, Yaliang Li, and Bolin Ding. Improving LoRA in privacy-preserving federated learning. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=NLPzL6HWNl.

Canh T. Dinh, Nguyen Tran, and Josh Nguyen. Personalized federated learning with Moreau envelopes. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 21394–21405. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/f4f1f13c8289ac1b1ee0ff176b56fc60-Paper.pdf.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In Tal Linzen, Grzegorz Chrupała, and Afra Alishahi (eds.), *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 353–355, Brussels, Belgium, November 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5446. URL https://aclanthology.org/W18-5446/.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32, 2019.

Lei Wang, Jieming Bian, Letian Zhang, and Jie Xu. Adaptive LoRA experts allocation and selection for federated fine-tuning. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 76018–76045. Curran Associates, Inc., 2025. doi: 10.52202/085713-2553. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/6df1b2b45e64d402588746f79b68b82c-Paper-Conference.pdf.

Xi Wang, Soufiane Hayou, and Eric Nalisnick. The myth of expert specialization in MoEs: Why routing reflects geometry, not necessarily domain expertise. *arXiv preprint arXiv:2604.09780*, 2026.

Ziyao Wang, Zheyu Shen, Yexiao He, Guoheng Sun, Hongyi Wang, Lingjuan Lyu, and Ang Li. FLoRA: Federated fine-tuning large language models with heterogeneous low-rank adaptations. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 22513–22533. Curran Associates, Inc., 2024. doi: 10.52202/079017-0708. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/28312c9491d60ed0c77f7fff4ad86dd1-Paper-Conference.pdf.

Xun Wu, Shaohan Huang, and Furu Wei. Mixture of LoRA experts. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=uWvKBCYh4S.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin A Raffel, and Mohit Bansal. TIES-merging: Resolving interference when merging models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 7093–7115. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/1644c9af28ab7916874f6fd6228a9bcf-Paper-Conference.pdf.

Yiyuan Yang, Guodong Long, Tao Shen, Jing Jiang, and Michael Blumenstein. Dual-personalizing adapter for federated foundation models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 39409–39433. Curran Associates, Inc., 2024. doi: 10.52202/079017-1245. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/45a30141c6719e9cfedfb51f1c665a37-Paper-Conference.pdf.

Haoran Zhang, Dongjun Kim, Seohyeon Cha, and Haris Vikalo. Fedrot-loRA: Mitigating rotational misalignment in federated loRA. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=2X8Qi3VdjA.

Jianyi Zhang, Saeed Vahidian, Martin Kuo, Chunyuan Li, Ruiyi Zhang, Tong Yu, Guoyin Wang, and Yiran Chen. Towards building the FederatedGPT: Federated instruction tuning. In *ICASSP 2024-2024 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 6915–6919. IEEE, 2024.

Ziyu Zhao, Tao Shen, Didi Zhu, Zexi Li, Jing Su, Xuwu Wang, and Fei Wu. Merging LoRAs like playing LEGO: Pushing the modularity of LoRA to extremes through rank-wise clustering. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=j6fsbpAllN.

Shenghe Zheng, Hongzhi Wang, Chenyu Huang, Xiaohui Wang, Tao Chen, Jiayuan Fan, Shuyue Hu, and Peng Ye. Decouple and orthogonalize: A data-free framework for LoRA merging. *arXiv preprint arXiv:2505.15875*, 2025.

## APPENDIX CONTENTS

**A Related Work** **15**

**B Experimental Details and Additional Results** **16**

B.1 Synthetic Validation of the Theory . . . . . 16

B.2 Geometry and Routing Beyond the Exact Model . . . . . 18

B.3 Experimental Setup . . . . . 22

B.4 Detailed Results under Compound Heterogeneity . . . . . 26

B.5 Training-Capacity Analysis . . . . . 26

B.6 Component Ablations . . . . . 26

B.7 Client-Mixture Estimation . . . . . 28

**C FedSEE Algorithm and Implementation Details** **28**

C.1 Projection-Level Reduction . . . . . 28

C.2 Joint Extraction with Balanced Transport . . . . . 30

C.3 The Mixture Proxy and Capacity Assignment . . . . . 30

C.4 Router Parameterization and Aggregation . . . . . 31

**D Theoretical Results and Proofs** **31**

D.1 Barycentric Geometry and the Mixing Gap . . . . . 32

D.2 Supervised Expert Recovery and Finite-Sample Guarantees . . . . . 34

D.3 Identifiability Limits . . . . . 38

D.4 Mixture Estimation and Static Reconstruction . . . . . 41

D.5 Routing and Task-Aligned Specialization . . . . . 43

D.6 Extensions Beyond the Exact Squared-Loss Setting . . . . . 51

D.7 Additional Model-Class Instantiations . . . . . 55

D.8 Deep-Model Extension for FedSEE . . . . . 55

D.9 Recovery under Balanced Transport . . . . . 57

D.10 Stability of the Deployed Contribution Matrix . . . . . 58

**E Future Directions** **60**

# A Related Work

**Statistical Heterogeneity and Mixture Models.** The canonical challenge in federated learning is the divergence of client data distributions. Traditional methods mitigate this via proximal regularization (FedProx (Li et al., 2020)), control variates (SCAFFOLD (Karimireddy et al., 2020)), or meta-learning-based personalization (Per-FedAvg (Fallah et al., 2020); pFedMe (T. Dinh et al., 2020)). A related line of work explicitly models each client's data as a mixture of latent source distributions. FedEM (Marfoq et al., 2021) jointly learns shared latent components and client-specific mixture weights, while FedSoft (Ruan & Joe-Wong, 2022) learns cluster and personalized models under soft client membership. These methods estimate the latent components as part of federated optimization. FedSEE studies a different recovery problem in which local optimization has already produced mixed client adaptations and the server must recover task-indexed experts from the resulting upload pool. Hard clustering methods such as IFCA (Ghosh et al., 2020) instead assign each client to a single component. We evaluate this paradigm as IFCA+LoRA by pairing each cluster with its own dedicated adapter. Without task-indexed side information, latent components remain semantically unlabeled, and without sufficient boundary information even the unordered expert set may be non-identifiable (Theorems 3 and 4). By leveraging task-labeled public data, FedSEE fixes the task correspondence and, under the anchor condition, recovers task-indexed experts with the guarantee of Theorem 2.

**Federated LoRA Fine-Tuning.** Early federated PEFT frameworks such as FedIT (Zhang et al., 2024) apply standard federated aggregation to LoRA parameters. Because separately averaging the two LoRA factors does not generally reproduce the average of their products, $\bar{B}\bar{A} \neq \overline{BA}$, subsequent methods focus on reducing this aggregation mismatch. FedEx-LoRA (Singhal et al., 2024) corrects the residual aggregation error, FLoRA (Wang et al., 2024) stacks client updates along the rank dimension, FLoRA-NA (Nguyen et al., 2025) constructs surrogate aggregated factors, and FedRot-LoRA (Zhang et al., 2026) aligns equivalent low-rank factorizations before aggregation. These methods target an accurate global adapted model rather than the recovery of task-indexed experts. Other approaches exploit the asymmetric roles of the LoRA factors. FedSA-LoRA (Guo et al., 2025) shares only the $A$ factor, while FFA-LoRA (Sun et al., 2024) fixes one randomly initialized factor and optimizes only the other. FedSEE instead learns a shared projection $A^{\mathrm{global}}$ during initialization and then freezes it. Beyond reducing aggregation ambiguity, this common projection gives the theoretical model a shared feature space in which client optima in $B$-space become mixture-weighted combinations of task optima (Theorem 1).

**Federated Mixture-of-Experts.** A complementary line of work preserves heterogeneity through multiple adapters or experts. MoLE (Wu et al., 2024) learns gating functions over a collection of independently trained LoRA experts. In federated settings, FedLEASE (Wang et al., 2025) clusters client representations to construct domain-specific experts and subsequently uses an adaptive mixture-of-experts mechanism to select among them. Related personalized approaches such as FedDPA (Yang et al., 2024) maintain global and local adapters and combine them through instance-dependent weighting. FedSEE differs from clustering-based expert construction in two respects. First, each mixed client upload may contribute fractionally to multiple recovered task experts through the extraction weights and the resulting contribution matrix $C$. Second, the recovered experts are optimized against task-labeled public risks, which assigns them explicit task identities rather than identities induced by client clusters. Resource-heterogeneous FedLoRA methods such as FlexLoRA (Bai et al., 2024) instead adapt LoRA rank to client capabilities and use SVD-based redistribution of the aggregated update. FedSEE addresses a different form of system heterogeneity by separating the experts available for inference from those a client can actively update through its capacity-aware active/passive partition.

**Centralized Model and Adapter Composition.** A parallel non-federated literature studies combining independently trained task-specific models or adapters through weight arithmetic (Task Arithmetic (Ilharco et al., 2023)), conflict resolution (TIES-Merging (Yadav et al., 2023)), orthogonalization (ZipLoRA (Shah et al., 2025); DO-Merging (Zheng et al., 2025)), or modular decomposition (LoRA-LEGO (Zhao et al., 2025)). LoRAHub (Huang et al., 2024) uses a small number of examples from an unseen task to determine composition weights over a collection of already trained LoRA modules. These methods solve a forward composition problem in which task-specific models or adapters are already available and the goal is to combine them. FedSEE instead addresses an inverse extraction problem. In the federated mixture setting, task-specific experts are not directly observed

Table 3: Expert recovery error against anchor impurity.

| $\varepsilon$ | 0 | 0.05 | 0.10 | 0.20 | 0.30 | 0.40 |
|---|---|---|---|---|---|---|
| exact uploads | 0.000 | 0.228 | 0.455 | 0.911 | 1.366 | 1.821 |
| finite, $n_i = 10^3$ | 0.167 | 0.380 | 0.596 | 1.041 | 1.574 | 1.980 |
| finite, $n_i = 1.6 \times 10^4$ | 0.044 | 0.242 | 0.483 | 0.930 | 1.387 | 1.836 |
| bound $\kappa(\Phi)^{1/2}\varepsilon\,\mathrm{diam}_{\mathrm{F}}(\Theta)$ | 0.000 | 0.600 | 1.200 | 2.400 | 3.600 | 4.800 |

and the server receives adaptations that already mix information from multiple tasks. FedSEE uses task-labeled public data to recover and identify the latent task experts from this upload pool before learning how to route among them.

# B Experimental Details and Additional Results

This appendix provides the experimental details and additional results supporting Section 5. We first present controlled synthetic experiments that isolate the predictions of the theoretical analysis. We then give the complete specification of the natural-language experiments, followed by detailed benchmark results, the training-capacity analysis, component ablations, and client-mixture estimation results.

## B.1 Synthetic Validation of the Theory

This subsection provides controlled validation of the theoretical results in Section 3. We construct a synthetic setting in which the assumptions can be enforced directly and examine the barycentric characterization, expert-recovery guarantees, identifiability limits, and routing predictions. These experiments isolate the behavior predicted by the theory before turning to the transformer experiments of Section 5.

### B.1.1 Synthetic Setup and Barycentric Validation

We instantiate the LoRA model of Section 4 directly: $\phi(x) = Ax$ with $A \in \mathbb{R}^{r\times d}$ frozen, $f_0(x) = W_0x$, and $\theta = B$. Data for task $m$ is generated as $y = W_0x + \theta_m^*\phi(x) + \zeta$ with $\zeta \sim \mathcal{N}(0, \sigma^2 I)$, so Assumption 2 holds by construction. Every task shares the input second moment $\Sigma = I$, hence $\Phi_m = AA^\top = \Phi$ for all $m$ and Assumption 2(ii) holds exactly; $A$ is drawn with a prescribed singular spectrum so that $\kappa(\Phi)$ is set directly, and the experts $\{\theta_m^*\}$ are centered and rescaled to a prescribed $\mathrm{diam}_{\mathrm{F}}(\Theta)$. Unless noted we use $d = 60$, $r = 16$, $l = 12$, $M = 6$, $\kappa(\Phi) = 4$, $\mathrm{diam}_{\mathrm{F}}(\Theta) = 6$, $\sigma = 0.5$, $N = 12$ clients, and non-anchor mixtures drawn from $\mathrm{Dir}(0.5)$. For each task one anchor client is planted with $\alpha_{i_m,m} = 1 - \varepsilon$ and $\alpha_{i_m,m'} = \varepsilon/(M-1)$, so that $\varepsilon$ is directly controlled; the failure of Assumption 3 is imposed by drawing every client $\gamma$-interior instead. Clients solve their least-squares problem in closed form, and the server solves Eq. (8) and Eq. (57) as quadratic programs on the simplex by FISTA with exact simplex projection. On an adversarial instance admitting a non-vertex optimum the solver agrees with exact SLSQP to four decimals ($w = (0.085, 0.744, 0.172)$, objective $0.4404$ against $4.0400$ at the anchor), confirming that the vertex solutions reported below are not artifacts of the optimizer.

**Barycentric convergence (Theorem 1).** Theorem 1 predicts that the population client optimum equals the mixture-weighted barycenter of the task experts. Mean error $\|\hat{\theta}_i - \sum_m \alpha_{im}\theta_m^*\|_{\mathrm{F}}$ over 12 clients, against local sample size: 0.801, 0.366, 0.166, 0.083, 0.036 at $n_i = 2\times10^2, 10^3, 5\times10^3, 2\times10^4, 10^5$. Successive ratios are 2.19, 2.21, 1.99, and 2.29, closely matching the $n^{-1/2}$ predictions 2.24, 2.24, 2.00, and 2.24.

### B.1.2 Expert Recovery and Finite-Sample Behavior

**Anchor impurity and finite-sample recovery (Theorem 2 and Proposition 3).** Theorem 2 and Proposition 3 predict complementary effects of anchor impurity and finite sampling. Figure 4(a) and Table 3 show that expert-recovery error is exactly linear in anchor impurity under exact uploads

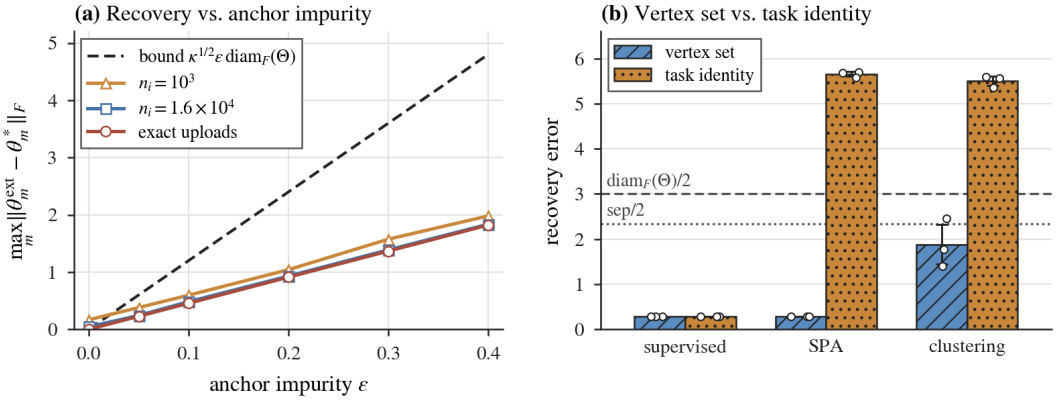



Figure 4: **Synthetic validation of expert recovery.** (a) Expert-recovery error against anchor impurity, with the bound of Theorem 2. (b) Vertex-set versus task-identity recovery; dashed and dotted lines mark $\mathrm{diam}_F(\Theta)/2$ and $\mathrm{sep}/2$ (Theorem 3). Bars show mean $\pm$ standard deviation over three seeds; circles are individual seeds.

and vanishes at $\varepsilon = 0$. The finite-sample floor decomposes as Proposition 3 predicts: holding $n_i = 2\times10^4$ and varying the public set, the error falls $0.087 \to 0.039$ as $n_m$ goes $10^2 \to 1.6\times10^4$; holding $n_m = 1.6\times10^4$ and varying local data, it falls $0.352 \to 0.043$ as $n_i$ goes $250 \to 1.6\times10^4$, halving for each fourfold increase.

**Effect of the conditioning factor.** The bound of Theorem 2 is never violated, and its $\varepsilon$-dependence is tight, but the conditioning factor behaves in two distinct regimes. When each anchor's residual mass is spread evenly over the remaining tasks, program Eq. (8) places weight 1.00 on the anchor: the recovered expert *is* the anchor upload, the error equals $\varepsilon\,\|\theta_m^* - \frac{1}{M-1}\sum_{m'\neq m}\theta_{m'}^*\|_{\mathrm{F}}$, and is therefore independent of $\Phi$ in this regime. Measured error is then invariant to $\kappa(\Phi)$ across 1 to 1024, and remains so when the experts are placed on eigendirections of $\Phi$ so that the Gram matrix induced in mixture coordinates has condition number equal to $\kappa(\Phi)$ exactly. The factor becomes active precisely when the pool admits a convex combination strictly better than any anchor, so that the optimum lies on a face rather than a vertex: with anchor impurity concentrated on a single high-curvature coordinate and complementary clients present, the minimizer has $\max_j \omega_{m,j} \in [0.39, 0.75]$ and the error rises monotonically with conditioning, $1.200 \to 1.894 \to 3.679 \to 4.194$ at $\kappa(\Phi) = 1, 16, 256, 4096$. At $\kappa(\Phi) = 1$ the measured error 1.2000 coincides with the bound 1.200, so the constant is attained. Thus the $\varepsilon$-dependence is tight in this regime, while for larger $\kappa(\Phi)$ the bound remains conservative.

### B.1.3 Identifiability and Scaling Limits

**Task identity and vertex-set recovery (Theorems 3 and 4).** Figure 4(b) and Table 4 separate vertex-set recovery from task-identity recovery. Here $\mathrm{sep}/2 = 2.333$ and $\mathrm{diam}_{\mathrm{F}}(\Theta)/2 = 3.000$.

Table 4: Recovery error with and without anchors, three seeds, $N = 12$, $n_i = 4000$, $n_m = 1000$.

| | anchors, $\varepsilon = 0.05$ | | no anchors, $\gamma = 1$ | |
|---|---|---|---|---|
| Method | vertex set | task identity | vertex set | task identity |
| supervised | **0.287** | **0.287** | 3.760 | 3.760 |
| SPA | **0.287** | 5.658 | 3.770 | 3.818 |
| clustering | 1.877 | 5.507 | 3.780 | 3.815 |

Increasing the federation size does not reduce the unanchored error floor. At $N = 20$ the no-anchor error is $3.754, 3.789, 3.813, 3.816$ as $n_i$ goes $500 \to 3.2\times10^4$, and remains nearly unchanged as $N$ grows from 12 to 60; the exact-anchor error decreases from 0.251 to 0.033 over the same range at the $n_i^{-1/2}$ rate (Figure 5(a)).

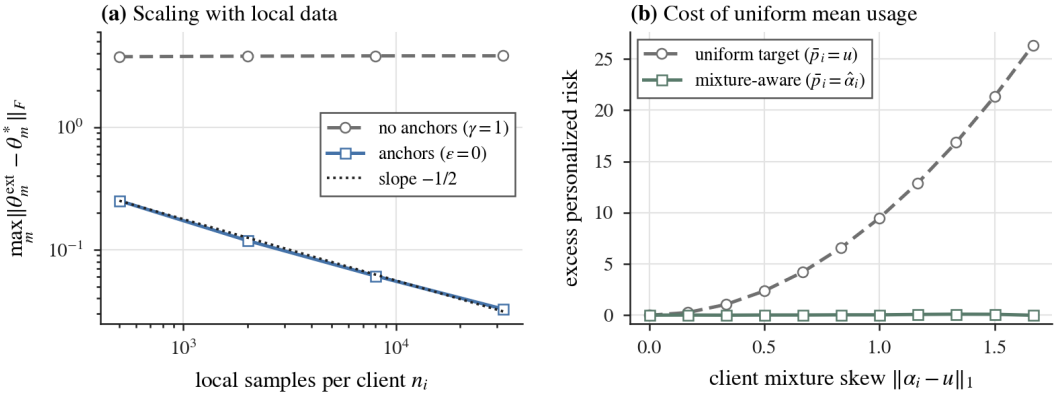



Figure 5: **Synthetic validation of identifiability limits and routing.** (a) Recovery error against local data size with and without anchors (Theorem 4). (b) Excess personalized risk of constant routers under uniform (load-balancing) versus mixture-aware mean usage (Theorem 5).

### B.1.4 Mixture-Aware Routing

**Mean calibration and balanced routing (Theorem 5 and Corollary 5).** To isolate the mean-calibration component of Theorem 5, we evaluate constant routing policies, for which $V_i(p_i) = 0$. The resulting excess risk is available in closed form as $\sum_m \alpha_{im} \| \sum_k p_k \theta_k^* - \theta_m^* \|_\Phi^2$ minus its value at $p = \alpha_i$, so the comparison does not depend on an optimization procedure. Numerical minimization confirms the minimizer is $\alpha_i$ to three decimals. Sweeping a client from uniform toward a vertex, the excess risk of a balanced router is $0, 1.053, 4.212, 9.477, 16.847, 26.324$ — exactly $26.32\,\tau^2$ in the interpolation parameter, and thus quadratic in $\|\alpha_i - u\|_1$ (Figure 5(b)). Using the full pipeline instead, with experts from Eq. (8) and mixtures from Eq. (57), over the 24 non-anchor clients of four seeds: $\|\hat{\alpha}_i - \alpha_i\|_1 = 0.056$, excess risk $0.031$ when anchored to $\tilde{c}_i$ and $7.526$ under load balancing.

## B.2 Geometry and Routing Beyond the Exact Model

The exact barycentric characterization in Section 3 is exact under the shared-feature squared-loss model. We next examine whether the same structure remains visible in a substantially different regime with a deep transformer, cross-entropy loss, and LoRA adaptation. We focus on two questions motivated directly by the theory. First, whether mixed-client updates retain a task-simplex organization when the exact barycentric relation no longer holds. Second, whether the client-level task mixture is sufficient for specialization, or whether mixed clients still require input-dependent routing.

**Controlled setup.** We use QNLI, SST-2, and WiC and construct nine clients at prescribed locations of the resulting three-task simplex, with three pure clients, three edge clients, and three interior clients. We use `distilroberta-base` with rank-4 LoRA adapters applied to the query and value projections. In an initialization round, each client trains both LoRA factors and the server averages the resulting $A$ matrices to obtain a shared $A^{\text{global}}$. The averaged initialization update in $B$ is absorbed into the frozen base model, after which $A^{\text{global}}$ remains fixed and all residual $B$ matrices are reinitialized to zero. We then train one pooled expert for each task and one local adapter for each client for exactly 240 optimizer steps under the same learning-rate schedule. This common optimization clock makes their learned updates directly comparable without differences caused only by training duration. We repeat the experiment over three random seeds. For the geometry analysis, we compare the six value-projection $B$ blocks layer by layer and normalize distances within each layer by the maximum pairwise distance between the three task experts. The routing diagnostic subsequently freezes these task experts and trains only the input-dependent router.

**Expert-plane geometry.** We first examine how the learned client updates are organized relative to the task experts. For each layer and seed, the three task-expert updates define a two-dimensional affine plane in the LoRA parameter space. We construct an orthonormal basis of this plane by taking the QNLI-to-SST-2 direction as the first axis and the component toward WiC orthogonal to it as the

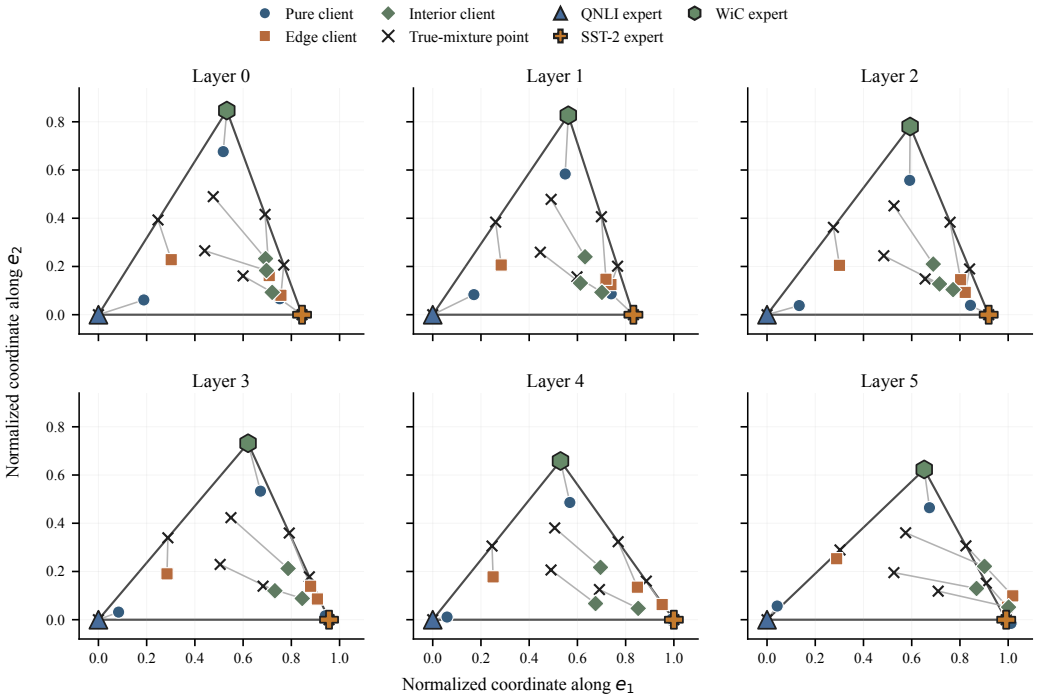



Figure 6: **Expert-plane geometry across layers.** Colored markers show projected client updates and crosses show the locations induced by their realized task mixtures, averaged over three seeds.

second, and then project each client update onto this plane. Because the basis is orthonormal, the resulting expert triangle preserves the pairwise distances between the three task experts for that layer. To compare the geometry across seeds and layers, coordinates are normalized by the maximum pairwise expert distance within each seed and layer before averaging across seeds. The crosses in Figure 6 show the locations induced by the realized client mixtures, while the colored markers show the corresponding learned client projections.

Figure 6 reveals a consistent task-simplex organization throughout the adapted layers. Pure clients remain associated with their corresponding task vertices, edge clients follow the faces spanned by their active tasks, and interior clients occupy the interior of the expert triangle. Moreover, the relative organization of these client types remains visible despite changes in the shape of the expert triangle across layers. This persistence suggests that the task-mixture structure predicted by the theoretical analysis continues to organize the learned updates well beyond the exact shared-feature squared-loss setting.

**Quantifying the simplex approximation.** We next quantify how well the expert simplex explains the full client updates, including components that are not visible in the two-dimensional projections. For client $i$ and layer $\ell$, we define the normalized reconstruction residual as

$$R_i^\ell(w) = \frac{\left\| B_i^\ell - \sum_{m=1}^{3} w_m B_m^\ell \right\|_{\mathrm{F}}}{D^\ell}, \qquad D^\ell = \max_{m \neq m'} \left\| B_m^\ell - B_{m'}^\ell \right\|_{\mathrm{F}}.$$

Each row of Table 5 corresponds to a different choice of the reconstruction weights $w$. The true-mixture residual uses $w = \alpha_i$ and measures the discrepancy from the barycentric location specified by the realized client mixture. The best convex-hull fit minimizes the residual over $w \in \Delta^2$ and therefore gives the closest point in the expert simplex. The best affine fit minimizes the same residual subject only to $\sum_m w_m = 1$, allowing the reconstruction to move anywhere in the affine plane spanned by the experts. As reference controls, the uniform mixture uses $w = (1/3, 1/3, 1/3)$, while

Table 5: Normalized reconstruction residuals of learned client updates. Results are mean ± standard deviation over three seeds of the per-seed median across clients and layers.

| Reference | Normalized residual |
|---|---|
| Best affine fit | $0.444 \pm 0.029$ |
| Best convex-hull fit | $0.444 \pm 0.029$ |
| True mixture $\alpha_i$ | $0.493 \pm 0.023$ |
| Uniform mixture | $0.579 \pm 0.015$ |
| Shuffled mixture | $0.650 \pm 0.012$ |

Table 6: Validation task cross-entropy under static expert mixtures and input-dependent routing. Lower is better.

| Client type | Uniform mixture | True mixture | Input-dependent |
|---|---|---|---|
| Pure | 0.672 | 0.487 | **0.481** |
| Edge | 0.684 | 0.625 | **0.497** |
| Interior | 0.691 | 0.679 | **0.554** |

the shuffled mixture permutes the entries of $\alpha_i$ across the three experts, preserving the mixture proportions while breaking their correspondence with task identity.

The expert simplex provides a substantially better reconstruction than the uniform and shuffled controls. In particular, the shuffled-mixture residual shows that preserving the mixture proportions is not sufficient when those proportions are assigned to the wrong task experts. The nearly identical affine and convex-hull residuals provide an additional geometric indication that allowing the reconstruction to leave the convex region spanned by the experts offers essentially no improvement. Together with Figure 6, these results show that the task simplex remains a meaningful organizing structure for the learned updates in the deep-model setting.

**Mixture-coordinate recovery.** The optimal weights of the convex-hull reconstruction also provide an estimate of the client's task composition. Specifically, for each client $i$ and layer $\ell$, let

$$\hat{\alpha}_i^\ell \in \arg\min_{w \in \Delta^2} \left\| B_i^\ell - \sum_{m=1}^{3} w_m B_m^\ell \right\|_{\mathrm{F}}.$$

Figure 7 compares each recovered coordinate $\hat{\alpha}_{im}^\ell$ with the corresponding realized mixture weight $\alpha_{im}$ across clients, tasks, layers, and seeds. The recovered coordinates closely track the realized proportions, with a correlation of $0.814 \pm 0.040$ and a mean absolute error of $0.157 \pm 0.017$ across seeds. The support structure is particularly well preserved. For edge clients, whose mixtures contain exactly two tasks, the recovered mass assigned to the absent task is only $0.080 \pm 0.021$. Thus, the convex representation captures not only the overall simplex organization observed above, but also meaningful information about the client's underlying task composition.

**Input-dependent routing for mixed clients.** We next ask whether knowing the client-level task mixture is sufficient for specialization once the task experts are fixed. We compare three expert combinations. The uniform baseline uses the same weights $p_i(x) = u$ for every input, while the true-mixture baseline fixes $p_i(x) = \alpha_i$. Both are static combinations and therefore contain no input-level routing information. The input-dependent router instead produces $p_i(x)$ from the current example.

The three settings reveal two complementary effects. First, the uniform combination performs worst across all client types, showing that treating the experts equally ignores useful information about the client's task composition. Replacing the uniform weights with the realized mixture $\alpha_i$ substantially improves the pure and edge cases and also improves the interior clients. Second, for mixed clients, the true mixture remains a static client-level summary and therefore cannot determine which expert should process an individual example. Input-dependent routing further reduces the cross-entropy from $0.625$ to $0.497$ for edge clients and from $0.679$ to $0.554$ for interior clients. In contrast, for

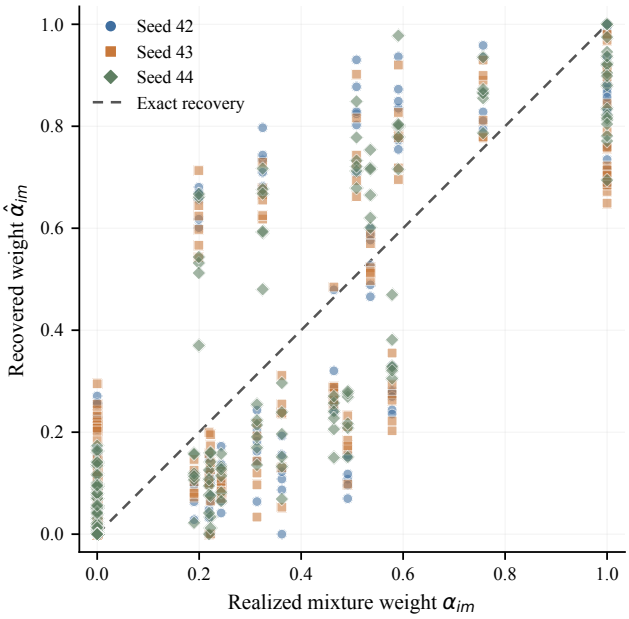



Figure 7: **Recovery of client mixture coordinates.** Recovered simplex coefficients versus realized task proportions across clients, tasks, layers, and seeds. The dashed line denotes exact recovery.

pure clients $\alpha_i$ is one-hot and already identifies the relevant expert, making the true-mixture combination and the input-dependent router nearly identical. These results separate the roles of client composition and input-level specialization: expert usage, while mixed clients additiona

**Mixture-aware router calibration.** We finally isolate the role of the calibration target in the routing objective. We train two otherwise identical input-dependent routers and change only the target of the KL regularizer on the mean expert usage, using either the uniform distribution or the realized client mixture $\alpha_i$. This comparison therefore separates average-usage calibration from the input-level discrimination learned through the task loss.

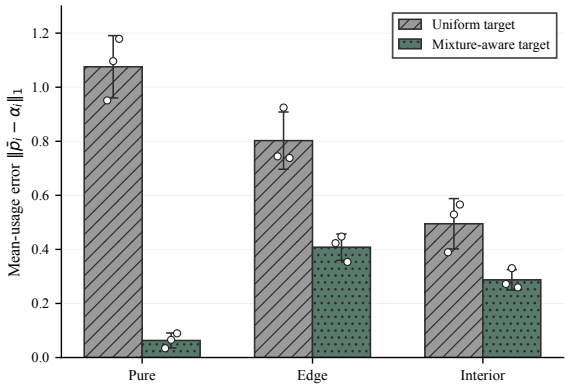



Figure 8: **Mixture-aware calibration of average expert usage.** Mean usage mismatch under uniform and mixture-aware KL targets. Bars show mean $\pm$ standard deviation over three seeds.

Replacing the uniform target with $\alpha_i$ reduces the overall mean-usage mismatch $\|\bar{p}_i - \alpha_i\|_1$ from $0.791$ to $0.253$. Figure 8 shows that this improvement is consistent across pure, edge, and interior clients and is particularly pronounced for structured nonuniform mixtures. For pure clients, the mean probability assigned to the correct expert increases from $0.462$ to $0.968$. For edge clients, the mean probability assigned to the absent expert decreases from $0.289$ to $0.024$. In contrast, the validation task cross-entropy remains nearly unchanged at $0.509$ and $0.511$ for uniform and mixture-aware calibration, respectively. Unlike the full-training ablation in Appendix B.6, this diagnostic freezes the task experts and optimizes only the router, isolating mean-usage calibration from its effect on expert training. These results indicate that the task loss supplies the input-dependent discrimination required for specialization, while the mixture-aware KL term aligns the router's average expert usage with the client composition.

Together, these experiments show that the task-simplex structure remains visible in the deep-model setting and that the client mixture continues to provide meaningful information about expert composition. At the same time, specialization of mixed clients requires input-dependent routing, while mixture-aware calibration controls the router's average expert usage.

### B.3 Experimental Setup

This subsection specifies the data construction, evaluation protocol, baseline implementations, model configurations, and optimization settings used in Section 5. Unless stated otherwise, these settings are held fixed within the corresponding experimental protocol.

#### B.3.1 Tasks, Public Data, and Evaluation

**Task pool.** We use six classification tasks from GLUE and SuperGLUE (Wang et al., 2018; 2019): SST-2, QNLI, CoLA, QQP, WiC, and BoolQ. These tasks form a common pool of $M = 6$ task distributions. The same task pool is used in the headline comparison, heterogeneity sweep, and ablations.

**Public reference data.** For each task $m$, we reserve $1{,}000$ labeled examples as the public reference set $\mathcal{D}_m^s$. These examples are disjoint from all client training, validation, and test data and are never used for local model training or evaluation. FedSEE uses them only at the server to evaluate the task-indexed public losses required by the supervised extraction program Eq. (13). The evaluated baselines do not use these public examples. The public-set-size ablation varies this resource explicitly to quantify the sensitivity of FedSEE to the amount of server-side supervision.

**Validation and personalized evaluation.** Each client is assigned disjoint validation and test sets of 200 examples each, with task compositions matched to its realized training mixture. The validation set is used only for hyperparameter and model selection. The test set is not used for local training, server-side extraction, router calibration, or model selection and is evaluated only after the experimental configuration has been fixed. For each random seed, all compared methods use the same client training, validation, and test partitions. Overall accuracy is the mean of the per-client test accuracies. We additionally report the mean test accuracy of the worst-served $50\%$ and $25\%$ of clients to measure personalization quality away from the population average.

#### B.3.2 Compound-Client Construction

The client partitions are constructed to control both within-client and across-client heterogeneity while keeping the underlying set of tasks fixed. Each task contributes at most $5{,}000$ training examples to the allocation pool, or all available examples when its source training set is smaller. In particular, WiC contributes $4{,}428$ examples.

**Pure-client endpoint.** The condition denoted $\alpha_{\mathrm{Dir}} = 0$ is not sampled from a Dirichlet distribution. It is a deterministic pure-client reference in which every client receives examples from exactly one task. In the $N = 12$ heterogeneity sweep, two clients are assigned to each of the six tasks. Each task's available pool is divided between its assigned clients, with any integer-division remainder assigned to the final client. This realizes the vertex regime of the simplex geometry in Section 3.

**Mixed-client regimes.** For $\alpha_{\mathrm{Dir}} \in \{0.1, 0.5, 1.0, 50\}$, one task-proportion vector is drawn for each client from the symmetric distribution $\mathrm{Dirichlet}(\alpha_{\mathrm{Dir}}\mathbf{1})$. The $\alpha_{\mathrm{Dir}} = 50$ condition is denoted *uniform* in the figures and tables because its draws are strongly concentrated around equal task proportions. Smaller concentration parameters produce more peaked and more diverse client mixtures, whereas larger values move clients toward similar, near-uniform compositions.

Before converting a sampled mixture into integer data allocations, task weights below $0.05$ are set to zero and the remaining entries are renormalized. This preprocessing rule is applied identically to the client partitions used by all compared methods.

**Supply-constrained allocation.** Each client's mixture vector is scaled by a common nominal sample budget obtained from the total available task supply and the number of clients. This produces

a requested number of examples from each task. For a task whose aggregate requests exceed its available supply, all client requests for that task are reduced proportionally until the total matches the available pool. When aggregate demand falls below supply, the unused remainder is left unallocated rather than redistributed. Consequently, the realized amount of training data varies with the sampled client mixtures. Table 7 reports the realized pooled client training–validation allocation at each heterogeneity level.

Table 7: Realized pooled client training–validation allocation per task across the heterogeneity sweep. "Uniform" denotes $\alpha_{\mathrm{Dir}} = 50$.

| **Task** | $\alpha_{\mathrm{Dir}} = 0$ | 0.1 | 0.5 | 1.0 | **uniform** |
|---|---|---|---|---|---|
| SST-2 | 5000 | 4997 | 4995 | 4993 | 4716 |
| QNLI | 5000 | 4169 | 4996 | 4686 | 4728 |
| QQP | 5000 | 2901 | 4439 | 3045 | 4987 |
| CoLA | 5000 | 4998 | 2794 | 4665 | 4876 |
| WiC | 4428 | 4427 | 4425 | 3618 | 4422 |
| BoolQ | 5000 | 4047 | 3639 | 4994 | 4994 |

### B.3.3 Baseline Implementations

We compare against six federated LoRA methods spanning global aggregation, personalization, and clustering. For a given experiment, all methods use the same client partitions, pretrained backbone, communication-round budget, local-epoch budget, batch size, maximum sequence length, LoRA rank, and adapted transformer modules unless the method itself requires a different treatment of the LoRA factors. Method-specific aggregation and personalization follow the respective published procedures.

**FedIT.** FedIT (Zhang et al., 2024) applies standard federated averaging to both LoRA factors $A$ and $B$, producing a single global adapter shared by all clients.

**FFA-LoRA.** FFA-LoRA (Sun et al., 2024) keeps the LoRA projection $A$ fixed and aggregates the trainable $B$ updates, producing a single shared adapter without averaging two independently changing LoRA factors.

**FedSA-LoRA.** FedSA-LoRA (Guo et al., 2025) exploits the asymmetric roles of the two LoRA factors by aggregating $A$ while retaining a personalized client-side $B$ component.

**FedDPA-T.** FedDPA-T (Yang et al., 2024) maintains global and personalized adapter components and combines them at inference through its test-time personalization procedure. We use the test-time variant throughout the reported comparisons.

**IFCA+LoRA.** We instantiate IFCA (Ghosh et al., 2020) with LoRA by maintaining a separate LoRA adapter for each cluster and iteratively assigning each client to a single cluster using the IFCA hard-assignment procedure. A client updates only the adapter associated with its assigned cluster.

**FedLEASE.** FedLEASE (Wang et al., 2025) clusters client $B$ updates using cosine similarity, constructs one expert per cluster, and applies its published routing mechanism over the resulting expert pool.

The conceptual relationships among these methods and FedSEE are discussed separately in Appendix A. The description here concerns their experimental instantiation.

### B.3.4 FEDSEE CONFIGURATION

**Initialization and shared projection.** FedSEE begins with one initialization round. The server broadcasts a common randomly initialized LoRA pair $(A^{(0)}, B^{(0)})$, and each client trains both factors for $E_{\text{init}}$ local epochs. The server then forms $A^{\text{global}} = N^{-1} \sum_i A_i$ and keeps this projection fixed for all subsequent task-specific experts. The complete initialization and communication procedure is given in Algorithm 1.

**Expert extraction.** The server solves Eq. (13) once per communication round using 50 coefficient-update steps at learning rate 0.1. The coefficient matrix is projected onto the balanced-transport constraints using 20 Sinkhorn iterations every 5 updates, and the resulting contribution matrix is smoothed across rounds with EMA rate $\lambda = 0.9$. We use an optimization regularization coefficient of 0.01 during the coefficient updates. The extraction procedure itself is described in Appendix C.2.

**Inference and training capacity.** FedSEE uses $M = 6$ recovered experts and makes all six available at inference, so $U_i = 6$. The default configuration updates at most one expert per client, $L_i = 1$, which we denote **FedSEE (L1)**. The full-capacity variant **FedSEE (L6)** sets $L_i = 6$ while retaining the same inference pool. Experts with normalized client-specific contribution below $\tau = 0.05$ are excluded from the active set, so $L_i$ remains an upper bound rather than a strict quota. The capacity construction is defined in Appendix C.3, and its empirical effect is examined in Appendix B.5.

**Router and shared expert.** We set the router-calibration weight to $\beta_{\text{KL}} = 10^{-3}$ and use router learning rate $\eta_r = 10\,\eta_B$. The shared LoRA expert is trained at learning rate $\eta_B$ and aggregated across clients using uniform FedAvg. Router parameterization and aggregation are described in Appendix C.4.

Table 8 summarizes the settings shared across methods and the additional FedSEE-specific settings. Following the NLU protocol of FedLEASE (Wang et al., 2025), all reported transformer runs use AdamW with a frozen pretrained backbone and a classification head that is frozen after initialization.

### B.3.5 HEADLINE COMPOUND-HETEROGENEITY PROTOCOL

The headline comparison in Section 5.2 uses $\alpha_{\text{Dir}} = 0.1$ and RoBERTa-large (Liu et al., 2019), with $N = 10$ fully participating clients. All methods train for $T = 10$ communication rounds with $E = 5$ local epochs per round. The supply-constrained allocation yields approximately 2,000–2,500 training examples per client. For each method, we select the learning rate from $\{5 \times 10^{-4}, 10^{-3}, 5 \times 10^{-3}\}$ using the mean client validation accuracy on one fixed tuning seed. The selected learning rate is then fixed and evaluated on the held-out test sets using the tuning seed and two additional seeds, giving three reported runs per method. We report the mean and standard deviation across these runs.

### B.3.6 HETEROGENEITY-SWEEP PROTOCOL

The heterogeneity sweep in Section 5.3 uses `distilroberta-base`, $N = 12$ clients, $T = 10$ communication rounds, and $E = 5$ local epochs. We evaluate five population regimes: the deterministic pure-client endpoint denoted $\alpha_{\text{Dir}} = 0$, Dirichlet concentrations $\alpha_{\text{Dir}} \in \{0.1, 0.5, 1.0\}$, and the near-uniform $\alpha_{\text{Dir}} = 50$ endpoint.

All methods use the fixed expert learning rate $\eta_B = 5 \times 10^{-4}$ and the three shared random seeds $\{42, 43, 44\}$. For a given seed, every method uses the same realized client partition. Between-method differences are therefore computed seed by seed. For each method, we report the mean paired difference from FedSEE and its sample standard deviation across the three matched seeds.

**Local-only and centralized references.** The sweep additionally includes two non-federated reference conditions. *Local-only* trains each client independently on its own data and therefore measures performance without information sharing across clients. The *centralized* reference pools the available training examples by task and trains one task-specific model on each pooled task dataset, which is then evaluated on examples from the corresponding task. These conditions are included as empirical references rather than as theoretical lower or upper bounds.

Table 8: Complete shared and FedSEE-specific hyperparameter configuration.

| | |
|---|---|
| *Shared across compared methods* | |
| LoRA rank $r$ | 4 |
| LoRA scaling $\alpha_{\mathrm{LoRA}}$ | 8 |
| LoRA dropout | 0.1 |
| Target modules | query and value projections in all self-attention layers |
| Backbone parameters | frozen; LoRA-MoE query/value projections use $W_0$ without bias |
| Optimizer | AdamW |
| Weight decay | 0.01 |
| Gradient-norm clipping | 1.0 |
| Batch size | 64 |
| Maximum sequence length | 128 tokens |
| Local epochs per round $E$ | 5 |
| Communication rounds $T$ | 10 |
| Personal validation-set size | 200 examples per client |
| Personal test-set size | 200 examples per client |
| *FedSEE-specific* | |
| Task experts $M$ | 6 |
| Initialization epochs $E_{\mathrm{init}}$ | 5 |
| Inference capacity $U_i$ | 6 |
| Training capacity $L_i$ | 1 by default; 6 in the L6 ablation |
| Extraction steps | 50 per communication round |
| Extraction learning rate | 0.1 |
| Extraction regularization coefficient | 0.01 |
| Sinkhorn projection | 20 iterations every 5 extraction steps |
| Contribution EMA rate $\lambda$ | 0.9 |
| Router calibration weight $\beta_{\mathrm{KL}}$ | $10^{-3}$ |
| Router learning rate | $\eta_r = 10\,\eta_B$ |
| Active-set contribution floor $\tau$ | 0.05 |
| Shared-expert learning rate | $\eta_B$ |
| Shared-expert aggregation | uniform FedAvg |
| Public reference set | 1,000 examples per task for FedSEE |

### B.3.7 Ablation Protocol

Unless an ablation directly changes the corresponding component, the component ablations follow the headline compound-heterogeneity setting of Section 5.2: RoBERTa-large, $\alpha_{\mathrm{Dir}} = 0.1$, $N = 10$ clients, $T = 10$ communication rounds, and $E = 5$ local epochs. This keeps the remaining experimental conditions fixed while isolating each design choice.

The public-supervision ablation varies $|\mathcal{D}_m^s| \in \{100, 250, 500, 1000\}$ examples per task, with 1,000 as the default. The router-calibration ablation compares the default client-specific calibration with uniform mean-usage calibration and with removal of the first-moment calibration term. The expert-count ablation varies the task-labeled public reference sets: $M = 3$ retains SST-2, QNLI, and CoLA, $M = 6$ covers all client tasks, and $M = 9$ additionally includes RTE, MRPC, and WNLI, which are absent from the clients; for WNLI, all 635 available training examples are used. The shared-expert ablation removes the always-active pair $(A^{\mathrm{s}}, B^{\mathrm{s}})$ while leaving the routed expert mechanism unchanged.

Training capacity is evaluated separately with `distilroberta-base` across the heterogeneity sweep. We compare $L_i = 1$ and $L_i = 6$ while holding $U_i = 6$ fixed. The corresponding results are reported in Appendix B.5.

### B.3.8 COMPUTATIONAL ENVIRONMENT

All reported transformer experiments were executed on a GPU cluster with NVIDIA H100 and A100 accelerators. Each experimental run used a single-process federated simulation loop.

## B.4 DETAILED RESULTS UNDER COMPOUND HETEROGENEITY

Table 9 gives the per-task breakdown for the RoBERTa-large experiment in Section 5.2. The main text focuses on client-level overall and tail metrics because these directly evaluate personalized performance under compound heterogeneity. The task-level results provide a complementary view of where the aggregate gains arise. FedSEE obtains the highest accuracy on QNLI, WiC, and BoolQ and remains competitive on SST-2, CoLA, and QQP, while producing the strongest overall and client-tail performance.

Table 9: Per-task and personalized accuracy (%) under compound heterogeneity ($\alpha_{\mathrm{Dir}} = 0.1$) with RoBERTa-large. Results are reported as mean $\pm$ standard deviation over three seeds.

| Method | SST-2 | QNLI | CoLA | QQP | WiC | BoolQ | Overall | Worst-50% | Worst-25% |
|---|---|---|---|---|---|---|---|---|---|
| FedIT | $91.76 \pm 2.35$ | $86.16 \pm 1.77$ | $78.28 \pm 3.25$ | $70.81 \pm 25.73$ | $61.62 \pm 1.88$ | $72.41 \pm 5.30$ | $78.06 \pm 2.04$ | $69.35 \pm 5.17$ | $64.44 \pm 6.09$ |
| FFA-LoRA | $89.86 \pm 2.87$ | $77.67 \pm 7.35$ | $75.32 \pm 3.03$ | $71.80 \pm 15.43$ | $61.25 \pm 0.70$ | $62.90 \pm 2.80$ | $73.43 \pm 1.09$ | $64.64 \pm 2.27$ | $62.21 \pm 2.49$ |
| IFCA+LoRA | $93.04 \pm 1.13$ | $83.69 \pm 6.18$ | $78.90 \pm 1.78$ | $\mathbf{86.02 \pm 1.95}$ | $61.85 \pm 3.45$ | $67.82 \pm 6.72$ | $79.42 \pm 0.99$ | $70.95 \pm 2.42$ | $67.37 \pm 1.49$ |
| FedSA-LoRA | $\mathbf{93.86 \pm 1.66}$ | $86.66 \pm 4.40$ | $\mathbf{81.00 \pm 1.36}$ | $79.98 \pm 4.39$ | $63.95 \pm 1.61$ | $71.97 \pm 9.16$ | $79.75 \pm 1.37$ | $71.89 \pm 2.60$ | $66.78 \pm 2.83$ |
| FedDPA-T | $92.08 \pm 1.07$ | $87.64 \pm 3.53$ | $77.82 \pm 2.08$ | $73.91 \pm 18.85$ | $64.32 \pm 2.89$ | $75.47 \pm 3.82$ | $79.20 \pm 1.00$ | $71.40 \pm 3.74$ | $67.34 \pm 4.20$ |
| FedLEASE | $91.40 \pm 2.57$ | $83.21 \pm 1.66$ | $79.63 \pm 5.07$ | $71.66 \pm 14.99$ | $65.67 \pm 2.75$ | $75.28 \pm 10.52$ | $79.52 \pm 1.80$ | $72.46 \pm 3.99$ | $69.10 \pm 3.77$ |
| **FedSEE (L1, Ours)** | $92.54 \pm 1.39$ | $\mathbf{92.29 \pm 1.95}$ | $80.47 \pm 0.90$ | $84.67 \pm 1.74$ | $\mathbf{66.24 \pm 4.96}$ | $\mathbf{77.73 \pm 3.86}$ | $\mathbf{82.65 \pm 0.35}$ | $\mathbf{76.54 \pm 1.65}$ | $\mathbf{72.75 \pm 2.21}$ |

## B.5 TRAINING-CAPACITY ANALYSIS

Appendix C.3 defines the active and passive expert sets controlled by the training capacity $L_i$. Here we examine how this choice affects empirical performance across different heterogeneity regimes.

**Training-capacity tradeoff.** We evaluate the effect of training capacity by fixing the expert pool at $M = 6$ and the inference capacity at $U_i = 6$, while varying only the maximum number of task-specific experts updated by each client. Specifically, we compare the default $L_i = 1$ configuration with full-capacity training at $L_i = 6$. Both variants retain all six experts during inference and therefore use the same inference capacity and routing architecture. Table 10 reports the comparison across the full heterogeneity sweep.

FedSEE (L1) is strongest at the pure-client endpoint, improving over full-capacity training by $0.89$ points at $\alpha_{\mathrm{Dir}} = 0$. As client mixtures become broader, L6 obtains small gains of $0.25$, $0.92$, $0.05$, and $0.31$ points at $\alpha_{\mathrm{Dir}} = 0.1$, $0.5$, $1.0$, and the near-uniform regime, respectively. This behavior is consistent with the role of the client-specific contribution profile. For concentrated mixtures, local updates can be focused on the most relevant expert, whereas broader mixtures provide useful local signal for a larger portion of the expert pool. Importantly, the two configurations remain within one percentage point across every regime despite L1 updating at most one task-specific expert per client instead of six. We therefore use $L_i = 1$ as the default configuration, which retains the full six-expert inference pool while substantially reducing task-specific training and communication cost.

Table 10: Training-capacity ablation: overall accuracy (%) across the heterogeneity sweep.

| Variant | $\alpha_{\mathrm{Dir}}=0$ | 0.1 | 0.5 | 1.0 | **uniform** |
|---|---|---|---|---|---|
| FedSEE (L1) | $\mathbf{76.46 \pm 0.30}$ | $74.95 \pm 0.63$ | $71.28 \pm 1.76$ | $72.10 \pm 0.80$ | $69.97 \pm 1.37$ |
| FedSEE (L6) | $75.57 \pm 0.23$ | $\mathbf{75.20 \pm 0.40}$ | $\mathbf{72.20 \pm 1.12}$ | $\mathbf{72.15 \pm 0.93}$ | $\mathbf{70.28 \pm 0.41}$ |

## B.6 COMPONENT ABLATIONS

We isolate the main design choices of FedSEE under the headline compound-heterogeneity setting of Section 5.2. Unless an ablation directly modifies the corresponding component, we use RoBERTa-large, $\alpha_{\mathrm{Dir}} = 0.1$, $N = 10$ clients, $T = 10$ communication rounds, and $E = 5$ local epochs,

following the protocol in Appendix B.3. Each ablation changes one component of the default FedSEE configuration while keeping the remaining settings fixed. Table 11 reports overall personalized accuracy and the change relative to the default configuration. The default row corresponds to the configuration used in Section 5.2, with $|\mathcal{D}_m^s| = 1{,}000$, mixture-aware router calibration, six recovered experts, and the shared expert.

Table 11: Component ablations under the headline compound-heterogeneity setting. Results are reported as mean $\pm$ standard deviation over three seeds. $\Delta$ denotes the change in mean accuracy relative to the default configuration.

| **Component** | **Setting** | **Overall (%)** | $\Delta$ |
|---|---|---|---|
| Default | FedSEE (L1) | $\mathbf{82.65 \pm 0.35}$ | – |
| Public reference set | $\lvert\mathcal{D}_m^s\rvert = 100$ | $81.27 \pm 0.74$ | $-1.38$ |
| | $\lvert\mathcal{D}_m^s\rvert = 250$ | $81.92 \pm 0.90$ | $-0.73$ |
| | $\lvert\mathcal{D}_m^s\rvert = 500$ | $82.49 \pm 0.39$ | $-0.16$ |
| Router calibration | uniform target | $80.85 \pm 0.42$ | $-1.80$ |
| | none | $81.35 \pm 0.75$ | $-1.30$ |
| Expert count | 3 experts | $82.01 \pm 0.59$ | $-0.64$ |
| | 9 experts | $82.31 \pm 0.54$ | $-0.34$ |
| Shared expert | removed | $82.03 \pm 0.19$ | $-0.62$ |

**Public supervision.** We vary the task-labeled public reference set used for server-side extraction in Eq. (13) from 100 to the default 1,000 examples per task. Reducing the public reference set produces a gradual decrease in accuracy, from $82.65\%$ with 1,000 examples per task to $81.27\%$ with 100. Importantly, the 100-example setting uses only one tenth of the default public supervision, yet still achieves $81.27\%$ overall accuracy. This remains $1.52$ points above the strongest evaluated baseline, FedSA-LoRA at $79.75\%$, under the same headline setting reported in Table 1. Increasing the reference set to 500 examples per task recovers most of the default performance, reaching $82.49\%$, only $0.16$ points below the 1,000-example setting. These results indicate that additional public supervision is beneficial, while the performance advantage over the evaluated baselines is retained even when the public reference set is reduced substantially.

**Mixture-aware router calibration.** We compare the default client-specific calibration target $\tilde{c}_i$ in Eq. (14) with a uniform mean-usage target and with removing the mean-usage calibration term entirely. Replacing the client-specific target with a uniform target reduces accuracy from $82.65\%$ to $80.85\%$, while removing the calibration term gives $81.35\%$. Mixture-aware calibration therefore improves over no calibration by $1.30$ points in this setting, whereas uniform calibration is $0.50$ points below using no calibration term. This ordering is consistent with the routing analysis in Section 3.3 and Proposition 1, where uniform mean expert usage can be mismatched to clients with non-uniform task mixtures. Empirically, the results suggest that the choice of calibration target matters rather than mean-usage calibration being beneficial by itself.

**Expert count.** The default configuration uses six public task anchors matching the six client tasks. With $M = 3$, public supervision is retained only for SST-2, QNLI, and CoLA, while $M = 9$ additionally introduces RTE, MRPC, and WNLI, which are absent from the clients. The six-expert default achieves $82.65\%$, compared with $82.01\%$ for $M = 3$ and $82.31\%$ for $M = 9$. The modest reductions indicate robustness to both incomplete and extraneous task coverage in the public reference set.

**Shared expert.** FedSEE includes the always-active shared LoRA pair $(A^s, B^s)$ in Eq. (12) to capture adaptation that need not be assigned to a task-specific expert. Removing this component while leaving the routed expert mechanism unchanged reduces overall accuracy from $82.65\%$ to $82.03\%$, a decrease of $0.62$ percentage points. This modest reduction indicates that the shared expert provides a useful complementary adaptation pathway, allowing structure common across tasks to be

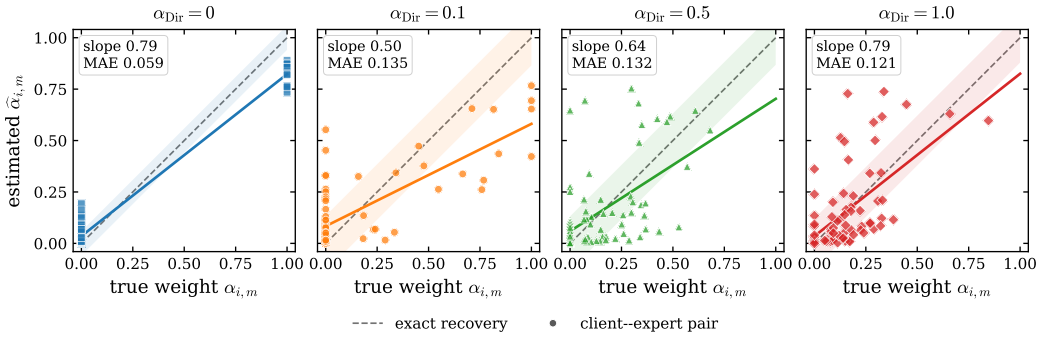



Figure 9: **Client-mixture estimation across heterogeneity regimes.** Contribution-based estimate $\hat{c}_{i,m}$ versus the ground-truth task proportion $\alpha_{i,m}$ for each client–task pair. The dashed diagonal indicates exact recovery, the solid line shows the least-squares fit, and each panel reports the fitted slope and mean absolute error (MAE).

represented without being absorbed into the routed expert pool. At the same time, performance remains close to the full model after its removal, suggesting that the task-specific routed experts account for the majority of the adaptation benefit rather than relying on the shared component.

### B.7 Client-Mixture Estimation

Proposition 5 characterizes client-mixture recovery through barycentric coordinates of the recovered task experts. Here we instead examine whether FedSEE's contribution-based mixture proxy tracks the known proportions used to construct the federated partitions.

For each client, we normalize its column of the smoothed contribution matrix $C$ to unit sum, yielding a contribution-based estimate $\hat{c}_i$, and average this estimate across the ten communication rounds. We compare these estimates with the task proportions used by the Dirichlet partitioner, which are not provided to the server during training. Figure 9 reports all client–task pairs. At the pure-client endpoint $\alpha_{\mathrm{Dir}} = 0$, the dominant task is correctly identified for every client, with a mean absolute error of $0.059$. In the mixed regimes, the corresponding MAEs are $0.135$, $0.132$, and $0.121$ for $\alpha_{\mathrm{Dir}} \in \{0.1, 0.5, 1.0\}$, respectively. Across all regimes, the contribution-based estimates remain positively associated with the underlying task proportions, showing that FedSEE's extraction contributions retain meaningful information about each client's mixture composition.

## C FedSEE Algorithm and Implementation Details

Section 4 introduces the main components of FedSEE. This appendix provides the corresponding algorithmic and implementation details, including the complete training procedure, the projection-level reduction underlying the LoRA instantiation, the joint balanced-extraction solver, the contribution-based capacity assignment, and the router parameterization and aggregation. Theoretical guarantees for these components are developed in Appendix D, while the experimental configurations used in our evaluation are reported in Appendix B.

Algorithm 1 states one full run of FedSEE. Extract solves the supervised extraction program jointly over all $M$ experts and returns the recovered experts together with the raw client-contribution matrix $C$, whose rows sum to one and whose client columns sum to $M/N$ under the balanced-transport constraint. Assign selects client $i$'s inference set from $c_{\cdot i}$, normalizes the retained contributions to obtain the unit-sum target $\tilde{c}_i$, and partitions the inference set into active and passive experts. ClientUpdate then runs $E$ local epochs on Eq. (14), taking full gradients on the active blocks and stop-gradient on the passive blocks. Only active blocks are uploaded.

### C.1 Projection-Level Reduction

Equation (12) describes one adapted linear projection, with the layer and module indices suppressed. Conditional on its incoming representation $x$, the routed LoRA contribution is linear in each $B_m$

**Algorithm 1** FedSEE: Federated Supervised Expert Extraction

**Require:** Public data $\{\mathcal{D}_m^s\}_{m=1}^M$, capacities $\{L_i, U_i\}_{i=1}^N$, communication rounds $T$
**Require:** Epochs $E_{\text{init}}, E$; learning rates $\eta_B, \eta_r$; EMA rate $\lambda$; calibration weight $\beta_{\text{KL}}$; contribution floor $\tau$
**Ensure:** Experts $\{B_m^{\text{expert}}\}_{m=1}^M$, global projection $A^{\text{global}}$, shared pair $(A^{\text{s}}, B^{\text{s}})$, router $\psi$

**Initialize:** broadcast LoRA pair $(A^{(0)}, B^{(0)})$
Clients train for $E_{\text{init}}$ epochs and return $\{(A_i, B_i)\}_{i=1}^N$
$A^{\text{global}} \leftarrow \frac{1}{N}\sum_{i=1}^N A_i$ ▷ fixed thereafter
Each client initializes $\psi_i^{(0)}$ and $(A_i^{\text{s}}, B_i^{\text{s}})$ locally, with $A_i^{\text{s}} \leftarrow \text{Kaiming}()$ and $B_i^{\text{s}} \leftarrow \mathbf{0}$
$\big(\{B_m^{\text{expert}}\}, C^{(1)}\big) \leftarrow \textsc{Extract}\big(\{B_i\}_{i=1}^N, W_0\big)$ ▷ initial experts

**for** $t = 1, \ldots, T$ **do**
  Server broadcasts $\{B_m^{\text{expert}}\}_{m=1}^M$ and, for $t > 1$, $(A^{\text{s}}, B^{\text{s}})$ and $\psi^{(t-1)}$
  **for** client $i \in \{1, \ldots, N\}$ **in parallel do**
    $(\mathcal{U}_i, \mathcal{A}_i, \mathcal{V}_i, \tilde{c}_i) \leftarrow \textsc{Assign}\big(C_{\cdot i}^{(t)}, U_i, L_i, \tau\big)$
    $\big(\{B_m^i\}_{m\in\mathcal{A}_i}, (A_i^{\text{s}}, B_i^{\text{s}}), \psi_i\big) \leftarrow \textsc{ClientUpdate}\big(i, \mathcal{U}_i, \mathcal{A}_i, \tilde{c}_i, t\big)$
  **end for**
  $(A^{\text{s}}, B^{\text{s}}) \leftarrow \frac{1}{N}\sum_{i=1}^N (A_i^{\text{s}}, B_i^{\text{s}}), \quad \psi^{(t)} \leftarrow \frac{1}{N}\sum_{i=1}^N \psi_i$
  $\big(\{B_m^{\text{expert}}\}, C^*\big) \leftarrow \textsc{Extract}\big(\{B_m^i\}, W_0 + sB^{\text{s}}A^{\text{s}}\big)$ ▷ update experts
  $C^{(t+1)} \leftarrow (1-\lambda)C^{(t)} + \lambda C^*$ ▷ EMA of contributions
**end for**

**procedure** $\textsc{ClientUpdate}(i, \mathcal{U}_i, \mathcal{A}_i, \tilde{c}_i, t)$
  $B_m \leftarrow B_m^{\text{expert}}$ for $m \in \mathcal{U}_i$; for $t > 1$, $(A_i^{\text{s}}, B_i^{\text{s}}) \leftarrow (A^{\text{s}}, B^{\text{s}})$ and $\psi_i \leftarrow \psi^{(t-1)}$
  Let $\Pi_m[\cdot]$ be the identity for $m \in \mathcal{A}_i$ and stop-gradient for $m \in \mathcal{V}_i$
  **for** $e = 1, \ldots, E$ **do**
    $h(x) \leftarrow \Big(W_0 + sB_i^{\text{s}}A_i^{\text{s}} + s\sum_{m\in\mathcal{U}_i} p_{im}(x)\,\Pi_m[B_m]\,A^{\text{global}}\Big)x$
    Compute $\mathcal{L}$ by Eq. (14)
    Update $\{B_m\}_{m\in\mathcal{A}_i}$, $(A_i^{\text{s}}, B_i^{\text{s}})$ with $\eta_B$ and $\psi_i$ with $\eta_r$
  **end for**
  **return** $\{B_m\}_{m\in\mathcal{A}_i}$, $(A_i^{\text{s}}, B_i^{\text{s}})$, $\psi_i$
**end procedure**

**procedure** $\textsc{Assign}(c_{\cdot i}, U_i, L_i, \tau)$
  $\mathcal{U}_i \leftarrow \text{Top-}U_i\{c_{mi}\}_{m=1}^M$
  $\tilde{c}_{mi} \leftarrow c_{mi} \big/ \sum_{k\in\mathcal{U}_i} c_{ki}$ for $m \in \mathcal{U}_i$
  $\mathcal{A}_i \leftarrow \text{Top-}L_i\{m \in \mathcal{U}_i : \tilde{c}_{mi} \geq \tau\}$
  $\mathcal{V}_i \leftarrow \mathcal{U}_i \setminus \mathcal{A}_i$
  **return** $(\mathcal{U}_i, \mathcal{A}_i, \mathcal{V}_i, \tilde{c}_i)$
**end procedure**

**procedure** $\textsc{Extract}(\mathcal{P}, \tilde{W})$ ▷ $\mathcal{P}$: upload pool
  Solve by projected gradient descent with Sinkhorn projection (Appendix C.2):
  $\boldsymbol{\Omega}^* \in \arg\min_{\boldsymbol{\Omega}} \sum_{m=1}^M \widehat{\mathcal{L}}_m^s\big(\sum_{p\in\mathcal{P}} \omega_{mp}B_p;\, A^{\text{global}}, \tilde{W}\big)$
  **s.t.** $\omega_{mp} \geq 0$, $\sum_{p\in\mathcal{P}} \omega_{mp} = 1\ \forall m$, $\sum_{m=1}^M c_{mi} = M/N\ \forall i$, where $c_{mi} := \sum_{p:\,\text{client}(p)=i} \omega_{mp}$
  **return** $\big\{B_m^{\text{expert}} = \sum_p \omega_{mp}^* B_p\big\}_{m=1}^M$ and $C^* = (c_{mi}^*)$
**end procedure**

because $A^{\text{global}}$ is fixed. The always-on shared term can therefore be absorbed exactly into $\tilde{f}_0(x) = W_0x + sB^{\text{s}}A^{\text{s}}x$ at that projection. Hence the reduction to Eq. (5) is exact at the frozen-feature projection level.

The corresponding extension from this projection-level reduction to the full transformer is analyzed in Appendix D.8.

### C.2 Joint Extraction with Balanced Transport

**Solver.** The optimization problem in Eq. (13) is solved by warm-started projected gradient descent. We differentiate the public loss through the reconstructed expert $B_m$ only and reach $\mathbf{\Omega}$ by the chain rule

$$\frac{\partial \widehat{\mathcal{L}}_m^s}{\partial \omega_{m,(i,m')}} = \left\langle B_{m'}^i, \frac{\partial \widehat{\mathcal{L}}_m^s}{\partial B_m} \right\rangle.$$

Gradients with respect to each reconstructed $B_m$ and each row of $\mathbf{\Omega}$ are clipped to norm $0.1$, and the coefficients are clipped to be nonnegative between Sinkhorn projections. Thus, each optimization step requires one forward and backward pass per task, independent of the size of the upload pool.

**Balanced coupling.** FedSEE couples the $M$ taskwise extraction problems through client-level contribution constraints to prevent different experts from repeatedly concentrating on the same clients. For task $m$ and client $i$, define

$$c_{mi} := \sum_{p:\mathrm{client}(p)=i} \omega_{mp}.$$

Since the extraction weights for each task sum to one, $\sum_i c_{mi} = 1$. The total contribution mass across the $M$ experts is therefore $M$. FedSEE distributes this mass uniformly across the $N$ clients by imposing

$$\sum_m c_{mi} = \frac{M}{N},$$

and enforces the resulting row and column marginals using Sinkhorn projection.

**Client normalization.** The balanced-transport constraint is imposed on the raw contribution matrix $C$ during expert extraction. After extraction, client $i$ normalizes the contributions of the experts retained in its inference set:

$$\tilde{c}_{mi} = \frac{c_{mi}}{\sum_{k\in\mathcal{U}_i} c_{ki}}, \qquad m \in \mathcal{U}_i,$$

so that $\sum_{m\in\mathcal{U}_i} \tilde{c}_{mi} = 1$. When $\mathcal{U}_i = [M]$, this reduces to $\tilde{c}_{mi} = (N/M)c_{mi}$. This client-side normalization does not alter the balanced-transport constraints or the recovered experts. The effect of balanced coupling on expert recovery is analyzed in Proposition 11 of Appendix D.9.

### C.3 The Mixture Proxy and Capacity Assignment

The barycentric analysis of Section 3 motivates a client-specific mixture profile over the recovered experts. FedSEE obtains this profile directly from the client contributions produced by the joint extraction procedure. These contributions provide the client-specific relevance signal used for both capacity assignment and router calibration.

**Temporal smoothing.** FedSEE does not use the raw contribution matrix from each extraction round directly. Instead, it maintains a temporally smoothed contribution matrix for capacity assignment and router calibration. The smoothing is applied only to the contribution matrix, not to the extraction weights $\mathbf{\Omega}$ or the recovered expert parameters. Let $C^{*,(t)} \in \mathbb{R}^{M\times N}$ denote the raw balanced contribution matrix returned by the extraction program at round $t$. The deployed update is

$$C^{(t+1)} = (1-\lambda)C^{(t)} + \lambda C^{*,(t)}, \qquad \lambda \in (0,1). \tag{15}$$

The feasibility and tracking properties of this update are established in Proposition 12 of Appendix D.10.

For client $i$, the column $c_{\cdot i}^{(t)} = (c_{1i}^{(t)}, \ldots, c_{Mi}^{(t)})$ summarizes its contribution across the recovered experts. After restricting this profile to the client's inference set, the retained entries are normalized to obtain $\tilde{c}_i$, which is used to rank experts for capacity assignment and to calibrate the router.

**Active, passive, and inference sets.** FedSEE separates training and inference capacity through two per-client budgets. The training capacity $L_i \in \{1, \dots, M\}$ limits the number of experts updated by backpropagation, while the inference capacity $U_i \in \{L_i, \dots, M\}$ limits the number of experts participating in the forward pass. Ranking the recovered experts by their client-specific contributions gives

$$\mathcal{U}_i = \text{Top-}\, U_i\{c_{mi}\}_m, \quad \tilde{c}_{mi} = \frac{c_{mi}}{\sum_{m' \in \mathcal{U}_i} c_{m'i}}, \quad \mathcal{A}_i = \text{Top-}\, L_i\{\tilde{c}_{mi} : \tilde{c}_{mi} \geq \tau\}, \quad \mathcal{V}_i = \mathcal{U}_i \setminus \mathcal{A}_i, \tag{16}$$

where $\mathcal{U}_i$, $\mathcal{A}_i$, and $\mathcal{V}_i$ denote the inference, active, and passive expert sets, respectively. The normalization is performed after applying the inference cap, so $\tilde{c}_i$ remains a unit-sum distribution over $\mathcal{U}_i$ when $U_i < M$. The threshold $\tau$ is then applied together with $L_i$ to determine the active set. Consequently, $L_i$ is an upper bound rather than a strict quota, and clients with concentrated contribution profiles may update fewer than $L_i$ experts.

Active experts receive gradients for both the expert parameters and the router. Passive experts remain in the forward pass, but their $B_m$ blocks are detached during backpropagation. Their routing probabilities therefore continue to receive task-loss gradients while their expert parameters remain unchanged on client $i$. Only active expert blocks are uploaded, so $\mathcal{A}_i$ also determines the expert-specific communication cost of the round. The effect of training capacity on empirical performance is evaluated in Appendix B.5.

### C.4 Router Parameterization and Aggregation

FedSEE uses a lightweight linear router at each routed layer. For an input token representation $x$, the router produces expert logits through $W_r^\ell x + b_r^\ell$. Experts outside the client-specific inference set $\mathcal{U}_i$ are masked before applying the softmax, yielding routing probabilities over the experts available to client $i$. The KL term in Eq. (14) is applied at each routed layer and averaged over $\mathcal{P}$. Its mean-routing statistic $\bar{p}_i^\ell$ is estimated on each minibatch by averaging the routing probabilities over its examples and token positions. The router is optimized with a separate learning rate $\eta_r$.

After each communication round, the server averages the router parameters across participating clients and broadcasts the resulting shared router for the next round. Each client retains its own inference set $\mathcal{U}_i$ and mixture-calibration target $\tilde{c}_i$, so the shared router is trained under different client-specific masks and calibration targets across the federation. The theoretical justification for using a shared routing function is given in Appendix D.5.

## D Theoretical Results and Proofs

This appendix provides the complete proofs and supporting theoretical results for the framework and its FedSEE instantiation. We first develop the exact squared-loss analysis, including the barycentric geometry, expert recovery, identifiability limits, mixture estimation, and routing guarantees. We then extend the framework beyond the exact quadratic setting and analyze the additional approximations introduced by the FedSEE instantiation.

Appendix D.1 develops the barycentric geometry and the mixing gap. Appendix D.2 establishes the population and finite-sample recovery guarantees, Appendix D.3 gives the identifiability limits, Appendix D.4 studies mixture estimation and static reconstruction, and Appendix D.5 analyzes task-aligned routing. Appendix D.6 extends these results beyond the squared-loss setting, while Appendix D.7 discusses additional model-class instantiations. Finally, Appendices D.8, D.9, and D.10 analyze the deep-model approximation, balanced expert extraction, and temporal smoothing used by FedSEE.

**Notation.** Recall that $\|C\|_\Phi^2 := \operatorname{tr}(C\Phi C^\top)$. Since $\Phi \succ 0$,

$$\sigma_{\min}(\Phi)\|C\|_{\mathrm{F}}^2 \leq \|C\|_\Phi^2 \leq \sigma_{\max}(\Phi)\|C\|_{\mathrm{F}}^2.$$

Hence conversion between the $\Phi$-weighted and Frobenius norms introduces at most the factor $\kappa(\Phi)^{1/2}$, where $\kappa(\Phi) = \sigma_{\max}(\Phi)/\sigma_{\min}(\Phi)$. We use $[K] := \{1, \dots, K\}$ for positive integers $K$.

### D.1 Barycentric Geometry and the Mixing Gap

**Proof of Theorem 1 (Mixture recovery in $\theta$-space). Step 1. Specialize the mixture-risk decomposition.** By Eq. (3), the client population risk decomposes as

$$\mathcal{L}_i(\theta) = \sum_{m=1}^{M} \alpha_{im} \mathcal{L}_m(\theta).$$

Under the squared loss used in Section 3, this becomes

$$\mathcal{L}_i(\theta) = \sum_{m=1}^{M} \alpha_{im} \mathbb{E}_{(x,y)\sim\mathcal{P}_m} \left[ \|y - f_0(x) - \theta\phi(x)\|_2^2 \right]. \tag{17}$$

**Step 2. Apply the task-wise realizability model.** By Assumption 2(i), under $\mathcal{P}_m$,

$$y - f_0(x) = \theta_m^* \phi(x) + \zeta_m.$$

Substituting into Eq. (17) gives

$$\mathcal{L}_i(\theta) = \sum_{m=1}^{M} \alpha_{im} \mathbb{E}_{\mathcal{P}_m} \left[ \|(\theta_m^* - \theta)\phi(x) + \zeta_m\|_2^2 \right]. \tag{18}$$

Since $\mathbb{E}[\zeta_m \mid x] = 0$, the cross term $2\mathbb{E}[\langle(\theta_m^* - \theta)\phi(x),\, \zeta_m\rangle]$ vanishes. Expanding the remaining square gives

$$\mathcal{L}_i(\theta) = \sum_{m=1}^{M} \alpha_{im} \Big[ \mathbb{E}_{\mathcal{P}_m} \left[ \|(\theta_m^* - \theta)\phi(x)\|^2 \right] + \sigma_m^2 l \Big]. \tag{19}$$

**Step 3. Evaluate the quadratic term using the feature covariance.** Using $\|Cv\|^2 = \mathrm{tr}(Cvv^\top C^\top)$, linearity of trace and expectation, and Assumption 2(ii) ($\Phi_m = \Phi$), we obtain

$$\begin{aligned} \mathbb{E}_{\mathcal{P}_m} \left[ \|(\theta - \theta_m^*)\phi(x)\|^2 \right] &= \mathrm{tr}\big((\theta - \theta_m^*)\mathbb{E}_{\mathcal{P}_m}[\phi\phi^\top](\theta - \theta_m^*)^\top\big) \\ &= \mathrm{tr}\big((\theta - \theta_m^*)\Phi(\theta - \theta_m^*)^\top\big) = \|\theta - \theta_m^*\|_\Phi^2, \end{aligned} \tag{20}$$

where $\Phi = \mathbb{E}[\phi\phi^\top]$. Denoting $\Lambda_i = \sum_m \alpha_{im}\sigma_m^2 l$ as the constant term:

$$\mathcal{L}_i(\theta) = \sum_{m=1}^{M} \alpha_{im} \|\theta - \theta_m^*\|_\Phi^2 + \Lambda_i. \tag{21}$$

**Step 4. Solve the first-order optimality condition.** Using $\nabla_\theta \|\theta - C\|_\Phi^2 = 2(\theta - C)\Phi$, we obtain

$$\nabla_\theta \mathcal{L}_i(\theta) = 2 \sum_{m=1}^{M} \alpha_{im}(\theta - \theta_m^*)\Phi = 2\Big(\theta - \sum_{m=1}^{M} \alpha_{im}\theta_m^*\Big)\Phi. \tag{22}$$

Setting to zero and right-multiplying by $\Phi^{-1}$, which exists by Assumption 2(iii), gives

$$\theta_i^* = \sum_{m=1}^{M} \alpha_{im}\theta_m^*. \tag{23}$$

**Step 5. Uniqueness and global minimality.** The Hessian of $\mathcal{L}_i$ with respect to $\mathrm{vec}(\theta)$ is $2(\Phi \otimes I_l) \succ 0$, since $\Phi \succ 0$ by Assumption 2(iii). Hence $\mathcal{L}_i$ is strictly convex in $\theta$, and the stationary point above is the unique global minimizer. ■

**Remark 1 (Heterogeneous input covariance)** *Without Assumption 2(ii), the gradient in Step 4 yields a covariance-weighted average rather than a pure mixture average:*

$$\theta_i^* = \Big(\sum_m \alpha_{im}\theta_m^*\Phi_m\Big)\Big(\sum_m \alpha_{im}\Phi_m\Big)^{-1}, \quad \Phi_m := \mathbb{E}_{\mathcal{P}_m}[\phi\phi^\top]. \tag{24}$$

*The bias relative to the true mixture* $\bar{\theta}_i^{\mathrm{mix}} := \sum_m \alpha_{im}\theta_m^*$ *admits a clean closed form. Writing* $\bar{\Phi}_i := \sum_m \alpha_{im}\Phi_m$ *for the mixture-averaged feature covariance and using* $\sum_m \alpha_{im}(\Phi_m - \bar{\Phi}_i) = 0$*, the optimality condition gives*

$$\theta_i^* - \bar{\theta}_i^{\mathrm{mix}} = \Big(\sum_m \alpha_{im}(\theta_m^* - \bar{\theta}_i^{\mathrm{mix}})(\Phi_m - \bar{\Phi}_i)\Big)\,\bar{\Phi}_i^{-1}, \tag{25}$$

*so that, bounding* $\|\theta_m^* - \bar{\theta}_i^{\mathrm{mix}}\|_{\mathrm{F}} \le \mathrm{diam}_{\mathrm{F}}(\Theta)$ *and* $\sigma_{\min}(\bar{\Phi}_i) \ge \min_m \sigma_{\min}(\Phi_m)$ *(Weyl's inequality),*

$$\|\bar{\theta}_i^{\mathrm{mix}} - \theta_i^*\|_{\mathrm{F}} \;\le\; \rho_\Phi \cdot \mathrm{diam}_{\mathrm{F}}(\Theta), \qquad \rho_\Phi \;:=\; \frac{\max_{m,m'} \|\Phi_m - \Phi_{m'}\|_2}{\min_m \sigma_{\min}(\Phi_m)}. \tag{26}$$

*The controlling quantity is the* dispersion *of the per-task feature covariances, not the conditioning of any single one:* $\rho_\Phi$ *depends on how much the* $\Phi_m$ *differ from one another, and vanishes exactly when they coincide. As the task feature covariances align,* $\rho_\Phi \to 0$ *and Theorem 1 is recovered in the limit. In particular, when all* $\Phi_m$ *are equal the bias is exactly zero even if the shared* $\Phi$ *is ill-conditioned — a regime the worst-case condition number* $\max_m \kappa(\Phi_m)$ *would have spuriously penalized.*

**Proposition 2 (Simplex faces and mixing-gap geometry)** *Suppose the task experts are affinely independent, equivalently* $h > 0$*, and let* $S_i := \{m : \alpha_{im} > 0\}$*. Then*

$$\theta_i^* \in \mathrm{relint}\,\mathrm{conv}\{\theta_m^* : m \in S_i\}.$$

*Hence* $\theta_i^*$ *lies in the relative interior of the full simplex iff* $\alpha_{im} > 0$ *for every* $m$*, and is a vertex iff client* $i$ *is pure.*

*Moreover, the mixing gap satisfies*

$$G_i = \frac{1}{2}\sum_{m,k} \alpha_{im}\alpha_{ik}\|\theta_m^* - \theta_k^*\|_\Phi^2, \tag{27}$$

*and consequently*

$$\frac{\sigma_{\min}(\Phi)\mathrm{sep}^2}{2}\left(1 - \|\alpha_i\|_2^2\right) \le G_i \le \frac{\sigma_{\max}(\Phi)\mathrm{diam}_{\mathrm{F}}(\Theta)^2}{2}\left(1 - \|\alpha_i\|_2^2\right). \tag{28}$$

*Thus* $G_i = 0$ *iff client* $i$ *is pure.*

**Proof of Proposition 2.** Since $h > 0$, the task experts are affinely independent and their convex hull is an $(M-1)$-simplex. By Theorem 1,

$$\theta_i^* = \sum_m \alpha_{im}\theta_m^*.$$

Barycentric coordinates in a non-degenerate simplex are unique. A point lies in the relative interior of the face generated by a set of vertices exactly when its coefficients on those vertices are strictly positive and all remaining coefficients vanish. The face, full-simplex, and vertex claims therefore follow.

For the mixing gap, let $\bar{\theta}_i := \sum_m \alpha_{im}\theta_m^* = \theta_i^*$. The weighted variance identity under the inner product induced by $\Phi$ yields

$$\begin{aligned} G_i &= \sum_m \alpha_{im}\|\theta_m^* - \bar{\theta}_i\|_\Phi^2 \\ &= \frac{1}{2}\sum_{m,k} \alpha_{im}\alpha_{ik}\|\theta_m^* - \theta_k^*\|_\Phi^2. \end{aligned}$$

For $m \neq k$,

$$\sigma_{\min}(\Phi)\mathrm{sep}^2 \le \|\theta_m^* - \theta_k^*\|_\Phi^2 \le \sigma_{\max}(\Phi)\mathrm{diam}_{\mathrm{F}}(\Theta)^2.$$

Finally,

$$\sum_{m \neq k} \alpha_{im}\alpha_{ik} = 1 - \sum_m \alpha_{im}^2 = 1 - \|\alpha_i\|_2^2,$$

which gives Eq. (28). Because $\mathrm{sep} > 0$, equality $G_i = 0$ holds exactly when $\|\alpha_i\|_2^2 = 1$, i.e., when $\alpha_i$ is a simplex vertex. ■

Proposition 2 shows that the gap between static adaptation and task-specific specialization is determined by the client's mixture structure and vanishes exactly for pure clients. We next study when the underlying task experts can be recovered from mixed client uploads.

### D.2 Supervised Expert Recovery and Finite-Sample Guarantees

**Proof of Theorem 2 (Population expert recovery). Step 1. Bound the anchor client's distance from the task expert.** By Theorem 1, anchor client $i_m$ satisfies:

$$\theta^*_{i_m} - \theta^*_m = \sum_{m' \neq m} \alpha_{i_m,m'}(\theta^*_{m'} - \theta^*_m). \tag{29}$$

Taking Frobenius norms and applying the triangle inequality:

$$\|\theta^*_{i_m} - \theta^*_m\|_{\mathrm{F}} \leq \underbrace{\sum_{m' \neq m} \alpha_{i_m,m'}}_{=1-\alpha_{i_m,m} \leq \varepsilon} \cdot \mathrm{diam}_{\mathrm{F}}(\Theta) \leq \varepsilon \cdot \mathrm{diam}_{\mathrm{F}}(\Theta). \tag{30}$$

**Step 2. Translate to the $\Phi$-norm.** Using $\|C\|_\Phi \leq \sigma_{\max}(\Phi)^{1/2}\|C\|_{\mathrm{F}}$:

$$\|\theta^*_{i_m} - \theta^*_m\|_\Phi \leq \sigma_{\max}(\Phi)^{1/2} \cdot \varepsilon \cdot \mathrm{diam}_{\mathrm{F}}(\Theta). \tag{31}$$

**Step 3. Apply the projection bound.** The server minimizes $\|\theta - \theta^*_m\|^2_\Phi$ over $\mathrm{conv}(\{\theta^*_i\})$. Since $\theta^*_{i_m} \in \mathrm{conv}(\{\theta^*_i\})$ is feasible and $\theta^{\mathrm{ext}}_m$ is the minimizer:

$$\|\theta^{\mathrm{ext}}_m - \theta^*_m\|_\Phi \leq \|\theta^*_{i_m} - \theta^*_m\|_\Phi \leq \sigma_{\max}(\Phi)^{1/2} \cdot \varepsilon \cdot \mathrm{diam}_{\mathrm{F}}(\Theta). \tag{32}$$

**Step 4. Convert back to Frobenius norm.** Using $\|C\|_{\mathrm{F}} \leq \sigma_{\min}(\Phi)^{-1/2}\|C\|_\Phi$:

$$\|\theta^{\mathrm{ext}}_m - \theta^*_m\|_{\mathrm{F}} \leq \frac{\sigma_{\max}(\Phi)^{1/2}}{\sigma_{\min}(\Phi)^{1/2}} \cdot \varepsilon \cdot \mathrm{diam}_{\mathrm{F}}(\Theta) = \kappa(\Phi)^{1/2} \cdot \varepsilon \cdot \mathrm{diam}_{\mathrm{F}}(\Theta). \tag{33}$$

■

**Corollary 1 (Exact recovery with pure clients)** *If $\varepsilon = 0$, then $\theta^{\mathrm{ext}}_m = \theta^*_m$ for all $m \in [M]$.*

$\varepsilon = 0$ implies $\theta^*_{i_m} = \theta^*_m$ by Theorem 1. Since $\theta^*_m$ is feasible, the projection objective achieves value $\|\theta^*_m - \theta^*_m\|^2_\Phi = 0$, uniquely attained at $\theta^{\mathrm{ext}}_m = \theta^*_m$.

In the finite-sample setting, clients upload estimated parameters $\hat{\theta}_i$ rather than population optima $\theta^*_i$, and the server evaluates empirical task losses on finite public reference sets. The following boundedness and tail conditions provide the concentration required for the finite-sample recovery analysis.

**Assumption 4 (Boundedness)** *The feature map is bounded, $\|\phi(x)\|_2 \leq Q$ almost surely, and the uploaded parameters satisfy $\|\hat{\theta}_i\|_{\mathrm{F}} \leq \theta_{\max}$ for all $i \in [N]$.*

**Assumption 5 (Finite-Sample Public Data)** *In addition to Assumption 1, each public set $\mathcal{D}^s_m$ is independent of the client-side data used to construct the uploaded parameters. Moreover, conditional on $x$, the task noise is $\sigma_m$-sub-Gaussian: for every $v \in \mathbb{R}^l$ and $t \in \mathbb{R}$,*

$$\mathbb{E}\big[\exp\big(t\, v^\top \zeta_m\big) \mid x\big] \leq \exp\left(\frac{t^2\sigma^2_m\|v\|^2_2}{2}\right).$$

**Proposition 3 (Finite-sample expert recovery)** *Under Assumptions 1–3, 4, and 5, suppose the upload pool is fixed independently of the public sample used for extraction and*

$$\|\hat{\theta}_i - \theta^*_i\|_{\mathrm{F}} \leq \delta_{\mathrm{loc}} \qquad \text{for all } i \in [N].$$

*Condition on the uploaded parameters $\{\hat{\theta}_i\}_{i=1}^N$ and define*

$$\mathcal{S} := \mathrm{span}\{\hat{\theta}_i - \hat{\theta}_1 : i = 2, \ldots, N\}, \qquad d_{\mathcal{S}} := \dim(\mathcal{S}) \leq N - 1,$$

*together with*

$$\chi_\Phi := \frac{Q^2}{\sigma_{\min}(\Phi)}.$$

*There exist universal constants $C_0, C_1 > 0$ such that, whenever*

$$n_m \geq C_1 \chi_\Phi \left( d_{\mathcal{S}} + \log \frac{2}{\delta} \right),$$

*with probability at least $1 - \delta$ over the public sample $\mathcal{D}_m^s$,*

$$\begin{aligned} \|\theta_m^{\mathrm{ext}} - \theta_m^*\|_{\mathrm{F}} \leq\ & \underbrace{\kappa(\Phi)^{1/2} \varepsilon \operatorname{diam}_{\mathrm{F}}(\Theta)}_{\text{(I) anchor error}} \\ & + \underbrace{\kappa(\Phi)^{1/2} \sqrt{\big(2\varepsilon \operatorname{diam}_{\mathrm{F}}(\Theta) + \delta_{\mathrm{loc}}\big)\delta_{\mathrm{loc}}}}_{\text{(II) client estimation}} \\ & + \underbrace{\frac{C_0 \sqrt{\sigma_m^2 + Q^2 \big(\operatorname{diam}_{\mathrm{F}}(\Theta) + \delta_{\mathrm{loc}}\big)^2}}{\sigma_{\min}(\Phi)^{1/2}} \sqrt{\frac{d_{\mathcal{S}} + \log(2/\delta)}{n_m}}}_{\text{(III) public-data estimation}}. \end{aligned} \tag{34}$$

*In particular, when $\varepsilon = 0$, Term* (II) *reduces to $\kappa(\Phi)^{1/2}\delta_{\mathrm{loc}}$. More generally,*

$$\sqrt{\big(2\varepsilon \operatorname{diam}_{\mathrm{F}}(\Theta) + \delta_{\mathrm{loc}}\big)\delta_{\mathrm{loc}}} \leq \delta_{\mathrm{loc}} + \sqrt{2\varepsilon \operatorname{diam}_{\mathrm{F}}(\Theta)\, \delta_{\mathrm{loc}}},$$

*so Terms* (I) *and* (II) *both vanish as $\varepsilon, \delta_{\mathrm{loc}} \to 0$, while Term* (III) *vanishes at the parametric $n_m^{-1/2}$ rate for fixed upload geometry.*

**Proof of Proposition 3.** We use the $\Phi$-projection characterization throughout. By Eq. (21), the population public loss for task $m$ obeys $\mathcal{L}_m(\theta) - \mathcal{L}_m(\theta_m^*) = \|\theta - \theta_m^*\|_\Phi^2$, so every population minimizer over a convex feasible set is the $\Phi$-projection of $\theta_m^*$ onto that set. Recall the norm equivalence $\sigma_{\min}(\Phi)^{1/2}\|\cdot\|_{\mathrm{F}} \leq \|\cdot\|_\Phi \leq \sigma_{\max}(\Phi)^{1/2}\|\cdot\|_{\mathrm{F}}$.

**A ladder of intermediate experts.** Let

$$\begin{aligned} \theta_m^{(0)} &:= \arg\min_{\theta \in \operatorname{conv}\{\theta_i^*\}} \mathcal{L}_m(\theta) && \text{(population loss, } \textit{true} \text{ uploads)}, \\ \theta_m^{(1)} &:= \arg\min_{\theta \in \operatorname{conv}\{\hat{\theta}_i\}} \mathcal{L}_m(\theta) && \text{(population loss, } \textit{noisy} \text{ uploads)}, \\ \theta_m^{(2)} &:= \arg\min_{\theta \in \operatorname{conv}\{\hat{\theta}_i\}} \widehat{\mathcal{L}}_m^s(\theta) = \theta_m^{\mathrm{ext}} && \text{(empirical loss, noisy uploads)}. \end{aligned}$$

By the triangle inequality,

$$\|\theta_m^{\mathrm{ext}} - \theta_m^*\|_{\mathrm{F}} \leq \underbrace{\|\theta_m^{(0)} - \theta_m^*\|_{\mathrm{F}}}_{(A)} + \underbrace{\|\theta_m^{(1)} - \theta_m^{(0)}\|_{\mathrm{F}}}_{(B1)} + \underbrace{\|\theta_m^{(2)} - \theta_m^{(1)}\|_{\mathrm{F}}}_{(B2)}. \tag{35}$$

**Term (A) — anchor error.** This is exactly Theorem 2:

$$(A) \leq \kappa(\Phi)^{1/2} \cdot \varepsilon \cdot \operatorname{diam}_{\mathrm{F}}(\Theta), \tag{36}$$

which is Term (I). We also record the $\Phi$-norm form $\|\theta_m^{(0)} - \theta_m^*\|_\Phi \leq \sigma_{\max}(\Phi)^{1/2}\varepsilon \operatorname{diam}_{\mathrm{F}}(\Theta)$, used below.

**Term (B1) — client noise via a variational projection estimate.** Both $\theta_m^{(0)}$ and $\theta_m^{(1)}$ are $\Phi$-projections of the *same* point $z := \theta_m^*$ onto two convex sets, $K := \operatorname{conv}\{\theta_i^*\}$ and $\hat{K} := \operatorname{conv}\{\hat{\theta}_i\}$. Their Hausdorff distance satisfies $d_H(K, \hat{K}) \leq \max_i \|\hat{\theta}_i - \theta_i^*\|_{\mathrm{F}} \leq \delta_{\mathrm{loc}}$ in Frobenius norm, hence $\eta := d_H^\Phi(K, \hat{K}) \leq \sigma_{\max}(\Phi)^{1/2}\delta_{\mathrm{loc}}$ in the $\Phi$-metric. We *cannot* invoke non-expansiveness here, because the projected point is fixed while the set moves — the resulting dependence is Hölder-$\frac{1}{2}$, not Lipschitz. Instead, write $p := \theta_m^{(0)} = \Pi_K^\Phi(z)$ and $p' := \theta_m^{(1)} = \Pi_{\hat{K}}^\Phi(z)$, and choose $q \in K$ with $\|q - p'\|_\Phi \leq \eta$ and $q' \in \hat{K}$ with $\|q' - p\|_\Phi \leq \eta$ (possible since $d_H^\Phi \leq \eta$). The variational characterizations of the two projections, $\langle z - p,\, q - p\rangle_\Phi \leq 0$ and $\langle z - p',\, q' - p'\rangle_\Phi \leq 0$, give

$$\langle z - p,\, p - p'\rangle_\Phi = \langle z - p,\, p - q\rangle_\Phi + \langle z - p,\, q - p'\rangle_\Phi \ \geq\ -\|z - p\|_\Phi\, \eta, \tag{37}$$

$$\langle z - p',\, p - p'\rangle_\Phi = \langle z - p',\, p - q'\rangle_\Phi + \langle z - p',\, q' - p'\rangle_\Phi \ \leq\ \|z - p'\|_\Phi\, \eta. \tag{38}$$

Subtracting,

$$\|p-p'\|_\Phi^2 = \langle z-p',\, p-p'\rangle_\Phi - \langle z-p,\, p-p'\rangle_\Phi \le \big(\|z-p\|_\Phi + \|z-p'\|_\Phi\big)\,\eta. \tag{39}$$

For $\|z-p'\|_\Phi$, the perturbed anchor $\hat\theta_{i_m} \in \hat K$ is feasible with $\|\theta_m^* - \hat\theta_{i_m}\|_{\mathrm F} \le \varepsilon\,\mathrm{diam}_{\mathrm F}(\Theta) + \delta_{\mathrm{loc}}$, so $\|z-p'\|_\Phi \le \sigma_{\max}(\Phi)^{1/2}(\varepsilon\,\mathrm{diam}_{\mathrm F}(\Theta) + \delta_{\mathrm{loc}})$. Substituting this and the Term (A) bound on $\|z-p\|_\Phi$ together with $\eta$ into Eq. (39),

$$\|p-p'\|_\Phi^2 \le \sigma_{\max}(\Phi)\,\big(2\varepsilon\,\mathrm{diam}_{\mathrm F}(\Theta) + \delta_{\mathrm{loc}}\big)\,\delta_{\mathrm{loc}}. \tag{40}$$

Converting to Frobenius norm via $\|\cdot\|_{\mathrm F} \le \sigma_{\min}(\Phi)^{-1/2}\|\cdot\|_\Phi$,

$$(B1) = \|\theta_m^{(1)} - \theta_m^{(0)}\|_{\mathrm F} \le \kappa(\Phi)^{1/2}\sqrt{\big(2\varepsilon\,\mathrm{diam}_{\mathrm F}(\Theta) + \delta_{\mathrm{loc}}\big)\,\delta_{\mathrm{loc}}}\,, \tag{41}$$

which is Term (II). At $\varepsilon = 0$ this is exactly $\kappa(\Phi)^{1/2}\delta_{\mathrm{loc}}$, recovering the Lipschitz rate.

**Term (B2) — empirical versus population loss over a fixed feasible set.** Now $\theta_m^{(1)}$ and $\theta_m^{(2)}$ minimize, respectively, the population and empirical public losses over the same convex set

$$\hat K := \mathrm{conv}\{\hat\theta_1, \dots, \hat\theta_N\}.$$

Conditional on the uploaded parameters, $\hat K$ is fixed and the public sample remains i.i.d.

Define

$$\mathcal S := \mathrm{span}\{\hat\theta_i - \hat\theta_1 : i = 2, \dots, N\}, \qquad d_{\mathcal S} := \dim(\mathcal S),$$

and let

$$\Delta := \theta_m^{(2)} - \theta_m^{(1)}.$$

Since both parameters lie in $\hat K$,

$$\Delta \in \mathcal S.$$

For a public sample $\mathcal D_m^s = \{(x_j, y_j)\}_{j=1}^{n_m}$, write

$$\widehat{\mathbb E}_m[h] := \frac{1}{n_m}\sum_{j=1}^{n_m} h(x_j, y_j), \qquad \mathbb E_m[h] := \mathbb E_{(x,y)\sim\mathcal P_m}[h(x,y)],$$

and define the residual at the population in-class optimum

$$r_1(x,y) := y - f_0(x) - \theta_m^{(1)}\phi(x).$$

Because $\theta_m^{(2)}$ minimizes the empirical squared loss over $\hat K$,

$$\widehat{\mathbb E}_m\|\Delta\phi(x)\|_2^2 \le 2\widehat{\mathbb E}_m\langle r_1(x,y), \Delta\phi(x)\rangle. \tag{42}$$

On the other hand, first-order optimality of $\theta_m^{(1)}$ for the population loss over the convex set $\hat K$ gives

$$\Big\langle \nabla\mathcal L_m(\theta_m^{(1)}), \Delta\Big\rangle_{\mathrm F} \ge 0.$$

Since

$$\nabla\mathcal L_m(\theta_m^{(1)}) = -2\mathbb E_m[r_1(x,y)\phi(x)^\top],$$

this is equivalent to

$$\mathbb E_m\langle r_1(x,y), \Delta\phi(x)\rangle \le 0.$$

Combining with Eq. (42),

$$\widehat{\mathbb E}_m\|\Delta\phi(x)\|_2^2 \le 2\,\Big|(\widehat{\mathbb E}_m - \mathbb E_m)\langle r_1(x,y), \Delta\phi(x)\rangle\Big|\,. \tag{43}$$

*Step (i): restricted empirical quadratic form.* Choose a $\Phi$-orthonormal basis $U_1, \dots, U_{d_{\mathcal S}}$ of $\mathcal S$, i.e.

$$\langle U_j, U_k\rangle_\Phi := \mathrm{tr}(U_j\Phi U_k^\top) = \mathbf 1\{j=k\}.$$

For $a \in \mathbb R^{d_{\mathcal S}}$, write

$$\Delta(a) := \sum_{k=1}^{d_{\mathcal S}} a_k U_k, \qquad \|\Delta(a)\|_\Phi = \|a\|_2,$$

and define

$$V(x) := \big[U_1\phi(x), \dots, U_{d_{\mathcal{S}}}\phi(x)\big] \in \mathbb{R}^{l\times d_{\mathcal{S}}}.$$

Then

$$\mathbb{E}_m[V(x)^\top V(x)] = I_{d_{\mathcal{S}}}.$$

Furthermore,

$$\|V(x)^\top V(x)\|_{\mathrm{op}} = \sup_{\|a\|_2=1} \|\Delta(a)\phi(x)\|_2^2 \le \frac{Q^2}{\sigma_{\min}(\Phi)} = \chi_\Phi.$$

Therefore, a matrix Chernoff bound for sums of independent positive semidefinite matrices gives, with probability at least $1-\delta/2$,

$$\widehat{\mathbb{E}}_m\|\Gamma\phi(x)\|_2^2 \ge \frac{1}{2}\|\Gamma\|_\Phi^2 \qquad \text{for every } \Gamma\in\mathcal{S}, \tag{44}$$

provided

$$n_m \ge C\chi_\Phi \log\frac{2d_{\mathcal{S}}}{\delta}.$$

The sample-size condition in the proposition is stronger than this requirement and therefore suffices.

*Step (ii): the centered linear process.* Let

$$R_m := \|\theta_m^{(1)} - \theta_m^*\|_{\mathrm{F}}.$$

Because $\theta_m^{(1)} \in \hat{K}$, there is a convex combination of the true uploads within $\delta_{\mathrm{loc}}$ of $\theta_m^{(1)}$. Since every true client optimum lies in $\mathrm{conv}(\Theta)$,

$$R_m \le \mathrm{diam}_{\mathrm{F}}(\Theta) + \delta_{\mathrm{loc}}. \tag{45}$$

Writing

$$r_1(x,y) = (\theta_m^* - \theta_m^{(1)})\phi(x) + \zeta_m,$$

consider any $\Gamma\in\mathcal{S}$ with $\|\Gamma\|_\Phi = 1$. The centered random variable

$$Z_\Gamma := \langle r_1(x,y), \Gamma\phi(x)\rangle - \mathbb{E}_m\langle r_1(x,y), \Gamma\phi(x)\rangle$$

has variance bounded by

$$\mathrm{Var}(Z_\Gamma) \le \sigma_m^2 + Q^2R_m^2. \tag{46}$$

Indeed, the noise contribution has variance

$$\mathbb{E}_m[\sigma_m^2\|\Gamma\phi(x)\|_2^2] = \sigma_m^2,$$

while the signal contribution satisfies

$$\mathbb{E}_m\langle(\theta_m^* - \theta_m^{(1)})\phi(x), \Gamma\phi(x)\rangle^2 \le Q^2R_m^2\mathbb{E}_m\|\Gamma\phi(x)\|_2^2 = Q^2R_m^2.$$

The cross term vanishes because $\mathbb{E}[\zeta_m \mid x] = 0$.

Moreover,

$$\|\Gamma\|_{\mathrm{F}} \le \sigma_{\min}(\Phi)^{-1/2},$$

so

$$\|\Gamma\phi(x)\|_2 \le \sqrt{\chi_\Phi}.$$

Assumption 5 therefore makes the noise component sub-Gaussian at scale at most $\sigma_m\sqrt{\chi_\Phi}$, while the centered signal component is bounded at scale at most a universal constant times $QR_m\sqrt{\chi_\Phi}$. Consequently $Z_\Gamma$ satisfies a Bernstein-type tail bound with variance proxy

$$v_m^2 := \sigma_m^2 + Q^2R_m^2$$

and scale at most $C\sqrt{\chi_\Phi}\,v_m$.

Apply this scalar Bernstein bound on a $1/2$-net of the unit sphere in the $d_{\mathcal{S}}$-dimensional space $\mathcal{S}$. Such a net has cardinality at most $5^{d_{\mathcal{S}}}$. A union bound and the standard net extension therefore give, with probability at least $1-\delta/2$,

$$\sup_{\substack{\Gamma\in\mathcal{S}\\ \|\Gamma\|_\Phi=1}} \Big|(\widehat{\mathbb{E}}_m - \mathbb{E}_m)\langle r_1(x,y), \Gamma\phi(x)\rangle\Big|$$

$$\le Cv_m\left[\sqrt{\frac{d_{\mathcal{S}} + \log(2/\delta)}{n_m}} + \sqrt{\chi_\Phi}\,\frac{d_{\mathcal{S}} + \log(2/\delta)}{n_m}\right]. \tag{47}$$

Under the proposition's sample-size condition

$$n_m \geq C_1 \chi_\Phi \left( d_{\mathcal{S}} + \log \frac{2}{\delta} \right),$$

the second term in brackets is dominated by the first. Hence, after enlarging the universal constant,

$$\left| (\widehat{\mathbb{E}}_m - \mathbb{E}_m) \langle r_1(x,y), \Gamma\phi(x) \rangle \right| \leq C v_m \|\Gamma\|_\Phi \sqrt{\frac{d_{\mathcal{S}} + \log(2/\delta)}{n_m}} \tag{48}$$

simultaneously for every $\Gamma \in \mathcal{S}$.

*Step (iii): parameter error.* Applying Eq. (44) and Eq. (48) to $\Delta = \theta_m^{(2)} - \theta_m^{(1)}$ in Eq. (43) gives

$$\frac{1}{2}\|\Delta\|_\Phi^2 \leq 2 C v_m \|\Delta\|_\Phi \sqrt{\frac{d_{\mathcal{S}} + \log(2/\delta)}{n_m}}.$$

If $\Delta = 0$ there is nothing to prove. Otherwise, dividing by $\|\Delta\|_\Phi$ and absorbing numerical constants yields

$$\|\Delta\|_\Phi \leq C_0 v_m \sqrt{\frac{d_{\mathcal{S}} + \log(2/\delta)}{n_m}}.$$

Using Eq. (45) and $\|\Delta\|_{\mathrm{F}} \leq \sigma_{\min}(\Phi)^{-1/2} \|\Delta\|_\Phi$, we obtain

$$(B2) \leq \frac{C_0 \sqrt{\sigma_m^2 + Q^2 \big(\mathrm{diam}_{\mathrm{F}}(\Theta) + \delta_{\mathrm{loc}}\big)^2}}{\sigma_{\min}(\Phi)^{1/2}} \sqrt{\frac{d_{\mathcal{S}} + \log(2/\delta)}{n_m}}, \tag{49}$$

which is Term (III).

**Collecting all terms.** Combining (A), (B1), and (B2),

$$\begin{aligned} \|\theta_m^{\mathrm{ext}} - \theta_m^*\|_{\mathrm{F}} \leq\ & \underbrace{\kappa(\Phi)^{1/2} \varepsilon\, \mathrm{diam}_{\mathrm{F}}(\Theta)}_{\text{(I)}} \\ & + \underbrace{\kappa(\Phi)^{1/2} \sqrt{\big(2\varepsilon\, \mathrm{diam}_{\mathrm{F}}(\Theta) + \delta_{\mathrm{loc}}\big)\delta_{\mathrm{loc}}}}_{\text{(II)}} \\ & + \underbrace{\frac{C_0 \sqrt{\sigma_m^2 + Q^2 \big(\mathrm{diam}_{\mathrm{F}}(\Theta) + \delta_{\mathrm{loc}}\big)^2}}{\sigma_{\min}(\Phi)^{1/2}} \sqrt{\frac{d_{\mathcal{S}} + \log(2/\delta)}{n_m}}}_{\text{(III)}}. \end{aligned} \tag{50}$$

This is exactly the bound in Proposition 3. ■

The bound separates three sources of recovery error: anchor impurity, local estimation error, and finite public-sample error. These terms vanish respectively as $\varepsilon \to 0$, $\delta_{\mathrm{loc}} \to 0$, and $n_m \to \infty$.

**Remark 2 (On the coupling of the error terms)** *Term (II) is not fully decoupled from the anchor error: when $\varepsilon > 0$ the perturbed feasible hull $\mathrm{conv}\{\hat{\theta}_i\}$ is displaced from a set whose $\Phi$-projection of $\theta_m^*$ is itself $O(\varepsilon)$ away, producing the cross-term $\sqrt{\varepsilon\, \delta_{\mathrm{loc}}}$. This interaction is intrinsic to projecting a fixed point onto a moving convex set — the dependence on the set is Hölder-$\frac{1}{2}$ in Hausdorff distance, not Lipschitz — and it vanishes both as $\varepsilon \to 0$ (pure anchors, Lipschitz recovery) and as $\delta_{\mathrm{loc}} \to 0$ (exact uploads). The interaction therefore does not alter consistency of the finite-sample recovery bound: together with Term* (III) *of Proposition 3, the full recovery error vanishes as $\varepsilon \to 0$, $\delta_{\mathrm{loc}} \to 0$, and $n_m \to \infty$ for fixed upload geometry.*

### D.3 Identifiability Limits

Throughout, an *instance* $\mathcal{I} = (\{\theta_m^*\}_{m=1}^M, \{\alpha_i\}_{i=1}^N)$ satisfies Assumptions 2, with task $m$ generated by $y = f_0(x) + \theta_m^* \phi(x) + \zeta_m$ over the common input marginal; by Theorem 1 it induces the upload pool $\theta_i^* = \sum_m \alpha_{im} \theta_m^*$.

**Proof of Theorem 3 (Label indeterminacy). Step 1. The permuted instance is observationally equivalent.** Write $\theta'_m(\mathcal{I}^\pi) = \theta^*_{\pi(m)}$, $\alpha'_{im} = \alpha_{i\pi(m)}$, and $\mathcal{P}'_m = \mathcal{P}_{\pi(m)}$. The three label-free observables coincide. *(i) Client data.* Under $\mathcal{I}^\pi$, client $i$ draws from $\sum_m \alpha'_{im}\mathcal{P}'_m = \sum_m \alpha_{i\pi(m)}\mathcal{P}_{\pi(m)} = \sum_k \alpha_{ik}\mathcal{P}_k$ (reindex $k = \pi(m)$), exactly client $i$'s distribution under $\mathcal{I}$. *(ii) Upload pool.* By Theorem 1, $\theta^*_i(\mathcal{I}^\pi) = \sum_m \alpha'_{im}\theta'_m = \sum_k \alpha_{ik}\theta^*_k = \theta^*_i(\mathcal{I})$. *(iii) Unlabeled corpus.* $\frac{1}{M}\sum_m \mathcal{P}'_m = \frac{1}{M}\sum_k \mathcal{P}_k$ is permutation-invariant.

**Step 2. Two-point argument.** Choose a pair $(m, m')$ attaining

$$\|\theta^*_m - \theta^*_{m'}\|_{\mathrm{F}} = \mathrm{diam}_{\mathrm{F}}(\Theta),$$

and let $\pi$ transpose this pair. By Step 1, $\mathcal{I}$ and $\mathcal{I}^\pi$ induce identical observables, so $\mathcal{A}$ has the same output distribution on both instances. Couple the two runs to have the same realization $(\hat\theta_1, \dots, \hat\theta_M)$. Then

$$\begin{aligned}\max_k \|\hat\theta_k - \theta^*_k(\mathcal{I})\|_{\mathrm{F}} + \max_k \|\hat\theta_k - \theta^*_k(\mathcal{I}^\pi)\|_{\mathrm{F}} \\ \geq \|\hat\theta_m - \theta^*_m\|_{\mathrm{F}} + \|\hat\theta_m - \theta^*_{m'}\|_{\mathrm{F}} \\ \geq \|\theta^*_m - \theta^*_{m'}\|_{\mathrm{F}} = \mathrm{diam}_{\mathrm{F}}(\Theta).\end{aligned}$$

Taking expectations shows that at least one of the two observationally equivalent instances has task-indexed error at least $\mathrm{diam}_{\mathrm{F}}(\Theta)/2$. Since both belong to $\mathcal{O}(\mathcal{I})$, Eq. (9) follows.

**Remark 3 (Scope)** *The argument uses only that the abstract tasks are exchangeable within the linear model. If, in a given application, tasks carry a priori distinguishable structure known to the estimator (e.g., disjoint label vocabularies), the permutation symmetry is broken by that side information—which is itself a form of task supervision, consistent with our thesis.*

**Proof of Theorem 4 (Vertex-set non-identifiability). Step 1. Shrunk instance.** Let $\bar\theta := \frac{1}{M}\sum_m \theta^*_m$ and, for $t \in (0, 1)$, set

$$\tilde\theta_m := (1-t)\,\theta^*_m + t\,\bar\theta, \qquad \tilde\alpha_{im} := \frac{\alpha_{im} - t/M}{1-t}. \tag{51}$$

Since each client is $\gamma$-interior and $t \leq \gamma$, $\alpha_{im} \geq \gamma/M \geq t/M$, so $\tilde\alpha_{im} \geq 0$; and $\sum_m \tilde\alpha_{im} = (1 - M \cdot t/M)/(1-t) = 1$. Thus $\tilde{\mathcal{I}} := (\{\tilde\theta_m\}, \{\tilde\alpha_i\})$ is a valid instance.

**Step 2. Identical pools.** For every $i$,

$$\begin{aligned}\sum_m \tilde\alpha_{im}\tilde\theta_m = \sum_m \frac{\alpha_{im} - t/M}{1-t}\big[(1-t)\theta^*_m + t\bar\theta\big] = \sum_m (\alpha_{im} - t/M)\theta^*_m + \frac{t}{1-t}\bar\theta\sum_m(\alpha_{im} - t/M) \\ = \theta^*_i - t\bar\theta + \frac{t}{1-t}\bar\theta(1-t) = \theta^*_i,\end{aligned} \tag{52}$$

so $\mathcal{P}(\tilde{\mathcal{I}}) = \mathcal{P}(\mathcal{I})$ and any pool-only estimator has the same output law on both.

**Step 3. Vertex-set geometry.** With $\bar\theta_{-m} := \frac{1}{M-1}\sum_{k\neq m}\theta^*_k$, $\|\tilde\theta_m - \theta^*_m\|_{\mathrm{F}} = t\|\theta^*_m - \bar\theta\|_{\mathrm{F}} = t\frac{M-1}{M}\|\theta^*_m - \bar\theta_{-m}\|_{\mathrm{F}} \geq t\frac{M-1}{M}h$, since $\bar\theta_{-m} \in \mathrm{aff}\{\theta^*_k : k \neq m\}$. Also $\|\theta^*_m - \bar\theta\|_{\mathrm{F}} \leq \mathrm{diam}_{\mathrm{F}}(\Theta)$, so the displacement is at most $t\,\mathrm{diam}_{\mathrm{F}}(\Theta) \leq \mathrm{sep}/2$ by the choice of $t$. For $k \neq m$, $\|\theta^*_m - \tilde\theta_k\|_{\mathrm{F}} \geq \mathrm{sep} - t\,\mathrm{diam}_{\mathrm{F}}(\Theta) \geq \mathrm{sep}/2$. Hence the minimal distance between the two vertex sets is

$$d := \min_{m,k}\|\theta^*_m - \tilde\theta_k\|_{\mathrm{F}} \;\geq\; t\frac{M-1}{M}h \;>\; 0. \tag{53}$$

**Step 4. Matching.** For any common output $\hat{\mathcal{V}}$, let $e_1 := \min_\pi \max_m \|\hat\theta_{\pi(m)} - \theta^*_m\|_{\mathrm{F}}$ and $e_2 := \min_\pi \max_m \|\hat\theta_{\pi(m)} - \tilde\theta_m\|_{\mathrm{F}}$. Let $\pi_1$ and $\pi_2$ attain these minima. For any $m$, set $k := \pi_2^{-1}(\pi_1(m))$. Then

$$\|\theta^*_m - \tilde\theta_k\|_{\mathrm{F}} \leq \|\theta^*_m - \hat\theta_{\pi_1(m)}\|_{\mathrm{F}} + \|\hat\theta_{\pi_2(k)} - \tilde\theta_k\|_{\mathrm{F}} \leq e_1 + e_2.$$

By Eq. (53), $e_1 + e_2 \geq d$ for every realization. Therefore $\mathbb{E}_{\mathcal{A}}[e_1] + \mathbb{E}_{\mathcal{A}}[e_2] \geq d$, so

$$\max\{\mathbb{E}_{\mathcal{A}}[e_1], \mathbb{E}_{\mathcal{A}}[e_2]\} \geq \frac{d}{2} \geq \frac{M-1}{2M}\,t\,h,$$

which proves the claim. ■

**Remark 4 (Relation between interiority and anchor quality)** *Under the $\gamma$-interior condition, every client's purity is at most $1-(M-1)\gamma/M$, so the best achievable anchor impurity satisfies $\varepsilon \geq (M-1)\gamma/M$. For $\gamma \leq \mathrm{sep}/(2\,\mathrm{diam}_{\mathrm{F}}(\Theta))$, Theorem 4 gives a pool-only lower bound linear in $\gamma$. Hence, on families for which $\varepsilon = \Theta(\gamma)$, pool-only vertex recovery has a linear lower bound in anchor impurity, matching the linear dependence of Theorem 2 up to conditioning and simplex geometry. This lower bound concerns upload-only recovery; supervised recovery is distinguished separately in Remark 5.*

**Remark 5 (Distinct roles of anchor information)** *For pool-only estimators, Theorem 4 gives an information-theoretic obstruction when the population remains $\gamma$-interior: without anchors the upload pool alone is consistent with multiple vertex sets (Theorem 4). The supervised program is not subject to this particular obstruction — the instances $\mathcal{I}$ and $\tilde{\mathcal{I}}$ of Theorem 4 induce* different *labeled public losses, since $\tilde{\theta}_m \neq \theta^*_m$, so the $\mathbf{\Omega}$ program can distinguish them. The reason the program nonetheless requires anchors is* algorithmic rather than information-theoretic*: its expert is constrained to $\mathrm{conv}\{$uploads$\}$, and when no near-pure client exists for task $m$, the true vertex $\theta^*_m$ lies strictly* outside *this hull, so no convex combination of uploads can reach it — exactly the $O(\varepsilon)$ projection gap (Term (I)) of Theorem 2. These mechanisms are distinct. Theorem 4 establishes an information-theoretic limitation of pool-only recovery, whereas the anchor requirement of the supervised extraction program is a feasibility limitation induced by its convex-hull constraint.*

**Proposition 4 (Supervised identification margin)** *Let $\mathcal{V} = \{V_1, \ldots, V_M\}$ be any candidate vertex set with $\max_m \min_k \|V_k - \theta^*_m\|_{\mathrm{F}} \leq \eta$ for*

$$\eta \;\leq\; \frac{\mathrm{sep}}{2\big(1+\kappa(\Phi)^{1/2}\big)}, \tag{54}$$

*and define the supervised assignment $\hat{\sigma}(m) := \arg\min_{k\in[M]} \mathcal{L}_m(V_k)$.* (a) *With the population public loss, $\hat{\sigma}$ is exactly correct, and the loss margin over every incorrect candidate is at least $\frac{1}{4}\,\sigma_{\min}(\Phi)\,\mathrm{sep}^2$.* (b) *Under Assumptions 2 and 4 with sub-Gaussian noise $\|\zeta_m\|_{\psi_2} \leq \sigma_m$, there are universal constants $c, C > 0$ such that if*

$$n_m \;\geq\; C\,(K/G)^2\,\log(M^2/\delta), \quad K := Q^2\mathrm{diam}_{\mathrm{F}}(\Theta)^2 + \sigma_m Q\,\mathrm{diam}_{\mathrm{F}}(\Theta), \quad G := \tfrac{1}{4}\sigma_{\min}(\Phi)\,\mathrm{sep}^2, \tag{55}$$

*then with probability at least $1-\delta$ the empirical assignment $\arg\min_k \widehat{\mathcal{L}}^s_m(V_k)$ is correct simultaneously for all $m \in [M]$.*

**Proof of Proposition 4 (Supervised identification margin).** **Part (a).** From Eq. (21) specialized to the pure task-$m$ distribution, $\mathcal{L}_m(\theta) - \mathcal{L}_m(\theta^*_m) = \|\theta - \theta^*_m\|^2_\Phi$. Let $V_{k(m)}$ satisfy $\|V_{k(m)} - \theta^*_m\|_{\mathrm{F}} \leq \eta$, and let $V_j$ be within $\eta$ of some $\theta^*_{m'}$, $m' \neq m$ (once $\eta < \mathrm{sep}/2$, guaranteed by Eq. (54), the candidate-to-vertex assignment is injective). Then

$$\mathcal{L}_m(V_j) - \mathcal{L}_m(V_{k(m)}) = \|V_j - \theta^*_m\|^2_\Phi - \|V_{k(m)} - \theta^*_m\|^2_\Phi \geq \sigma_{\min}(\Phi)(\mathrm{sep}-\eta)^2 - \sigma_{\max}(\Phi)\eta^2, \tag{56}$$

using $\|V_j - \theta^*_m\|_{\mathrm{F}} \geq \mathrm{sep} - \eta$ and norm equivalence. By Eq. (54), $\eta \leq \mathrm{sep}/(2(1+\kappa^{1/2})) \leq \mathrm{sep}/4$, so $\sigma^{1/2}_{\min}(\mathrm{sep}-\eta) \geq \frac{3}{4}\sigma^{1/2}_{\min}\mathrm{sep}$ while $\sigma^{1/2}_{\max}\eta = \sigma^{1/2}_{\min}\kappa^{1/2}\eta \leq \frac{1}{2}\sigma^{1/2}_{\min}\mathrm{sep}$. Therefore the gap is at least $(\frac{9}{16} - \frac{1}{4})\sigma_{\min}(\Phi)\mathrm{sep}^2 \geq \frac{1}{4}\sigma_{\min}(\Phi)\mathrm{sep}^2 = G$, so the population argmin selects $V_{k(m)}$.

**Part (b).** Fix $m$ and a pair $(V_{k(m)}, V_j)$. For $(x,y) \sim \mathcal{P}_m$ let $g(x,y) := \ell(V_j, x, y) - \ell(V_{k(m)}, x, y)$. With $\Delta_j := V_j - \theta^*_m$, $\Delta_k := V_{k(m)} - \theta^*_m$ and $y - f_0(x) = \theta^*_m\phi(x) + \zeta_m$, $\mathbb{E}[g] = \|\Delta_j\|^2_\Phi - \|\Delta_k\|^2_\Phi \geq G$ by part (a). By Assumption 4 the quadratic terms are bounded by $C_1 Q^2 \mathrm{diam}_{\mathrm{F}}(\Theta)^2$ a.s., and the cross term is conditionally $\sigma_m$-sub-Gaussian scaled by at most $C_1 Q\,\mathrm{diam}_{\mathrm{F}}(\Theta)$; hence $\|g - \mathbb{E}g\|_{\psi_1} \leq C_2 K$ with $K = Q^2\mathrm{diam}_{\mathrm{F}}(\Theta)^2 + \sigma_m Q\,\mathrm{diam}_{\mathrm{F}}(\Theta)$. Bernstein's inequality gives, for the empirical mean $\bar{g}$ over $n_m$ samples, $\mathbb{P}(\bar{g} \leq 0) \leq 2\exp[-c\,n_m \min(G^2/K^2, G/K)]$. A union bound over the at most $M(M-1)$ ordered pairs shows all empirical assignments are simultaneously correct with probability $\geq 1-\delta$ provided $n_m \geq C(K/G)^2\log(M^2/\delta)$. ■

**Relation to supervised recovery.** The preceding results distinguish three aspects of expert recovery: recovery accuracy under task supervision, recovery of the unordered vertex set from client

uploads, and identification of the expert-to-task correspondence. We next relate the supervised upper bound of Theorem 2 to the two identifiability lower bounds.

Theorem 4 addresses recovery of the vertex *set* from the upload pool alone. Under the $\gamma$-interior condition its lower bound is linear in $\gamma$ whenever $\gamma \le \mathrm{sep}/(2\,\mathrm{diam}_{\mathrm{F}}(\Theta))$. Moreover, every $\gamma$-interior population has anchor impurity at least $(M-1)\gamma/M$. Thus the lower bound captures the same linear dependence on distance from the simplex boundary as the anchor term in Theorem 2; for families in which $\varepsilon = \Theta(\gamma)$, the two scale identically up to conditioning and simplex geometry. Theorem 4 applies to pool-only recovery. FedSEE's supervised program uses labeled public loss to identify task directions, while its anchor condition controls the geometric projection of each task vertex onto the convex hull of the uploads (Remark 5).

Furthermore, a sharp distinction exists between recovering the vertex set and identifying its task correspondence. Boundary information can make the unordered vertex set recoverable, but it cannot by itself determine which vertex corresponds to which task. As anchor clients approach purity, the supervised recovery error vanishes, whereas the task-indexed label-blind floor of Theorem 3 remains fixed. The task-labeled public loss supplies an identification margin: Proposition 4 shows that the population loss for task $m$ separates its correct vertex from the alternatives by at least $\frac{1}{4}\sigma_{\min}(\Phi)\,\mathrm{sep}^2$. Thus task-indexed side information is necessary to establish the true vertex–task correspondence; in FedSEE, this information is supplied by the labeled public data.

### D.4 Mixture Estimation and Static Reconstruction

Given a client upload $\hat{\theta}_i$ and the recovered experts, define its barycentric mixture estimate by

$$\hat{\alpha}_i := \operatorname*{arg\,min}_{a \in \Delta^{M-1}} \left\| \hat{\theta}_i - \sum_{m=1}^{M} a_m \theta_m^{\mathrm{ext}} \right\|_{\Phi}^2 . \tag{57}$$

The following result quantifies the accuracy of this estimator in terms of the local upload error, expert-recovery error, feature conditioning, and simplex geometry. We first record a geometric inequality used in the analysis.

**Lemma 1 (Centered combinations)** *For any $c \in \mathbb{R}^M$ with $\mathbf{1}^\top c = 0$, $\|\sum_m c_m \theta_m^*\|_{\mathrm{F}} \ge \|c\|_\infty h$.*

**Proof.** If $c = 0$, the claim is immediate. Otherwise, let $m^* := \arg\max_m |c_m|$. Since $\mathbf{1}^\top c = 0$, $\sum_m c_m \theta_m^* = c_{m^*}(\theta_{m^*}^* - \sum_{k \ne m^*} \beta_k \theta_k^*)$ with $\beta_k := -c_k/c_{m^*}$, $\sum_k \beta_k = 1$, an affine combination of $\{\theta_k^* : k \ne m^*\}$. Thus

$$\left\| \sum_m c_m \theta_m^* \right\|_{\mathrm{F}} = |c_{m^*}| \left\| \theta_{m^*}^* - \sum_{k \ne m^*} \beta_k \theta_k^* \right\|_{\mathrm{F}} \ge |c_{m^*}|\, \mathrm{dist}_{\mathrm{F}}\left(\theta_{m^*}^*, \mathrm{aff}\{\theta_k^* : k \ne m^*\}\right) \ge \|c\|_\infty h.$$

■

**Proposition 5 (Barycentric mixture estimation)** *Under Assumptions 2 and 4 with affinely independent experts ($h > 0$), let $\hat{\alpha}_i$ solve the barycentric projection Eq. (57) using an upload with $\|\hat{\theta}_i - \theta_i^*\|_{\mathrm{F}} \le \delta_{\mathrm{loc}}$ and recovered experts with $\max_m \|\theta_m^{\mathrm{ext}} - \theta_m^*\|_{\mathrm{F}} \le \xi_{\max}$ (Proposition 3). Then*

$$\|\hat{\alpha}_i - \alpha_i\|_1 \;\le\; \frac{2M}{h}\Big(\kappa(\Phi)^{1/2}\,(\delta_{\mathrm{loc}} + \xi_{\max}) + \xi_{\max}\Big) \;=:\; \eta_C. \tag{58}$$

*In particular, $\eta_C \to 0$ as $\delta_{\mathrm{loc}} \to 0$ and $\xi_{\max} \to 0$, and $\hat{\alpha}_i = \alpha_i$ when $\delta_{\mathrm{loc}} = \xi_{\max} = 0$.*

**Proof.** Write $\hat{\theta}_m := \theta_m^{\mathrm{ext}}$, $E_m := \hat{\theta}_m - \theta_m^*$ with $\|E_m\|_{\mathrm{F}} \le \xi_{\max}$, and $e_i := \hat{\theta}_i - \theta_i^*$ with $\|e_i\|_{\mathrm{F}} \le \delta_{\mathrm{loc}}$. Let $a^* := \alpha_i$, $\hat{a} := \hat{\alpha}_i$, and $c := \hat{a} - a^*$, so that $\mathbf{1}^\top c = 0$ (both lie in $\Delta^{M-1}$).

**Step 1. Residual at the truth.** Using $\theta_i^* = \sum_m \alpha_{im} \theta_m^*$ (Theorem 1), the residual of the projection objective Eq. (57) at $a^*$ is

$$\rho^* := \sum_m a_m^* \hat{\theta}_m - \hat{\theta}_i = \sum_m \alpha_{im} E_m - e_i, \qquad \|\rho^*\|_\Phi \le \sigma_{\max}(\Phi)^{1/2}(\xi_{\max} + \delta_{\mathrm{loc}}) =: r^*, \tag{59}$$

using $\sum_m \alpha_{im} = 1$ and $\|\cdot\|_\Phi \le \sigma_{\max}(\Phi)^{1/2}\|\cdot\|_{\mathrm{F}}$.

**Step 2. Near-optimal residual.** By optimality of $\hat{a}$ and feasibility of $a^*$ for the convex program Eq. (57), $\|\sum_m \hat{a}_m\hat{\theta}_m - \hat{\theta}_i\|_\Phi \le \|\rho^*\|_\Phi \le r^*$. Since $\sum_m \hat{a}_m\hat{\theta}_m - \hat{\theta}_i = \sum_m c_m\hat{\theta}_m + \rho^*$, the triangle inequality gives

$$\Big\|\sum_m c_m\hat{\theta}_m\Big\|_\Phi \le 2r^*, \qquad \text{hence} \qquad \Big\|\sum_m c_m\hat{\theta}_m\Big\|_{\mathrm{F}} \le \frac{2r^*}{\sigma_{\min}(\Phi)^{1/2}} = 2\kappa(\Phi)^{1/2}(\xi_{\max} + \delta_{\mathrm{loc}}). \tag{60}$$

**Step 3. Lower bound via the simplex height.** Since $\mathbf{1}^\top c = 0$, Lemma 1 gives $\|\sum_m c_m\theta^*_m\|_{\mathrm{F}} \ge \|c\|_\infty h$. Using $\|\sum_m c_m E_m\|_{\mathrm{F}} \le \|c\|_1\xi_{\max} \le 2\xi_{\max}$ (as $\|c\|_1 \le 2$ on the simplex),

$$\Big\|\sum_m c_m\hat{\theta}_m\Big\|_{\mathrm{F}} \ge \Big\|\sum_m c_m\theta^*_m\Big\|_{\mathrm{F}} - \Big\|\sum_m c_m E_m\Big\|_{\mathrm{F}} \ge \|c\|_\infty h - 2\xi_{\max}. \tag{61}$$

**Step 4. Combine.** Chaining Steps 2 and 3, $\|c\|_\infty h \le 2\kappa(\Phi)^{1/2}(\xi_{\max} + \delta_{\mathrm{loc}}) + 2\xi_{\max}$, and using $\|c\|_1 \le M\|c\|_\infty$,

$$\|\hat{\alpha}_i - \alpha_i\|_1 = \|c\|_1 \le \frac{2M}{h}\Big(\kappa(\Phi)^{1/2}(\delta_{\mathrm{loc}} + \xi_{\max}) + \xi_{\max}\Big) = \eta_C. \tag{62}$$

■

The dependence on $h$ shows that greater affine separation among the task experts improves the conditioning of barycentric mixture estimation. The same geometric quantity also appears in the pool-only vertex-recovery lower bound, although it plays a different role in that result. We next propagate the mixture-estimation and expert-recovery errors to the reconstruction of the client's optimal static adapter.

**Corollary 2 (Static barycentric reconstruction)** *Under Assumptions 2 and 4, with the barycentric mixture estimate $\hat{\alpha}_i$ satisfying $\|\hat{\alpha}_i - \alpha_i\|_1 \le \eta_C$ (Proposition 5), the reconstructed static adapter satisfies*

$$\mathcal{L}_i\left(\sum_m \hat{\alpha}_{im}\theta^{\mathrm{ext}}_m\right) - \mathcal{L}_i(\theta^*_i) \;\le\; \sigma_{\max}(\Phi)\Delta_i^2, \tag{63}$$

*where*

$$\Delta_i := \eta_C\,\mathrm{diam}_{\mathrm{F}}(\Theta) + \xi_{\max}.$$

*Hence $\Delta_i \to 0$ yields recovery of the optimal static adapter $\theta^*_i$.*

**Proof.** Define the reconstructed static adapter as

$$\theta^{\mathrm{rec}}_i := \sum_{m=1}^{M} \hat{\alpha}_{im}\theta^{\mathrm{ext}}_m, \tag{64}$$

and let

$$D_\Theta := \mathrm{diam}_{\mathrm{F}}(\Theta), \qquad \xi_{\max} := \max_{m\in[M]} \|\theta^{\mathrm{ext}}_m - \theta^*_m\|_{\mathrm{F}}. \tag{65}$$

By Proposition 5,

$$\|\hat{\alpha}_i - \alpha_i\|_1 \le \eta_C. \tag{66}$$

Under Assumption 2, the quadratic form in Eq. (21) and the barycentric identity $\theta^*_i = \sum_m \alpha_{im}\theta^*_m$ imply

$$\mathcal{L}_i(\theta) - \mathcal{L}_i(\theta^*_i) = \|\theta - \theta^*_i\|^2_\Phi \tag{67}$$

for every $\theta$. Therefore,

$$\mathcal{L}_i(\theta^{\mathrm{rec}}_i) - \mathcal{L}_i(\theta^*_i) = \|\theta^{\mathrm{rec}}_i - \theta^*_i\|^2_\Phi \le \sigma_{\max}(\Phi)\|\theta^{\mathrm{rec}}_i - \theta^*_i\|^2_{\mathrm{F}}. \tag{68}$$

It remains to bound the parameter-space reconstruction error. Using $\theta_i^* = \sum_m \alpha_{im}\theta_m^*$ and adding and subtracting $\sum_m \hat{\alpha}_{im}\theta_m^*$ gives

$$\theta_i^{\mathrm{rec}} - \theta_i^* = \sum_{m=1}^{M} (\hat{\alpha}_{im} - \alpha_{im})\theta_m^* + \sum_{m=1}^{M} \hat{\alpha}_{im}(\theta_m^{\mathrm{ext}} - \theta_m^*). \tag{69}$$

For the first term, since both $\hat{\alpha}_i$ and $\alpha_i$ belong to $\Delta^{M-1}$,

$$\sum_{m=1}^{M} (\hat{\alpha}_{im} - \alpha_{im}) = 0.$$

Fix any $m_0 \in [M]$. Then

$$\left\| \sum_{m=1}^{M} (\hat{\alpha}_{im} - \alpha_{im})\theta_m^* \right\|_{\mathrm{F}} = \left\| \sum_{m=1}^{M} (\hat{\alpha}_{im} - \alpha_{im})(\theta_m^* - \theta_{m_0}^*) \right\|_{\mathrm{F}} \tag{70}$$

$$\leq \sum_{m=1}^{M} |\hat{\alpha}_{im} - \alpha_{im}| \, \|\theta_m^* - \theta_{m_0}^*\|_{\mathrm{F}} \tag{71}$$

$$\leq D_\Theta \|\hat{\alpha}_i - \alpha_i\|_1 \tag{72}$$

$$\leq \eta_C D_\Theta. \tag{73}$$

For the second term, because $\hat{\alpha}_i \in \Delta^{M-1}$,

$$\left\| \sum_{m=1}^{M} \hat{\alpha}_{im}(\theta_m^{\mathrm{ext}} - \theta_m^*) \right\|_{\mathrm{F}} \leq \sum_{m=1}^{M} \hat{\alpha}_{im} \|\theta_m^{\mathrm{ext}} - \theta_m^*\|_{\mathrm{F}} \tag{74}$$

$$\leq \xi_{\max}. \tag{75}$$

Combining Eq. (69)–Eq. (75) yields

$$\|\theta_i^{\mathrm{rec}} - \theta_i^*\|_{\mathrm{F}} \leq \eta_C D_\Theta + \xi_{\max}. \tag{76}$$

Define

$$\Delta_i := \eta_C D_\Theta + \xi_{\max}. \tag{77}$$

Substituting this bound into Eq. (68) gives

$$\mathcal{L}_i\left( \sum_{m=1}^{M} \hat{\alpha}_{im}\theta_m^{\mathrm{ext}} \right) - \mathcal{L}_i(\theta_i^*) \leq \sigma_{\max}(\Phi)\Delta_i^2. \tag{78}$$

Hence, whenever $\eta_C \to 0$ and $\xi_{\max} \to 0$, we have $\Delta_i \to 0$, and the reconstructed static adapter converges to the optimal static adapter in both parameter error and population excess risk. ■

### D.5 Routing and Task-Aligned Specialization

We first formalize task-aligned specialization and then characterize the routing conditions required to approach it.

**Definition 1 (Task-aligned specialization)** *Fix a mixture of $M$ task distributions $\{\mathcal{P}_m\}_{m=1}^M$ whose true experts $\{\theta_m^*\}_{m=1}^M$ form a non-degenerate simplex ($h > 0$). A recovered system—a candidate expert set $\mathcal{V} = \{V_1, \ldots, V_M\}$ together with per-client routers $\{p_i(\cdot)\}_{i=1}^N$, produced by* any *procedure—is* $(\eta, \beta)$-specialized *if there exists a permutation $\pi \in S_M$ such that:*

(i) Expert recovery (parameter space): $\max_{m \in [M]} \|V_{\pi(m)} - \theta_m^*\|_{\mathrm{F}} \leq \eta$.

(ii) Routing concentration (function space): *for every client $i$ and every task $m$ in its support,* $\mathbb{E}_{x \sim \mathcal{P}_m}\big[p_{i,\pi(m)}(x)\big] \geq 1 - \beta$.

The permutation $\pi$ accounts for the standard identifiability up to global relabeling that arises in mixture models and separable non-negative matrix factorization (Arora et al., 2012). Accordingly, Definition 1 is invariant to a global permutation of the recovered experts. FedSEE additionally seeks task-indexed alignment, corresponding to $\pi = \mathrm{id}$. The task-identity result shows that label-free observations alone cannot determine this correspondence, whereas task-labeled recovery resolves it under the stated recovery conditions.

Condition (ii) is defined at the input level rather than at the client level. Thus, a mixed client may route inputs from different tasks to different experts rather than being assigned as a whole to a single component. In particular, Condition (ii) implies

$$\|\bar{p}_i - \Pi_\pi \alpha_i\|_1 \le 2\beta, \qquad \bar{p}_i := \mathbb{E}_{x\sim\mathcal{D}_i}[p_i(x)].$$

**Remark 6 (Ideal specialization)** *The task-indexed ideal corresponds to $(\eta, \beta) = (0, 0)$ with $\pi = \mathrm{id}$. Under Definition 1, each task-$m$ input is then routed to $\theta^*_m$ and the predictor attains the oracle risk*

$$\Lambda_i := \sum_{m=1}^{M} \alpha_{im}\sigma_m^2 l.$$

*By the mixing-gap characterization established above, the best static adapter incurs an additional risk $G_i$, which is strictly positive for every genuinely mixed client and vanishes for a pure client.*

The results below separate two requirements for approaching this ideal. Mean expert usage must be calibrated to the client's mixture, while the routing policy must remain sufficiently input dependent to specialize across tasks.

**Proof of Theorem 5.** We separate the argument into the static contribution determined by $\bar{p}_i$ and the additional contribution induced by input-dependent routing.

**Step 1: Risk of the constant mean router.** For any $q \in \Delta^{M-1}$, let

$$\theta(q) := \sum_{m=1}^{M} q_m \theta^\star_m.$$

Under Assumption 2, the quadratic client risk satisfies the variance identity

$$\mathcal{L}_i(\theta) - \mathcal{L}_i(\theta^\star_i) = \|\theta - \theta^\star_i\|^2_\Phi,$$

where

$$\theta^\star_i = \sum_{m=1}^{M} \alpha_{im}\theta^\star_m$$

by Theorem 1. Hence

$$\mathcal{L}_i(\theta(q)) - \mathcal{L}_i(\theta^\star_i) = \left\| \sum_{m=1}^{M} (q_m - \alpha_{im})\theta^\star_m \right\|^2_\Phi. \tag{79}$$

Set $c := q - \alpha_i$. Since both $q$ and $\alpha_i$ lie in the simplex,

$$\mathbf{1}^\top c = 0.$$

By norm equivalence and Lemma 1,

$$\left\| \sum_m c_m \theta^\star_m \right\|^2_\Phi \ge \sigma_{\min}(\Phi) \left\| \sum_m c_m \theta^\star_m \right\|^2_F \ge \sigma_{\min}(\Phi) h^2 \|c\|^2_\infty.$$

Therefore

$$\mathcal{L}_i(\theta(q)) - \mathcal{L}_i(\theta^\star_i) \ge \sigma_{\min}(\Phi) h^2 \|q - \alpha_i\|^2_\infty. \tag{80}$$

For the upper bound, let

$$s := \frac{1}{2}\|c\|_1.$$

If $s = 0$ the claim is immediate. Otherwise define

$$a_m := \frac{(c_m)_+}{s}, \qquad b_m := \frac{(-c_m)_+}{s}.$$

Because $\mathbf{1}^\top c = 0$, both $a$ and $b$ belong to $\Delta^{M-1}$, and

$$\sum_m c_m \theta_m^\star = s\left(\sum_m a_m \theta_m^\star - \sum_m b_m \theta_m^\star\right).$$

Both terms in parentheses lie in $\mathrm{conv}\{\theta_m^\star\}_{m=1}^M$. The diameter of this convex hull is $D_\Theta$, so

$$\left\|\sum_m c_m \theta_m^\star\right\|_F \le s D_\Theta = \frac{D_\Theta}{2}\|c\|_1.$$

Consequently,

$$\mathcal{L}_i(\theta(q)) - \mathcal{L}_i(\theta_i^\star) \le \frac{\sigma_{\max}(\Phi) D_\Theta^2}{4}\|q - \alpha_i\|_1^2. \tag{81}$$

**Step 2: Reduction of an arbitrary router to its mean.** Now consider an arbitrary routing policy $p_i(\cdot)$ and set

$$q := \bar{p}_i.$$

For brevity write

$$\theta_p(x) := \theta_{p_i}(x), \qquad \bar{\theta}_p := \theta(\bar{p}_i).$$

Let $T \in [M]$ denote the latent task of a sample from client $i$. Conditioned on $T = m$, Assumption 2 gives

$$y = f_0(x) + \theta_m^\star \phi(x) + \zeta_m, \qquad \mathbb{E}[\zeta_m \mid x] = 0.$$

Since $p_i(x)$ is a function of the input and does not depend on $\zeta_m$, the noise cross-term vanishes conditionally on $x$. Therefore

$$\mathcal{R}_i - \mathcal{L}_i(\bar{\theta}_p) = \mathbb{E}\Big[\|(\theta_T^\star - \theta_p(x))\phi(x)\|_2^2 - \|(\theta_T^\star - \bar{\theta}_p)\phi(x)\|_2^2\Big]. \tag{82}$$

For every $x$, both $\theta_p(x)$ and $\bar{\theta}_p$ lie in the convex hull of the experts. Hence

$$\|\theta_T^\star - \theta_p(x)\|_F \le D_\Theta, \qquad \|\theta_T^\star - \bar{\theta}_p\|_F \le D_\Theta.$$

Using $\|\phi(x)\|_2 \le Q$, both corresponding signal residuals have Euclidean norm at most $Q D_\Theta$. Thus

$$\begin{aligned}\left|\mathcal{R}_i - \mathcal{L}_i(\bar{\theta}_p)\right| &\le 2Q D_\Theta\, \mathbb{E}\left[\|(\theta_p(x) - \bar{\theta}_p)\phi(x)\|_2\right] \\ &\le 2Q^2 D_\Theta\, \mathbb{E}\left[\|\theta_p(x) - \bar{\theta}_p\|_F\right].\end{aligned} \tag{83}$$

It remains to relate parameter variation to routing variation. For any two probability vectors $a, b \in \Delta^{M-1}$, the same positive–negative decomposition used above gives

$$\left\|\sum_m (a_m - b_m)\theta_m^\star\right\|_F \le \frac{D_\Theta}{2}\|a - b\|_1. \tag{84}$$

Applying Eq. (84) pointwise with $a = p_i(x)$ and $b = \bar{p}_i$ yields

$$\|\theta_p(x) - \bar{\theta}_p\|_F \le \frac{D_\Theta}{2}\|p_i(x) - \bar{p}_i\|_1.$$

Substituting into Eq. (83),

$$\left|\mathcal{R}_i - \mathcal{L}_i(\theta(\bar{p}_i))\right| \le Q^2 D_\Theta^2 V_i(p_i). \tag{85}$$

**Step 3: Combine the two components.** Using

$$\mathcal{R}_i - \mathcal{L}_i(\theta_i^\star) = \big[\mathcal{L}_i(\theta(\bar{p}_i)) - \mathcal{L}_i(\theta_i^\star)\big] + \big[\mathcal{R}_i - \mathcal{L}_i(\theta(\bar{p}_i))\big],$$

the lower bound follows by combining Eq. (80) with the negative side of Eq. (85), and the upper bound follows by combining Eq. (81) with the positive side of Eq. (85). ■

The preceding result characterizes the roles of mean calibration and routing variation relative to the optimal static adapter. We next give a complementary guarantee showing when recovered experts and task-aligned routing approach the routed oracle.

**Corollary 3 (Oracle risk from expert recovery and routing)** *Under Assumptions 2 and 4, suppose*

$$\max_m \|\theta_m^{\mathrm{ext}} - \theta_m^*\|_{\mathrm{F}} \le \xi_{\max},$$

*and, for every task $m$ in client $i$'s support, the router satisfies*

$$\mathbb{E}_{x\sim\mathcal{P}_m}[p_{im}(x)] \ge 1-\beta.$$

*Let $D_\Theta := \mathrm{diam}_{\mathrm{F}}(\Theta)$. The population risk $\mathcal{R}_i$ of the routed predictor Eq. (5) using the recovered experts satisfies*

$$\mathcal{R}_i - \Lambda_i \;\le\; Q^2\big[\xi_{\max}^2 + \big(2\xi_{\max}D_\Theta + D_\Theta^2\big)\beta\big]. \tag{86}$$

*Consequently, $\xi_{\max}\to 0$ and $\beta\to 0$ imply $\mathcal{R}_i\to\Lambda_i$.*

**Proof.** For an input from task $m$, define $r_m(x) := 1 - p_{im}(x)$. By Assumption 2 and $\mathbb{E}[\zeta_m \mid x] = 0$, subtracting the oracle noise risk gives

$$\mathcal{R}_i - \Lambda_i = \sum_m \alpha_{im}\,\mathbb{E}_{x\sim\mathcal{P}_m}\left[\left\|\left(\sum_k p_{ik}(x)\theta_k^{\mathrm{ext}} - \theta_m^*\right)\phi(x)\right\|_2^2\right]. \tag{87}$$

Using $\|\phi(x)\|_2 \le Q$,

$$\mathcal{R}_i - \Lambda_i \le Q^2\sum_m \alpha_{im}\,\mathbb{E}_{x\sim\mathcal{P}_m}\left\|\sum_k p_{ik}(x)\theta_k^{\mathrm{ext}} - \theta_m^*\right\|_{\mathrm{F}}^2. \tag{88}$$

For task $m$,

$$\left\|\sum_k p_{ik}(x)\theta_k^{\mathrm{ext}} - \theta_m^*\right\|_{\mathrm{F}} \le \left\|\sum_k p_{ik}(x)(\theta_k^{\mathrm{ext}} - \theta_k^*)\right\|_{\mathrm{F}} + \left\|\sum_{k\ne m} p_{ik}(x)(\theta_k^* - \theta_m^*)\right\|_{\mathrm{F}} \tag{89}$$

$$\le \xi_{\max} + D_\Theta r_m(x). \tag{90}$$

Since $0\le r_m(x)\le 1$, we have $r_m(x)^2 \le r_m(x)$, while the routing-concentration assumption gives $\mathbb{E}_{\mathcal{P}_m}[r_m(x)] \le \beta$. Therefore

$$\mathbb{E}_{\mathcal{P}_m}\left[\big(\xi_{\max} + D_\Theta r_m(x)\big)^2\right] \le \xi_{\max}^2 + \big(2\xi_{\max}D_\Theta + D_\Theta^2\big)\beta. \tag{91}$$

Averaging over $m$ with weights $\alpha_{im}$ and using $\sum_m \alpha_{im} = 1$ yields Eq. (86). ■

We next examine the consequence of imposing a fixed mean-usage target. Proposition 1 in the main text shows that uniform mean usage conflicts with task-aligned specialization whenever the client mixture is non-uniform.

**Proof of Proposition 1.** Let

$$q_{im} := \mathbb{E}_{x\sim\mathcal{P}_m}[p_i(x)] \in \Delta^{M-1}.$$

If client $i$ satisfies Definition 1(ii) with permutation $\pi$, then

$$q_{im,\pi(m)} \ge 1-\beta.$$

Moreover,

$$\bar{p}_i = \sum_m \alpha_{im} q_{im}, \qquad \Pi_\pi \alpha_i = \sum_m \alpha_{im} e_{\pi(m)}.$$

Hence

$$\|\bar{p}_i - \Pi_\pi \alpha_i\|_1 \le \sum_m \alpha_{im} \|q_{im} - e_{\pi(m)}\|_1 \tag{92}$$

$$= 2 \sum_m \alpha_{im} \big(1 - q_{im,\pi(m)}\big) \le 2\beta. \tag{93}$$

If additionally $\|\bar{p}_i - u\|_1 \le \delta_{\mathrm{bal}}$, then the reverse triangle inequality gives

$$2\beta \ge \|\bar{p}_i - \Pi_\pi \alpha_i\|_1 \ge \|\Pi_\pi \alpha_i - u\|_1 - \|\bar{p}_i - u\|_1.$$

Since $u$ is permutation invariant,

$$\|\Pi_\pi \alpha_i - u\|_1 = \|\alpha_i - u\|_1.$$

Therefore

$$\beta \ge \frac{1}{2} \big(\|\alpha_i - u\|_1 - \delta_{\mathrm{bal}}\big)_+.$$

For exact uniform mean usage, $\delta_{\mathrm{bal}} = 0$, and hence

$$\beta \ge \frac{1}{2} \|\alpha_i - u\|_1.$$

Thus $\beta > 0$ for every non-uniform client. ■

**Corollary 4 (Cost of anchoring to a fixed target)** *Let $t \in \Delta^{M-1}$ and suppose $\|\bar{p}_i - t\|_\infty \le \chi$. Then*

$$\mathcal{R}_i - \mathcal{L}_i(\theta_i^*) \ge \sigma_{\min}(\Phi) h^2 \big(\|\alpha_i - t\|_\infty - \chi\big)_+^2 - Q^2 D_\Theta^2 V_i(p_i). \tag{94}$$

**Proof.** By the reverse triangle inequality,

$$\|\bar{p}_i - \alpha_i\|_\infty \ge \big(\|\alpha_i - t\|_\infty - \|\bar{p}_i - t\|_\infty\big)_+ \ge \big(\|\alpha_i - t\|_\infty - \chi\big)_+.$$

Substituting this inequality into the lower bound of Theorem 5 yields Eq. (94). ■

**Corollary 5 (Personalization cost of balanced routing)** *Taking $t = u = \frac{1}{M}\mathbf{1}$, any policy satisfying $\|\bar{p}_i - u\|_\infty \le \chi$ obeys*

$$\mathcal{R}_i - \mathcal{L}_i(\theta_i^*) \ge \sigma_{\min}(\Phi) h^2 \big(\|\alpha_i - u\|_\infty - \chi\big)_+^2 - Q^2 D_\Theta^2 V_i(p_i). \tag{95}$$

*Hence the routed predictor remains separated from the optimal static adapter whenever the mean-mismatch term exceeds the routing-variation term.*

**Proof.** Apply Corollary 4 with $t = u = \frac{1}{M}\mathbf{1}$. ■

These results show that a fixed uniform target can conflict with both task-aligned specialization and personalized performance. We next study a separate limitation concerning how the router representation itself can distinguish different client mixtures. We first record a standard Lipschitz property of the softmax map that will be used in the routing-sensitivity analysis.

**Lemma 2 (Softmax is $\frac{1}{2}$-Lipschitz in $\ell_2$)** *For all $a, b$, $\|\mathrm{softmax}(a) - \mathrm{softmax}(b)\|_2 \le \frac{1}{2}\|a - b\|_2$.*

**Proof.** The Jacobian at logits $z$ with $p = \mathrm{softmax}(z)$ is $J(z) = \mathrm{diag}(p) - pp^\top$. For a unit $u$, $u^\top J u = \mathrm{Var}_{e\sim p}(u_e) \le \frac{1}{4}(\max_e u_e - \min_e u_e)^2 \le \frac{1}{2}\|u\|_2^2$ by Popoviciu's inequality, so $\|J(z)\|_2 \le \frac{1}{2}$; integrate along the segment from $a$ to $b$. ■

**Sensitivity of routing statistics to task geometry.** To characterize when mean routing statistics can distinguish client mixtures, consider the stylized router-input model

$$x = \mu + \delta_m + \xi \qquad \text{under task } m, \tag{96}$$

where $\mu$ is shared across tasks, $\delta_m$ is task specific, and $\xi$ is centered with a task-independent distribution. Let $\Pi_r$ project onto the top-$r$ principal subspace of the pooled second moment, and define

$$\tau := \max_{m \neq m'} \|\Pi_r(\delta_m - \delta_{m'})\|_2, \qquad \rho_\perp := \max_{m \neq m'} \|(I - \Pi_r)(\delta_m - \delta_{m'})\|_2.$$

**Proposition 6 (Routing-geometry sensitivity)** *Let $g(x) = W_r x + b_r$ be any linear router, $p(x) = \mathrm{softmax}(g(x))$, and $\bar{p}_m := \mathbb{E}[p(x) \mid \text{task } m]$. Then for all $m \neq m'$,*

$$\|\bar{p}_m - \bar{p}_{m'}\|_2 \leq \tfrac{1}{2}\big(\|W_r\Pi_r\|_2\,\tau + \|W_r(I - \Pi_r)\|_2\,\rho_\perp\big) =: D_{\mathrm{rt}}. \tag{97}$$

*Moreover a client with mixture $\alpha$ has $\bar{p}(\alpha) = \sum_m \alpha_m \bar{p}_m$, and for any $\alpha, \alpha'$,*

$$\|\bar{p}(\alpha) - \bar{p}(\alpha')\|_2 \leq \|\alpha - \alpha'\|_1 \cdot D_{\mathrm{rt}}. \tag{98}$$

**Proof.** Since $\xi$ has the same distribution under every task, couple the task-$m$ and task-$m'$ inputs using the same realization of $\xi$:

$$x_m = \mu + \delta_m + \xi, \qquad x_{m'} = \mu + \delta_{m'} + \xi.$$

By Jensen's inequality and Lemma 2, (the bias $b_r$ cancels),

$$\|\bar{p}_m - \bar{p}_{m'}\|_2 \leq \mathbb{E}_\xi \|\mathrm{softmax}(g(x_m)) - \mathrm{softmax}(g(x_{m'}))\|_2 \leq \tfrac{1}{2}\|W_r(\delta_m - \delta_{m'})\|_2.$$

**Subspace split.** Let $\Delta := \delta_m - \delta_{m'}$. Decomposing $\Delta$ into its components inside and outside the principal subspace gives

$$\|W_r\Delta\|_2 \leq \|W_r\Pi_r\|_2\|\Pi_r\Delta\|_2 + \|W_r(I-\Pi_r)\|_2\|(I-\Pi_r)\Delta\|_2 \leq \|W_r\Pi_r\|_2\tau + \|W_r(I-\Pi_r)\|_2\rho_\perp,$$

proving Eq. (97). For the client-level statement, linearity of expectation gives $\bar{p}(\alpha) = \sum_m \alpha_m \bar{p}_m$. Since $\sum_m (\alpha_m - \alpha'_m) = 0$,

$$\|\bar{p}(\alpha) - \bar{p}(\alpha')\|_2 = \|\sum_m (\alpha_m - \alpha'_m)(\bar{p}_m - \bar{p}_1)\|_2 \leq \|\alpha - \alpha'\|_1 D_{\mathrm{rt}},$$

which proves the result. ■

The second inequality is a forward sensitivity bound. It shows that a change in the client mixture of size $\|\alpha - \alpha'\|_1$ changes the mean routing statistic by at most $D_{\mathrm{rt}}\|\alpha - \alpha'\|_1$. Consequently, when $D_{\mathrm{rt}}$ is small, substantially different mixtures may induce similar mean-routing statistics. This motivates obtaining mixture information from supervised expert recovery rather than inferring it solely from router statistics.

**Mixture-aware calibration in the small-logit regime.** We now relate the mean router response to the mixture-aware calibration target used by FedSEE. Fix a client $i$ and a router layer; let the router input have client mean $\bar{h}_i$, logits $z(h) = W_r h + b_r$, $p(h) = \mathrm{softmax}(z(h))$, $\bar{p}_i := \mathbb{E}[p(h)]$, and $u := \frac{1}{M}\mathbf{1}$. With $\Pi_M := I_M - \frac{1}{M}\mathbf{1}\mathbf{1}^\top$, the router operates in the *$\nu$-small-logit regime* if $\|\Pi_M z(h)\|_2 \leq \nu$ a.s.

**Corollary 6 (Routing variation in the small-logit regime)** *If $p(h) = \mathrm{softmax}(z(h))$ satisfies $\|\Pi_M z(h)\|_2 \leq \nu$ almost surely, then*

$$V_i(p) := \mathbb{E}\|p(h) - \bar{p}_i\|_1 \leq \frac{\sqrt{M}}{2}\,\nu.$$

*Consequently, Theorem 5 has routing-variation slack at most*

$$\frac{\sqrt{M}}{2}\,Q^2 D_\Theta^2 \nu$$

*in this regime.*

**Proof.** Softmax is invariant to adding the same constant to all logits, so

$$p(h) = \mathrm{softmax}(\Pi_M z(h)).$$

Since

$$u = \mathrm{softmax}(0),$$

Lemma 2 gives, pointwise,

$$\|p(h) - u\|_2 \le \frac{1}{2}\|\Pi_M z(h)\|_2 \le \frac{\nu}{2}.$$

Because $\bar{p}_i = \mathbb{E}[p(h)]$, the Euclidean variance identity yields

$$\mathbb{E}\|p(h) - \bar{p}_i\|_2^2 = \mathbb{E}\|p(h) - u\|_2^2 - \|\bar{p}_i - u\|_2^2 \le \frac{\nu^2}{4}.$$

Therefore, by Jensen's inequality and $\|v\|_1 \le \sqrt{M}\|v\|_2$,

$$V_i(p) = \mathbb{E}\|p(h) - \bar{p}_i\|_1 \le \sqrt{M}\,\mathbb{E}\|p(h) - \bar{p}_i\|_2 \le \frac{\sqrt{M}}{2}\nu.$$

Substituting this bound into Theorem 5 yields the stated routing-variation term. ■

**Lemma 3 (Second-order softmax expansion)** *For all* $z$, $\|\mathrm{softmax}(z) - u - \frac{1}{M}\Pi_M z\|_2 \le \frac{3}{4}\|\Pi_M z\|_2^2$. *Hence in the* $\nu$*-small-logit regime,*

$$\bar{p}_i = u + \tfrac{1}{M}\Pi_M(W_r\bar{h}_i + b_r) + r_i, \qquad \|r_i\|_2 \le \tfrac{3}{4}\nu^2. \tag{99}$$

**Proof.** Let $\sigma(z) := \mathrm{softmax}(z)$, $J(z) = \mathrm{diag}(\sigma) - \sigma\sigma^\top$. At the origin $\sigma(0) = u$, $J(0) = \frac{1}{M}\Pi_M$. Using $\|J(z) - J(z')\|_2 \le 3\|p - p'\|_2 \le \frac{3}{2}\|z - z'\|_2$ (Lemma 2), $J$ is $\frac{3}{2}$-Lipschitz. Applying Taylor's theorem to $\tilde{z} := \Pi_M z$ and using $\sigma(z) = \sigma(\tilde{z})$ gives $\|\sigma(z) - u - J(0)\tilde{z}\|_2 \le \frac{3}{4}\|\tilde{z}\|_2^2$. Taking expectations over $h$ and using $\mathbb{E}[z(h)] = W_r\bar{h}_i + b_r$ gives Eq. (99). ■

**Proposition 7 (KL-anchor calibration)** *In the* $\nu$*-small-logit regime:* (a) *if* $\mathrm{KL}(\tilde{c}_i\|\bar{p}_i) \le \rho$ *then*

$$\|\Pi_M(W_r\bar{h}_i + b_r) - M(\tilde{c}_i - u)\|_2 \le M(\sqrt{2\rho} + \tfrac{3}{4}\nu^2);$$

(b) *any router satisfying*

$$\Pi_M(W_r\bar{h}_i + b_r) = M(\tilde{c}_i - u)$$

*has*

$$\|\bar{p}_i - \tilde{c}_i\|_2 \le \tfrac{3}{4}\nu^2.$$

**Proof.** For part (a), Pinsker's inequality gives

$$\|\tilde{c}_i - \bar{p}_i\|_2 \le \|\tilde{c}_i - \bar{p}_i\|_1 \le \sqrt{2\rho}.$$

Rearranging Eq. (99),

$$\left\| \frac{1}{M}\Pi_M(W_r\bar{h}_i + b_r) - (\tilde{c}_i - u) \right\|_2 = \|\bar{p}_i - \tilde{c}_i - r_i\|_2.$$

Using $\|r_i\|_2 \le \frac{3}{4}\nu^2$ therefore yields

$$\left\| \frac{1}{M}\Pi_M(W_r\bar{h}_i + b_r) - (\tilde{c}_i - u) \right\|_2 \le \sqrt{2\rho} + \frac{3}{4}\nu^2.$$

Multiplying by $M$ proves part (a).

For part (b), substituting

$$\Pi_M(W_r\bar{h}_i + b_r) = M(\tilde{c}_i - u)$$

into Eq. (99) gives

$$\bar{p}_i - \tilde{c}_i = r_i.$$

Hence

$$\|\bar{p}_i - \tilde{c}_i\|_2 \le \frac{3}{4}\nu^2.$$

■

**Remark 7 (Uniform versus mixture-aware calibration)** *Under the simplified load-balancing objective $\|\bar{p}_i - u\|_2^2$ of Wang et al. (2026), the local expansion Eq. (99) encourages the router's mean response toward the uniform target $u$. In contrast, FedSEE's KL term calibrates the same first moment toward the task-indexed target $\tilde{c}_i$. The uniform-usage result above shows that calibration toward $u$ is incompatible with exact specialization for a non-uniform client, whereas the result below quantifies the mean-routing error obtained from calibration toward $\tilde{c}_i$.*

**Corollary 7 (KL control of mean-routing mismatch)** *Let $\hat{\alpha}_i$ satisfy $\|\hat{\alpha}_i - \alpha_i\|_1 \le \eta_C$ and define $\delta_i^{\mathrm{proxy}} := \|\tilde{c}_i - \hat{\alpha}_i\|_1$. If $\mathrm{KL}(\tilde{c}_i \| \bar{p}_i) \le \rho$, then*

$$\|\bar{p}_i - \alpha_i\|_1 \le \sqrt{2\rho} + \eta_C + \delta_i^{\mathrm{proxy}}. \tag{100}$$

*Under the assumptions of Theorem 5 and when routing over the true experts, this further implies*

$$\mathcal{R}_i - \mathcal{L}_i(\theta_i^*) \le \frac{\sigma_{\max}(\Phi) D_\Theta^2}{4} \left( \sqrt{2\rho} + \eta_C + \delta_i^{\mathrm{proxy}} \right)^2 + Q^2 D_\Theta^2 V_i(p_i). \tag{101}$$

**Proof.** Pinsker's inequality gives

$$\|\tilde{c}_i - \bar{p}_i\|_1 \le \sqrt{2\,\mathrm{KL}(\tilde{c}_i \| \bar{p}_i)} \le \sqrt{2\rho}.$$

By the definition of $\delta_i^{\mathrm{proxy}}$ and Proposition 5,

$$\|\tilde{c}_i - \alpha_i\|_1 \le \|\tilde{c}_i - \hat{\alpha}_i\|_1 + \|\hat{\alpha}_i - \alpha_i\|_1 \le \delta_i^{\mathrm{proxy}} + \eta_C.$$

Hence

$$\|\bar{p}_i - \alpha_i\|_1 \le \|\bar{p}_i - \tilde{c}_i\|_1 + \|\tilde{c}_i - \alpha_i\|_1 \le \sqrt{2\rho} + \eta_C + \delta_i^{\mathrm{proxy}}.$$

Substituting this bound into the upper inequality of Theorem 5 gives Eq. (101). ■

The bound separates three sources of mean-routing error: the KL calibration error $\rho$, the certified barycentric mixture-estimation error $\eta_C$, and the proxy error $\delta_i^{\mathrm{proxy}}$. The first two vanish under the corresponding optimization and recovery limits, while the third quantifies the additional approximation introduced by using the contribution-based proxy $\tilde{c}_i$ in place of the certified barycentric estimate.

**Scope of the routing guarantee.** Theorem 5 separates two effects in routed risk. The mean-mismatch term quantifies calibration relative to the client mixture, while $V_i(p_i)$ controls how far input-dependent routing can move the risk from the corresponding static adapter. The latter term has no prescribed sign in the exact decomposition. Input-dependent routing may therefore improve upon the static floor and is precisely the mechanism needed to close the mixing gap.

FedSEE's KL anchor controls the mean-calibration component through Corollary 7. Reaching the routed oracle additionally requires accurate expert recovery and task-aligned input-level routing. Corollary 3 shows that as the routing concentration error $\beta$ and expert-recovery error $\xi_{\max}$ vanish, the routed risk approaches the oracle risk. Thus the KL anchor calibrates mean usage, while the task loss remains responsible for input-level specialization.

**Shared-router aggregation.** Condition (ii) of Definition 1 is task-conditioned rather than client-specific. Consequently, a single router can satisfy the specialization condition for all clients simultaneously. In particular, suppose $p^\star$ satisfies

$$\mathbb{E}_{x \sim \mathcal{P}_m}[p_m^\star(x)] \ge 1 - \beta$$

for every task $m$. Then, for client $i$,

$$\bar{p}_i = \sum_m \alpha_{im} \mathbb{E}_{x \sim \mathcal{P}_m}[p^\star(x)],$$

and therefore

$$\|\bar{p}_i - \alpha_i\|_1 \le 2\beta.$$

Thus, in the ideal task-aligned regime of Remark 6, separate client-specific routers are not required to satisfy the routing-concentration condition.

A shared router can nevertheless be difficult to calibrate across clients when task information is weak in its input representation. Proposition 6 bounds the separation between task-conditioned routing statistics, so substantially different client mixtures may induce similar mean expert usage. In this regime, Theorem 5 identifies the resulting mean-calibration mismatch, while its variation term $V_i(p_i)$ captures the additional effect of input-dependent routing. The theorem therefore does not imply that router aggregation necessarily incurs a positive excess-risk penalty.

### D.6 Extensions Beyond the Exact Squared-Loss Setting

The preceding results exploit the exact quadratic geometry induced by the squared-loss model, while the experiments in Section 5 optimize softmax cross-entropy. We therefore extend the analysis to general losses. The client risk remains an exact mixture of task risks, while the client optimum becomes a perturbed barycenter controlled by the curvature-dispersion parameter $\varkappa$. We then propagate this perturbation to expert recovery, specialize the analysis to softmax cross-entropy over a frozen representation, and discuss the corresponding implications for mixture estimation, routing, and identifiability.

#### D.6.1 Setup and Assumptions

We retain the model class defined in Eq. (2), $f_\theta(x) = f_0(x) + \theta\,\phi(x)$, where $(f_0, \phi)$ is frozen and $\theta \in \mathbb{R}^{l\times r}$ remains the sole trainable parameter block. We replace the squared error with a general loss function $\ell(f_\theta(x), y)$. Let $\mathcal{L}_m(\theta) := \mathbb{E}_{\mathcal{P}_m}[\ell(f_\theta(x), y)]$ denote the population risk for task $m$, and let $\Theta := \{\theta_m^*\}_{m=1}^M$ represent the set of true experts for $M \geq 2$. Since $\mathcal{D}_m^s$ contains i.i.d. samples from $\mathcal{P}_m$ (Assumption 1), $\widehat{\mathcal{L}}_m^s$ is the empirical counterpart of $\mathcal{L}_m$.

For analytical convenience, we vectorize the parameters: let $\vartheta := \mathrm{vec}(\theta) \in \mathbb{R}^{lr}$, such that $\nabla^2\mathcal{L}_m(\theta) \in \mathbb{R}^{lr\times lr}$ and $\|\theta\|_{\mathrm{F}} = \|\vartheta\|_2$. We write $\mathcal{L}_m(\vartheta)$ and $\mathcal{L}_m(\theta)$ interchangeably. All Hessians in this section are defined with respect to this vectorization, and all matrix norms denote the spectral norm. When the loss is invariant along a subspace of the parameter block, as occurs for softmax cross-entropy under common shifts of all logits, the strong-convexity statements below are understood on the corresponding invariant complement $\mathcal{V}$. Appendix D.6.3 makes this restriction explicit for the cross-entropy objective.

**Assumption 6 (Regular per-task risk)** *Let $\bar{\mathcal{C}} \subseteq \mathbb{R}^{l\times r}$ be a closed and convex set with interior $\mathcal{C} := \operatorname{int}\bar{\mathcal{C}}$. Assume (i) for every task $m$ and every client $i$, the unique minimizers $\theta_m^* := \arg\min_{\theta\in\bar{\mathcal{C}}}\mathcal{L}_m(\theta)$ and $\theta_i^* := \arg\min_{\theta\in\bar{\mathcal{C}}}\mathcal{L}_i(\theta)$ exist and lie strictly within $\mathcal{C}$, ensuring that $\nabla\mathcal{L}_m(\theta_m^*) = 0$ and $\nabla\mathcal{L}_i(\theta_i^*) = 0$; and (ii) on $\mathcal{C}$, each $\mathcal{L}_m$ is twice continuously differentiable and satisfies, for all $\theta, \theta' \in \mathcal{C}$,*

$$\mu I \preceq \nabla^2\mathcal{L}_m(\theta) \preceq LI, \qquad \left\|\nabla^2\mathcal{L}_m(\theta) - \nabla^2\mathcal{L}_m(\theta')\right\| \leq L_H\,\|\theta - \theta'\|_{\mathrm{F}}, \tag{102}$$

*with $0 < \mu \leq L < \infty$ and $L_H \geq 0$.*

**Definition 2 (Curvature dispersion)** *The curvature dispersion of the task family and its relative counterpart are defined as:*

$$\Delta_H := \max_{m\neq m'}\sup_{\theta\in\mathcal{C}}\left\|\nabla^2\mathcal{L}_m(\theta) - \nabla^2\mathcal{L}_{m'}(\theta)\right\|, \qquad \varkappa := \frac{\Delta_H + \frac{1}{2}L_H\,\mathrm{diam}_{\mathrm{F}}(\Theta)}{\mu}. \tag{103}$$

Two observations contextualize the curvature dispersion $\varkappa$. First, $\Delta_H$ evaluates task divergence at a shared parameter; it characterizes the structural mismatch of the task landscapes rather than the spatial distance between their optima. Second, under the squared loss $\ell(\theta; x, y) = \|y - f_\theta(x)\|_2^2$, the Hessian evaluates to $\nabla^2\mathcal{L}_m(\theta) = 2\,\Phi_m \otimes I_l$ globally. Consequently, if tasks share an input covariance ($\Phi_m = \Phi$), this yields $\Delta_H = 0$ and $L_H = 0$, resulting in $\varkappa = 0$. Even under a heterogeneous squared loss ($\Phi_m \neq \Phi$), we have $\Delta_H = 2\max_{m\neq m'}\|\Phi_m - \Phi_{m'}\|_2$, $L_H = 0$, and $\mu = 2\min_m\sigma_{\min}(\Phi_m)$, such that $\varkappa = \rho_\Phi$. Thus, $\varkappa$ exactly reproduces the covariance-dispersion bias $\rho_\Phi$ from Remark 1 in Appendix D.1, directly confirming the internal consistency of the two analyses.

#### D.6.2 General-Loss Geometry and Expert Recovery

**Proposition 8 (Mixture risk decomposition)** *For any measurable loss function $\ell$ and any $\theta$ for which the expectations exist,*

$$\mathcal{L}_i(\theta) = \sum_{m=1}^M \alpha_{im}\,\mathcal{L}_m(\theta). \tag{104}$$

**Proof.** This follows directly from Eq. (1) and the linearity of expectation: the client distribution $\mathcal{D}_i$ is identically the mixture $\sum_m \alpha_{im}\mathcal{P}_m$, and the loss $\ell$ operates independently of the client index $i$. ■

The risk decomposition therefore survives exactly beyond squared loss. What changes is the geometry of its minimizer. In general, $\theta_i^*$ need not coincide with the barycenter of the task optima, but the deviation can be controlled by the curvature dispersion introduced above.

**Theorem 6 (Perturbed barycenter under a general loss)** *Let Assumption 6 hold. Define the exact barycenter $\theta_i^{\mathrm{bary}} := \sum_m \alpha_{im}\theta_m^*$ and the local radius $D_i := \max_m \|\theta_m^* - \theta_i^{\mathrm{bary}}\|_{\mathrm{F}} \le \mathrm{diam}_{\mathrm{F}}(\Theta)$. The client population optimum $\theta_i^*$ satisfies:*

$$\big\|\theta_i^* - \theta_i^{\mathrm{bary}}\big\|_{\mathrm{F}} \;\le\; \varkappa\, D_i \;\le\; \varkappa\, \mathrm{diam}_{\mathrm{F}}(\Theta). \tag{105}$$

*This bound is non-vacuous in the regime where $\varkappa < 1$. Consequently, $\mathrm{dist}_{\mathrm{F}}\big(\theta_i^*, \mathrm{conv}\{\theta_m^*\}\big) \le \varkappa\, \mathrm{diam}_{\mathrm{F}}(\Theta)$, and when $\varkappa = 0$, the identity $\theta_i^* = \sum_m \alpha_{im}\theta_m^*$ of Theorem 1 is recovered exactly.*

**Proof.** By Proposition 8 and Assumption 6, the aggregate risk $\mathcal{L}_i = \sum_m \alpha_{im}\mathcal{L}_m$ is $\mu$-strongly convex on $\mathcal{C}$.

*Step 1 (Integral representation).* Because interiority ensures $\nabla\mathcal{L}_m(\theta_m^*) = 0$, we apply the fundamental theorem of calculus to $\nabla\mathcal{L}_m$ along the segment $[\theta_m^*, \theta_i^*] \subset \mathcal{C}$. Utilizing the vectorized notation $\vartheta = \mathrm{vec}(\theta)$, we obtain:

$$\nabla\mathcal{L}_m(\vartheta_i^*) \;=\; G_m\,(\vartheta_i^* - \vartheta_m^*), \qquad G_m \;:=\; \int_0^1 \nabla^2\mathcal{L}_m\big(\vartheta_m^* + t(\vartheta_i^* - \vartheta_m^*)\big)\,\mathrm{d}t. \tag{106}$$

Because each $G_m$ is an integrated average of Hessians evaluated within $\mathcal{C}$, it satisfies $\mu I \preceq G_m \preceq LI$. This representation remains exact.

*Step 2 (First-order condition).* The optimality of $\theta_i^*$ for $\mathcal{L}_i$ dictates $\sum_m \alpha_{im}\nabla\mathcal{L}_m(\vartheta_i^*) = 0$. Substituting Eq. (106) yields $\sum_m \alpha_{im}G_m(\vartheta_i^* - \vartheta_m^*) = 0$. Defining $\bar{G} := \sum_m \alpha_{im}G_m \succeq \mu I$ and rearranging terms:

$$\bar{G}\,\vartheta_i^* \;=\; \sum_m \alpha_{im}G_m\vartheta_m^*. \tag{107}$$

*Step 3 (Barycentric alignment).* Noting that $\sum_m \alpha_{im}(G_m - \bar{G}) = 0$, we subtract $\bar{G}\,\vartheta_i^{\mathrm{bary}}$ from both sides of Eq. (107) to isolate the deviation:

$$\bar{G}\big(\vartheta_i^* - \vartheta_i^{\mathrm{bary}}\big) \;=\; \sum_m \alpha_{im}\,(G_m - \bar{G})\,\vartheta_m^* \;=\; \sum_m \alpha_{im}\,(G_m - \bar{G})\,\big(\vartheta_m^* - \vartheta_i^{\mathrm{bary}}\big). \tag{108}$$

*Step 4 (Bounding curvature variation).* Fix indices $m, k$. The integrands defining $G_m$ and $G_k$ evaluate to a parameter-space displacement of $(1-t)(\vartheta_m^* - \vartheta_k^*)$. Decoupling this into a task difference at a shared point and the integrated displacement yields:

$$\|G_m - G_k\| \;\le\; \Delta_H \;+\; L_H \int_0^1 (1-t)\,\|\vartheta_m^* - \vartheta_k^*\|_2\,\mathrm{d}t \;\le\; \Delta_H + \tfrac{1}{2}L_H\,\mathrm{diam}_{\mathrm{F}}(\Theta). \tag{109}$$

Because $\bar{G}$ is a convex combination of the matrices $G_k$, convexity of the spectral norm ensures this bound applies uniformly to $\|G_m - \bar{G}\|$.

*Step 5 (Conclusion).* Taking the Euclidean norm $\|\cdot\|_2$ in Eq. (108) and utilizing $\|\bar{G}^{-1}\| \le 1/\mu$, we find:

$$\big\|\vartheta_i^* - \vartheta_i^{\mathrm{bary}}\big\|_2 \;\le\; \frac{1}{\mu}\Big(\Delta_H + \tfrac{1}{2}L_H\,\mathrm{diam}_{\mathrm{F}}(\Theta)\Big)\sum_m \alpha_{im}\big\|\vartheta_m^* - \vartheta_i^{\mathrm{bary}}\big\|_2 \;\le\; \varkappa\, D_i. \tag{110}$$

Translating the Euclidean norm back to the equivalent Frobenius norm over the matrices satisfies Eq. (105). ■

**Remark 8 (Structural boundaries of exactness)** *Equation (108) shows that exactness $\theta_i^* = \theta_i^{\mathrm{bary}}$ holds precisely when the directional condition $\sum_m \alpha_{im}\,(G_m - \bar{G})\,(\vartheta_m^* - \vartheta_i^{\mathrm{bary}}) = 0$ is met; for $M = 2$ this reduces to $(G_1 - G_2)(\vartheta_1^* - \vartheta_2^*) = 0$. This is a condition on the* averaged *operators $G_m$ along a single direction, and is strictly weaker than $\Delta_H = 0$: non-quadratic families with uniformly separated Hessians can still be exactly barycentric for every $\alpha \in \Delta^{M-1}$. The bound Eq. (105) is therefore sufficient rather than necessary—it is the tightest control available from curvature magnitudes alone, and $\varkappa$ should be read as an upper bound on the deviation, not as a measure of it.*

**Remark 9 (Identifiability under general losses)** *The task-identity indeterminacy of Theorem 3 follows from permutation symmetry and is therefore loss agnostic. In contrast, the vertex-set non-identifiability construction of Theorem 4 uses the exact quadratic barycentric geometry. Under general losses, Theorem 6 provides only an approximate simplex geometry controlled by $\varkappa$. Establishing a corresponding approximate vertex-set indistinguishability result would therefore require an additional perturbation argument and is not claimed here.*

The same perturbation also affects supervised expert extraction. For an $\varepsilon$-anchor client, the distance to its target task expert now has two sources: anchor impurity and curvature-induced deviation from the exact barycenter. This yields the following general-loss recovery guarantee.

**Theorem 7 (Expert recovery, general loss)** *Let Assumptions 6 and 3 hold, with exact client uploads $\{\theta_i^*\}$. The extraction program Eq. (8) returns:*

$$\left\|\theta_m^{\mathrm{ext}} - \theta_m^*\right\|_{\mathrm{F}} \;\le\; \sqrt{L/\mu}\;\cdot\;(\varepsilon + \varkappa)\;\cdot\;\mathrm{diam}_{\mathrm{F}}(\Theta). \tag{111}$$

**Proof.** Let $i_m$ designate the exact anchor client for task $m$. By the triangle inequality and Theorem 6,

$$\begin{aligned}\|\theta_{i_m}^* - \theta_m^*\|_{\mathrm{F}} \;&\le\; \|\theta_{i_m}^{\mathrm{bary}} - \theta_m^*\|_{\mathrm{F}} + \varkappa\,\mathrm{diam}_{\mathrm{F}}(\Theta)\\ &\le\; \Big\|\sum_{m'\neq m}\alpha_{i_m,m'}(\theta_{m'}^* - \theta_m^*)\Big\|_{\mathrm{F}} + \varkappa\,\mathrm{diam}_{\mathrm{F}}(\Theta) \;\le\; (\varepsilon+\varkappa)\,\mathrm{diam}_{\mathrm{F}}(\Theta).\end{aligned} \tag{112}$$

The simplex vertex allocating all probability mass to $\theta_{i_m}^*$ is feasible for the program Eq. (8), ensuring $\mathcal{L}_m(\theta_m^{\mathrm{ext}}) \le \mathcal{L}_m(\theta_{i_m}^*)$. Utilizing the $L$-smoothness of $\mathcal{L}_m$ at its minimizer $\theta_m^*$ and its $\mu$-strong convexity, we obtain:

$$\tfrac{\mu}{2}\|\theta_m^{\mathrm{ext}} - \theta_m^*\|_{\mathrm{F}}^2 \;\le\; \mathcal{L}_m(\theta_m^{\mathrm{ext}}) - \mathcal{L}_m(\theta_m^*) \;\le\; \mathcal{L}_m(\theta_{i_m}^*) - \mathcal{L}_m(\theta_m^*) \;\le\; \tfrac{L}{2}\|\theta_{i_m}^* - \theta_m^*\|_{\mathrm{F}}^2. \tag{113}$$

Substituting the anchor distance and taking the square root completes the proof. ■

**Remark 10 (Other downstream implications)** *Beyond expert recovery, the perturbation in Theorem 6 also affects downstream quantities that rely on the exact barycentric geometry. For mixture estimation, if $\|\hat{\theta}_i - \theta_i^*\|_{\mathrm{F}} \le \delta_{\mathrm{loc}}$, then*

$$\left\|\hat{\theta}_i - \theta_i^{\mathrm{bary}}\right\|_{\mathrm{F}} \le \delta_{\mathrm{loc}} + \varkappa\,\mathrm{diam}_{\mathrm{F}}(\Theta),$$

*so the curvature-induced deviation enters as an additive upload error. The routing results of Appendix D.5 rely more directly on the quadratic risk geometry and therefore do not transfer verbatim. Extending them to general losses would require controlling the same curvature-induced deviation together with the strong-convexity and smoothness constants. When $\varkappa = 0$, the exact barycentric geometry of the squared-loss analysis is recovered.*

The preceding results reduce the general-loss extension to controlling the curvature and conditioning quantities $\mu$, $L$, $\Delta_H$, and $L_H$. We next instantiate these quantities for softmax cross-entropy over the frozen-feature model; the composed-transformer approximation is treated separately in Appendix D.8.

### D.6.3 Softmax Cross-Entropy over a Frozen Representation

We specialize the general-loss analysis to softmax cross-entropy over the frozen representation in Eq. (2). Assume $l$ classes and let

$$z_\theta(x) = f_0(x) + \theta\phi(x) \in \mathbb{R}^l, \qquad \ell(\theta; x, y) = -\log\big[\mathrm{softmax}(z_\theta(x))\big]_y.$$

Writing $p_\theta(x) = \mathrm{softmax}(z_\theta(x))$, define the softmax covariance

$$S(z) := \mathrm{diag}(p) - pp^\top.$$

Because the softmax is invariant to adding the same scalar to every logit, we use the centered parameterization $\mathbf{1}^\top\theta = 0$. The corresponding invariant complement is $\mathcal{V} = \mathbb{R}^r \otimes \mathbf{1}^\perp$, and all curvature bounds below are understood on this subspace.

**Proposition 9 (Cross-entropy curvature over a frozen representation)** *For every $\theta$, the Hessian evaluates to:*

$$\nabla^2\mathcal{L}_m(\theta) \;=\; \mathcal{H}_m(\theta) \;:=\; \mathbb{E}_{\mathcal{P}_m}\Big[\,\phi(x)\phi(x)^\top \otimes S\big(z_\theta(x)\big)\Big]. \tag{114}$$

*Suppose Assumption 4 holds and each taskwise feature second moment $\Phi_m := \mathbb{E}_{\mathcal{P}_m}[\phi(x)\phi(x)^\top]$ is positive definite. Define $\lambda_- := \min_m \sigma_{\min}(\Phi_m)$ and $\lambda_+ := \max_m \sigma_{\max}(\Phi_m)$. Suppose further that $\mathcal{C} \subseteq \{\|\theta\|_{\mathrm{F}} \le \theta_{\max}\}$ and let $B_0 := \sup_x \|f_0(x)\|_\infty$. Then $\|z_\theta(x)\|_\infty \le B_0 + \theta_{\max}Q$, so every class probability satisfies*

$$\pi_{\min} \ge \frac{1}{l}\exp\big(-2(B_0 + \theta_{\max}Q)\big).$$

*On the centered subspace $\mathcal{V}$,*

$$l\,\pi_{\min}^2\lambda_- I \;\preceq\; \mathcal{H}_m(\theta) \;\preceq\; \tfrac{1}{2}\lambda_+ I, \qquad \frac{L}{\mu} \le \frac{\lambda_+}{2l\pi_{\min}^2\lambda_-}. \tag{115}$$

*Furthermore, the curvature dispersion $\Delta_H$ satisfies:*

$$\Delta_H \;=\; \max_{m\neq m'}\,\sup_{\theta\in\mathcal{C}}\big\|\mathcal{H}_m(\theta) - \mathcal{H}_{m'}(\theta)\big\| \;\le\; Q^2\cdot\max_{m\neq m'}\mathrm{TV}\big(\mathcal{P}_m^x, \mathcal{P}_{m'}^x\big), \tag{116}$$

*where $\mathcal{P}_m^x$ denotes the input marginal of task $m$. The Hessian-Lipschitz constant, measured in spectral norm against a Frobenius parameter displacement, satisfies $L_H \le 2Q^3$.*

**Proof.** *Hessian.* Defining $z = f_0(x) + \theta\phi(x)$ and $\vartheta = \mathrm{vec}(\theta)$ yields $\partial z/\partial\vartheta = \phi(x)^\top \otimes I_l$. Because the Hessian of the softmax cross-entropy with respect to its logits is $S(z)$, the chain rule yields $\nabla^2_\vartheta \ell = (\phi\phi^\top) \otimes S(z)$ pointwise. Taking expectations recovers Eq. (114).

*Upper and Lower Bounds.* Because $S(z)\mathbf{1} = 0$, we define the orthogonal projector $P := I_l - \frac{1}{l}\mathbf{1}\mathbf{1}^\top$ onto $\mathbf{1}^\perp$. For any unit vector $v$, shift-invariance of the variance allows us to assume $v$ is centered ($v^\top\mathbf{1} = 0$). The geometric constraints $\max v \ge 0 \ge \min v$ and $(\max v)^2 + (\min v)^2 \le \|v\|_2^2 = 1$ guarantee $\max_c v_c - \min_c v_c \le \sqrt{2}$. Popoviciu's inequality therefore dictates $\mathrm{Var}_{c\sim p}(v_c) \le \frac{1}{4}(\max - \min)^2 \le \frac{1}{2}$, ensuring $S(z) \preceq \frac{1}{2}I_l$. Leveraging the logit bound $p_c \ge \pi_{\min}$, we simultaneously derive the lower bound: $\mathrm{Var}_{c\sim p}(v_c) = \frac{1}{2}\sum_{c,d} p_c p_d (v_c - v_d)^2 \ge \frac{\pi_{\min}^2}{2}\sum_{c,d}(v_c - v_d)^2 = l\pi_{\min}^2\|v\|_2^2$. This establishes $S(z) \succeq l\pi_{\min}^2 P$. Because $\phi(x)\phi(x)^\top \succeq 0$, tensoring preserves both orderings pointwise. Taking expectations gives $\mathcal{H}_m(\theta) \succeq l\pi_{\min}^2(\Phi_m \otimes P)$ and $\mathcal{H}_m(\theta) \preceq \frac{1}{2}(\Phi_m \otimes I_l)$. Restricting to $\mathcal{V} = \mathbb{R}^r \otimes \mathbf{1}^\perp$ and using the definitions of $\lambda_-$ and $\lambda_+$ yields Eq. (115).

*Dispersion.* Evaluated at a common parameter $\theta$, the integrand $\Psi_\theta(x) := \phi(x)\phi(x)^\top \otimes S(z_\theta(x))$ constitutes a fixed function bounded by $\|\Psi_\theta(x)\| \le \frac{1}{2}Q^2$. For bounded functions $\Psi$ and distributions $P_1, P_2$, the relation $\|\mathbb{E}_{P_1}\Psi - \mathbb{E}_{P_2}\Psi\| \le 2\sup_x\|\Psi(x)\|\,\mathrm{TV}(P_1, P_2)$ under the convention $\mathrm{TV}(P_1, P_2) = \sup_A |P_1(A) - P_2(A)|$ guarantees Eq. (116).

*Hessian-Lipschitz.* Using the Jacobian bound from the proof of Lemma 3, $\|S(z) - S(z')\|_2 \le \frac{3}{2}\|z - z'\|_2$, and hence $\|\mathcal{H}_m(\theta) - \mathcal{H}_m(\theta')\|_2 \le \frac{3}{2}Q^3\|\theta - \theta'\|_{\mathrm{F}} \le 2Q^3\|\theta - \theta'\|_{\mathrm{F}}$. ■

**Remark 11 (Interpretation of the cross-entropy constants)** *The lower curvature bound deteriorates as the logit range increases, since $\mu \ge l\,\pi_{\min}^2\lambda_-$ and $\pi_{\min}$ decreases exponentially with $B_0 + \theta_{\max}Q$. Consequently, the resulting upper bounds on the condition number $L/\mu$ and the perturbation parameter $\varkappa$ can become loose when the model is highly confident. The analysis is therefore most informative in a bounded-logit regime, enforced here through the parameter set $\mathcal{C}$ and the bound $\|\theta\|_{\mathrm{F}} \le \theta_{\max}$. Under these conditions, softmax cross-entropy satisfies the local strong-convexity, smoothness, and Hessian-regularity requirements used by the preceding general-loss results.*

### D.7 Additional Model-Class Instantiations

The preceding analysis extends the framework along the loss dimension. A complementary question is how the shared-representation abstraction applies beyond the LoRA instantiation used by FedSEE. For the squared-loss results of Section 3, the relevant quantities are determined jointly by the induced representation and the resulting expert geometry:

- $\sigma_{\min}(\Phi)$ and $\sigma_{\max}(\Phi)$ characterize the conditioning of the shared feature geometry;
- $h$ and $\mathrm{diam}_{\mathrm{F}}(\Theta)$ characterize the geometry of the task-expert simplex and enter the recovery and routing guarantees;
- $Q$, the feature bound of Assumption 4, controls the finite-sample and routing-variation terms.

Any model whose induced $(f_0, \phi)$ representation and task experts satisfy the corresponding assumptions inherits the guarantees of Section 3 with the associated constants. Section 4 develops the LoRA instantiation. We give two additional examples below.

**Linear probing and last-layer personalization.** Let $\phi$ be a frozen pre-trained encoder and let $\theta$ denote the trainable linear head. In this case, Assumption 2(i) has the usual interpretation that each task is well specified within the shared representation. Assumption 2(ii) additionally requires the tasks to induce a common feature second moment, which need not hold for a general pre-trained encoder because different task distributions may induce different feature covariances. When these covariances are only approximately shared, the covariance-mismatch analysis of Remark 1 in Appendix D.1 provides the corresponding perturbed interpretation through $\rho_\Phi$.

**Random features and fixed kernel representations.** Let $\phi$ be a fixed finite-dimensional random-feature map, or a fixed feature representation associated with a kernel, and let $\theta$ denote the corresponding linear coefficient block. Bounded features directly satisfy the representation bound of Assumption 4, while feature non-degeneracy gives the corresponding covariance lower bound. The remaining conditions are not automatic. Task realizability requires the task predictors to lie in the span of the chosen feature map, and Assumption 2(ii) requires the induced feature covariance to be shared across tasks. When the latter holds only approximately, Remark 1 again provides the appropriate covariance-perturbation interpretation.

These examples illustrate that the framework is not specific to LoRA, while its guarantees remain conditional on the induced representation and task geometry satisfying the stated assumptions. Directions for relaxing these assumptions are discussed in Appendix E.

### D.8 Deep-Model Extension for FedSEE

We next return to the deep FedSEE instantiation. The preceding analysis applies to the frozen-feature model in Eq. (2). At a single adapted projection, fixing $A^{\mathrm{global}}$ makes the routed contribution linear in the $B_m$ blocks, so the reduction is exact conditional on the incoming representation, as detailed in Appendix C.1. In the full transformer, however, an adapted projection changes the representations passed to subsequent layers, and query adaptation may also modify the attention weights. Proposition 10 controls the resulting deviation from a local first-order model, and Corollary 8 translates this approximation into a bound on the departure from exact barycentric geometry.

**Proposition 10 (Deep-model perturbation)** *Let $\vartheta$ collect the routed $B$ blocks across all adapted projections and let $F_\vartheta$ denote the resulting network. Fix a reference $\vartheta_0$ and suppose the parameter Jacobian is $H$-Lipschitz on a convex neighborhood $\|\vartheta - \vartheta_0\|_{\mathcal{B}} \le R_B$, where $\|\vartheta\|_{\mathcal{B}}^2$ is the sum of squared Frobenius norms of its constituent blocks. Then, for every $\Delta$ such that $\vartheta_0 + \Delta$ remains in this neighborhood,*

$$F_{\vartheta_0+\Delta}(x) = F_{\vartheta_0}(x) + J_0(x)\Delta + r_\Delta(x), \qquad \|r_\Delta(x)\|_2 \le \frac{H}{2}\|\Delta\|_{\mathcal{B}}^2, \tag{117}$$

*where $J_0(x)$ is the Jacobian of $F_\vartheta(x)$ with respect to $\vartheta$ evaluated at $\vartheta_0$.*

**Proof.** Fix an input $x$ and define the path

$$g(t) := F_{\vartheta_0 + t\Delta}(x), \qquad t \in [0,1].$$

Because the line segment $\{\vartheta_0 + t\Delta : t \in [0,1]\}$ lies in the assumed neighborhood, the fundamental theorem of calculus gives

$$F_{\vartheta_0+\Delta}(x) - F_{\vartheta_0}(x) = \int_0^1 J_{\vartheta_0+t\Delta}(x)\Delta \, dt.$$

Adding and subtracting $J_{\vartheta_0}(x)\Delta$ yields

$$F_{\vartheta_0+\Delta}(x) = F_{\vartheta_0}(x) + J_{\vartheta_0}(x)\Delta + r_\Delta(x),$$

where

$$r_\Delta(x) := \int_0^1 \left(J_{\vartheta_0+t\Delta}(x) - J_{\vartheta_0}(x)\right)\Delta \, dt.$$

Let $\|\cdot\|_{\mathrm{op}}$ denote the operator norm induced by $\|\cdot\|_{\mathcal{B}}$ on parameter perturbations and $\|\cdot\|_2$ on network outputs. By the $H$-Lipschitz assumption on the Jacobian,

$$\|J_{\vartheta_0+t\Delta}(x) - J_{\vartheta_0}(x)\|_{\mathrm{op}} \le Ht\|\Delta\|_{\mathcal{B}}.$$

Therefore,

$$\begin{aligned}
\|r_\Delta(x)\|_2 &\le \int_0^1 \|J_{\vartheta_0+t\Delta}(x) - J_{\vartheta_0}(x)\|_{\mathrm{op}} \, \|\Delta\|_{\mathcal{B}} \, dt \\
&\le \int_0^1 Ht\|\Delta\|_{\mathcal{B}}^2 \, dt \\
&= \frac{H}{2}\|\Delta\|_{\mathcal{B}}^2.
\end{aligned}$$

Finally, for fixed $\vartheta_0$ and $x$, $J_{\vartheta_0}(x)\Delta$ is linear in the concatenated parameter perturbation $\Delta$. Thus the first-order network is a generalized affine predictor in $\Delta$ with input-dependent design operator $J_0(x)$. This form need not have the separable output-block structure $\theta\phi(x)$ of Eq. (2). ■

**Assumption 7 (Local first-order geometry)** *After vectorizing $\Delta$, write the first-order predictor as*

$$F_{\mathrm{lin},v}(x) = F_{\vartheta_0}(x) + J_0(x)v.$$

*For each task $m$, there exists $v_m^*$ such that*

$$y = F_{\vartheta_0}(x) + J_0(x)v_m^* + \zeta_m, \qquad \mathbb{E}[\zeta_m \mid x] = 0.$$

*On the analyzed parameter subspace, the taskwise Jacobian Gram matrices satisfy*

$$G_m := \mathbb{E}_{\mathcal{P}_m}[J_0(x)^\top J_0(x)] = G \succ 0$$

*for all $m$.*

Under Assumption 7, the same quadratic argument as in Theorem 1 gives $v_i^* = \sum_m \alpha_{im} v_m^*$ for the first-order client optimum. Since $\vartheta = \vartheta_0 + v$ and $\sum_m \alpha_{im} = 1$, the corresponding first-order optima also satisfy $\vartheta_{i,\mathrm{lin}}^* = \sum_m \alpha_{im}\vartheta_{m,\mathrm{lin}}^*$. The following corollary relates this geometry to the optima of the full deep model.

**Corollary 8 (Approximate barycentric geometry of the deep model)** *Suppose Assumption 7 holds and that the task and client minimizers of both the deep and first-order risks lie in the parameter neighborhood of Proposition 10. Suppose further that, on this neighborhood, each task risk and each client risk is uniformly within $\tau_{\mathrm{deep}}$ of its corresponding first-order risk, and that the first-order risks are $\mu$-strongly convex on the analyzed parameter subspace. Then*

$$\|\vartheta_m^* - \vartheta_{m,\mathrm{lin}}^*\|_{\mathcal{B}} \le 2\sqrt{\frac{\tau_{\mathrm{deep}}}{\mu}}$$

*for every task $m$, and likewise*

$$\|\vartheta_i^* - \vartheta_{i,\mathrm{lin}}^*\|_{\mathcal{B}} \le 2\sqrt{\frac{\tau_{\mathrm{deep}}}{\mu}}$$

*for every client $i$. Consequently,*

$$\left\|\vartheta_i^* - \sum_{m=1}^M \alpha_{im}\vartheta_m^*\right\|_{\mathcal{B}} \le 4\sqrt{\frac{\tau_{\mathrm{deep}}}{\mu}}. \tag{118}$$

**Proof.** Let $R$ denote one of the deep risks and $R_{\mathrm{lin}}$ its first-order counterpart, with

$$\sup_{\vartheta} |R(\vartheta) - R_{\mathrm{lin}}(\vartheta)| \le \tau_{\mathrm{deep}}.$$

Let $\vartheta^*$ and $\vartheta^*_{\mathrm{lin}}$ minimize $R$ and $R_{\mathrm{lin}}$, respectively. Then

$$R_{\mathrm{lin}}(\vartheta^*) \le R(\vartheta^*) + \tau_{\mathrm{deep}} \le R(\vartheta^*_{\mathrm{lin}}) + \tau_{\mathrm{deep}} \le R_{\mathrm{lin}}(\vartheta^*_{\mathrm{lin}}) + 2\tau_{\mathrm{deep}}.$$

By $\mu$-strong convexity of $R_{\mathrm{lin}}$,

$$R_{\mathrm{lin}}(\vartheta^*) - R_{\mathrm{lin}}(\vartheta^*_{\mathrm{lin}}) \ge \frac{\mu}{2}\|\vartheta^* - \vartheta^*_{\mathrm{lin}}\|^2_{\mathcal{B}},$$

and hence

$$\|\vartheta^* - \vartheta^*_{\mathrm{lin}}\|_{\mathcal{B}} \le 2\sqrt{\frac{\tau_{\mathrm{deep}}}{\mu}}.$$

Applying this once to client $i$ and once to every task, and using Assumption 7 and the generalized affine barycentric identity established above,

$$\vartheta^*_{i,\mathrm{lin}} = \sum_m \alpha_{im}\vartheta^*_{m,\mathrm{lin}},$$

gives

$$\begin{aligned}\left\|\vartheta^*_i - \sum_m \alpha_{im}\vartheta^*_m\right\|_{\mathcal{B}} &\le \|\vartheta^*_i - \vartheta^*_{i,\mathrm{lin}}\|_{\mathcal{B}} \\ &\quad + \sum_m \alpha_{im}\|\vartheta^*_{m,\mathrm{lin}} - \vartheta^*_m\|_{\mathcal{B}} \\ &\le 4\sqrt{\frac{\tau_{\mathrm{deep}}}{\mu}},\end{aligned}$$

since $\sum_m \alpha_{im} = 1$. ■

### D.9 Recovery under Balanced Transport

Theorem 2 analyzes the supervised extraction problem in Eq. (8) independently for each task. FedSEE instantiates this extractor through Eq. (13) and optimizes the taskwise programs jointly under a balanced-transport constraint on the client contributions. This coupling reduces the joint feasible set relative to independent extraction and can therefore increase the taskwise extraction loss. The following proposition quantifies the resulting effect on expert recovery through the coupling excess $\delta^{\mathrm{BT}}_m$. The corresponding optimization procedure is described in Appendix C.2.

**Proposition 11 (Recovery under balanced coupling)** *Under Assumptions 1, 2, and 3, with exact client uploads and population public loss, let $\theta^{\mathrm{ind}}_m$ denote the independent extractor certified by Theorem 2, and let $\theta^{\mathrm{BT}}_m$ denote expert $m$ produced by the jointly balanced program. Define the coupling excess*

$$\delta^{\mathrm{BT}}_m := \mathcal{L}_m(\theta^{\mathrm{BT}}_m) - \mathcal{L}_m(\theta^{\mathrm{ind}}_m) \ge 0.$$

*Then*

$$\|\theta^{\mathrm{BT}}_m - \theta^*_m\|^2_{\mathrm{F}} \le \kappa(\Phi)\,\varepsilon^2 \mathrm{diam}_{\mathrm{F}}(\Theta)^2 + \frac{\delta^{\mathrm{BT}}_m}{\sigma_{\min}(\Phi)}. \tag{119}$$

*Hence the independent recovery guarantee is retained whenever the transport constraint is nonbinding ($\delta^{\mathrm{BT}}_m = 0$). In particular, transport-compatible pure anchors give exact population recovery.*

**Proof.** Because each row of the balanced solution remains a convex combination of the upload pool, $\theta^{\mathrm{BT}}_m$ is feasible for the corresponding independent extraction problem. Since $\theta^{\mathrm{ind}}_m$ minimizes the task loss over this feasible set,

$$\mathcal{L}_m(\theta^{\mathrm{BT}}_m) \ge \mathcal{L}_m(\theta^{\mathrm{ind}}_m),$$

and hence $\delta^{\mathrm{BT}}_m \ge 0$.

Under Assumption 2, the population task loss satisfies

$$\mathcal{L}_m(\theta) - \mathcal{L}_m(\theta_m^*) = \|\theta - \theta_m^*\|_\Phi^2.$$

By the definition of $\delta_m^{\mathrm{BT}}$,

$$\begin{aligned}\|\theta_m^{\mathrm{BT}} - \theta_m^*\|_\Phi^2 &= \mathcal{L}_m(\theta_m^{\mathrm{BT}}) - \mathcal{L}_m(\theta_m^*) \\ &= \mathcal{L}_m(\theta_m^{\mathrm{ind}}) - \mathcal{L}_m(\theta_m^*) + \delta_m^{\mathrm{BT}} \\ &= \|\theta_m^{\mathrm{ind}} - \theta_m^*\|_\Phi^2 + \delta_m^{\mathrm{BT}}.\end{aligned}$$

The population recovery guarantee for the independent extractor gives

$$\|\theta_m^{\mathrm{ind}} - \theta_m^*\|_\Phi \le \sigma_{\max}(\Phi)^{1/2}\varepsilon\,\mathrm{diam}_{\mathrm{F}}(\Theta).$$

Therefore,

$$\|\theta_m^{\mathrm{BT}} - \theta_m^*\|_\Phi^2 \le \sigma_{\max}(\Phi)\varepsilon^2\mathrm{diam}_{\mathrm{F}}(\Theta)^2 + \delta_m^{\mathrm{BT}}.$$

Finally, using

$$\|A\|_{\mathrm{F}}^2 \le \sigma_{\min}(\Phi)^{-1}\|A\|_\Phi^2$$

gives

$$\|\theta_m^{\mathrm{BT}} - \theta_m^*\|_{\mathrm{F}}^2 \le \kappa(\Phi)\varepsilon^2\mathrm{diam}_{\mathrm{F}}(\Theta)^2 + \frac{\delta_m^{\mathrm{BT}}}{\sigma_{\min}(\Phi)},$$

which is Eq. (119). ■

### D.10 Stability of the Deployed Contribution Matrix

FedSEE applies temporal smoothing to the balanced contribution matrices produced across communication rounds before using them for capacity assignment and router calibration. We next characterize the feasibility and tracking behavior of this smoothing step.

Define the balanced-transport polytope

$$\mathcal{C} := \left\{ C \in \mathbb{R}_+^{M\times N} : C\mathbf{1}_N = \mathbf{1}_M,\ C^\top \mathbf{1}_M = \frac{M}{N}\mathbf{1}_N \right\}. \tag{120}$$

Every raw matrix $C^{*,(t)}$ produced by the balanced extraction lies in $\mathcal{C}$, and the initialization matrix $C^{(1)}$ is obtained from the same extraction program. Since $\mathcal{C}$ is convex, the EMA recursion in Proposition 12 preserves these constraints at every round. The proposition also quantifies how closely the smoothed matrix tracks a changing sequence of raw extraction matrices.

**Proposition 12 (Stability of the deployed contribution EMA)** *Let $C^{*,(t)} \in \mathcal{C}$ denote the raw balanced contribution matrix returned by extraction at round $t$, let $C^{(1)} \in \mathcal{C}$, and let the deployed matrix follow*

$$C^{(t+1)} = (1-\lambda)C^{(t)} + \lambda C^{*,(t)}, \qquad \lambda \in (0,1).$$

*Then $C^{(t)} \in \mathcal{C}$ for every $t$.*

*Suppose additionally that the roundwise raw extraction matrices satisfy*

$$\|C^{*,(t+1)} - C^{*,(t)}\|_{\mathrm{F}} \le \zeta_C^{(t)},$$

*and define*

$$\bar{\zeta}_C := \limsup_{t\to\infty} \zeta_C^{(t)}.$$

*Then*

$$\limsup_{t\to\infty} \|C^{(t)} - C^{*,(t)}\|_{\mathrm{F}} \le \frac{\bar{\zeta}_C}{\lambda}. \tag{121}$$

*In particular, if $\zeta_C^{(t)} \to 0$, then*

$$\|C^{(t)} - C^{*,(t)}\|_{\mathrm{F}} \to 0.$$

*If the raw extraction matrix is stationary, $C^{*,(t)} \equiv C^*$, then the stronger geometric bound holds:*

$$\|C^{(t)} - C^*\|_{\mathrm{F}} \le (1-\lambda)^{t-1}\|C^{(1)} - C^*\|_{\mathrm{F}}. \tag{122}$$

**Proof.** Because $\mathcal{C}$ is convex and both $C^{(t)}$ and $C^{*,(t)}$ lie in $\mathcal{C}$,

$$C^{(t+1)} = (1-\lambda)C^{(t)} + \lambda C^{*,(t)} \in \mathcal{C}.$$

Since $C^{(1)} \in \mathcal{C}$, induction proves feasibility for every round.

For the tracking result, define

$$E_C^{(t)} := C^{(t)} - C^{*,(t)}.$$

Subtracting $C^{*,(t+1)}$ from the EMA recursion gives

$$\begin{aligned} E_C^{(t+1)} &= (1-\lambda)C^{(t)} + \lambda C^{*,(t)} - C^{*,(t+1)} \\ &= (1-\lambda)E_C^{(t)} - \big(C^{*,(t+1)} - C^{*,(t)}\big). \end{aligned} \tag{123}$$

Therefore,

$$\|E_C^{(t+1)}\|_{\mathrm{F}} \le (1-\lambda)\|E_C^{(t)}\|_{\mathrm{F}} + \zeta_C^{(t)}. \tag{124}$$

Iterating this inequality yields

$$\|E_C^{(t)}\|_{\mathrm{F}} \le (1-\lambda)^{t-1}\|E_C^{(1)}\|_{\mathrm{F}} + \sum_{s=1}^{t-1}(1-\lambda)^{t-1-s}\zeta_C^{(s)}. \tag{125}$$

To control the final term, fix $\epsilon > 0$. By the definition of $\bar{\zeta}_C$, there exists $S$ such that

$$\zeta_C^{(s)} \le \bar{\zeta}_C + \epsilon \qquad \text{for all } s \ge S.$$

Split the convolution in Eq. (125) at $S$. The finite head satisfies

$$\sum_{s=1}^{S-1}(1-\lambda)^{t-1-s}\zeta_C^{(s)} \longrightarrow 0$$

as $t \to \infty$, while the tail is bounded by

$$\begin{aligned} \sum_{s=S}^{t-1}(1-\lambda)^{t-1-s}\zeta_C^{(s)} &\le (\bar{\zeta}_C + \epsilon)\sum_{j=0}^{\infty}(1-\lambda)^j \\ &= \frac{\bar{\zeta}_C + \epsilon}{\lambda}. \end{aligned} \tag{126}$$

The initialization term in Eq. (125) also vanishes geometrically. Taking $\limsup_{t\to\infty}$ and then letting $\epsilon \downarrow 0$ proves Eq. (121).

If $\zeta_C^{(t)} \to 0$, then $\bar{\zeta}_C = 0$, giving exact tracking. If $C^{*,(t)} \equiv C^*$, the drift term in Eq. (123) vanishes identically, and direct iteration gives Eq. (122). ■

**Relation to recovery and mixture estimation.** Proposition 12 concerns only the temporal behavior of the smoothed contribution matrix. It guarantees that the balanced-transport constraints are preserved and bounds the discrepancy between the smoothed matrix $C^{(t)}$ and the roundwise extraction matrix $C^{*,(t)}$. It does not imply that either matrix recovers the true client mixtures.

Mixture estimation is instead characterized by Proposition 5, which bounds the error of the barycentric estimate $\hat{\alpha}_i$. FedSEE uses the normalized contribution vector $\tilde{c}_i$ as a computational proxy for this estimate, with discrepancy $\delta_i^{\mathrm{proxy}} = \|\tilde{c}_i - \hat{\alpha}_i\|_1$.

Similarly, Proposition 3 applies to an upload pool satisfying its shared-representation, anchor, and estimation conditions. These are conditions on the upload pool rather than guarantees of the FedSEE training procedure. In particular, initialization trains $B_i$ jointly with client-specific $A_i$ before forming $A^{\mathrm{global}}$, while later uploads arise from joint routed training. The proposition applies whenever the resulting upload pool satisfies the stated conditions.

# E FUTURE DIRECTIONS

The present analysis suggests two natural directions for extending the framework. First, the exact barycentric characterization holds under the squared-loss model, while the general-loss analysis of Appendix D.6 introduces a perturbation controlled by the variation in task curvature. A sharper characterization of this perturbation, particularly for deep models trained with cross-entropy, would further connect the theoretical framework to practical fine-tuning settings. Second, the framework assumes that the number of underlying tasks $M$ is known. In practical federated settings, this quantity may itself need to be inferred from the client population. Extending supervised expert extraction to jointly estimate the number of latent tasks and their associated experts is an important direction for future work.