FEDTAU: CORRECTING OBSERVATION-INDUCED TARGET MISMATCH IN FEDERATED LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Federated learning (FL) trains a shared predictor from heterogeneous client data, but existing methods mainly address heterogeneity after local objectives have already been defined. We identify an earlier source of discrepancy: clients may observe the same underlying prediction task through different sampling or recording mechanisms, causing their local supervision to encode different prediction targets. We refer to this phenomenon as *observation-induced target mismatch*. To address this, we formulate FL as learning a shared reference response distribution through client-specific observation mechanisms and propose **FedTau**, a unified τ-correction that corrects local supervision where the mismatch first occurs. For each observed record, FedTau evaluates the shared predictor through the corresponding observation mechanism and defines τ as the difference between the observed-data negative log-likelihood and the conventional supervised loss. Theoretically, we show that the corrected client objectives can achieve a common predictive optimum and analyze the effect of observation-mechanism estimation errors on local updates and aggregation. Experiments on nine datasets against twenty-three distinct baselines demonstrate consistent improvements across all three tasks. In particular, FedTau improves CIFAR-10 accuracy by 12.32% over the second-best under severe class-prior heterogeneity, reduces RMSE by 6.67%–37.61% across three right-censored regression datasets, and improves both token likelihood and token accuracy on all three sequence-prediction datasets.

1 INTRODUCTION

Federated Learning (FL) is a well-known collaborative Machine Learning (ML) training algorithm that trains a shared/global predictor by combining client updates (McMahan et al., 2017; Kairouz & McMahan, 2021). However, its intrinsic property of *diverse modes of observation* from different clients leads to the performance degradation of the global predictor. A vivid analogy is the parable of the "Six blind men touching an elephant". Each man touches the same elephant (underlying object), but through different observation mechanisms (touching the legs, nose, ears, etc.). Each man may think his own touching (local observation) is a complete description of the elephant, leading to incompatible results. The same thing occurs in FL. Clients perform the same underlying prediction task, but different sampling or recording processes in training make identical local loss functions focus on different apparent local prediction targets. To address such heterogeneity, existing approaches develop local regularization, control variates, adaptive aggregation, or information sharing (Li et al., 2020; Karimireddy et al., 2020; Acar et al., 2021; Reddi et al., 2021). However, they optimize the training after local objectives are specified (Li et al., 2021; Tan et al., 2022; Zhu et al., 2021), whereas an *earlier source of discrepancy* has already occurred and is left to be addressed: *how the local observation discrepancy affect the model performance and whether the local observation process itself can determine what target is encoded/processed by local supervision*?

To help dive into real-world ML training problems, we list some examples. Class-selective sampling changes local label probabilities. A reporting threshold replaces some continuous responses with a bound. A storage limit removes sequence suffixes. If these recorded observations are treated directly as ground truth, the resulting local objectives may encode different tasks. Squared-error training on capped response targets the capped conditional mean rather than the original conditional mean. Assigning an artificial end-of-sequence token to a truncated sequence teaches the model to stop at the recording limit. These discrepancies still remain even under exact local optimization. Better

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.

optimization can solve the specified local objective more accurately, but cannot correct a target that was defined from a distorted observation beforehand. We call this phenomenon *observation-induced target mismatch*. *Therefore, modifying local updates or their aggregation alone (by existing methods) does not resolve how heterogeneous observations should be interpreted during training*.

To this end, for the first time, we propose our conceptual formulation of FL as learning a shared/global reference response distribution through client-specific observation mechanisms. The reference distribution defines the prediction task of interest at deployment, while each client's sampling or recording mechanism specifies how the shared response is transformed into local observations available for training. This separates *what a client observes* from *what the model is intended to predict*, and makes observation-induced target mismatch an explicit part of local objective design.

Building on this formulation, we develop **FedTau**, a unified observation-aware correction for local supervision, namely $\boldsymbol{\tau}$**-correction**. The high-level idea of FedTau is to correct the discrepancy when it first occurs (observation mismatch). Technically, for each observed record, FedTau evaluates the shared predictor through the *client-specific observation mechanism* and defines τ as the correction from the conventional supervised loss to the observation negative log-likelihood. The correction is fully determined by the observation mechanism, which is specified from training statistics or recording metadata, rather than being introduced as an independently weighted regularizer (like mainstream methods). To realize this, FedTau yields prior correction for class-selective sampling, bound-aware supervision for censored regression, and continuation-aware supervision for truncated sequences. Please note that FedTau corrects supervisions by only changing the local training objectives. Other subsequent training components (model architecture, weighted model aggregation, and direct reference-task inference, etc.) remain unchanged, making it possible to leverage joint optimization with other existing algorithms. In our theoretical analysis, we establish when such a single proposed correction can give heterogeneous clients a common/unified predictive optimum and quantify the effect of mechanism estimation on local updates and aggregation.

We validate FedTau across nine datasets against twenty-three baselines on three mainstream ML tasks (image classification, censored regression, and sequence prediction). Across all three tasks, FedTau consistently improves reference-task prediction under our proposed heterogeneous client observation mechanisms. Moreover, under severe class-prior heterogeneity on CIFAR-10 (100 clients, 10% participation, Dirichlet $\alpha = 0.1$, and one local epoch), FedTau outperforms the second best by 12.32% and FedAvg by 17.08%. Across three right-censored regression benchmarks, FedTau reduces RMSE by 6.67%–37.61% relative to the second best. On all three sequence-prediction benchmarks, FedTau improves both token likelihood and token accuracy. Additional classification experiments across IID and non-IID partitions, different model architectures, and varying local-training budgets further demonstrate the robustness and scalability of FedTau's τ-correction mechanism.

Our contributions are threefold. (i) **Observation-induced target mismatch.** For the first time, we identify how client-specific sampling and recording processes can make local supervision encode different prediction targets for the same underlying task, and formalize this mismatch using a shared reference distribution and client-specific observation mechanisms. (ii) **A unified $\boldsymbol{\tau}$-correction.** We derive FedTau from observation likelihoods and establish its common-target property and sensitivity to mechanism-estimation errors. (iii) **Cross-task validation.** We evaluate FedTau across nine datasets against twenty-three baselines on three ML tasks and show consistent improvements under heterogeneous observation mechanisms, with ablations on client-mechanism matching, reference targets, and correction placement.

2 Related Work

Federated optimization. FedAvg combines local training with model averaging (McMahan et al., 2017). FedProx, SCAFFOLD, and FedDyn address local optimization through proximal penalties, control variates, and dynamic regularization, respectively (Li et al., 2020; Karimireddy et al., 2020; Acar et al., 2021). FedNova corrects objective inconsistency from unequal local work (Wang et al., 2020). FedOpt develops adaptive server updates (Reddi et al., 2021), and FedMuon corrects bias in local linear-minimization-oracle updates (Takezawa et al., 2026). These methods optimize a supplied loss, but they do not specify how to score a censored bound or a truncated prefix. FedTau derives that loss from the recording process while retaining model averaging.

Representation and knowledge sharing. MOON uses model-contrastive representation learning (Li et al., 2021). FedProto and FedTGP exchange class prototypes (Tan et al., 2022; Zhang et al., 2024), whereas FedGen, FedFTG, and FedLMG transfer knowledge through generated features or images (Zhu et al., 2021; Zhang et al., 2022b; Yang et al., 2025). These methods transfer representations or predictions rather than model the recording process. FedTau corrects local supervision through the observation likelihood, without auxiliary prototypes or a generator.

Class priors and prediction targets. Balanced Softmax and logit adjustment incorporate class frequencies into the classification objective (Ren et al., 2020; Menon et al., 2021). FedLC calibrates class-dependent margins (Zhang et al., 2022a), while Fed-RoD separates a balanced generic predictor from a personalized predictor (Chen & Chao, 2022). VRLS estimates test-to-train importance ratios using entropy regularization for distributed importance-weighted risk minimization (Wu et al., 2025). Unlike VRLS, which reweights examples, FedTau's likelihood formulation evaluates the recorded response under the client observation mechanism.

Observation likelihoods and loss correction. Forward loss correction accounts for label corruption in the predicted probabilities before evaluating cross-entropy (Patrini et al., 2017). Classical censored and grouped-data likelihoods likewise evaluate the recorded event, rather than treating a bound or interval as an exact response (Tobin, 1958; Turnbull, 1976). FedTau develops this likelihood-based approach to client-dependent observation mechanisms, covering both altered sampling probabilities and incomplete responses. Our τ definition yields prior, survival, and continuation corrections before model aggregation. Its common-target analysis establishes a shared predictive optimum without requiring identical client gradients under the stated assumptions (Appendix A.5).

3 Observation Mechanisms and Local Target Mismatch

FL combines models trained on different client datasets. However, a client may observe a transformed version of the response required at deployment. Directly fitting these observations can change what the model is trained to predict. FedTau addresses this target mismatch by accounting for the observation mechanism in the local loss, while retaining ordinary model averaging. We formalize the distinction between the response predicted at deployment and the observation that is actually available at each client during training.

Let Client $k \in \{1, \dots, K\}$ hold $\mathcal{D}_k = \{(x_i, o_i)\}_{i=1}^{n_k}$, where x is the input, o is the recorded observation, and n_k is the number of data samples in client k. Let y denote the reference response of interest, which may be only partially or indirectly observed. We use $p^*(y \mid x)$ to denote the shared reference response distribution and $q_\theta(y \mid x)$ the predictive model. Each client is associated with an observation mechanism $\mathcal{T}_k$, which maps a response distribution to the corresponding recorded observation distribution using $\mathcal{T}_k[\cdot]$. Thus, the observed-data distribution at client k satisfies

$$p_k^{\mathrm{obs}}(\cdot \mid x) = \mathcal{T}_k[p^*](\cdot \mid x). \tag{1}$$

The transformation $\mathcal{T}_k$ may represent response-dependent sampling, censoring, truncation, or other recording processes, and need not be linear or invertible. Clients may also have different input marginal distributions p_k^X, i.e., Non-Independent and Identically Distributed (Non-IID). At the population level, $\mathcal{T}_k$ denotes the underlying observation mechanism. In practice, its parameters can be specified by the data-collection protocol or estimated from available training statistics and recording metadata.

Observation-induced target mismatch. Directly fitting the recorded observation can change the population prediction target. Consider a client that records a capped response $u = \min(y, c_k)$. Squared-error training on u targets

$$\mathbb{E}[\min(Y, c_k) \mid X = x] = \mathbb{E}[Y \mid X = x] - \mathbb{E}[\max(Y - c_k, 0) \mid X = x]. \tag{2}$$

Whenever Y can exceed c_k with nonzero probability, the second term is positive, so the optimal predictor for the recorded response differs from that for the reference response. Moreover, because c_k may vary across clients, different clients can be driven toward different population optima even though they share the same underlying prediction task.

This mismatch cannot be removed by optimizing the original local loss more accurately. The issue lies in how the recorded observation is interpreted when defining the local objective, rather than in

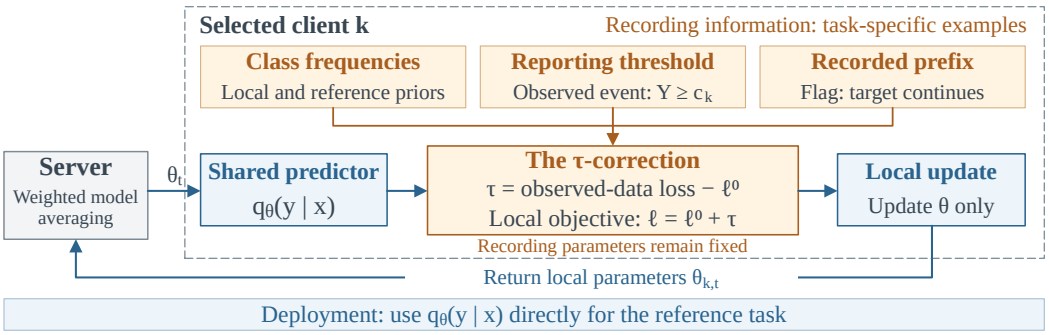


Figure 1: Core idea of **FedTau**. Observation-induced target mismatch arises when client-specific observations are directly used to define local supervision. FedTau corrects the mismatch at this point through the τ-correction, before local updates and model aggregation, while keeping the predictive model and inference unchanged.

optimization error itself. Therefore, our goal is to construct local objectives whose predictive optima are consistent with the shared reference task, without requiring client gradients to be identical at every parameter value. In Section 4, we develop the **τ-correction** to achieve this goal when the mismatch first occurs before model aggregation.

4 FedTau: A Unified Correction for Local Supervision

FedTau corrects local supervision at the stage when observation-induced target mismatch first arises. Given a client-specific observation mechanism, we derive a unified τ-correction from the observed-data likelihood and then apply the same principle for classification, regression, and sequence prediction. The illustration of FedTau is shown in Fig. 1.

4.1 Deriving the τ-Correction

In Section 3, we define the true/underlying observation mechanism $\mathcal{T}_k$. Nevertheless, $\mathcal{T}_k$ is not always known. In practice, we actually use an implemented/estimated observation mechanism $\widehat{\mathcal{T}}_k$, which is estimated by the data-collection protocol, available training statistics, and recording metadata, etc. When the observation mechanism is known exactly, $\widehat{\mathcal{T}}_k = \mathcal{T}_k$. Therefore, we have the corresponding predicted observation distribution at client k as $q^{\text{obs}}_{k,\theta} = \widehat{\mathcal{T}}_k[q_\theta]$. Let $\ell^0_k(\theta; x, o)$ be the uncorrected baseline supervised loss applied directly to a recorded observation o. The **τ-correction** is then the observation negative log-likelihood under $q^{\text{obs}}_{k,\theta}$ with the empirical local objective $\widehat{F}_k(\theta)$:

$$\begin{aligned} \tau_k(\theta; x, o) &= -\log q^{\text{obs}}_{k,\theta}(o \mid x) - \ell^0_k(\theta; x, o), \\ \widehat{F}_k(\theta) &= \frac{1}{n_k} \sum_{(x_i, o_i) \in D_k} \big[\ell^0_k(\theta; x_i, o_i) + \tau_k(\theta; x_i, o_i)\big]. \end{aligned} \tag{3}$$

Please note τ_k is not an independently weighted regularizer or an additional optimization penalty. It is uniquely determined by the observation likelihood and exactly converts the conventional local supervision into the corresponding observed-data negative log-likelihood. Although the parameters of $\widehat{T}_k$ remain fixed during local training, τ_k depends on the current prediction q_θ and is therefore differentiated with respect to θ. Consequently, its gradient is propagated through all trainable components of the predictive model. Details can be found in Appendix A.5.1.

4.2 Task-Specific Forms of the τ-Correction

The τ-correction is actually scalable to various ML activities, as long as the observation processes exist. We now specify the τ-correction in Eq. (3) for three mainstream ML tasks, each with a specific observation process. In general, the predictive model q_θ remains unchanged. only the client-specific observation mechanism and the resulting local supervision differ.

Classification: class-selective sampling. Consider class-selective sampling under label shift, where $p_k(x \mid y) = p^*(x \mid y)$. Let $\pi_k(c)$ and $\pi^*(c)$ denote the local and reference class priors for class $c \in \{1, \dots, C\}$ at client k, respectively. By Bayes' rule, the observation mechanism transforms a reference prediction $q(c \mid x)$ as $\mathcal{T}_k[q](c \mid x) \propto q(c \mid x)\frac{\pi_k(c)}{\pi^*(c)}$. For implementation, we estimate the priors, denoted by $\widetilde{\pi}_k(c)$ and $\widetilde{\pi}^*(c)$, using add-one-smoothed training counts and define the logits offset $b_{k,c} = \log \widetilde{\pi}_k(c) - \log \widetilde{\pi}^*(c)$. Define $z_\theta(\cdot)$ as uncorrected classifier logits and let $q_\theta = \mathrm{softmax}(z_\theta)$. The uncorrected loss is $\ell_k^0 = \mathrm{CE}(z_\theta(x), y)$. Applying Eq. (3) gives

$$\begin{aligned} \tau_k^{\mathrm{cls}}(\theta; x, y) &= \log \sum_c q_\theta(c \mid x) e^{b_{k,c}} - b_{k,y}, \\ \ell_k^{\mathrm{FedTau}} &= \ell_k^0 + \tau_k^{\mathrm{cls}} = \mathrm{CE}(z_\theta(x) + b_k, y). \end{aligned} \tag{4}$$

where e is equivalent to $\exp(\cdot)$. Here, FedTau corrects the client-specific sampling prior inside local training. Although b_k is fixed, τ_k^{cls} remains prediction-dependent through the normalization term. If the local and reference priors are identical, $b_k = 0$ and the correction vanishes. A uniform reference prior recovers the client-side Balanced Softmax objective (Ren et al., 2020). Appendix A.5.2 provides the corresponding prediction-error and prior-estimation analysis.

Regression: right-censored responses. Consider a client that records $u = \min(y, c_k)$ and let $\delta = \mathbf{1}\{y \geq c_k\}$, where c_k is the client-specific reporting threshold. The observation likelihood is $q_\theta(u \mid x)$ for an uncensored record and $\int_{c_k}^{\infty} q_\theta(v \mid x)\, dv$ for a censored one. Using the working model $q_\theta = \mathcal{N}(\mu_\theta(x), \sigma^2)$ and the uncorrected point loss $\ell_k^{\mathrm{pt}} = -\log q_\theta(u \mid x)$, we have

$$\begin{aligned} \tau_k^{\mathrm{reg}} &= \delta \left[-\log \Phi\left(\frac{\mu_\theta(x) - c_k}{\sigma} \right) - \ell_k^{\mathrm{pt}} \right], \\ \ell_k^{\mathrm{FedTau}} &= \ell_k^{\mathrm{pt}} + \tau_k^{\mathrm{reg}}, \end{aligned} \tag{5}$$

where Φ is the standard normal CDF, $\mu_\theta(x)$ is the predicted mean, and $\sigma > 0$ is a fixed working scale used in the Gaussian likelihood. Thus, for a censored record, FedTau replaces the capped point target with the likelihood of the observed event $Y \geq c_k$, rather than treating c_k as the exact response (Tobin, 1958). Appendix A.5.3 derives this likelihood and shows how the correction removes the corresponding mean-score bias. Implementation parameters are specified in Appendix A.7.2.

Sequence prediction: truncated targets. Consider a recorded target prefix $y_{1:L}$ and let δ represent a continuation flag, where L is the recorded prefix length except End-of-Sequence (EOS) and $\delta = 1$ indicates that the true target continues beyond the recorded prefix. Let $p_{\mathrm{EOS}} = q_\theta(\mathrm{EOS} \mid x, y_{1:L})$. The conventional loss treats the end of the recorded prefix as an EOS event: $\ell_k^{\mathrm{EOS}} = -\sum_{j=1}^{L} \log q_\theta(y_j \mid x, y_{<j}) - \log p_{\mathrm{EOS}}$. For a truncated target, the recorded boundary indicates continuation rather than termination. The continuation probability is therefore the total probability of all non-EOS next tokens, i.e., $1 - p_{\mathrm{EOS}}$, rather than p_{EOS}. We have

$$\begin{aligned} \tau_k^{\mathrm{seq}} &= \delta \left[\log p_{\mathrm{EOS}} - \log(1 - p_{\mathrm{EOS}}) \right], \\ \ell_k^{\mathrm{FedTau}} &= \ell_k^{\mathrm{EOS}} + \tau_k^{\mathrm{seq}}. \end{aligned} \tag{6}$$

Complete targets retain their true EOS supervision, whereas truncated targets are trained using the observed continuation event without imputing the missing suffix. Appendices A.5.3 and A.5.5 provide the corresponding marginal-likelihood derivation and normalization details.

4.3 A Common Target Before Aggregation

The task-specific corrections in Section 4.2 all follow the same principle: each client optimizes the likelihood of its observed records under the shared predictor and its observation mechanism. To characterize the population target, let $F_k(\theta) = \mathbb{E}_{X \sim p_k^X,\, O \sim \mathcal{T}_k[p^*](\cdot \mid X)} \left[-\log \mathcal{T}_k[q_\theta](O \mid X) \right]$ denote the population observation risk (the expected local loss) of client k under the true mechanism $\mathcal{T}_k$.

Proposition 1 (Common predictive target). Assume there exists θ^* such that $q_{\theta^*} = p^*$. For every client k, we have

$$F_k(\theta) - F_k(\theta^*) = \mathbb{E}_{X \sim p_k^X} D_{\mathrm{KL}} \left(\mathcal{T}_k[p^*](\cdot \mid X) \,\|\, \mathcal{T}_k[q_\theta](\cdot \mid X) \right) \geq 0. \tag{7}$$

Therefore, although clients have different input distributions p_k^X and different observation mechanisms, our proposed τ-correction can still minimize every client's local objective, referred to as a common predictive target. If the clients' observation mechanisms jointly provide sufficient information to identify the reference task, minimizing any positively weighted combination of these risks directs the shared reference distribution $p*$.

The insight is that FedTau makes it possible for every client to have a common/consistent predictive optimum θ^*, rather than diverse local optima, no matter how the gradients are changing during each training step. Appendix A.5.1 provides the proof, while Appendix A.5.4 analyzes the effect of estimated observation mechanisms on local updates and aggregation.

For the training process, as illustrated in Fig. 1, each participating client starts from the current global model θ_t, optimizes the corrected local objective in Eq. (3), and uploads $\theta_{k,t}$ to the server. The server then implements FedAvg (McMahan et al., 2017) for model aggregation:

$$\theta_{t+1} = \sum_{k \in S_t} \frac{n_k}{\sum_{j \in S_t} n_j} \theta_{k,t}, \tag{8}$$

where S_t is the set of participating clients at the training round t. The observation-mechanism parameters remain fixed during local training, and the inference uses the shared predictor q_θ directly. The detailed algorithm implementation (Alg. 1–4) is attached in Appendix A.2.

5 Experiments

5.1 Experimental design

Baselines. We compare FedTau against a total of twenty-three baselines across eleven classification, twelve regression, and nine sequence baselines. FedAvg (McMahan et al., 2017) is the direct averaging reference. FedProx (Li et al., 2020) and FedDyn (Acar et al., 2021) test whether local regularization addresses the same mismatch. For classification, we also compare representation and class-structure methods (MOON (Li et al., 2021), FedProto (Tan et al., 2022), FedTGP (Zhang et al., 2024), FedCPD (He et al., 2024), and FedRG (Wen et al., 2026)) and generative transfer methods (FedGen (Zhu et al., 2021), FedFTG (Zhang et al., 2022b), and FedLMG (Yang et al., 2025)). For regression, we add update normalization, control variates, adaptive aggregation, and regression-specific alternatives (Wang et al., 2020; Karimireddy et al., 2020; Hsu et al., 2019; Reddi et al., 2021; Tang, 2026; Passeri et al., 2023). For sequence prediction, we cover server momentum/adaptation, drift correction, extrapolation, and sharpness-aware updates (Hsu et al., 2019; Reddi et al., 2021; Gao et al., 2022; Jhunjhunwala et al., 2023; Sun et al., 2023; Fan et al., 2024). Details of these baselines and task adaptations can be found in Appendices A.6–A.8.

Datasets and federated settings. The experiment is conducted on a total of nine datasets, with three datasets for each task. (i) For classification, CIFAR-10/100 (Krizhevsky, 2009) and TinyImageNet (Le et al., 2015) use a ResNet-18 (He et al., 2016) trained from scratch with the settings 100 clients, 10% participation, local epochs $E \in \{1, 3\}$, and 500 rounds. Class-wise Dirichlet partition with $\alpha \in \{0.1, 0.6\}$ is utilized to control the level of non-IID. The IID reference uses equal-size random subsets. The complete results, including $E = 5$, StageCNN, input preprocessing, and optimizer settings, are in Appendix A.6. (ii) For regression, California Housing, Superconductivity, and hourly Bike Sharing (Pace & Barry, 1997; Hamidieh, 2018; Fanaee-T & Gama, 2014) use approximately equal-size random client partitions, 100 clients, 10% participation, and 500 rounds with five local minibatch updates per selected client. Details are in Appendix A.7.2. (iii) For sequence prediction, XSum (Narayan et al., 2018), SAMSum (Gliwa et al., 2019), and CNN/DailyMail (See et al., 2017) use FLAN-T5-small (Chung et al., 2024), with $4000/500/500$ examples sampled from the official training/validation/test splits. Ten clients receive 400 training records each, with two participate per round for 50 rounds and ten local minibatch updates each. Recording caps range from 8–32 tokens on XSum/SAMSum and 32–96 on CNN/DailyMail. Longer targets retain their prefixes and continuation flags. Details are in Appendix A.8.2.

Evaluation metrics and implementation. We evaluate the shared/global predictor using top-1 accuracy for classification, original-scale RMSE and R^2 for regression, and reference-target token NLL and teacher-forced token accuracy for sequence prediction. MAE, generation-level ROUGE,

Table 1: ResNet-18 accuracy (%). Best/second-best results are bold/boxed. ↑ Δ is the performance increase over the second best.

Method	α	CIFAR-10 Centralized: 95.42±0.08 $E=1$	CIFAR-10 $E=3$	CIFAR-100 Centralized: 76.12±0.18 $E=1$	CIFAR-100 $E=3$	TinyImageNet Centralized: 60.61±0.14 $E=1$	TinyImageNet $E=3$
FedAvg	0.1	62.06±6.14	73.52±1.04	47.61±1.69	56.84±1.49	43.09±1.22	47.02±1.14
	0.6	77.43±2.46	85.82±2.00	54.26±0.92	61.91±0.93	46.52±0.52	50.43±0.65
	IID	80.19±1.33	86.40±1.48	54.98±0.92	62.52±0.72	47.35±0.71	50.65±0.61
FedProx	0.1	65.34±4.61	73.29±1.13	48.42±1.67	57.34±1.50	42.87±1.03	47.49±1.03
	0.6	79.48±2.58	86.79±2.09	55.13±1.22	62.45±1.14	46.97±0.49	50.94±0.65
	IID	81.88±0.39	88.63±0.77	55.71±0.32	62.97±0.51	47.22±0.17	51.15±0.21
MOON	0.1	61.57±5.80	71.55±2.42	48.84±1.78	57.11±1.76	44.42±0.88	47.36±1.13
	0.6	77.85±3.09	85.12±2.61	54.27±0.89	62.33±1.12	47.21±0.88	51.48±0.68
	IID	79.35±0.29	88.98±0.79	54.77±0.17	63.91±0.11	48.06±0.19	51.98±0.10
FedDyn	0.1	64.42±2.66	76.43±2.25	49.97±1.09	59.37±1.29	32.60±0.43	42.55±0.81
	0.6	81.43±1.83	88.29±1.77	55.39±0.96	63.37±1.14	44.32±1.35	45.04±1.38
	IID	81.99±0.93	90.14±0.32	55.90±1.06	63.79±0.46	45.17±0.59	45.63±0.42
FedGen	0.1	66.79±5.12	74.13±3.31	47.86±1.86	57.20±1.95	43.53±1.13	47.51±1.42
	0.6	78.69±1.20	86.49±1.52	55.14±1.27	61.71±1.50	48.59±1.39	51.37±1.56
	IID	80.33±0.27	87.53±0.87	56.73±0.15	63.05±0.13	49.81±0.41	51.83±0.51
FedFTG	0.1	64.04±4.14	72.24±2.29	46.52±1.51	59.29±1.57	41.62±1.34	47.64±1.40
	0.6	79.65±1.91	86.59±1.83	56.15±0.89	63.54±0.50	47.77±0.72	51.87±0.97
	IID	82.08±0.08	88.41±0.26	57.07±0.37	63.79±0.24	49.23±0.38	52.55±0.42
FedLMG	0.1	65.59±5.79	76.25±2.31	50.54±0.87	60.21±0.45	41.33±1.58	48.49±1.56
	0.6	78.11±1.39	86.72±1.49	56.13±1.22	63.49±0.72	49.27±0.43	51.29±0.79
	IID	83.13±0.14	88.79±0.41	56.89±0.28	64.08±0.27	50.32±0.41	51.78±0.39
FedProto	0.1	66.08±5.42	70.31±1.91	43.33±0.99	52.88±1.17	41.00±1.27	47.59±1.30
	0.6	77.39±1.16	84.43±1.28	53.43±2.17	59.67±1.88	48.62±0.85	50.27±1.00
	IID	78.44±0.68	87.51±0.97	54.16±1.03	60.13±0.88	49.09±0.31	50.46±0.36
FedTGP	0.1	66.82±3.61	73.47±1.45	49.04±0.74	60.83±0.94	39.58±2.47	47.55±0.71
	0.6	79.09±1.38	86.57±1.35	55.91±0.52	63.59±0.74	48.65±0.71	51.37±0.84
	IID	81.28±0.23	88.64±0.32	56.49±0.22	64.10±0.24	49.21±0.14	51.86±0.17
FedCPD	0.1	63.79±5.01	72.92±3.04	48.10±0.90	57.00±1.28	43.08±0.89	47.83±1.22
	0.6	79.45±1.53	86.65±0.91	56.15±0.84	62.05±0.75	47.59±0.82	51.46±1.21
	IID	80.17±0.55	87.43±0.49	56.43±0.31	63.41±0.28	48.97±0.23	51.93±0.41
FedRG	0.1	65.73±2.31	75.47±1.84	49.05±0.74	59.40±0.58	43.52±0.73	47.69±0.58
	0.6	79.51±1.53	87.32±0.82	56.02±0.33	62.49±0.62	48.86±0.76	50.91±0.57
	IID	81.69±0.48	88.41±0.24	56.77±0.21	63.08±0.35	49.19±0.54	51.13±0.47
FedTau	0.1	**79.14**±2.05	**84.72**±1.23	**53.79**±1.11	**63.33**±0.68	**47.89**±1.33	**51.54**±0.81
	Δ	↑**12.32**	↑**8.29**	↑**3.25**	↑**2.50**	↑**3.47**	↑**3.05**
	0.6	**85.65**±1.21	**90.94**±0.79	**58.17**±1.08	**64.84**±0.62	**51.23**±0.93	**53.67**±0.36
	Δ	↑**4.22**	↑**2.65**	↑**2.02**	↑**1.25**	↑**1.96**	↑**1.80**
	IID	**87.00**±1.35	**92.37**±0.41	**58.91**±0.79	**65.27**±0.28	**51.99**±0.63	**53.97**±0.45
	Δ	↑**3.87**	↑**2.23**	↑**1.84**	↑**1.17**	↑**1.67**	↑**1.42**

and checkpoint diagnostics are shown in Appendices A.7 and A.8. All experiments are implemented in PyTorch. The hardware platform consists of two Intel Xeon Gold 6548Y+ CPUs and an NVIDIA RTX 5880 Ada Generation GPU. More experiments can be found in Appendices A.6–A.8.

5.2 Effectiveness Across Three Tasks

We evaluate FedTau's τ-correction principle across three response types (classification, regression, and sequence prediction). FedTau corrects the local supervision according to the client observation mechanism while keeping model averaging and reference-task inference unchanged.

Classification under heterogeneous data distribution (non-IID). Table 1 reports the highest mean accuracy for FedTau in all 18 dataset-partition-epoch settings. At $\alpha = 0.1$ and $E = 1$, FedTau reaches 79.14%, 53.79%, and 47.89% on CIFAR-10, CIFAR-100, and TinyImageNet, exceeding the strongest respective baselines by 12.32%, 3.25%, and 3.47%. The corresponding gains over FedAvg are 17.08%, 6.18%, and 4.80%. This demonstrates the superiority of FedTau across different levels of non-IID and data complexities. The same performance trend persists with other settings (e.g.,

Table 2: Regression under heterogeneous right censoring. Best/second-best results are bold/boxed. Δ reports the RMSE reduction or R^2 increase.

Method	California Housing		Superconductivity		Bike Sharing	
	RMSE↓ $\times 10^{-2}$	R^2 ↑ $\times 10^{-2}$	RMSE↓ $\times 10^{0}$	R^2 ↑ $\times 10^{-2}$	RMSE↓ $\times 10^{0}$	R^2 ↑ $\times 10^{-2}$
FedAvg	78.58±0.91	54.82±1.05	19.3033±1.3273	68.03±4.40	117.1295±3.8439	58.99±2.69
FedNova	78.50±0.72	55.00±0.83	19.3033±1.4188	68.03±4.70	117.0783±3.7508	59.04±2.63
MOON	78.30±0.57	55.20±0.65	19.2689±1.3224	68.14±4.37	116.9834±2.6003	59.09±1.82
FedDAAW	78.10±0.63	55.42±0.72	19.2267±1.3713	68.28±4.52	116.7718±3.0736	59.24±2.15
FedProx	77.50±0.85	56.09±0.96	19.5166±1.1630	67.29±3.90	117.2855±2.6488	58.89±1.86
FedAdagrad	77.00±0.25	56.67±0.28	19.0122±1.1491	68.99±3.75	115.7164±3.8825	59.97±2.69
SCAFFOLD	76.50±0.11	57.22±0.12	18.9456±2.4857	69.20±8.08	115.5793±5.0637	60.07±3.50
FedAvgM	76.20±0.37	57.56±0.41	18.7186±1.1163	69.94±3.59	113.9367±3.6418	61.20±2.48
FedDyn	75.50±0.29	58.34±0.32	18.6602±1.9943	70.12±6.39	113.6863±2.7325	61.37±1.86
FedAdam	74.80±0.57	59.11±0.62	18.4212±2.4793	70.89±7.84	112.3403±3.7662	62.28±2.53
FedYogi	74.40±0.05	59.53±0.05	18.3792±1.1539	71.02±3.64	112.1450±2.8385	62.41±1.90
FedLSBT	73.00±0.08	61.05±0.09	18.6179±1.5529	70.26±4.96	113.8025±2.3751	61.29±1.62
FedTau	**61.98**±0.04	**71.98**±0.04	**17.1534**±0.8736	**77.46**±2.30	**69.9637**±2.0160	**86.13**±0.80
Δ	↓11.02	↑10.93	↓1.2258	↑6.44	↓42.1813	↑23.72

$\alpha = 0.1, 0.6$, and IID, and $E = 3$). The complete results from total of 54 different settings, including more local training epochs $E = 5$ and more training models (StageCNN), are shown in Appendix A.6. These results show that FedTau can correct client-specific class-prior mismatch while still benefiting from additional local training. Note that random IID partitioning may still produce unequal empirical class priors/distributions. Thus, FedTau still works in IID settings. In the exact identity-mechanism case (client observation is identical to reference task), the FedTau correction can be zero. Details are in Appendix A.5.1.

Regression with heterogeneous reporting limits. Table 2 tests continuous responses using a censored likelihood. FedTau has the lowest RMSE on all three datasets. Against the strongest baselines, RMSE decreases from 0.7300 to 0.6198 on California Housing, from 18.3792 to 17.1534 on Superconductivity, and from 112.1450 to 69.9637 on Bike Sharing (relative reductions of 15.10%, 6.67%, and 37.61%). The respective R^2 increases by 0.1093, 0.0644, and 0.2372. The results verify the target-mismatch argument in Eq. (2): conventional regression treats a censored response as if its true value were exactly c_k, whereas FedTau uses the correct information that the true value is at least c_k. The comparison with incorrect thresholds tests the mechanism specification in Table 12 in Appendix. MAE results, imputation controls, selection, and interval quantization are in Appendix A.7.

Sequence prediction with incomplete targets. Table 3 compares variable-length outputs in news and dialogue summarization. In sequence prediction, the observation mechanism affects whether a recorded boundary is interpreted as true termination or truncation. FedTau replaces the artificial EOS supervision at a truncated boundary with a continuation event, without imputing the missing suffix or modifying the encoder–decoder. Statistically, FedTau has the lowest NLL and the highest token accuracy on all three datasets. On XSum, NLL decreases from the strongest baseline value of 2.4409 to 2.3836 (2.35%), while accuracy improves from 50.62% to 51.43% ($+0.81\%$). The corresponding improvements are 1.10% lower NLL and $+0.57\%$ accuracy on SAMSum, and 0.38% lower NLL and $+0.09\%$ accuracy on CNN/DailyMail. Both metrics consistently support the effectiveness of the FedTau correction.

Among all three ML tasks, FedTau provides a consistent performance improvement across categorical, continuous, and autoregressive reference predictions. Appendix Tables 16–19 separately evaluate selected checkpoints, fixed endpoints, and generated summaries.

5.3 Ablation Study

Contribution of the local and reference terms. Table 4 isolates the components of the classification correction while keeping the data, local training budget, and aggregation fixed. For the CIFAR-10 ablation, we use a non-uniform pooled reference with imbalance factor $50, 100$ clients, 10% participation, and five local minibatch updates per round. Full FedTau improves top-1 accuracy

Table 3: Sequence-to-sequence results. Entries are the mean ± sample standard deviation of the per-seed best test scores across five independent random seeds. Each run completes 50 communication rounds. Final evaluation follows the per-seed reporting protocol described in Appendix A.8.

	XSum		SAMSum		CNN/DailyMail	
Method	NLL↓	Acc.↑	NLL↓	Acc.↑	NLL↓	Acc.↑
FedAvg	2.4551±0.0006	49.02±0.04	1.7301±0.0025	59.78±0.08	2.0187±0.0012	60.20±0.07
FedProx	2.4553±0.0006	49.03±0.04	1.7301±0.0030	59.80±0.06	2.0185±0.0008	60.21±0.04
FedAvgM	2.4534±0.0003	49.06±0.02	1.7302±0.0012	59.74±0.04	2.0539±0.0007	60.19±0.03
FedAdam	2.4617±0.0003	49.03±0.02	1.7296±0.0010	59.83±0.05	2.0255±0.0006	60.02±0.05
FedDyn	2.4418±0.0014	50.48±0.04	1.7348±0.0018	59.58±0.06	2.0359±0.0005	60.10±0.02
FedDC	2.4409±0.0005	50.51±0.01	1.7350±0.0006	59.60±0.04	2.0531±0.0006	60.13±0.04
FedExP	2.4412±0.0008	50.49±0.03	1.7348±0.0010	59.57±0.07	2.0187±0.0009	59.88±0.03
FedSpeed	2.4410±0.0006	50.62±0.05	1.7350±0.0008	59.70±0.07	2.2274±0.0005	54.87±0.06
FedLESAM	2.4412±0.0010	50.57±0.06	1.7349±0.0005	59.68±0.04	2.0208±0.0007	59.90±0.04
FedTau	**2.3836**±0.0008	**51.43**±0.02	**1.7105**±0.0014	**60.40**±0.05	**2.0109**±0.0010	**60.30**±0.05

Table 4: Ablation study on CIFAR-10 at 200 round budget. Δ denotes the top-1 accuracy difference from the no-mechanism baseline. ψ is a fixed random permutation of client indices. All corrections are applied inside the local objective except the post-local variant. Complete metrics and longer-horizon results are reported in Appendix A.9.

Variant	Applied correction	Top-1 (%) ↑	Δ (p.p.)	NLL ↓
No mechanism (FedAvg)	0	46.07	0.00	1.4860
Local term only	$\log \widetilde{\pi}_k$	57.47	+11.39	1.3470
Reference term only	$-\log \widetilde{\pi}_*$	42.69	-3.38	1.7277
Shuffled client mechanism	$\log \widetilde{\pi}_{\psi(k)} - \log \widetilde{\pi}_*$	46.64	+0.56	1.6115
Reverse ratio	$\log \widetilde{\pi}_* - \log \widetilde{\pi}_k$	40.41	-5.67	1.8478
Post-local bias correction	$-b_k$ after local training	20.72	-25.35	14.3658
FedTau (full)	$\log \widetilde{\pi}_k - \log \widetilde{\pi}_*$	**59.95**	**+13.88**	**1.2261**

from 46.07% to 59.95% ($+13.88\%$) and reduces NLL from 1.4860 to 1.2261 (17.49%). Using only the local term reaches 57.47%, showing that it accounts for most of the improvement. Adding the reference term further improves accuracy by 2.48% and reduces NLL by 8.98%. This indicates that the pooled reference prior also contributes to prediction under the non-uniform target.

Correct mechanism specification matters. The benefit does not arise from adding an arbitrary logit offset. Shuffling client mechanisms reduces accuracy to 46.64%, close to the no-mechanism result, while reversing the prior ratio further reduces it to 40.41%. These results show FedTau requires the correction to match each client's observation mechanism and use the correct correction direction.

Correction placement matters. Applying the correction only after local training is not equivalent to incorporating it directly into the local objective. The post-local bias correction achieves only 20.72% accuracy and substantially increases NLL, showing that correcting the classifier bias after local training does not reproduce the effect of correcting local supervision during optimization. Additional longer training results, reporting windows, macro accuracy, and mechanism diagnostics are provided in Appendix A.9.

6 Conclusion

In this work, we identify *observation-induced target mismatch* as an earlier source of heterogeneity in FL: client-specific sampling and recording mechanisms can make local supervision encode different prediction targets even when clients share the same underlying task. To address this issue, we propose FedTau, a unified τ-correction that accounts for client-specific observation mechanisms where the mismatch first occurs. We show that the corrected client objectives can share a common predictive optimum and analyze the effect of mechanism-estimation errors. Experiments on nine datasets against twenty-three baselines demonstrate consistent improvements across classification, regression, and sequence prediction.

AI USE STATEMENT

This paper only uses LLMs to polish text and not for any other purpose.

REFERENCES

Durmus Alp Emre Acar, Yue Zhao, Ramon Matas, Matthew Mattina, Paul Whatmough, and Venkatesh Saligrama. Federated learning based on dynamic regularization. In *International Conference on Learning Representations*, 2021.

Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for privacy-preserving machine learning. In *proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1175–1191, 2017.

Hong-You Chen and Wei-Lun Chao. On bridging generic and personalized federated learning for image classification. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=I1hQbx10Kxn.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *Journal of machine learning research*, 25(70):1–53, 2024.

Ziqing Fan, Shengchao Hu, Jiangchao Yao, Gang Niu, Ya Zhang, Masashi Sugiyama, and Yanfeng Wang. Locally estimated global perturbations are better than local perturbations for federated sharpness-aware minimization. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 12858–12881. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/fan24c.html.

Hadi Fanaee-T and Joao Gama. Event labeling combining ensemble detectors and background knowledge. *Progress in Artificial Intelligence*, 2(2):113–127, 2014.

Liang Gao, Huazhu Fu, Li Li, Yingwen Chen, Ming Xu, and Cheng-Zhong Xu. Feddc: Federated learning with non-iid data via local drift decoupling and correction. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 10102–10111. IEEE, 2022.

Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. Samsum corpus: A human-annotated dialogue dataset for abstractive summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pp. 70–79, 2019.

Kam Hamidieh. A data-driven statistical model for predicting the critical temperature of a superconductor. *Computational Materials Science*, 154:346–354, 2018.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Zaobo He, Yusen Li, Daehee Seo, and Zhipeng Cai. Fedcpd: Addressing label distribution skew in federated learning with class proxy decoupling and proxy regularization. *Information Fusion*, 110:102481, 2024.

James J Heckman. Sample selection bias as a specification error. *Econometrica: Journal of the econometric society*, pp. 153–161, 1979.

Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335*, 2019.

Divyansh Jhunjhunwala, Shiqiang Wang, and Gauri Joshi. Fedexp: Speeding up federated averaging via extrapolation. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=IPrzNbddXV.

Peter Kairouz and H Brendan McMahan. Advances and open problems in federated learning. *Foundations and trends in machine learning*, 14(1-2):1–210, 2021.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International conference on machine learning*, pp. 5132–5143. PMLR, 2020.

Jared L Katzman, Uri Shaham, Alexander Cloninger, Jonathan Bates, Tingting Jiang, and Yuval Kluger. Deepsurv: personalized treatment recommender system using a cox proportional hazards deep neural network. *BMC medical research methodology*, 18(1):24, 2018.

Alex Krizhevsky. Learning multiple layers of features from tiny images. pp. 32–33, 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

Yann Le, Xuan Yang, et al. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10708–10717. IEEE, 2021.

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.

Bill Yuchen Lin, Chaoyang He, Zihang Ze, Hulin Wang, Yufen Hua, Christophe Dupuy, Rahul Gupta, Mahdi Soltanolkotabi, Xiang Ren, and Salman Avestimehr. Fednlp: Benchmarking federated learning methods for natural language processing tasks. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pp. 157–175, 2022.

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.

Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment, 2021. URL https://arxiv.org/abs/2007.07314.

Shashi Narayan, Shay B Cohen, and Mirella Lapata. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 1797–1807, 2018.

R Kelley Pace and Ronald Barry. Sparse spatial autoregressions. *Statistics & Probability Letters*, 33 (3):291–297, 1997.

Mattia Passeri, Andrea Agiollo, and Andrea Omicini. Peer-reviewed federated learning. In *CEUR WORKSHOP PROCEEDINGS*, volume 3579, pp. 49–65. CEUR-WS, 2023.

Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1944–1952, 2017.

Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečný, Sanjiv Kumar, and H. Brendan McMahan. Adaptive federated optimization. In *International Conference on Learning Representations*, 2021.

Jiawei Ren, Cunjun Yu, Xiao Ma, Haiyu Zhao, Shuai Yi, et al. Balanced meta-softmax for long-tailed visual recognition. *Advances in neural information processing systems*, 33:4175–4186, 2020.

Abigail See, Peter J Liu, and Christopher D Manning. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1073–1083, 2017.

Yan Sun, Li Shen, Tiansheng Huang, Liang Ding, and Dacheng Tao. Fedspeed: Larger local interval, less communication round, and higher generalization accuracy. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=bZjxxYURKT.

Yuki Takezawa, Anastasia Koloskova, Xiaowen Jiang, and Sebastian Stich. Fedmuon: Federated learning with bias-corrected lmo-based optimization. In *International Conference on Learning Representations*, volume 2026, pp. 39538–39569, 2026.

Yue Tan, Guodong Long, Lu Liu, Tianyi Zhou, Qinghua Lu, Jing Jiang, and Chengqi Zhang. Fedproto: Federated prototype learning across heterogeneous clients. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 8432–8440, 2022.

Jianqing Tang. FedDAAW: Dynamic client selection and accuracy-adaptive aggregation for cross-task federated learning under heterogeneity, 2026. Research Square preprint.

James Tobin. Estimation of relationships for limited dependent variables. *Econometrica: journal of the Econometric Society*, pp. 24–36, 1958.

Bruce W Turnbull. The empirical distribution function with arbitrarily grouped, censored and truncated data. *Journal of the Royal Statistical Society: Series B (Methodological)*, 38(3):290–295, 1976.

Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization. *Advances in neural information processing systems*, 33:7611–7623, 2020.

Tian Wen, Zhiqin Yang, Yonggang Zhang, Xuefeng Jiang, Hao Peng, Yuwei Wang, and Bo Han. Fedrg: Unleashing the representation geometry for federated learning with noisy clients, 2026. URL https://arxiv.org/abs/2603.19722.

Zhiyuan Wu, Changkyu Choi, Xiangcheng Cao, Volkan Cevher, and Ali Ramezani-Kebrya. Addressing label shift in distributed learning via entropy regularization. In *International Conference on Learning Representations*, volume 2025, pp. 51961–51996, 2025.

Mingzhao Yang, Shangchao Su, Bin Li, and Xiangyang Xue. One-shot heterogeneous federated learning with local model-guided diffusion models. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=PqJFVbJAMR.

Jianqing Zhang, Yang Liu, Yang Hua, and Jian Cao. Fedtgp: Trainable global prototypes with adaptive-margin-enhanced contrastive learning for data and model heterogeneity in federated learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 16768–16776, 2024.

Jie Zhang, Zhiqi Li, Bo Li, Jianghe Xu, Shuang Wu, Shouhong Ding, and Chao Wu. Federated learning with label distribution skew via logits calibration. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 26311–26329. PMLR, 17–23 Jul 2022a. URL https://proceedings.mlr.press/v162/zhang22p.html.

Lin Zhang, Li Shen, Liang Ding, Dacheng Tao, and Ling-Yu Duan. Fine-tuning global model via data-free knowledge distillation for non-iid federated learning. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10164–10173. IEEE, 2022b.

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

Zhuangdi Zhu, Junyuan Hong, and Jiayu Zhou. Data-free knowledge distillation for heterogeneous federated learning. In *International conference on machine learning*, pp. 12878–12889. PMLR, 2021.

A APPENDIX

Appendix Contents

A.1 Notation . . . **14**
A.2 Algorithmic procedures . . . **16**
A.3 Communication and computational overhead . . . **17**
A.3.1 Profiling protocol. . . . 17
A.3.2 Measurement. . . . 18
A.3.3 Communication. . . . 18
A.3.4 Computation. . . . 19
A.4 Information requirements and privacy . . . **19**
A.5 Methodology details . . . **20**
A.5.1 Common predictive target and identification . . . 20
A.5.2 Classification: prior correction and its gradients . . . 21
A.5.3 Regression and sequence observation likelihoods . . . 22
A.5.4 Mechanism error and federated local updates . . . 23
A.5.5 Loss normalization and empirical approximation . . . 24
A.6 Classification results . . . **24**
A.6.1 Experimental design . . . 24
A.6.2 Baseline rationale and comparison protocol . . . 26
A.6.3 Classification performance . . . 27
A.6.4 Robustness to heterogeneity and local epochs . . . 27
A.7 Regression results . . . **28**
A.7.1 Evaluation objective and protocol . . . 28
A.7.2 Data splits, target coordinates, and local training . . . 37
A.7.3 Reference-response metrics. . . . 38
A.7.4 Baseline selection and controlled comparison . . . 39
A.7.5 Comparison control. . . . 39
A.7.6 Primary comparison: heterogeneous right censoring . . . 40
A.7.7 Outcome-dependent selection and inverse-propensity weighting . . . 43
A.7.8 Symmetric interval quantization . . . 46
A.7.9 Joint interpretation. . . . 48
A.8 NLP sequence-to-sequence results . . . **49**
A.8.1 Evaluation objective. . . . 49
A.8.2 Datasets and experimental setup. . . . 49
A.8.3 Sequence-specific implementation of τ. . . . 50
A.8.4 Baseline selection. . . . 50
A.8.5 Why classification-specific methods are not direct substitutes. . . . 50
A.8.6 Controlled comparison and evaluation. . . . 50
A.8.7 Aggregate prediction results. . . . 51
A.8.8 Single-run SAMSum: improvements in prediction and generation. . . . 52
A.8.9 Single-run SAMSum: the benefit persists at the fixed endpoint. . . . 53
A.8.10 Single-run CNN/DailyMail: consistent reference-prediction gains. . . . 53
A.8.11 Joint interpretation. . . . 54
A.9 Ablation study . . . **54**

A.1 NOTATION

Tables 5 and 6 summarize the notation used in the method, its analysis, and the procedures below. A superscript obs refers to the recorded observation, whereas $*$ refers to the reference prediction task. The mechanism parameters are fixed during local optimization. The correction τ_k is a scalar difference between the observed-data loss and uncorrected supervision. Its classification implementation uses the fixed logit vector b_k; the scalar correction can depend on the current prediction in all three tasks.

Table 5: Common notation and quantities used in the theoretical analysis.

Symbol	Definition
K, k	Number of clients and client index.
D_k, n_k	Observed training records held by client k and their number.
N	Total number of training records, $\sum_k n_k$.
x, y, o	Input, reference response, and recorded observation; y need not be fully observed.
$p_*(y \mid x)$	Reference conditional response distribution.
p_k^X, p_k^{obs}	Client input marginal and conditional distribution of recorded observations.
$q_\theta(y \mid x)$	Shared predictive model, parameterized by θ.
$\mathcal{T}_k$, $\widehat{\mathcal{T}}_k$	True observation mechanism and its protocol- or training-derived implementation.
$q_{k,\theta}^{\mathrm{obs}}$	Predicted local observation distribution: $\mathcal{T}_k[q_\theta]$ in theory, or $\widehat{\mathcal{T}}_k[q_\theta]$ in implementation.
ℓ_k^0, τ_k	Uncorrected record loss and scalar correction to the observed-data negative log-likelihood.
s_k, R_k, A	Sampling weight, conditional recording distribution, and measurable set of observations in Appendix A.5.1.
$\widehat{F}_k$, F_k, G_k	Empirical local risk; population risk with the true mechanism; population risk with its fixed estimate.
F, w_k	Weighted population risk $F = \sum_k w_k F_k$, with positive weights summing to one.
R, t, S_t	Communication horizon, round index, and participating clients at round t.
θ_t, $\theta_{k,t}$	Broadcast global parameters and the parameters returned by client k in round t.
$w_{k,t}$	Selected-cohort aggregation weight $n_k / \sum_{j \in S_t} n_j$.
$\mathcal{B}$, E, H_k	Local minibatch, local epoch count, and number of local minibatch updates.
$\mathcal{L}_k$	Task-specific minibatch objective returned to the local optimizer.
η	Gradient-descent step size in the local-update analysis.
θ_*	A parameter representing the reference prediction, $q_{\theta_*} = p_*$.
$\bar{p}^X$	Mixture of client input marginals, $\sum_k w_k p_k^X$.
β, m	Gradient-smoothness constant and positive lower Hessian bound in the conditional contraction result.
ε_k	Residual gradient norm $\|\nabla G_k(\theta_*)\|_2$ with the estimated mechanism.
ρ	Maximum inverse-prior weight ratio in Eq. (18).
$J_\theta(x)$, M	Logit Jacobian and its assumed operator-norm bound.
ϵ_F, ξ	Uniform empirical–population risk discrepancy and empirical optimization error in Eq. (29).

Table 6: Task-specific notation. Thresholds, interval widths, and likelihood scales use the same response coordinates as their local loss.

Symbol	Definition
Classification	
C, c	Number of classes and class index.
α	Dirichlet concentration controlling the experimental class-wise allocation.
$n_{k,c}$, N_c	Local and pooled training counts for class c; $N_c = \sum_k n_{k,c}$.
π_k, π_*	Population local and reference class priors.
$\widetilde{\pi}_k$, $\widetilde{\pi}_*$	Add-one-smoothed training estimates of the two priors.
$z_\theta(x)$	Uncorrected classifier logits, with $q_\theta = \mathrm{softmax}(z_\theta)$.
b_k	Cached logit offset $\log \widetilde{\pi}_k - \log \widetilde{\pi}_*$.
Regression	
g, m_y, s_y	Fixed target transformation and its training mean and standard deviation.
r, v	Standardized target $r = (g(y) - m_y)/s_y$ and, in the censoring experiment, its noise-perturbed value v.
u, c_k, δ	Recorded capped response, client threshold, and censoring flag; $\delta = 1$ indicates a bound rather than an exact response.
$\mu_\theta(x)$, σ	Predicted mean and fixed Gaussian working scale in model coordinates.
Φ, ϕ	Standard normal cumulative distribution and density functions.
I_i, m_i, h_k, d_k	Recorded interval, midpoint, bin width, and grid origin in the quantization experiment.
$a_k(r_i)$, Z_k	Assigned retention probability and mean assigned propensity in the nominal client population.
n_k^0, w_i	Pre-selection record count and the per-record IPW weight; w_i is distinct from a client aggregation weight.
Sequence-to-sequence learning	
T_i, L_i	Full target length and recorded prefix length, excluding EOS.
$y_{i,1:L_i}$, $y_{i,<j}$	Recorded target tokens and the teacher-forcing context before position j.
δ_i	Continuation flag: $\delta_i = 1$ certifies $T_i > L_i$; complete records have $\delta_i = 0$.
EOS, p_{EOS}	End-of-sequence token and its probability after the observed prefix.
κ	Fixed training-wide mean observed-event count, $N^{-1} \sum_i (L_i + 1)$.

A.2 ALGORITHMIC PROCEDURES

Algorithm 1 specifies the shared federated loop. Algorithms 2–4 specify its task-dependent preparation, minibatch objective, and inference rule. LOCALOPT denotes the configured local optimizer, including its state, regularization, and gradient-clipping rule; it does not assume that every task uses plain gradient descent. Only the model is optimized. Fixed observation-map parameters are not trainable, but gradients pass through the predicted observation probabilities.

Algorithm 1: FedTau — shared federated loop

Input: Observed client datasets $\{D_k\}_{k=1}^{K}$; initial model θ_0; R rounds; participation and local-update budgets; one task procedure below.

Output: Reference predictor q_{θ_R}.

Establish the reference task and run the task-specific preparation once using training information and recording metadata.
Cache the fixed mechanism parameters and any shared reference statistics.
for $t = 0, \dots, R-1$ **do**
  Select a nonempty client set S_t and broadcast θ_t.
  for each $k \in S_t$ **in parallel do**
    Set the local parameters $\theta \leftarrow \theta_t$ and initialize the optimizer as specified by the task configuration.
    for each minibatch update in the local budget **do**
      Draw $\mathcal{B}$ from the observed local records and compute the task-specific loss $\mathcal{L}_k(\theta; \mathcal{B})$.
      Update $\theta \leftarrow \text{LOCALOPT}(\theta, \nabla_\theta \mathcal{L}_k)$, holding the observation-mechanism parameters fixed.
    end for; return $\theta_{k,t} \leftarrow \theta$ and its aggregation count.
  end for
  Aggregate $\theta_{t+1} \leftarrow \sum_{k \in S_t} \frac{n_k}{\sum_{j \in S_t} n_j} \theta_{k,t}$.
end for; use the task-specific inference rule with θ_R.

The model-averaging expression follows Eq. (8); normalization buffers follow the corresponding implementation's state aggregation policy. In the IPW experiment only, the aggregation count is n_k^0, not the number of retained records. The local budget denotes E passes over the client data in the classification epoch sweep and the specified minibatch-update count in the regression and sequence experiments.

Algorithm 2: Classification with a client-prior mechanism

Input: Labeled records (x_i, y_i), C classes, and the shared classifier z_θ.

Preparation: Count $n_{k,c} = \sum_{i \in D_k} \mathbf{1}\{y_i = c\}$ and form $N_c = \sum_k n_{k,c}$, $N = \sum_c N_c$.
Compute $\widetilde{\pi}_k(c) = (n_{k,c} + 1)/(n_k + C)$ and $\widetilde{\pi}_*(c) = (N_c + 1)/(N + C)$.
Cache $b_{k,c} = \log \widetilde{\pi}_k(c) - \log \widetilde{\pi}_*(c)$. Subtracting its class-wise mean is optional and leaves softmax unchanged.
Minibatch loss: For every $(x_i, y_i) \in \mathcal{B}$, evaluate $z_\theta(x_i) + b_k$.
Return $\mathcal{L}_k = \dfrac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} \mathrm{CE}\big(z_\theta(x_i) + b_k, y_i\big)$.
Use this loss in Algorithm 1; do not apply a post-local classifier-bias correction.
Inference: Predict $\arg\max_c z_{\theta_R, c}(x)$ using uncorrected logits.

Algorithm 3: Regression with recorded response events

Input: Recorded responses and mechanism metadata in model coordinates; fixed target transform (g, m_y, s_y); regressor μ_θ and working scale $\sigma > 0$.

1 **Preparation:** Fix preprocessing and the recording mechanism before local optimization. Thresholds and intervals are expressed in the same coordinate as $\mu_\theta(x)$.
2 **Minibatch loss, right censoring:** For each record (x_i, u_i, δ_i), set $\mu_i \leftarrow \mu_\theta(x_i)$.
3 **if** $\delta_i = 0$, set $\ell_i \leftarrow (u_i - \mu_i)^2/(2\sigma^2) + \log\sigma$.
4 **else**, set $\ell_i \leftarrow -\log\Phi((\mu_i - c_k)/\sigma)$ using a stable log-CDF.
5 Return $\mathcal{L}_k = |\mathcal{B}|^{-1}\sum_{i\in\mathcal{B}}\ell_i$ and use the shared loop.
6 **Interval alternative:** For $I_i = [m_i - h_k/2, m_i + h_k/2]$, replace ℓ_i by
$$-\log\left[\Phi\left(\frac{m_i + h_k/2 - \mu_i}{\sigma}\right) - \Phi\left(\frac{m_i - h_k/2 - \mu_i}{\sigma}\right)\right],$$
7 evaluated using a stable log-CDF difference; average these record losses.
8 **Selection alternative:** For retained exact targets r_i and their assigned propensities, set $w_i \leftarrow \min\{20, Z_k/a_k(r_i)\}$.
9 Return $\mathcal{L}_k = \dfrac{\sum_{i\in\mathcal{B}} w_i(\mu_\theta(x_i) - r_i)^2}{\sum_{i\in\mathcal{B}} w_i}$ and use n_k^0 for aggregation.
10 **Inference:** Return $\hat{y} = g^{-1}(m_y + s_y\mu_{\theta_R}(x))$, with the evaluation-domain clipping specified in Appendix A.7.2.

The right-censoring and interval branches use observation likelihoods. The selection branch is the self-normalized IPW estimating objective in Eq. (42), rather than a likelihood substitution. The recorded (u_i, δ_i) are the outputs of the experiment's noise-and-censoring construction; neither branch supplies the unobserved response to local training.

Algorithm 4: Sequence learning with a censored-prefix mechanism

Input: Source x_i, observed non-EOS prefix $y_{i,1:L_i}$, continuation flag δ_i, and an autoregressive encoder–decoder q_θ.

1 **Preparation:** Fix each client's recording cap and retain its observed prefixes and continuation flags. Complete targets retain their genuine EOS event.
2 Compute and cache $\kappa = N^{-1}\sum_{i=1}^{N}(L_i + 1)$ from training records. Use the same value on every client.
3 **Minibatch loss:** Right-pad records and apply teacher forcing. A terminal EOS marker aligns the boundary position; for a censored record it is not an EOS target.
4 For each record, compute $\ell_i^{\text{prefix}} \leftarrow -\sum_{j=1}^{L_i}\log q_\theta(y_{i,j} \mid x_i, y_{i,<j})$, excluding padding.
5 Let $z_{i,v}$ be the boundary logit for vocabulary token v after $y_{i,1:L_i}$.
6 **if** $\delta_i = 0$, set $\ell_i^{\text{boundary}} \leftarrow -\log q_\theta(\text{EOS} \mid x_i, y_{i,1:L_i})$.
7 **else**, set $\ell_i^{\text{boundary}} \leftarrow \log\sum_v e^{z_{i,v}} - \log\sum_{v\neq\text{EOS}} e^{z_{i,v}}$.
8 Return $\mathcal{L}_k = \dfrac{1}{|\mathcal{B}|\kappa}\sum_{i\in\mathcal{B}}(\ell_i^{\text{prefix}} + \ell_i^{\text{boundary}})$.
9 Use this loss in Algorithm 1; optimize no hidden suffix tokens.
10 **Inference:** Generate from the unmodified shared distribution q_{θ_R} under the common decoding configuration, without a client recording cap or boundary correction.

Algorithm 4 specifies the fixed-scale `record` normalization in Eq. (28). The two boundary branches are evaluated directly, rather than by subtracting a large artificial-EOS loss from a large total loss. For a censored record, the boundary term equals $-\log(1 - p_{\text{EOS}})$ and is differentiated through the current logits.

A.3 Communication and computational overhead

A.3.1 Profiling protocol.

We profile seven reference implementations with a shared 6,506-parameter CNN, 32-dimensional features, and ten classes. The workload comprises 640 shape-controlled $3 \times 32 \times 32$ random tensors, divided equally among ten clients by IID assignment. Each of three communication rounds selects two clients, each performing two SGD updates with batch size 8, learning rate 0.01, momentum 0.9, and weight decay 5×10^{-4}. The initial model, partition, and participation schedule match across

methods, as verified by their recorded hashes. FedProto and FedTGP additionally extract prototypes over the complete local dataset; FedTGP uses two server-training epochs, and FedFTG uses one generator update and one distillation update per round. We use FP32 on an NVIDIA RTX 5880 Ada Generation GPU with PyTorch 2.2.0 and CUDA 12.1. This lightweight workload measures implementation costs separately from the main-backbone accuracy experiments.

A.3.2 MEASUREMENT.

Application payload counts include each recipient's uncompressed tensor values and aggregation metadata, excluding transport headers and the preloaded common model initialization. Training time sums the two participating clients' execution times and the server's execution time, including forward passes, backpropagation, and optimizer updates. CUDA is synchronized at timing boundaries; two backend warm-up updates are excluded. For each per-round quantity, we report its arithmetic mean and minimum–maximum range over the three rounds of the seed-0 profile. One-time setup is reported separately.

Table 7: Application payload of the profiled implementations. Uplink and downlink are per-round means; round totals show mean [minimum, maximum]. The final column sums all three rounds and one-time setup. 1 KiB $= 1024$ bytes.

Method	**Setup** (bytes)	**Uplink** (KiB/round)	**Downlink** (KiB/round)	**Round total** (KiB)	**Total** (KiB)
FedAvg	0	51.109	51.094	102.203 [102.203, 102.203]	306.609
FedProx	0	51.109	51.094	102.203 [102.203, 102.203]	306.609
FedDyn	0	51.109	51.094	102.203 [102.203, 102.203]	306.609
FedProto	0	2.656	1.771	4.427 [2.656, 5.312]	13.281
FedTGP	0	2.656	1.771	4.427 [2.656, 5.312]	13.281
FedFTG	800	51.109	51.094	102.203 [102.203, 102.203]	307.391
FedTau	1,200	51.109	51.094	102.203 [102.203, 102.203]	307.781

A.3.3 COMMUNICATION.

FedTau transmits the same 102.203 KiB per round as FedAvg, FedProx, and FedDyn. Its only additional payload is a one-time $1{,}200$-byte exchange: 800 bytes of class counts and 400 bytes for distributing the reference prior. Including setup, its total is 307.781 KiB versus 306.609 KiB for FedAvg, an increase of 0.38% over this three-round horizon. The cached correction therefore adds no recurring model payload. FedProto and FedTGP communicate fewer bytes because they exchange prototypes instead of model states; their first round has no global prototype download. These rows represent a different communication interface, not an additional payload on top of FedAvg.

Table 8: Computation of the profiled implementations. Client, server, and total times report per-round mean [minimum, maximum] in milliseconds; setup is a one-time cost. Relative time divides mean total time by FedAvg's, excluding setup. MMAC counts forward convolution and linear-layer multiply–accumulates per round, in millions; it excludes backward and other arithmetic.

Method	**Setup** (ms)	**Client** (ms)	**Server** (ms)	**Total** (ms)	**Relative time**	**Forward** MMAC
FedAvg	0.007	16.73 [8.09, 33.48]	0.76 [0.65, 0.83]	17.50 [8.90, 34.30]	1.000×	51.948
FedProx	0.007	27.23 [9.01, 63.23]	0.74 [0.65, 0.78]	27.96 [9.79, 64.01]	1.598×	51.948
FedDyn	0.007	27.53 [9.50, 63.19]	0.95 [0.84, 1.01]	28.48 [10.49, 64.20]	1.627×	51.948
FedProto	0.008	50.34 [22.38, 79.12]	2.25 [0.36, 6.05]	52.59 [22.74, 85.17]	3.006×	259.738
FedTGP	0.007	51.94 [21.96, 82.31]	80.45 [8.43, 224.18]	132.38 [30.69, 306.49]	7.565×	259.881
FedFTG	0.107	15.79 [7.94, 31.02]	64.46 [10.75, 171.85]	80.25 [18.72, 202.87]	4.586×	1525.747
FedTau	1.131	**15.64** [7.77, 31.10]	**0.67** [0.63, 0.74]	**16.31** [8.40, 31.84]	0.932×	**51.948**

A.3.4 COMPUTATION.

FedTau averages 16.31 ms per round, compared with 17.50 ms for FedAvg (0.932×), while requiring the same 51.948 forward MMAC per round. Its one-time calibration costs 1.131 ms. Including this setup, total profiled computation is 50.059 ms for FedTau and 52.503 ms for FedAvg. Peak allocated GPU memory is also comparable: 18.938 versus 18.933 MiB, a difference of 5 KiB. The mean round times of FedProx, FedDyn, FedProto, FedTGP, and FedFTG are 1.598×, 1.627×, 3.006×, 7.565×, and 4.586× that of FedAvg in this workload. All profiled rounds are retained, including the slower first round; the results characterize this short workload rather than steady-state latency. The direct efficiency result is that FedTau applies the correction without an additional network pass, a learned server module, or recurring metadata exchange, while maintaining computation comparable to ordinary averaging in the profiled configuration.

A.4 INFORMATION REQUIREMENTS AND PRIVACY

Local computation and communication. FedTau changes the local objective in Eq. (3) while retaining the model-averaging interface in Eq. (8). The client computes its correction from local observations and the specified reference statistics. Classification uses local class counts; censored regression uses the recorded response, threshold, and censoring indicator; sequence learning uses the observed prefix and its continuation flag. The selection and quantization extensions similarly use local propensities and recorded intervals. These quantities enter local loss evaluation and need not be transmitted as client-specific metadata. Once shared calibration quantities are established, the update exchange requires only the trained model parameters and aggregation weights. No raw examples, individual labels, per-example representations, or missing responses need to be exported for the correction.

Precedents for the required information. The relevant information types already appear in established federated methods. FedAvg uses local dataset sizes for weighted model aggregation (McMahan et al., 2017). FedLC uses class sample counts to construct its local logit calibration (Zhang et al., 2022a). More directly, Algorithm 1 of FedGen transmits local label counters to the server, which updates an empirical label distribution from those counters (Zhu et al., 2021). FedProto communicates class-wise feature prototypes rather than raw training examples (Tan et al., 2022). FedTau requires neither these prototypes nor per-example predictions: its categorical reference can be obtained from aggregate class counts. These precedents establish that the required statistics are compatible with existing federated protocols; they do not make the statistics intrinsically private.

Aggregate rather than client-specific reference information. Let $N_c = \sum_k n_{k,c}$ and $N = \sum_c N_c$. The add-one-smoothed reference used by FedTau is

$$\widetilde{\pi}_*(c) = \frac{N_c + 1}{N + C}, \qquad b_{k,c} = \log \widetilde{\pi}_k(c) - \log \widetilde{\pi}_*(c). \tag{9}$$

Thus, clients need the pooled reference, not the label histogram of any other client; the server need not inspect individual histograms. For fixed training data, these totals can be computed once before optimization. If the reference prior is specified publicly in advance, this aggregation is unnecessary; the prior must still represent the intended prediction task. The fixed sequence normalizer in Eq. (28) likewise depends only on the total observed-event count and total record count. Any shared preprocessing means or variances are additional aggregate statistics and should be included in the same information accounting.

Compatibility with secure aggregation. The required sums can be computed using secure aggregation (Bonawitz et al., 2017). For model averaging, each client can contribute the pair $(n_k\theta_{k,t}, n_k)$ to a secure sum, after which the server computes

$$\theta_{t+1} = \frac{\sum_{k \in S_t} n_k \theta_{k,t}}{\sum_{k \in S_t} n_k}. \tag{10}$$

The same primitive can aggregate class counts and normalization totals. Substituting these sums into Eq. (9) and Eq. (10) reproduces the original reference and aggregation rule. Consequently, hiding the individual summands does not require changing the FedTau objective, apart from any chosen numerical encoding. Against an honest-but-curious server, under the secure-aggregation protocol's cryptographic, collusion, and participation assumptions, its protocol transcript reveals no efficiently extractable information beyond the aggregate output and permitted side information. This is an optional protected implementation, not a guarantee obtained from the local correction alone.

Scope of the privacy claim. FedTau therefore does not require direct disclosure of individual training records or client-specific observation mechanisms. This information-flow property is distinct from a guarantee against inference from released models: shared gradients can reveal training information (Zhu et al., 2019), and an unchanged message format does not imply unchanged information content. Secure aggregation protects individual contributions beyond what its outputs reveal, but does not prevent inferences from the aggregates themselves. A record- or client-level differential-privacy guarantee would require a separately specified mechanism and accounting for calibration, preprocessing statistics, and model releases (Kairouz & McMahan, 2021). The common-target results in Appendix A.5 establish statistical correctness of the local objective; they do not assert zero information leakage.

A.5 Methodology details

We prove the common-target property in Proposition 1, derive the three corrections in Section 4, and quantify how mechanism error affects local optimization and aggregation. All population statements concern the likelihood objective; additional curvature assumptions are stated only where a contraction is derived.

A.5.1 Common predictive target and identification

Deriving the correction from sampling and recording. For a sampling-and-recording process, let $s_k(y, x) \geq 0$ be its fixed relative sampling weight and $\mathsf{R}_k(A \mid y, x)$ the conditional probability that the recorded observation lies in A. With a positive finite denominator, the implemented observation law is

$$\widehat{\mathcal{T}}_k[q](A \mid x) = \frac{\int \mathsf{R}_k(A \mid y, x) s_k(y, x) q(dy \mid x)}{\int s_k(y, x) q(dy \mid x)}. \tag{11}$$

Label shift retains the label and sets $s_k(c, x) = e^{b_{k,c}}$. Pure censoring and prefix truncation use $s_k = 1$, with R_k recording the corresponding point, bound, or prefix event. Thus sampling requires normalization, whereas incomplete records require marginalization. For a fixed baseline loss, subtraction in Eq. (3) uniquely specifies the scalar τ_k. The sum $\ell_k^0 + \tau_k$ is the observation negative

log-likelihood, not an approximation obtained by imputing an unobserved response. The identity mechanism gives $\tau_k = 0$ when ℓ_k^0 is its ordinary negative log-likelihood. The recording process and sampling model must justify this law; the construction does not infer an unknown mechanism from heterogeneity alone.

Assumptions and notation. Let $q_{k,\theta}^{\mathrm{obs}} = \mathcal{T}_k[q_\theta]$. Assume Eq. (1) and a parameter θ_* satisfying $q_{\theta_*} = p_*$ on the client input supports. The observation laws have densities with respect to a common dominating measure for each client, including a mixed measure for censored observations. Assume finite risks and positive predicted density wherever the true observation density is positive. Write $\mathbb{E}_k$ for expectation under $p_k^X(x)p_k^{\mathrm{obs}}(o \mid x)$ and define

$$F_k(\theta) = -\mathbb{E}_k \log q_{k,\theta}^{\mathrm{obs}}(O \mid X), \qquad F(\theta) = \sum_{k=1}^{K} w_k F_k(\theta), \qquad w_k > 0, \quad \sum_k w_k = 1. \tag{12}$$

Sample-size weights correspond to $w_k = n_k / \sum_j n_j$.

Proof of Proposition 1. Since $q_{k,\theta_*}^{\mathrm{obs}} = p_k^{\mathrm{obs}}$, direct subtraction gives

$$\begin{aligned} F_k(\theta) - F_k(\theta_*) &= \mathbb{E}_k \log \frac{p_k^{\mathrm{obs}}(O \mid X)}{q_{k,\theta}^{\mathrm{obs}}(O \mid X)} \\ &= \mathbb{E}_{X \sim p_k^X} D_{\mathrm{KL}}(\mathcal{T}_k[p_*](\cdot \mid X) \,\|\, \mathcal{T}_k[q_\theta](\cdot \mid X)) \geq 0. \end{aligned} \tag{13}$$

Thus θ_* minimizes every client risk, independently of the input marginals, client sizes, or invertibility of an individual map. For the deployment input law p_*^X, assume coverage $p_*^X \ll \bar{p}^X$, where $\bar{p}^X = \sum_k w_k p_k^X$. Joint identification means that, within the predictive family,

$$\big[\mathcal{T}_k[q] = \mathcal{T}_k[p_*] \quad p_k^X\text{-a.e. for every } k\big] \implies q = p_* \quad p_*^X\text{-a.e.} \tag{14}$$

Because all w_k are positive, $F(\theta) = F(\theta_*)$ holds exactly when every divergence in Eq. (13) vanishes. Equation (14) then yields $q_\theta = p_*$ on the deployment support. This identifies a predictive law, not a unique neural-network parameterization.

Why gradient matching is unnecessary. At an interior differentiable minimizer, $\nabla F_k(\theta_*) = 0$ for every client. If the gradients are β-Lipschitz on the relevant region, then

$$\sum_k w_k \|\nabla F_k(\theta)\|_2^2 \leq \beta^2 \|\theta - \theta_*\|_2^2, \qquad \sum_k w_k \|\nabla F_k(\theta_*)\|_2^2 = 0. \tag{15}$$

The inequality follows by subtracting $\nabla F_k(\theta_*) = 0$ and applying smoothness separately for each client. Hence the population gradient energy vanishes at the shared target, although local gradients and Hessians can differ elsewhere. Any number of population gradient steps starting at θ_* leaves each local model unchanged, so every selected-cohort average also leaves θ_* unchanged. An identity map reduces Eq. (3) to ordinary negative log-likelihood. With matched update rules, random draws, loss scaling, and model and optimizer states, equality of the corresponding trajectories follows by induction over local steps and communication rounds.

A.5.2 Classification: prior correction and its gradients

Derivation of the classification observation law. Under label shift and $\pi_*(c) > 0$, Bayes' rule gives

$$p_k(c \mid x) = \frac{\{\pi_k(c)/\pi_*(c)\} p_*(c \mid x)}{\sum_j \{\pi_k(j)/\pi_*(j)\} p_*(j \mid x)}. \tag{16}$$

For positive local priors, replacing p_* by $q_\theta = \mathrm{softmax}(z_\theta)$ shows that the true map is $\mathrm{softmax}(z_\theta + b_k^*)$, where $b_{k,c}^* = \log \pi_k(c) - \log \pi_*(c)$. Its negative log-likelihood is therefore the cross-entropy in Eq. (4), with the population offset replaced by its smoothed estimate b_k. The normalized observation loss is $-\log q_\theta(y \mid x) - b_{k,y} + \log \sum_c q_\theta(c \mid x) e^{b_{k,c}}$. Subtracting $\mathrm{CE}(z_\theta(x), y)$ gives the scalar τ_k^{cls} in Eq. (4). Thus the cached vector b_k and the prediction-dependent loss correction implement the

same gradients. For strictly positive local priors, the inverse is explicit:

$$q(c \mid x) = \frac{\{\pi_*(c)/\pi_k(c)\}\mathcal{T}_k[q](c \mid x)}{\sum_j \{\pi_*(j)/\pi_k(j)\}\mathcal{T}_k[q](j \mid x)}. \tag{17}$$

This establishes identification on that client's input support. Zero local priors require joint information or a restricted model; finite smoothing alone does not establish identification. If the reference prior is uniform, its log term is a common logit constant and drops out, giving client-side Balanced Softmax.

Prediction error under positive priors. Suppose every $\pi_k(c) > 0$ and let $\rho = \max_k[\max_c\{\pi_*(c)/\pi_k(c)\}/\min_c\{\pi_*(c)/\pi_k(c)\}]$. For probability vectors u, v and positive weights s_c, put $Z_u = \sum_c s_c u_c$ and $Z_v = \sum_c s_c v_c$. Adding and subtracting $s_c v_c / Z_u$ gives

$$\sum_c \left| \frac{s_c u_c}{Z_u} - \frac{s_c v_c}{Z_v} \right| \le \frac{\sum_c s_c |u_c - v_c| + |Z_u - Z_v|}{Z_u} \le 2\frac{\max_c s_c}{\min_c s_c}\|u - v\|_1.$$

Apply this to the inverse map with $s_c = \pi_*(c)/\pi_k(c)$, $u = \mathcal{T}_k[q]$, and $v = \mathcal{T}_k[p_*]$. Pinsker's inequality $\|u - v\|_1^2 \le 2D_{\mathrm{KL}}(v\|u)$ and Eq. (13) then yield

$$\mathbb{E}_{\bar{p}^X}\|q_\theta(\cdot \mid X) - p_*(\cdot \mid X)\|_1^2 \le 8\rho^2\big[F(\theta) - F(\theta_*)\big]. \tag{18}$$

For evaluation under $\bar{p}^X(x)p_*(y \mid x)$, the excess classification error of $\arg\max_c q_\theta(c \mid x)$ over the Bayes rule is at most $\sqrt{8\rho^2[F(\theta) - F(\theta_*)]}$. Indeed, its conditional excess is at most $\|q_\theta - p_*\|_1$, and Cauchy–Schwarz completes the bound. This makes the connection from corrected likelihood to reference prediction quantitative when the priors are positive.

Training-time correction. Let e_y denote the one-hot label and $J_\theta(x) = \partial z_\theta(x)/\partial\theta$. For any fixed offset b,

$$\nabla_\theta \,\mathrm{CE}(z_\theta(x) + b, y) = J_\theta(x)^\top \left[\mathrm{softmax}(z_\theta(x) + b) - e_y\right]. \tag{19}$$

With $b = b_k^*$ and $\theta = \theta_*$, the bracket has conditional expectation $p_k(\cdot \mid x) - p_k(\cdot \mid x) = 0$. Without correction, that expectation is $p_*(\cdot \mid x) - p_k(\cdot \mid x)$. The correction consequently changes gradients of all parameters entering the logits, including the feature extractor. Subtracting an offset from the final bias after local training does not generally reproduce these feature updates.

Sensitivity to estimated priors. If $\|J_\theta(x)\|_{\mathrm{op}} \le M$ on the relevant inputs, then

$$\|\nabla_\theta \,\mathrm{CE}(z_\theta + b_k, y) - \nabla_\theta \,\mathrm{CE}(z_\theta + b_k^*, y)\|_2 \le \frac{M}{2}\|b_k - b_k^*\|_2. \tag{20}$$

Indeed, the softmax Jacobian is $\mathrm{diag}(p) - pp^\top$. For any unit vector v, its quadratic form is $\mathrm{Var}_{c\sim p}(v_c) \le (\max_c v_c - \min_c v_c)^2/4 \le 1/2$. Integrating the Jacobian along the segment between the offsets proves Eq. (20). Thus bounded prior-offset error gives a bounded gradient perturbation, rather than an unquantified change in the training objective.

A.5.3 Regression and sequence observation likelihoods

Right-censored regression. For $U = \min(Y, c_k)$ and $\delta = \mathbf{1}\{Y \ge c_k\}$, the observation density and censoring mass are

$$q_{k,\theta}^{\mathrm{obs}}(u, \delta \mid x) = \begin{cases} q_\theta(u \mid x), & \delta = 0,\ u < c_k, \\ \displaystyle\int_{c_k}^{\infty} q_\theta(v \mid x)\, dv, & \delta = 1,\ u = c_k. \end{cases} \tag{21}$$

For $q_\theta = \mathcal{N}(\mu_\theta(x), \sigma^2)$, the second term equals $\Phi((\mu_\theta(x) - c_k)/\sigma)$. Taking negative logarithms gives Eq. (5), up to the parameter-independent Gaussian constant on uncensored records. This omitted constant changes neither the gradients nor the population minimizers.

The correction removes the mean-score bias explicitly. Fix x, a known $\sigma > 0$, and a correctly specified conditional Gaussian mean $\mu_* = \mu_{\theta_*}(x)$. Let $a = (c_k - \mu_*)/\sigma$ and let ϕ denote the

standard normal density. The uncensored and censored contributions satisfy

$$\mathbb{E}\left[(1-\delta)\frac{\mu_* - U}{\sigma^2} \mid x\right] = \frac{\phi(a)}{\sigma},$$
$$\mathbb{E}\left[-\delta\frac{\phi(a)}{\sigma\Phi(-a)} \mid x\right] = -\frac{\phi(a)}{\sigma}. \tag{22}$$

The first equality follows by integrating $-z\phi(z)/\sigma$ over $z < a$; the second uses $\Pr(\delta = 1 \mid x) = \Phi(-a)$. Their sum is zero. In contrast, the expected point-loss gradient at μ_* is $\mathbb{E}[(Y-c_k)_+ \mid x]/\sigma^2$, which is positive when the upper tail has positive expected excess. Thus point-target training pushes the mean downward even at the correct predictor. For fixed σ, the censoring probability is strictly increasing in μ, so Eq. (21) identifies the Gaussian mean. An unrestricted tail shape is not identified by its total mass.

Censored sequence prefixes. Let T be the number of non-EOS target tokens. Condition on the recording cap L and a protocol that reveals the prefix $y_{1:L}$ and the event $T > L$, but no further suffix information. For a proper autoregressive distribution over finite sequences, summing over all compatible continuations gives

$$\begin{aligned}\Pr_\theta(Y_{1:L} = y_{1:L}, T > L \mid x) &= \left[\prod_{j=1}^{L} q_\theta(y_j \mid x, y_{<j})\right] \sum_{v \neq \mathrm{EOS}} q_\theta(v \mid x, y_{1:L}) \\ &= \left[\prod_{j=1}^{L} q_\theta(y_j \mid x, y_{<j})\right](1 - p_{\mathrm{EOS}}),\end{aligned} \tag{23}$$

where $p_{\mathrm{EOS}} = q_\theta(\mathrm{EOS} \mid x, y_{1:L})$. The suffix probabilities after each next token sum to one. Negative logarithms therefore give $\ell_k^{\mathrm{FedTau}} = -\sum_{j=1}^{L} \log q_\theta(y_j \mid x, y_{<j}) - \log(1 - p_{\mathrm{EOS}})$, exactly Eq. (6). For the boundary EOS logit,

$$\frac{\partial[-\log p_{\mathrm{EOS}}]}{\partial z_{\mathrm{EOS}}} = p_{\mathrm{EOS}} - 1, \qquad \frac{\partial[-\log(1 - p_{\mathrm{EOS}})]}{\partial z_{\mathrm{EOS}}} = p_{\mathrm{EOS}}. \tag{24}$$

The first term encourages stopping at the cap; the second uses the observed continuation event. Complete records retain the true EOS likelihood. The likelihood is valid without observing suffix tokens; identifying their distribution additionally requires the joint information in Eq. (14).

A.5.4 Mechanism error and federated local updates

Let $G_k(\theta) = -\mathbb{E}_k \log(\widehat{\mathcal{T}}_k[q_\theta])(O \mid X)$ be the population risk with the estimated map fixed, distinct from the empirical $\widehat{F}_k$ in Eq. (3). Define the residual $\varepsilon_k = \|\nabla G_k(\theta_*)\|_2$. For correctly specified maps under Proposition 1, $G_k = F_k$ and $\varepsilon_k = 0$. For positive class priors, Eq. (20) gives $\varepsilon_k \leq (M/2)\|b_k - b_k^*\|_2$.

Local perturbation bound. Consider population gradient descent $\theta_k^{(h+1)} = \theta_k^{(h)} - \eta\nabla G_k(\theta_k^{(h)})$, starting from $\theta_k^{(0)} = \theta_t$, with $\eta \geq 0$. Suppose ∇G_k is β-Lipschitz, $\beta > 0$, on the trajectory and its connecting segments to θ_*. After $H_k \geq 1$ steps,

$$\|\theta_k^{(H_k)} - \theta_*\|_2 \leq (1+\eta\beta)^{H_k}\|\theta_t - \theta_*\|_2 + \frac{(1+\eta\beta)^{H_k} - 1}{\beta}\varepsilon_k. \tag{25}$$

Proof. Adding and subtracting $\eta\nabla G_k(\theta_*)$ gives $d_{h+1} \leq (1+\eta\beta)d_h + \eta\varepsilon_k$, where $d_h = \|\theta_k^{(h)} - \theta_*\|_2$. Iterating and summing the geometric series proves the bound. This quantifies accumulated residual error without assuming convexity.

Conditional contraction and aggregation. If, additionally, $mI \preceq \nabla^2 G_k \preceq \beta I$ on a convex region containing the iterates and θ_*, where $0 < m \leq \beta$, take $0 < \eta \leq 1/\beta$ and $r = 1 - \eta m$. The integrated Hessian has eigenvalues in $[m, \beta]$; hence the gradient-step map has Lipschitz constant at

most r. The same recursion now yields

$$\|\theta_k^{(H_k)} - \theta_*\|_2 \le r^{H_k}\|\theta_t - \theta_*\|_2 + \frac{1 - r^{H_k}}{m}\varepsilon_k. \tag{26}$$

The returned model is $\theta_{k,t} = \theta_k^{(H_k)}$. For $w_{k,t} = n_k / \sum_{j \in S_t} n_j$, the triangle inequality and Eq. (8) imply

$$\|\theta_{t+1} - \theta_*\|_2 \le \left(\sum_{k \in S_t} w_{k,t} r^{H_k}\right)\|\theta_t - \theta_*\|_2 + \sum_{k \in S_t} w_{k,t}\frac{1 - r^{H_k}}{m}\varepsilon_k. \tag{27}$$

With exact mechanisms, the residual term vanishes. For $H_k \ge 1$, every nonempty selected cohort is then contractive under these curvature assumptions. No unbiasedness assumption on selected-cohort-normalized weights is needed: the inequality holds for the realized cohort. The smooth bound is a perturbation bound; contraction additionally requires the stated curvature and is not a claim about arbitrary neural-network or momentum updates.

A.5.5 Loss normalization and empirical approximation

Record-level sequence normalization. In the `record` implementation, let L_i be the observed number of non-EOS tokens in training record i and set $N = \sum_k n_k$ and $\kappa = N^{-1}\sum_{i=1}^{N}(L_i + 1)$. The additional event is true EOS for a complete record and continuation for a censored record. A minibatch $\mathcal{B}$ uses

$$\mathcal{L}_{\mathcal{B}}(\theta) = \frac{1}{|\mathcal{B}|\kappa}\sum_{i \in \mathcal{B}} \ell_i^{\mathrm{FedTau}}(\theta), \qquad \kappa > 0. \tag{28}$$

The same training-derived κ is fixed across clients and updates. For uniformly sampled records at fixed parameters, the expected minibatch gradient equals the scaled empirical record-risk gradient. This common positive scale also preserves all population minimizers. The `legacy_token` option instead divides by the random minibatch event count; that ratio is not generally an unbiased record-risk gradient. Boundary events are evaluated stably as $\log\sum_v e^{z_v} - \log\sum_{v \ne \mathrm{EOS}} e^{z_v}$.

From empirical fit to population risk. For the same fixed weights, define $\widehat{F} = \sum_k w_k \widehat{F}_k$. On any event where $\sup_\theta |\widehat{F}(\theta) - F(\theta)| \le \epsilon_F$, with $\epsilon_F, \xi \ge 0$, an approximate empirical minimizer satisfying $\widehat{F}(\widehat{\theta}) \le \inf_\theta \widehat{F}(\theta) + \xi$ obeys

$$0 \le F(\widehat{\theta}) - F(\theta_*) \le 2\epsilon_F + \xi. \tag{29}$$

Indeed, $F(\widehat{\theta}) \le \widehat{F}(\widehat{\theta}) + \epsilon_F \le \widehat{F}(\theta_*) + \xi + \epsilon_F \le F(\theta_*) + \xi + 2\epsilon_F$. The uniform discrepancy includes sampling and mechanism-estimationerrors; the bound does not assume either error is zero.By Eq. (13), it controls the weighted observationKL divergence.The IPW selection extension is derived separately inAppendix A.7.7; its estimating objective is notidentified with the likelihood in Eq. (3).

A.6 Classification results

A.6.1 Experimental design

Objective and comparison grid. The classification experiments evaluate whether accounting for client-specific class frequencies during local training improves the shared predictor. Tables 9 and 10 report the ResNet-18 and StageCNN configurations, respectively. Each table covers three datasets, two non-IID partitions and an IID reference, and three local epoch counts. The resulting 54 settings test predictive performance and sensitivity to both client heterogeneity and the amount of local optimization.

Datasets and held-out evaluation. We use CIFAR-10, CIFAR-100 (Krizhevsky, 2009), and Tiny-ImageNet (Le et al., 2015), containing $C = 10$, 100, and 200 classes, respectively. For each CIFAR dataset, all $50{,}000$ official training images are available for federated training, and the $10{,}000$ official test images form the common evaluation set. For TinyImageNet, we use all $100{,}000$ training images and the $10{,}000$ labeled images in the official validation split as the held-out test set; the unlabeled test split is not used. Only training images are assigned to clients. No additional validation

subset is removed, and the main classification setting does not apply global long-tail subsampling. The pooled training and evaluation sets are class-balanced, even though individual clients can have highly unequal class frequencies. Deterministic views of the training images used for diagnostics are not additional held-out datasets.

Input resolution and preprocessing. The actual network inputs are 32×32 for CIFAR-10/100 and 64×64 for TinyImageNet. Training uses random cropping at the corresponding resolution with four-pixel padding, random horizontal flipping, tensor conversion, and channel normalization. Evaluation uses tensor conversion and the same normalization without random augmentation. The channel means and standard deviations are $(0.4914, 0.4822, 0.4465)$ and $(0.2470, 0.2435, 0.2616)$ for CIFAR-10; $(0.5071, 0.4867, 0.4408)$ and $(0.2675, 0.2565, 0.2761)$ for CIFAR-100; and $(0.4802, 0.4481, 0.3975)$ and $(0.2302, 0.2265, 0.2262)$ for TinyImageNet. The main partitioning experiment introduces neither label corruption nor client-specific image transformations.

Non-IID and IID construction. We distribute the training set among $K = 100$ clients. For each class, the non-IID construction shuffles its training indices, samples a K-dimensional Dirichlet allocation vector with concentration α, and divides the class indices at integer-valued cumulative allocation boundaries. Each training example belongs to exactly one client, and the partition remains fixed throughout training. We evaluate $\alpha \in \{0.1, 0.6\}$, with smaller α producing more concentrated class allocations (Hsu et al., 2019).

This class-wise construction does not enforce equal client dataset sizes. Both the local class proportions and the total sample counts n_k can vary, while the complete pooled training set is preserved. The average client size is 500 examples for either CIFAR dataset and $1{,}000$ for TinyImageNet; these are not fixed per-client quotas. The number of represented classes is also not fixed. For the IID reference, a seeded permutation of all training indices is split into equal-size client subsets. This removes deliberately imposed label skew but permits finite-sample differences in empirical class proportions. The equal-size Dirichlet construction used in the separate mechanism controls is not the partition used for these main non-IID comparisons.

Model implementations. The reference classification model is a small-image ResNet-18 (He et al., 2016) initialized from scratch. It uses a 3×3 convolutional stem, four residual stages with 64, 128, 256, and 512 channels and two BasicBlocks per stage, global average pooling, a 256-dimensional projection with batch normalization and ReLU, and a C-way linear classifier. Table 10 uses StageCNN. It has a 3×3 convolutional stem, four residual stages with 64, 128, 256, and 512 channels, and two BasicBlocks in each stage. The first block of each of the last three stages downsamples by stride two. Global average pooling is followed by a 512-to-256 linear projection, batch normalization, ReLU, and a C-class linear head. The staged interface exposes intermediate feature representations. Both classification configurations are trained from scratch and use a shared model architecture across the compared methods within each table. Their results test the correction in the specified ResNet-18 and StageCNN implementations, without relying on external pretraining or additional FedTau model capacity.

Participation and optimization. Ten clients are sampled uniformly without replacement in each of 500 communication rounds. The local epoch count is $E \in \{1, 3, 5\}$, where an epoch is one pass over a selected client's training data. FedTau uses local SGD with constant learning rate 0.01, momentum 0.9, and weight decay 5×10^{-4}. The local optimizer is initialized afresh at each client update. Training and evaluation batch sizes are 64 and 256, respectively. Gradient clipping at norm 5 starts after rounds 8, 12, and 14 for CIFAR-10, CIFAR-100, and TinyImageNet, respectively. The server uses sample-size-weighted aggregation.

Because non-IID clients can have different dataset sizes, a common E need not imply equal mini-batch counts across clients. However, the same client has the same nominal local data budget across methods within a matched setting. Increasing E changes local computation while holding communication rounds fixed; the epoch sweep is therefore a robustness comparison, not a fixed-computation comparison.

Classification-specific correction. Let $n_{k,c}$ be the number of training examples of class c at client k, with $n_k = \sum_c n_{k,c}$ and $N = \sum_k n_k$. The implementation uses add-one smoothing:

$$\widetilde{\pi}_k(c) = \frac{n_{k,c}+1}{n_k + C}, \qquad \widetilde{\pi}_*(c) = \frac{\sum_k n_{k,c}+1}{N + C}. \tag{30}$$

These training-only statistics define the fixed offset $b_{k,c} = \log \widetilde{\pi}_k(c) - \log \widetilde{\pi}_*(c)$ in Eq. (4). It is applied with unit strength at every local update, without an additional coefficient selected for each partition or epoch count. The shared predictor is evaluated without this client-specific offset. FedTau adds no classifier parameters, generator, or server-side knowledge-distillation stage.

Metric and five-seed reporting rule. For seed s and round t, global top-1 accuracy is

$$a_{s,t} = \frac{100}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} \mathbf{1}\left\{ \arg\max_c z_{\theta_{s,t},c}(x_i) = y_i \right\}. \tag{31}$$

For the five-seed best-score reporting convention, take the best accuracy within each seed before averaging:

$$b_s = \max_{1 \le t \le 500} a_{s,t}, \qquad \bar{b} = \frac{1}{5}\sum_{s=1}^{5} b_s, \qquad \text{SD} = \left[\frac{1}{4} \sum_{s=1}^{5} (b_s - \bar{b})^2 \right]^{1/2}. \tag{32}$$

This is a best-observed-test-accuracy summary, not selection of the best seed or a validation-selected checkpoint. The table margin Δ subtracts the strongest competing mean in the same setting from FedTau's mean and is expressed in percentage points. A five-seed trajectory is instead summarized pointwise across runs; its standard-deviation band is not the standard deviation of the five per-seed maxima.

A.6.2 Baseline rationale and comparison protocol

Aggregation and local optimization. FedAvg (McMahan et al., 2017) supplies the canonical averaging reference and the closest comparison for the local objective change. FedProx (Li et al., 2020) and FedDyn (Acar et al., 2021) provide established alternatives based on proximal and dynamic regularization. They test whether controlling local optimization can reproduce the benefit of explicitly accounting for client class frequencies. These comparisons distinguish the observation correction from a gain over unregularized averaging alone.

Representation and class-structure methods. MOON (Li et al., 2021) tests model-contrastive representation learning. FedProto (Tan et al., 2022) and FedTGP (Zhang et al., 2024) test prototype-based knowledge sharing, including trainable global prototypes in FedTGP. FedCPD (He et al., 2024) is particularly relevant because its class-proxy decoupling and proxy regularization explicitly address label-distribution skew and unreliable updates of absent-class proxies. FedRG (Wen et al., 2026) broadens the comparison to a recent representation-geometry approach developed for heterogeneous noisy clients. Its relevance is methodological; its published noisy-label setting is not identical to the label-skew experiment considered here. Together, these methods test alternatives that act on representations and class structure rather than only on parameter averaging.

Generative knowledge transfer. FedGen (Zhu et al., 2021), FedFTG (Zhang et al., 2022b), and FedLMG (Yang et al., 2025) provide generated-feature or generated-data knowledge-transfer alternatives. FedGen supplies generative knowledge to local learners, FedFTG fine-tunes the global model through data-free distillation, and FedLMG uses local-model-guided diffusion in its original one-shot framework. Their inclusion tests whether more elaborate knowledge transfer is needed to obtain the reported improvements. The present comparison concerns the evaluated shared-predictor configurations; it does not transfer communication-efficiency claims from their different original protocols.

Applicable comparison scope. The target is horizontal image classification with a shared predictor. Vertical federated methods require a different, feature-partitioned data layout, while methods returning only personalized predictors require a different evaluation target (Kairouz & McMahan,

2021). They are not direct additions to this global-accuracy comparison without specifying those adaptations. Within the present class-balanced pooled setting, the reference prior is uniform, so FedTau's categorical loss is equivalent to client-side Balanced Softmax with the same smoothing, as noted in the classification instance in Section 4. Algebraically, the uniform-reference term is a class-independent logit constant and drops out of cross-entropy. We do not count an identical loss implementation as a distinct algorithmic alternative. The role of a non-uniform reference is examined separately in Appendix A.9.

Fair comparison. The comparison fixes the training and evaluation data, client partition, participation schedule, classifier configuration, initialization, communication horizon, and nominal local data budget within each setting and seed. Method-specific regularization, auxiliary learning, and aggregation rules remain the competing algorithmic choices. Thus the comparison controls access to the prediction problem and training opportunities without equating different algorithms' total arithmetic costs. FedTau's correction uses training-label counts, not additional labeled images or held-out class statistics. Centralized training is reported as a pooled-data reference and is excluded from the federated ranking.

A.6.3 Classification performance

Consistent superiority across the reported grid. FedTau has the highest reported mean accuracy in all 54 settings. Against the strongest baseline chosen separately for each setting, the average margins are 3.00 percentage points in Table 9 and 6.26 points in Table 10. The corresponding ranges are 1.04–12.32 and 2.98–14.28 points. These averages assign equal weight to experimental settings; they are not accuracies on a pooled evaluation set. Selecting the strongest competitor independently in every setting makes the comparison more informative than comparing only with FedAvg or one fixed baseline.

Severe heterogeneity with ResNet-18. At $\alpha = 0.1$ and $E = 1$, FedTau reaches 79.14% on CIFAR-10, compared with 66.82% for the strongest baseline, FedTGP. The 12.32-point margin remains substantial even after replacing FedAvg with the best listed alternative; the gain over FedAvg itself is 17.08 points. On CIFAR-100, FedTau reaches 53.79% versus 50.54% for FedLMG, and on TinyImageNet it reaches 47.89% versus 44.42% for MOON. These are improvements of 3.25 and 3.47 points over the strongest baselines and 6.18 and 4.80 points over FedAvg. The strongest competitor changes across datasets, but the favorable ordering is preserved.

Consistent gains in the StageCNN results. At the same $\alpha = 0.1$ and $E = 1$, Table 10 reports 70.58%, 51.69%, and 44.74% for FedTau on CIFAR-10, CIFAR-100, and TinyImageNet. The strongest corresponding baseline accuracies are 56.30% (FedLMG), 42.55% (FedDyn), and 35.62% (FedRG). FedTau therefore improves by 14.28, 9.14, and 9.12 points, respectively. The substantial CIFAR-100 and TinyImageNet margins show that the reported benefit is not confined to a ten-class prediction problem.

Why the strong-skew comparison matters. Under the label-shift model, local cross-entropy learns a conditional distribution that includes the client's class prior. The FedTau offset accounts for this prior inside every local loss, so the shared predictor can represent the reference task. This changes the gradients received by both the classifier and the feature extractor before aggregation. Under the assumptions of Proposition 1, the corrected population objectives admit a common predictive target even though their gradients away from that target can differ. The strong-skew results are consistent with this correction being useful beyond parameter regularization, prototype supervision, and knowledge transfer. The argument concerns separate local objectives; it does not require the sample-weighted pooled cross-entropy objective to be incorrect.

A.6.4 Robustness to heterogeneity and local epochs

The advantage persists as imposed label skew decreases. Averaging over datasets and epoch counts, the margins over the strongest baselines under $\alpha = 0.1$, $\alpha = 0.6$, and IID are 5.07, 2.09, and 1.84 points for ResNet-18, and 7.48, 5.76, and 5.54 points for StageCNN. Thus, strong heterogeneity produces the largest average advantage, but the favorable comparison does not disappear under milder or random IID partitions.

FedTau also has a smaller IID-to-$\alpha = 0.1$ accuracy drop than FedAvg in all 18 matched dataset–configuration–epoch comparisons. For CIFAR-10 with $E = 1$, the drop is reduced from 18.13 to 7.86 points with ResNet-18 and from 22.96 to 14.59 points with StageCNN. The evidence therefore concerns both higher accuracy and lower sensitivity to the tested change in client partitioning. Because the non-IID construction also changes client sizes, this comparison measures robustness to the complete partitioning procedure, not an isolated label-skew effect at fixed client sizes.

IID is not the exact identity-mechanism case. Random IID assignment permits empirical-prior differences between finite client datasets. It therefore differs from explicitly enforcing $\widetilde{\pi}_k = \widetilde{\pi}_*$ for every client. Under that exact equality, the offset is zero and FedTau recovers ordinary cross-entropy and, with matched optimization states, the corresponding FedAvg updates. The IID rows show favorable performance under random partitioning; they do not imply an advantage when the correction is identically zero.

Longer local training retains the improvement. FedTau's reported accuracy increases from $E = 1$ to $E = 3$ and then $E = 5$ in every dataset–partition combination in both tables. Its first-place ranking is also preserved throughout the sweep. Under severe skew at $E = 5$, the margins over the strongest baselines remain 7.50, 2.29, and 3.00 points for ResNet-18 on CIFAR-10, CIFAR-100, and TinyImageNet, respectively. The corresponding margins in the second table are 2.98, 5.59, and 6.00 points.

These results distinguish robustness from invariance to local work. The baselines also benefit from additional training, so the advantage need not increase with E. Nevertheless, the correction remains useful after more local passes and does not prevent the model from exploiting them. For example, under $\alpha = 0.1$, CIFAR-100 accuracy increases from 53.79% to 64.86% with ResNet-18 and from 51.69% to 62.05% with StageCNN as E changes from one to five.

Relation to the centralized references. At $E = 5$ under IID partitioning, FedTau reaches 92.89% on CIFAR-10 with ResNet-18 and 90.72% with StageCNN, compared with centralized reference accuracies of 95.42% and 92.49%. The remaining differences are 2.53 and 1.77 points. These references contextualize prediction quality rather than enter the federated ranking or establish equal computational cost.

Interpretation across tasks and settings. Tables 9 and 10 support a consistent favorable accuracy ordering across the reported partitions and local epoch counts. Figures 2 and 3 organize the corresponding trajectory visualizations by the same experimental dimensions; the robustness comparisons above follow from the separately reported table settings rather than curve smoothness. The magnitude of improvement varies with the recognition task and configuration, as expected for a correction that addresses class-prior mismatch without replacing visual representation learning. The common pattern is that accounting for the local observation mechanism remains beneficial under severe skew, milder skew, and the reported IID partitions, while preserving the benefit of additional local training.

A.7 Regression results

A.7.1 Evaluation objective and protocol

What the experiments test. The regression experiments examine whether accounting for a client-specific observation mechanism improves prediction of the reference response. We consider California Housing (Pace & Barry, 1997), Superconductivity (Hamidieh, 2018), and Bike Sharing (Fanaee-T & Gama, 2014) under three recording mechanisms: right censoring, outcome-dependent label selection, and symmetric interval quantization. The prediction target remains the reference response y, rather than its client-specific recorded surrogate. This distinction is essential: accurately predicting a capped value or an interval midpoint does not necessarily imply accurate prediction of the response required at deployment.

The primary comparison uses heterogeneous right-censoring thresholds (Table 11 and Figure 4). Table 12 provides complementary controls for mechanism specification, point imputation, and fully observed training. The selection and quantization experiments (Tables 13 and 14; Figures 5 and 6)

Table 9: ResNet-18 classification accuracy (%). E denotes local epochs and α the Dirichlet concentration. Best/second-best means are bold/boxed; $\uparrow \Delta$ is the margin over the strongest federated baseline in percentage points.

		CIFAR-10			**CIFAR-100**			**TinyImageNet**		
		Centralized: **95.42±0.08**			Centralized: **76.12±0.18**			Centralized: **60.61±0.14**		
Method	α	E=1	E=3	E=5	E=1	E=3	E=5	E=1	E=3	E=5
FedAvg	0.1	62.06 ±6.14	73.52 ±1.04	75.34 ±1.44	47.61 ±1.69	56.84 ±1.49	58.76 ±1.51	43.09 ±1.22	47.02 ±1.14	47.24 ±1.19
	0.6	77.43 ±2.46	85.82 ±2.00	86.03 ±1.92	54.26 ±0.92	61.91 ±0.93	63.17 ±0.97	46.52 ±0.52	50.43 ±0.65	50.81 ±0.70
	IID	80.19 ±1.33	86.40 ±1.48	87.37 ±0.53	54.98 ±0.92	62.52 ±0.72	63.39 ±0.51	47.35 ±0.71	50.65 ±0.61	50.89 ±0.57
FedProx	0.1	65.34 ±4.61	73.29 ±1.13	75.27 ±1.21	48.42 ±1.67	57.34 ±1.50	59.17 ±1.46	42.87 ±1.03	47.49 ±1.03	47.29 ±1.04
	0.6	79.48 ±2.58	86.79 ±2.09	88.03 ±1.95	55.13 ±1.22	62.45 ±1.14	64.12 ±1.11	46.97 ±0.49	50.94 ±0.65	51.08 ±0.65
	IID	81.88 ±0.39	88.63 ±0.77	89.41 ±0.76	55.71 ±0.32	62.97 ±0.51	63.91 ±0.22	47.22 ±0.17	51.15 ±0.21	51.27 ±0.30
MOON	0.1	61.57 ±5.80	71.55 ±2.42	72.54 ±3.35	48.84 ±1.78	57.11 ±1.76	58.18 ±1.85	44.42 ±0.88	47.36 ±1.13	47.82 ±1.27
	0.6	77.85 ±3.09	85.12 ±2.61	85.90 ±2.58	54.27 ±0.89	62.33 ±1.12	62.92 ±1.24	47.21 ±0.88	51.48 ±0.68	52.24 ±0.62
	IID	79.35 ±0.29	88.98 ±0.79	89.22 ±0.84	54.77 ±0.17	63.91 ±0.11	64.08 ±0.09	48.06 ±0.19	51.98 ±0.10	52.35 ±0.11
FedDyn	0.1	64.42 ±2.66	76.43 ±2.25	77.69 ±2.19	49.97 ±1.09	59.37 ±1.29	61.12 ±1.26	32.60 ±0.43	42.55 ±0.81	43.93 ±0.82
	0.6	81.43 ±1.83	88.29 ±1.77	89.89 ±1.65	55.39 ±0.96	63.37 ±1.14	64.05 ±1.10	44.32 ±1.35	45.04 ±1.38	45.41 ±1.31
	IID	81.99 ±0.93	90.14 ±0.32	90.63 ±0.35	55.90 ±1.06	63.79 ±0.46	64.18 ±0.73	45.17 ±0.59	45.63 ±0.42	46.07 ±0.52
FedGen	0.1	66.79 ±5.12	74.13 ±3.31	77.58 ±3.01	47.86 ±1.86	57.20 ±1.95	59.91 ±1.84	43.53 ±1.13	47.51 ±1.42	48.01 ±1.37
	0.6	78.69 ±1.20	86.49 ±1.52	88.63 ±1.40	55.14 ±1.27	61.71 ±1.50	62.66 ±1.41	48.59 ±1.39	51.37 ±1.56	51.86 ±1.46
	IID	80.33 ±0.27	87.53 ±0.87	89.04 ±0.82	56.73 ±0.15	63.05 ±0.13	63.64 ±0.13	49.81 ±0.41	51.83 ±0.51	52.17 ±0.44
FedFTG	0.1	64.04 ±4.14	72.24 ±2.29	76.42 ±3.23	46.52 ±1.51	59.29 ±1.57	59.95 ±1.67	41.62 ±1.34	47.64 ±1.40	47.73 ±1.53
	0.6	79.65 ±1.91	86.59 ±1.83	88.64 ±1.86	56.15 ±0.89	63.54 ±0.50	64.17 ±0.62	47.77 ±0.72	51.87 ±0.97	52.16 ±1.09
	IID	82.08 ±0.08	88.41 ±0.26	90.73 ±0.17	57.07 ±0.37	63.79 ±0.24	64.55 ±0.27	49.23 ±0.38	52.55 ±0.42	52.75 ±0.55
FedLMG	0.1	65.59 ±5.79	76.25 ±2.31	76.39 ±2.11	50.54 ±0.87	60.21 ±0.45	61.29 ±1.11	41.33 ±1.58	48.49 ±1.56	48.68 ±1.50
	0.6	78.11 ±1.39	86.72 ±1.49	87.78 ±1.38	56.13 ±1.22	63.49 ±0.72	64.22 ±0.55	49.27 ±0.43	51.29 ±0.79	51.84 ±0.75
	IID	83.13 ±0.14	88.79 ±0.41	89.73 ±0.36	56.89 ±0.28	64.08 ±0.27	64.90 ±0.21	50.32 ±0.41	51.78 ±0.39	52.07 ±0.33
FedProto	0.1	66.08 ±5.42	70.31 ±1.91	76.40 ±2.64	43.33 ±0.99	52.88 ±1.17	56.88 ±1.27	41.00 ±1.27	47.59 ±1.30	47.97 ±1.40
	0.6	77.39 ±1.16	84.43 ±1.28	87.39 ±1.32	53.43 ±2.17	59.67 ±1.88	60.24 ±1.90	48.62 ±0.85	50.27 ±1.00	50.92 ±0.58
	IID	78.44 ±0.68	87.51 ±0.97	89.52 ±0.53	54.16 ±1.03	60.13 ±0.88	60.64 ±0.47	49.09 ±0.31	50.46 ±0.36	51.06 ±0.28
FedTGP	0.1	66.82 ±3.61	73.47 ±1.45	79.39 ±2.41	49.04 ±0.74	60.83 ±0.94	62.57 ±1.08	39.58 ±2.47	47.55 ±0.71	48.08 ±0.88
	0.6	79.09 ±1.38	86.57 ±1.35	89.22 ±1.41	55.91 ±0.52	63.59 ±0.74	63.90 ±0.87	48.65 ±0.71	51.37 ±0.84	51.78 ±0.96
	IID	81.28 ±0.23	88.64 ±0.32	90.97 ±0.14	56.49 ±0.22	64.10 ±0.24	64.87 ±0.25	49.21 ±0.14	51.86 ±0.17	52.10 ±0.19
FedCPD	0.1	63.79 ±5.01	72.92 ±3.04	76.51 ±2.01	48.10 ±0.90	57.00 ±1.28	59.56 ±1.12	43.08 ±0.89	47.83 ±1.22	48.16 ±1.07
	0.6	79.45 ±1.53	86.65 ±0.91	88.57 ±0.94	56.15 ±0.84	62.05 ±0.75	63.74 ±0.67	47.59 ±0.82	51.46 ±1.21	51.85 ±1.03
	IID	80.17 ±0.55	87.43 ±0.49	89.01 ±0.33	56.43 ±0.31	63.41 ±0.28	64.59 ±0.27	48.97 ±0.23	51.93 ±0.41	51.95 ±0.19
FedRG	0.1	65.73 ±2.31	75.47 ±1.84	78.15 ±0.43	49.05 ±0.74	59.40 ±0.58	61.06 ±0.52	43.52 ±0.73	47.69 ±0.58	47.87 ±0.37
	0.6	79.51 ±1.53	87.32 ±0.82	89.59 ±0.54	56.02 ±0.33	62.49 ±0.62	63.61 ±0.54	48.86 ±0.76	50.91 ±0.57	51.31 ±0.21
	IID	81.69 ±0.48	88.41 ±0.24	89.87 ±0.17	56.77 ±0.21	63.08 ±0.35	63.97 ±0.13	49.19 ±0.54	51.13 ±0.47	51.62 ±0.26
FedTau	0.1	**79.14** ±2.05 **↑12.32**	**84.72** ±1.23 **↑8.29**	**86.89** ±0.78 **↑7.50**	**53.79** ±1.11 **↑3.25**	**63.33** ±0.68 **↑2.50**	**64.86** ±0.86 **↑2.29**	**47.89** ±1.33 **↑3.47**	**51.54** ±0.81 **↑3.05**	**51.68** ±0.51 **↑3.00**
	0.6	**85.65** ±1.21 **↑4.22**	**90.94** ±0.79 **↑2.65**	**91.83** ±0.45 **↑1.94**	**58.17** ±1.08 **↑2.02**	**64.84** ±0.62 **↑1.25**	**65.40** ±0.47 **↑1.18**	**51.23** ±0.93 **↑1.96**	**53.67** ±0.36 **↑1.80**	**54.00** ±0.50 **↑1.76**
	IID	**87.00** ±1.35 **↑3.87**	**92.37** ±0.41 **↑2.23**	**92.89** ±0.69 **↑1.92**	**58.91** ±0.79 **↑1.84**	**65.27** ±0.28 **↑1.17**	**65.94** ±0.59 **↑1.04**	**51.99** ±0.63 **↑1.67**	**53.97** ±0.45 **↑1.42**	**54.15** ±0.11 **↑1.40**

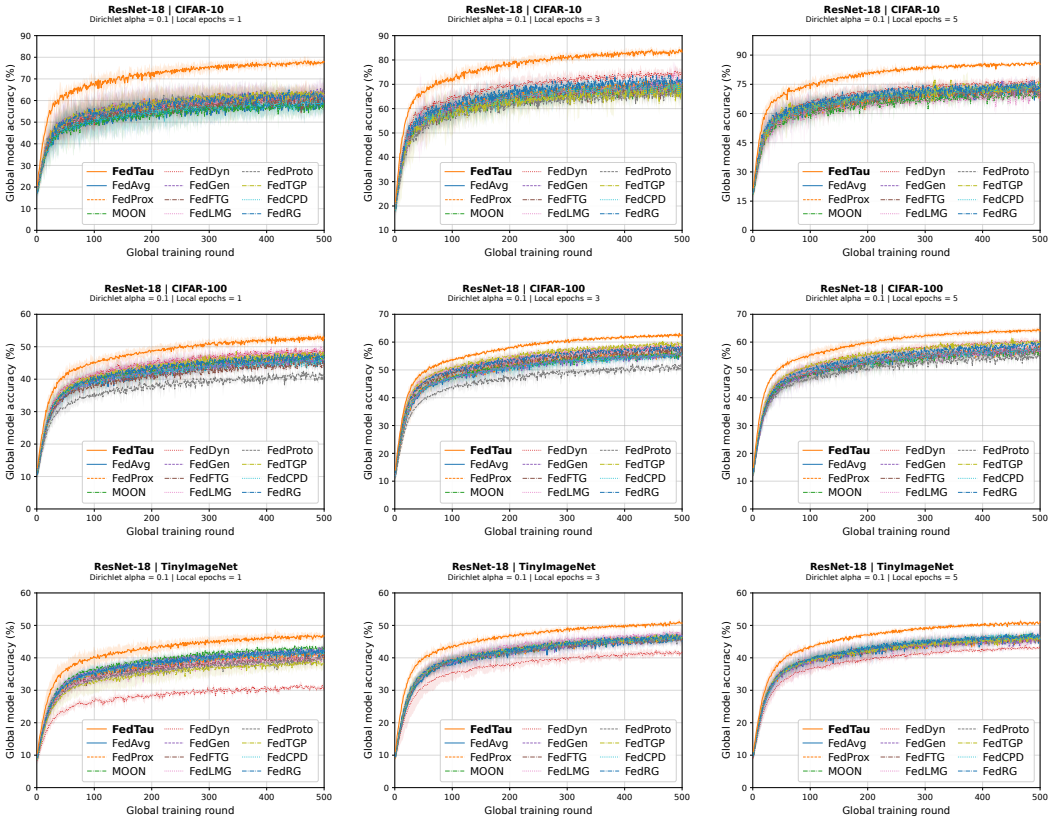


Figure 2: Illustrative training trajectory for the ResNet-18 classification settings. This page shows the Dirichlet $\alpha = 0.1$ setting. Within each row, the columns correspond to local epochs $E = 1$, $E = 3$, and $E = 5$, respectively. The rows correspond to CIFAR-10, CIFAR-100, and TinyImageNet.

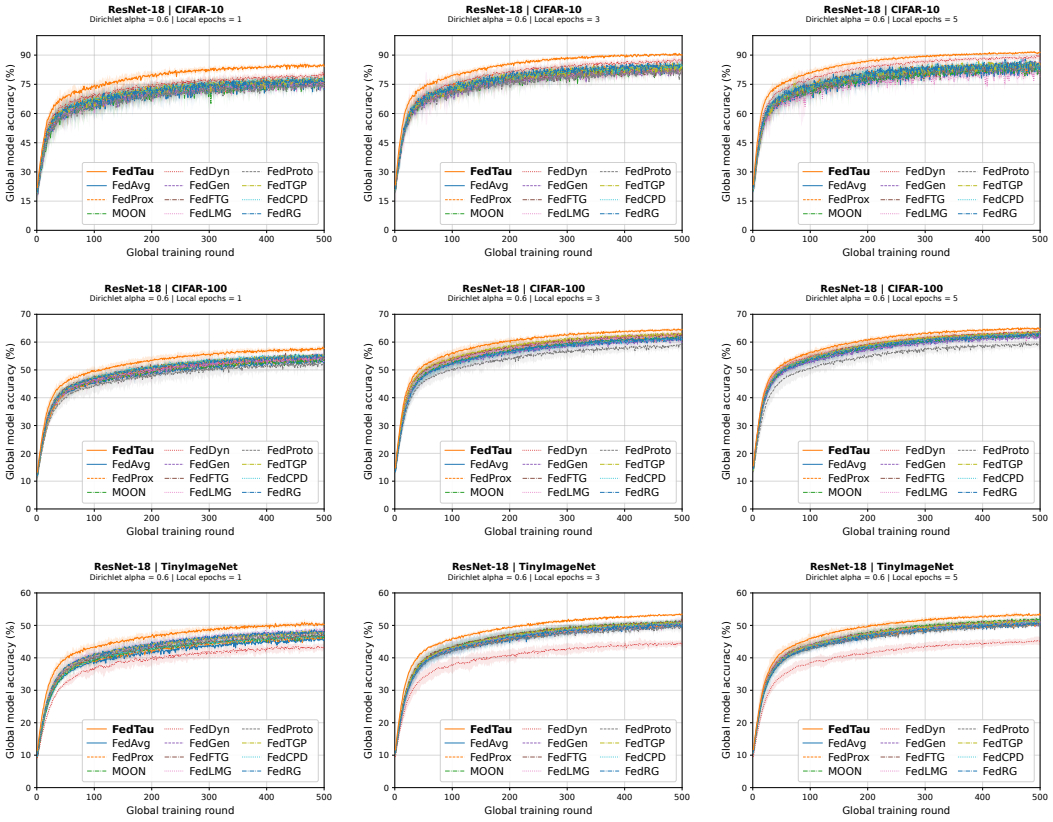


Figure 2: Illustrative trajectory reconstructions for the ResNet-18 classification settings. This page shows the Dirichlet $\alpha = 0.6$ setting. Within each row, the columns correspond to $E = 1$, $E = 3$, and $E = 5$, while the rows correspond to CIFAR-10, CIFAR-100, and TinyImageNet.

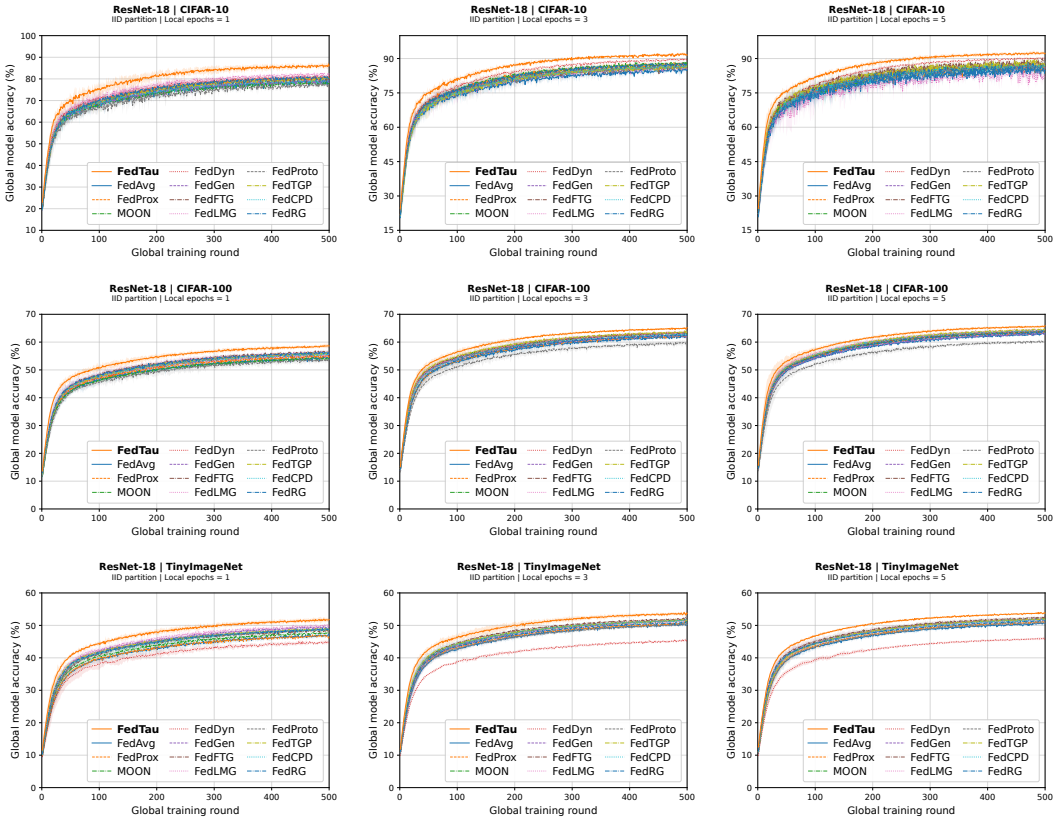


Figure 2: Illustrative trajectory reconstructions for the ResNet-18 classification settings. This page shows the IID setting. Within each row, the columns correspond to $E = 1$, $E = 3$, and $E = 5$, while the rows correspond to CIFAR-10, CIFAR-100, and TinyImageNet.

Table 10: StageCNN classification accuracy (%). E denotes local epochs and α the Dirichlet concentration. Best/second-best means are bold/boxed; ↑ Δ is the margin over the strongest federated baseline in percentage points. Centralized results are excluded from ranking.

		CIFAR-10			**CIFAR-100**			**TinyImageNet**		
		Centralized: **92.49±0.14**			Centralized: **70.34±0.40**			Centralized: **56.07±0.35**		
Method	α	E=1	E=3	E=5	E=1	E=3	E=5	E=1	E=3	E=5
FedAvg	0.1	53.23 ±3.47	60.41 ±2.38	62.74 ±2.62	40.09 ±2.18	52.77 ±1.68	53.07 ±1.22	32.22 ±1.91	34.06 ±1.86	35.19 ±1.77
	0.6	75.35 ±4.07	80.52 ±2.11	81.94 ±1.92	45.43 ±1.99	54.80 ±1.18	55.14 ±0.79	34.38 ±1.38	36.02 ±1.56	36.43 ±1.24
	IID	76.19 ±1.35	82.07 ±1.99	82.31 ±1.07	46.02 ±1.00	55.41 ±0.48	55.67 ±0.44	35.91 ±0.34	37.13 ±1.60	37.22 ±1.62
FedProx	0.1	52.49 ±4.31	61.96 ±2.32	63.85 ±4.04	39.85 ±2.28	53.59 ±0.94	54.28 ±1.33	32.71 ±1.47	35.11 ±1.36	35.92 ±1.13
	0.6	75.58 ±4.09	81.35 ±2.43	81.91 ±1.37	46.27 ±1.57	55.76 ±1.03	56.28 ±0.94	35.43 ±0.89	36.88 ±1.00	37.47 ±1.30
	IID	76.79 ±2.15	82.07 ±1.15	82.35 ±0.88	46.75 ±0.45	56.34 ±0.42	56.69 ±0.63	37.39 ±2.28	38.03 ±1.26	38.19 ±0.98
MOON	0.1	54.15 ±2.14	67.25 ±5.12	73.54 ±4.40	42.02 ±2.26	53.29 ±1.43	54.85 ±1.23	33.89 ±1.37	35.97 ±2.42	37.57 ±1.68
	0.6	77.24 ±1.15	84.68 ±3.55	85.37 ±2.30	44.90 ±0.90	54.85 ±0.74	55.24 ±0.82	37.11 ±1.43	38.62 ±1.48	39.52 ±1.53
	IID	77.90 ±1.13	85.00 ±0.28	85.44 ±0.25	45.02 ±0.74	55.37 ±0.55	55.64 ±0.67	38.90 ±2.28	40.35 ±1.44	40.76 ±1.20
FedDyn	0.1	52.59 ±2.35	72.41 ±4.55	74.47 ±5.41	42.55 ±1.01	54.73 ±1.39	56.46 ±0.74	22.92 ±1.01	27.55 ±1.49	29.75 ±1.08
	0.6	78.74 ±1.22	85.66 ±1.17	86.13 ±0.34	47.54 ±1.32	56.22 ±1.16	57.43 ±0.79	29.09 ±1.15	32.49 ±1.93	33.90 ±1.65
	IID	79.65 ±1.02	86.18 ±0.62	86.30 ±0.47	49.16 ±1.27	57.71 ±0.83	58.15 ±2.28	32.54 ±1.10	35.37 ±1.85	37.46 ±1.53
FedGen	0.1	52.31 ±4.04	70.13 ±4.20	75.48 ±5.34	39.22 ±1.80	51.43 ±1.74	54.69 ±0.86	33.77 ±1.34	35.68 ±1.26	36.71 ±1.05
	0.6	74.57 ±3.87	81.45 ±2.23	81.60 ±0.88	45.14 ±1.30	53.62 ±1.36	55.74 ±0.36	36.28 ±1.36	37.16 ±1.15	37.85 ±1.41
	IID	77.51 ±3.80	82.05 ±1.33	82.14 ±0.80	46.14 ±0.42	55.97 ±0.18	56.10 ±0.26	37.84 ±1.31	38.61 ±1.30	39.07 ±1.26
FedFTG	0.1	53.58 ±5.83	68.24 ±4.39	72.62 ±4.50	38.69 ±4.09	51.74 ±1.75	55.37 ±1.04	29.54 ±1.41	32.58 ±2.27	33.85 ±2.00
	0.6	75.92 ±3.39	80.96 ±1.88	81.24 ±1.79	47.64 ±1.04	56.88 ±1.31	57.60 ±1.47	34.95 ±2.08	36.29 ±2.03	36.81 ±1.19
	IID	76.58 ±1.62	81.57 ±0.62	81.78 ±0.64	48.23 ±0.28	58.32 ±1.19	58.41 ±0.95	38.21 ±1.38	39.22 ±1.40	39.97 ±1.51
FedLMG	0.1	56.30 ±2.73	72.25 ±4.17	74.40 ±4.25	36.71 ±3.13	53.47 ±1.65	55.75 ±1.02	30.68 ±1.41	34.52 ±1.87	37.20 ±1.63
	0.6	76.33 ±3.46	84.53 ±1.54	84.99 ±0.89	45.63 ±2.62	56.42 ±1.55	56.47 ±0.88	35.04 ±0.99	38.59 ±1.10	40.86 ±1.22
	IID	78.34 ±1.30	85.02 ±0.73	85.28 ±0.51	46.59 ±0.14	57.28 ±1.30	58.38 ±1.21	38.86 ±1.72	41.26 ±1.55	41.86 ±1.65
FedProto	0.1	53.82 ±4.73	66.71 ±3.95	74.99 ±5.02	36.76 ±1.01	52.29 ±2.32	54.03 ±1.35	30.51 ±1.07	33.10 ±1.04	34.04 ±1.68
	0.6	75.28 ±3.50	80.93 ±1.93	81.74 ±1.13	42.92 ±0.80	55.73 ±2.15	56.19 ±1.55	33.15 ±1.16	35.43 ±1.35	36.29 ±1.25
	IID	76.14 ±1.31	81.53 ±1.54	82.17 ±0.12	44.03 ±0.26	56.47 ±1.40	56.62 ±1.35	35.47 ±1.53	37.02 ±1.49	37.58 ±1.36
FedTGP	0.1	54.40 ±4.18	70.77 ±2.85	76.89 ±4.57	38.39 ±0.94	53.48 ±1.37	55.62 ±1.39	29.13 ±0.73	35.02 ±2.17	36.55 ±2.04
	0.6	75.81 ±1.79	85.58 ±0.87	85.93 ±0.17	46.55 ±0.68	56.27 ±0.93	56.50 ±1.01	34.18 ±0.40	38.94 ±1.15	39.61 ±1.05
	IID	76.13 ±0.34	85.72 ±0.34	86.12 ±0.09	47.00 ±0.19	56.77 ±0.42	57.00 ±0.46	39.66 ±0.56	40.68 ±1.90	41.25 ±1.53
FedCPD	0.1	56.18 ±4.46	69.85 ±3.67	74.21 ±2.88	41.76 ±1.12	54.13 ±1.05	54.51 ±0.97	33.23 ±1.20	35.82 ±2.11	37.10 ±2.44
	0.6	77.68 ±2.36	84.23 ±0.91	85.14 ±0.76	46.46 ±1.16	56.75 ±0.92	57.43 ±0.80	36.49 ±0.64	38.14 ±1.54	39.21 ±2.19
	IID	79.10 ±2.15	85.33 ±0.09	85.54 ±0.11	47.15 ±0.20	57.86 ±0.90	58.03 ±0.74	39.07 ±0.73	40.05 ±2.08	40.47 ±2.10
FedRG	0.1	54.72 ±2.44	64.38 ±2.39	65.63 ±2.41	42.27 ±1.36	53.82 ±1.41	54.22 ±1.17	35.62 ±1.36	38.87 ±1.74	40.53 ±1.99
	0.6	76.62 ±2.59	82.79 ±0.31	83.25 ±0.36	46.59 ±0.83	56.20 ±1.25	56.52 ±0.84	38.35 ±1.02	40.31 ±1.79	41.29 ±1.34
	IID	77.03 ±0.79	83.57 ±0.20	83.72 ±0.08	47.03 ±0.92	57.02 ±0.83	57.18 ±0.65	40.79 ±1.16	41.54 ±1.88	42.07 ±1.83
FedTau	0.1	**70.58** ±1.19 **↑14.28**	**79.00** ±1.02 **↑6.59**	**79.87** ±0.61 **↑2.98**	**51.69** ±2.18 **↑9.14**	**60.88** ±0.49 **↑6.15**	**62.05** ±0.34 **↑5.59**	**44.74** ±3.34 **↑9.12**	**46.36** ±2.86 **↑7.49**	**46.53** ±1.17 **↑6.00**
	0.6	**83.59** ±1.70 **↑4.85**	**89.74** ±0.59 **↑4.08**	**89.84** ±0.62 **↑3.71**	**54.23** ±0.84 **↑6.59**	**62.72** ±0.65 **↑5.84**	**63.02** ±0.54 **↑5.42**	**46.92** ±1.08 **↑8.57**	**47.06** ±0.59 **↑6.75**	**47.28** ±0.50 **↑5.99**
	IID	**85.17** ±1.20 **↑5.52**	**90.43** ±0.40 **↑4.25**	**90.72** ±0.22 **↑4.42**	**55.09** ±0.89 **↑5.93**	**63.41** ±0.62 **↑5.09**	**63.75** ±0.36 **↑5.34**	**47.68** ±1.20 **↑6.89**	**47.97** ±1.64 **↑6.43**	**48.03** ±1.77 **↑5.96**

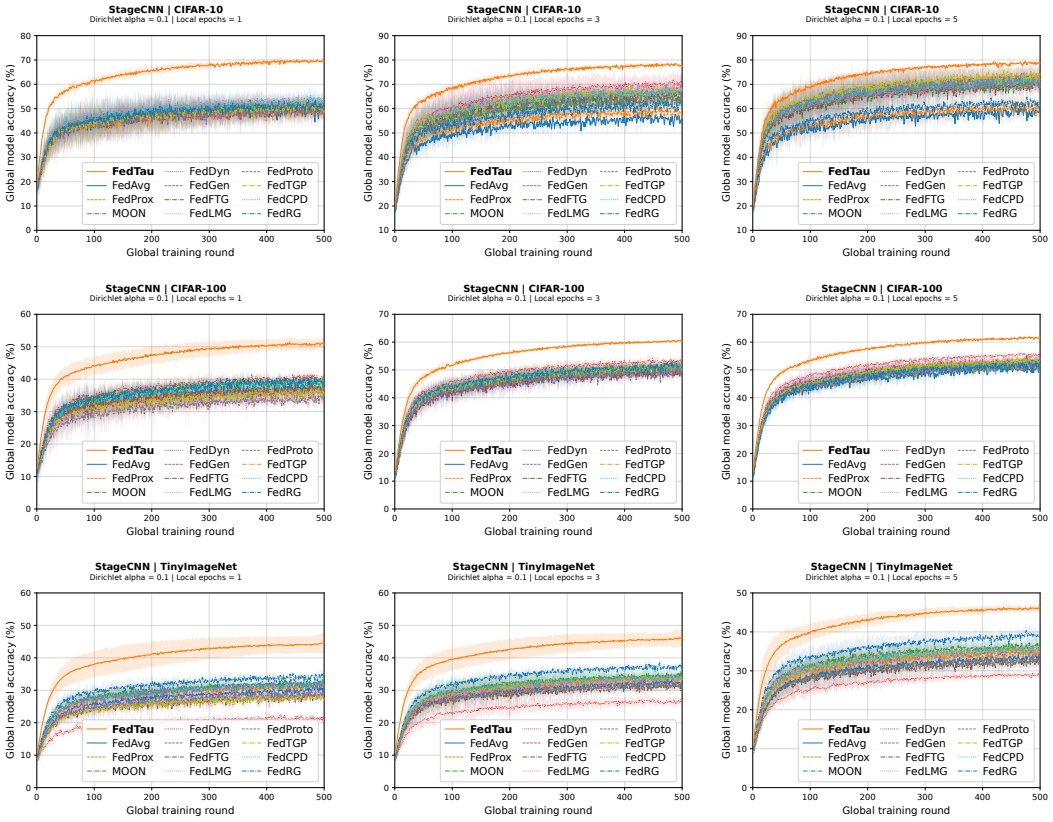


Figure 3: Illustrative training trajectory for the StageCNN classification settings. This page shows the Dirichlet $\alpha = 0.1$ setting. Within each row, the columns correspond to local epochs $E = 1$, $E = 3$, and $E = 5$, respectively. The rows correspond to CIFAR-10, CIFAR-100, and TinyImageNet.

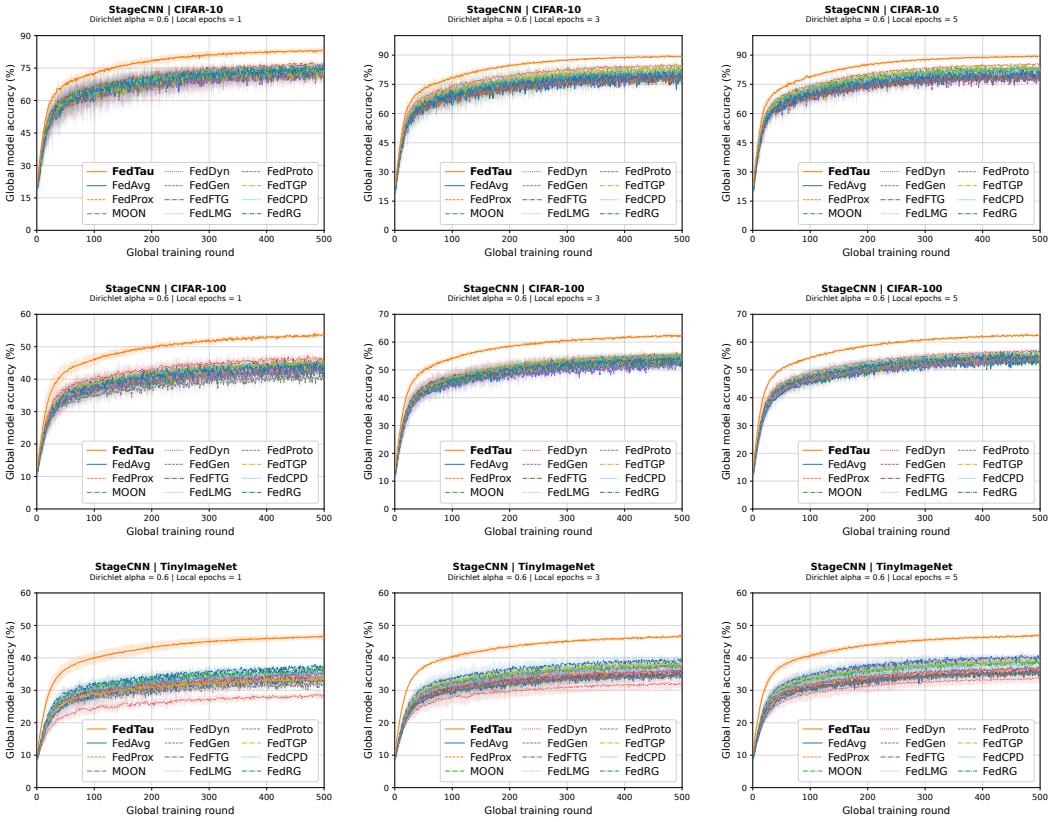


Figure 3: Illustrative trajectory reconstructions for the StageCNN classification settings. This page shows the Dirichlet $\alpha = 0.6$ setting. Within each row, the columns correspond to $E = 1$, $E = 3$, and $E = 5$, while the rows correspond to CIFAR-10, CIFAR-100, and TinyImageNet.

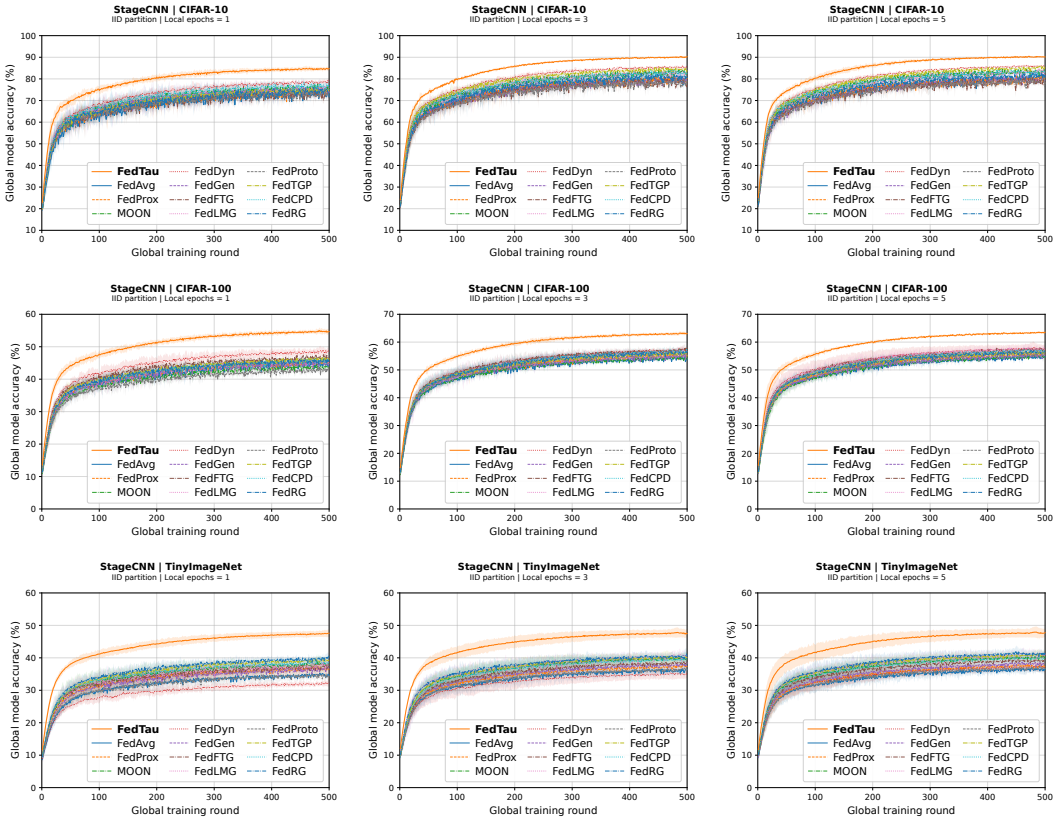


Figure 3: Illustrative trajectory reconstructions for the StageCNN classification settings. This page shows the IID setting. Within each row, the columns correspond to $E = 1$, $E = 3$, and $E = 5$, while the rows correspond to CIFAR-10, CIFAR-100, and TinyImageNet.

examine whether the benefit changes when the recording mechanism affects the regression target differently.

Why right censoring is the primary setting. Censoring is a standard observation model in limited-dependent-variable regression (Tobin, 1958). Right censoring also arises directly from incomplete follow-up in survival prediction, including established neural survival models (Katzman et al., 2018). These applications motivate distinguishing an observed bound from an observed response. Our primary setting tests precisely this distinction: a client records that a response exceeds its reporting limit, but an ordinary point-target objective treats the limit as the response itself.

The emphasis on right censoring follows the mechanism addressed by FedTau, rather than a claim that other incomplete-data problems are unimportant. Outcome-dependent selection and grouped or interval observations are established statistical settings (Heckman, 1979; Turnbull, 1976). Here they serve as complementary tests because retaining an exact selected response or a bounded interval need not produce the same one-sided target error as replacing an upper-tail response by a cap.

The censoring and quantization experiments use observed-event likelihoods. The outcome-selection experiment uses an IPW estimating objective, detailed in Appendix A.7.7; its population justification is a risk-reweighting identity rather than the KL identity for Eq. (3).

A.7.2 Data splits, target coordinates, and local training

Datasets and retained features. California Housing uses its eight numerical predictors and the released house-value target (Pace & Barry, 1997). Superconductivity uses the 81 predictors in `train.csv` to predict critical temperature (Hamidieh, 2018). Bike Sharing uses the hourly records in `hour.csv` and predicts `cnt` (Fanaee-T & Gama, 2014). Its input encodes season, month, hour, weekday, and weather category by one-hot vectors and retains year, holiday, working-day status, temperature, apparent temperature, humidity, and wind speed, giving 58 features. The component counts `casual` and `registered` are not predictors because their sum is the response. All splits are formed before the additional observation mechanism is applied, and all numerical normalization statistics are computed from the training split alone.

Right-censoring splits. California Housing uses a seeded random $80/20$ training/test split, with $16{,}512$ training and $4{,}128$ test examples and no separate validation subset. The Superconductivity and Bike Sharing right-censoring experiments use seeded random $70/10/20$ training/validation/test splits. The retained counts are $14{,}884/2{,}126/4{,}253$ for Superconductivity and $12{,}165/1{,}738/3{,}476$ for Bike Sharing. The Bike Sharing comparison is therefore a random-split regression experiment, not chronological forecasting. Validation and test responses are not subjected to the added noise or the client recording limit.

Model coordinates and reported units. Throughout this subsection, y denotes the released benchmark response in its original units, whereas r denotes the response used in local optimization. Let g be the target transformation, and let m_y and $s_y > 0$ be the training mean and standard deviation of $g(y)$. The model coordinate and its inverse are

$$r = \frac{g(y) - m_y}{s_y}, \qquad \hat{y} = g^{-1}(m_y + s_y \mu_\theta(x)) \,. \tag{33}$$

Each input feature is likewise standardized using its training mean and standard deviation; a nearly constant feature uses unit scale. For right censoring, $g(y) = y$ on California Housing and Superconductivity, while $g(y) = \log(1 + y)$ on Bike Sharing. The likelihood, the censoring threshold, and the added noise are all defined in the r coordinate. RMSE, MAE, and R^2 are evaluated after inversion, against the uncorrupted benchmark responses in their original units. The UCI right-censoring evaluator clips negative temperature or count predictions to zero; the Bike Sharing inverse additionally caps the log-coordinate argument at 20 before applying `expm1`. No client-specific recording map is applied at inference. For Bike Sharing, this inverse-transformed point prediction is not asserted to equal the original-scale conditional mean: the common-mean interpretation of the working likelihood is in the transformed coordinate.

Client allocation and optimization. The regression protocols use 100 clients and sample 10 clients uniformly without replacement in each round. Before censoring, selection, or quantization, a

seeded permutation of the training indices is divided into client subsets whose sizes differ by at most one. Censoring and quantization retain every assigned record; selection can reduce client sample counts by different amounts. Thus equal nominal client sizes do not imply identical recorded-data distributions or, after selection, equal observed sample counts. Each selected client performs five minibatch updates per round, using batch size 64, rather than five full local epochs. A loader is restarted when those updates exhaust its examples. The regression MLP has three ReLU hidden layers and a scalar output: hidden widths are $128, 128, 64$ for California Housing and Bike Sharing right censoring, and $256, 256, 128$ for Superconductivity right censoring. The FedTau implementation uses AdamW with learning rate 10^{-3}, weight decay 10^{-5}, and gradient clipping at norm 5. Its optimizer is initialized at each local update phase, and evaluation uses batch size 512. The right-censoring comparison has a 500-round horizon; selection and quantization use 200 rounds. Method-specific baseline regularizers and server updates are retained. The controlled comparison fixes the data, recorded observations, initialization, selected-client sequence, and local-update budget within each setting and seed.

Shared selection and quantization protocol. For the selection and quantization comparisons, the shared protocol specified here applies the California Housing settings to all three datasets: a seeded $80/20$ training/test split, no additional validation split, and direct standardization of the original response ($g(y) = y$). The resulting training/test counts are $16{,}512/4{,}128$, $17{,}010/4{,}253$, and $13{,}903/3{,}476$ for California Housing, Superconductivity, and Bike Sharing, respectively. All three use the $128, 128, 64$ MLP, 100 nominally balanced clients, 10% participation, five local minibatch updates, 200 rounds, and the AdamW and batch-size settings above. For these two mechanisms, inverse standardization is used directly, without the log transform or the UCI-specific output clipping. The selection parameters and quantizer parameters below are shared across datasets in standardized-target units, rather than retuned to their physical response scales. This protocol is separate from the dataset-specific right-censoring splits and the Bike Sharing log-target transformation.

A.7.3 Reference-response metrics.

For a test set containing N reference outcomes, let $\hat{y}_i$ denote the prediction for input x_i and let $\bar{y} = N^{-1}\sum_{i=1}^{N} y_i$. For FedTau, $\hat{y}_i$ is the original-scale prediction obtained from Eq. (33), including the specified output clipping. No client-specific recording map is applied at evaluation. The evaluation metrics are

$$\mathrm{RMSE} = \left[\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2\right]^{1/2}, \qquad \mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|\hat{y}_i - y_i|, \tag{34}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2}. \tag{35}$$

RMSE emphasizes larger prediction errors, whereas MAE measures average absolute deviation with less emphasis on large residuals. R^2 expresses squared error relative to the variation in the reference outcomes. On a fixed test set, RMSE and R^2 are deterministically related; their agreement should therefore not be interpreted as two independent statistical findings. Relative error reductions are computed within each dataset, avoiding comparisons between absolute errors measured on different response scales.

Five-seed best-score definition. For a five-seed best-score summary, the statistic is selected within each seed before averaging across seeds. Writing $M_{s,t}$ for metric M at evaluated round t in seed $s \in \{1, \ldots, 5\}$, define

$$b_s(M) = \begin{cases} \min_t M_{s,t}, & M \in \{\mathrm{RMSE}, \mathrm{MAE}\}, \\ \max_t M_{s,t}, & M = R^2. \end{cases} \tag{36}$$

The corresponding mean and sample standard deviation are

$$\bar{b}(M) = \frac{1}{5}\sum_{s=1}^{5} b_s(M), \qquad \mathrm{SD}(M) = \left[\frac{1}{4}\sum_{s=1}^{5}\big(b_s(M) - \bar{b}(M)\big)^2\right]^{1/2}. \tag{37}$$

Best values are selected separately for each metric, so the best MAE need not occur at the best-RMSE checkpoint. Checkpoint and late-window summaries retain their respective definitions.

A.7.4 BASELINE SELECTION AND CONTROLLED COMPARISON

Canonical and optimization-based alternatives. We compare against twelve federated baselines spanning the main applicable alternatives to mechanism-aware local training. FedAvg (McMahan et al., 2017) supplies the canonical model-averaging reference and the closest comparison for assessing the change in local supervision. FedProx (Li et al., 2020), FedDyn (Acar et al., 2021), and SCAFFOLD (Karimireddy et al., 2020) address heterogeneous local optimization through proximal regularization, dynamic regularization, and control variates, respectively. They are particularly relevant because they test whether reducing optimization inconsistency is sufficient when the recorded responses have client-dependent meanings.

FedNova (Wang et al., 2020) addresses objective inconsistency caused by heterogeneous local update counts through normalized aggregation. Its inclusion separates update-normalization effects from correction of the response observation model. FedAvgM (Hsu et al., 2019) and the adaptive server methods FedAdagrad, FedAdam, and FedYogi (Reddi et al., 2021) test momentum and coordinate-wise adaptation at the server. Together, these methods provide established, theoretically motivated comparisons beyond unmodified averaging. MOON (Li et al., 2021) adds a representation-level comparison through model-contrastive regularization. Its contrastive component acts on representations and can therefore be paired with a scalar prediction objective without introducing class-indexed target prototypes. FedDAAW (Tang, 2026) provides a recent cross-task alternative based on client selection and performance-adaptive aggregation; its reported applications include California Housing regression. Its relevance here is the use of client-performance information to adjust learning contributions, rather than an established state-of-the-art claim for censored regression. FedLSBT (Passeri et al., 2023) is included because it directly targets federated regression using least-squares boosted trees and was evaluated on California Housing in its original study. It provides a complementary model-family comparison, ensuring that the evaluation is not restricted to neural parameter-averaging methods. Its role differs from that of FedAvg: FedAvg is the direct averaging reference, whereas FedLSBT tests competitiveness against a regression-specific tree ensemble.

Why some classification methods are not direct baselines. The published class-prototype formulations of FedProto and FedTGP (Tan et al., 2022; Zhang et al., 2024) require discrete class identities. Applying them to a continuous response would require an additional discretization rule and a corresponding definition of regression prototypes. Similarly, the local-model-guided diffusion framework of FedLMG (Yang et al., 2025) is designed around synthetic image generation and one-shot knowledge transfer. Replacing these components with a tabular conditional generator would constitute a new adaptation rather than a direct implementation. We therefore prioritize established optimization methods and explicitly regression-capable alternatives whose relevance to the prediction task is clear.

A.7.5 COMPARISON CONTROL.

The shared-data and shared-budget protocol provides the basis for comparing the evaluated configurations. Method-specific penalties, server updates, and model families are the competing algorithmic choices, not quantities that must be made identical. In particular, FedLSBT need not be forced into a neural architecture to provide a meaningful regression comparison. Conversely, a fixed client-participation schedule evaluates a fixed-schedule configuration of any method whose full formulation also contains a client-selection policy; it does not separately assess that selection policy.

FedTau's right-censoring objective requires only the recorded input, the capped response, the censoring indicator, and the threshold. It does not require the unrecorded response above the threshold. Using this information to evaluate the probability of the recorded event is the intervention under study. The Clean oracle control serves a different purpose because its pre-noise, uncensored responses contain information unavailable in the ordinary recorded training set; it is therefore excluded from the ordinary baseline ranking.

The comparison concerns mechanism-aware supervision relative to the listed configurations. It does not treat the censored Gaussian likelihood itself as a new statistical estimator: that likelihood is classical (Tobin, 1958). FedTau places the appropriate observation model inside local training while retaining the federated update structure described in Section 4.

A.7.6 PRIMARY COMPARISON: HETEROGENEOUS RIGHT CENSORING

Observation construction. The recording mechanism is applied to a noisy standardized response, not directly to the original response y. For training example i assigned to client k, construct

$$v_i = r_i + \epsilon_i, \qquad \epsilon_i \sim \mathcal{N}(0, 0.15^2), \qquad u_i = \min(v_i, c_k), \qquad \delta_i = \mathbf{1}\{v_i \geq c_k\}. \tag{38}$$

The noise is generated once when the local dataset is built and is not resampled every round. To assign recording limits, form 100 equally spaced quantile levels between 0.60 and 0.90, randomly permute their client assignment, and set c_k to the corresponding empirical quantile of the pooled training values r_i before adding noise. An uncensored local record therefore contains the exact noisy value v_i, whereas a censored record contains only the bound $v_i \geq c_k$. All inputs and record counts are retained. The model is trained using (x_i, u_i, δ_i) and the known local threshold; the unobserved value above the threshold is not used in its local loss.

Likelihood and comparison target. In this experiment, the Gaussian working model is $V \mid X = x \sim \mathcal{N}(\mu_\theta(x), \sigma^2)$, with fixed model-coordinate scale $\sigma = 0.15$ in the default configuration. The event likelihood in Eq. (5) becomes

$$\ell_k^{\mathrm{FedTau}} = (1 - \delta_i) \left[\frac{(u_i - \mu_\theta(x_i))^2}{2\sigma^2} + \log \sigma \right] - \delta_i \log \Phi\left(\frac{\mu_\theta(x_i) - c_k}{\sigma} \right), \tag{39}$$

up to a parameter-independent constant. Uncensored records contribute a point density; censored records contribute the probability of the event $V \geq c_k$. The scale is a fixed working-likelihood choice, not an estimate that establishes the full predictive variance on these real datasets.

This objective addresses a directed target error. Because the added noise has zero conditional mean in model coordinates,

$$\mathbb{E}[U \mid X = x, k] = \mathbb{E}[R \mid X = x, k] - \mathbb{E}[(V - c_k)_+ \mid X = x, k]. \tag{40}$$

Point-target training on u_i therefore includes a client-dependent negative displacement even though the added noise is zero-mean. Changing the federated optimizer does not remove this displacement from the point-target population objective. FedTau instead uses the meaning of the recorded event in every local update, while performance is evaluated on the original, uncorrupted held-out responses after Eq. (33).

Mechanism controls. The Wrong threshold control evaluates the same censored records using $c_k + 0.75$ in place of c_k, with the likelihood scale unchanged. The point-imputation control uses $u_i + 0.5\delta_i$ as its squared-error target, with the offset measured in standardized-target units. We call this control *constant-offset imputation*; it is the row previously named Mean imputation, not an estimated conditional mean. Clean oracle uses the pre-noise, uncensored target r_i with the same client allocation and predictor. It supplies a fully observed training reference and is excluded from the ordinary baseline ranking. These controls separate correct event specification from a wrong threshold, a fixed point replacement, and access to uncorrupted labels.

Consistent advantages over the strongest listed baselines. Table 11 reports the best displayed mean for FedTau in all nine dataset–metric combinations. Importantly, the comparison remains favorable against the strongest baseline for each dataset, rather than only against FedAvg.

On California Housing, FedLSBT is the strongest listed baseline across the three metrics. FedTau reduces RMSE from 0.7300 to 0.6198 and MAE from 0.4698 to 0.4064, corresponding to relative reductions of 15.10% and 13.50%. The reported R^2 increases from 0.6105 to 0.7198. The advantage therefore extends to a regression-specific tree method, not just alternative neural aggregation rules.

On Superconductivity, the strongest listed baseline is FedYogi. FedTau lowers RMSE from 18.3792 to 17.1534 and MAE from 12.3551 to 10.7493, yielding reductions of 6.67% and 13.00%. The reported R^2 increases by 0.0644, from 0.7102 to 0.7746. Although the RMSE margin is smaller than on the other datasets, the improvement is not confined to one error criterion.

Bike Sharing exhibits the largest relative advantage. Against FedYogi, FedTau reduces RMSE from 112.1450 to 69.9637 and MAE from 59.1654 to 38.4882. These correspond to reductions of 37.61% and 34.95%, respectively. The reported R^2 increases from 0.6241 to 0.8613. The concurrent reduction in MAE shows that the advantage is not restricted to a metric that places disproportionate weight on large residuals.

Table 11: Comparison of FedTau with baselines under heterogeneous client-specific right-censoring mechanisms. Lower RMSE/MAE and higher R^2 are better. Boxed means mark the strongest baseline. The $\downarrow \Delta$ row reports FedTau's absolute improvement over the strongest baseline: a reduction in RMSE/MAE or an increase in R^2.

	California Housing			**Superconductivity**			**Bike Sharing**		
Method	RMSE↓	MAE↓	R^2 ↑	RMSE↓	MAE↓	R^2 ↑	RMSE↓	MAE↓	R^2 ↑
FedAvg	0.7858	0.4978	0.5482	19.3033	12.8355	0.6803	117.1295	61.2013	0.5899
	± 0.0091	± 0.0058	± 0.0105	± 1.3273	± 0.8826	± 0.0440	± 3.8439	± 2.0085	± 0.0269
FedNova	0.7850	0.4985	0.5500	19.3033	12.8355	0.6803	117.0783	61.1918	0.5904
	± 0.0072	± 0.0046	± 0.0083	± 1.4188	± 0.9434	± 0.0470	± 3.7508	± 1.9598	± 0.0263
MOON	0.7830	0.4975	0.5520	19.2689	12.8316	0.6814	116.9834	61.1884	0.5909
	± 0.0057	± 0.0036	± 0.0065	± 1.3224	± 0.8806	± 0.0437	± 2.6003	± 1.3601	± 0.0182
FedDAAW	0.7810	0.4964	0.5542	19.2267	12.8120	0.6828	116.7718	61.1152	0.5924
	± 0.0063	± 0.0040	± 0.0072	± 1.3713	± 0.9138	± 0.0452	± 3.0736	± 1.6086	± 0.0215
FedProx	0.7750	0.4933	0.5609	19.5166	12.9048	0.6729	117.2855	61.2771	0.5889
	± 0.0085	± 0.0054	± 0.0096	± 1.1630	± 0.7690	± 0.0390	± 2.6488	± 1.3839	± 0.0186
FedAdagrad	0.7700	0.4907	0.5667	19.0122	12.6982	0.6899	115.7164	60.6776	0.5997
	± 0.0025	± 0.0016	± 0.0028	± 1.1491	± 0.7675	± 0.0375	± 3.8825	± 2.0358	± 0.0269
SCAFFOLD	0.7650	0.4881	0.5722	18.9456	12.6604	0.6920	115.5793	60.6050	0.6007
	± 0.0011	± 0.0007	± 0.0012	± 2.4857	± 1.6611	± 0.0808	± 5.0637	± 2.6552	± 0.0350
FedAvgM	0.7620	0.4865	0.5756	18.7186	12.5441	0.6994	113.9367	59.9510	0.6120
	± 0.0037	± 0.0024	± 0.0041	± 1.1163	± 0.7481	± 0.0359	± 3.6418	± 1.9162	± 0.0248
FedDyn	0.7550	0.4829	0.5834	18.6602	12.5089	0.7012	113.6863	59.8274	0.6137
	± 0.0029	± 0.0019	± 0.0032	± 1.9943	± 1.3369	± 0.0639	± 2.7325	± 1.4380	± 0.0186
FedAdam	0.7480	0.4792	0.5911	18.4212	12.3799	0.7089	112.3403	59.2576	0.6228
	± 0.0057	± 0.0037	± 0.0062	± 2.4793	± 1.6662	± 0.0784	± 3.7662	± 1.9866	± 0.0253
FedYogi	0.7440	0.4771	0.5953	18.3792	12.3551	0.7102	112.1450	59.1654	0.6241
	± 0.0005	± 0.0003	± 0.0005	± 1.1539	± 0.7757	± 0.0364	± 2.8385	± 1.4975	± 0.0190
FedLSBT	0.7300	0.4698	0.6105	18.6179	12.4745	0.7026	113.8025	59.8243	0.6129
	± 0.0008	± 0.0005	± 0.0009	± 1.5529	± 1.0405	± 0.0496	± 2.3751	± 1.2486	± 0.0162
FedTau	**0.6198**	**0.4064**	**0.7198**	**17.1534**	**10.7493**	**0.7746**	**69.9637**	**38.4882**	**0.8613**
	± 0.0004	± 0.0003	± 0.0004	± 0.8736	± 0.5474	± 0.0230	± 2.0160	± 1.1090	± 0.0080
	↓ 0.1102	**↓ 0.0634**	**↓ 0.1093**	**↓ 1.2258**	**↓ 1.6058**	**↓ 0.0644**	**↓ 42.1813**	**↓ 20.6772**	**↓ 0.2372**

Across the three datasets, the reductions over the strongest listed baseline range from 6.67% to 37.61% in RMSE and from 13.00% to 34.95% in MAE. These within-dataset comparisons are more informative than comparing absolute error magnitudes across differently scaled responses. Their common direction is consistent with the same correction addressing client-dependent capping in multiple regression domains.

How Figure 4 supports interpretation. Figure 4 presents the three metrics side by side for each dataset. The displayed separation is especially pronounced on Bike Sharing, is also visible on California Housing, and is smaller on Superconductivity. This organization makes the magnitude and direction of the right-censoring comparison easier to inspect than an aggregate ranking alone.

The figure visualizes the reported separation between point-target training and event-aware supervision. Equation (40) explains the target mismatch: fitting capped responses introduces a threshold-dependent downward shift in the model response coordinate.

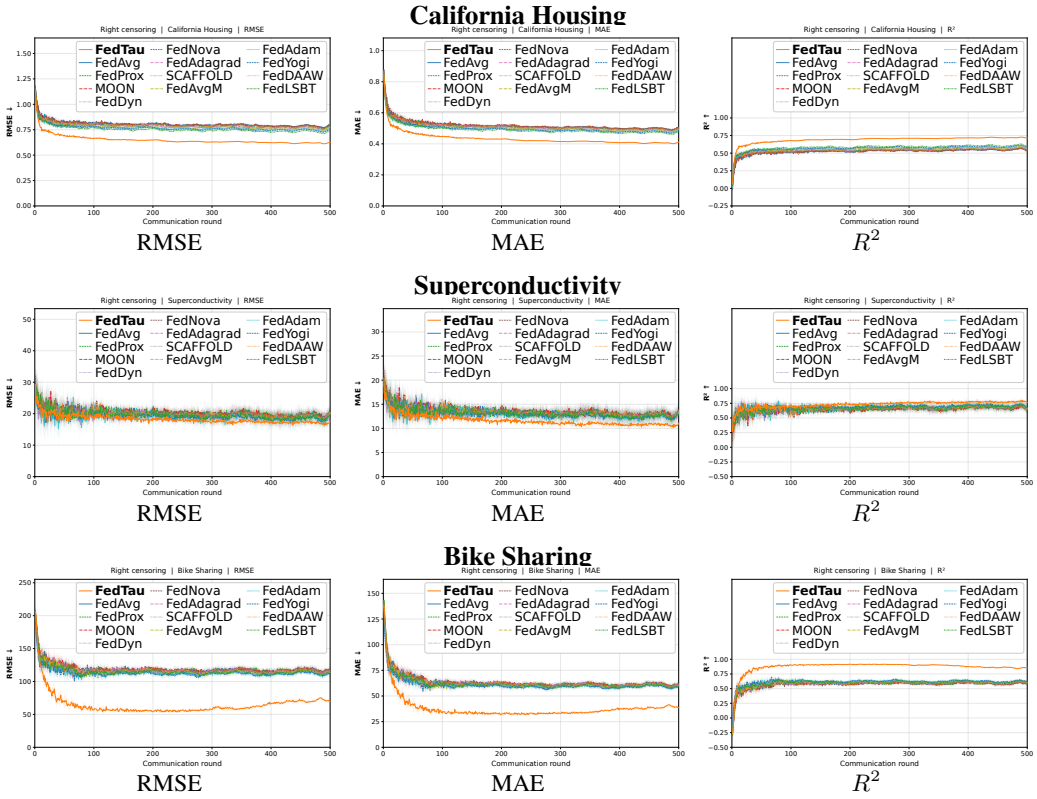


Figure 4: Training trajectories under heterogeneous client-specific right-censoring mechanisms. Each row corresponds to one regression dataset, while the columns report RMSE, MAE, and R^2, respectively. Shaded regions show their standard deviation.

Table 12: Comparison of FedTau with Wrong threshold, Constant-offset imputation, and Clean oracle under heterogeneous client-specific right-censoring mechanisms. Lower RMSE/MAE and higher R^2 are better. FedTau means are shown in bold.

	California Housing			Superconductivity			Bike Sharing		
Method	RMSE↓	MAE↓	R^2 ↑	RMSE↓	MAE↓	R^2 ↑	RMSE↓	MAE↓	R^2 ↑
Wrong threshold	0.6605	0.4333	0.6816	20.4592	13.0441	0.6697	133.9111	68.0152	0.4737
	± 0.0005	± 0.0003	± 0.0005	± 0.8535	± 0.5442	± 0.0276	± 2.0271	± 1.0296	± 0.0159
Constant-offset imputation	0.6479	0.4334	0.6940	18.7511	12.7700	0.6585	84.7096	45.3021	0.7946
	± 0.0008	± 0.0005	± 0.0008	± 0.8011	± 0.5213	± 0.0205	± 2.0074	± 1.0735	± 0.0097
FedTau	**0.6198**	**0.4064**	**0.7198**	**17.1534**	**10.7493**	**0.7746**	**69.9637**	**38.4882**	**0.8613**
	± 0.0004	± 0.0003	± 0.0004	± 0.8736	± 0.5474	± 0.0230	± 2.0160	± 1.1090	± 0.0080
Clean oracle	0.5167	0.3581	0.8054	15.0450	9.7911	0.8275	58.6466	33.9139	0.9038
	± 0.0007	± 0.0005	± 0.0005	± 0.8035	± 0.5229	± 0.0184	± 2.0418	± 1.1807	± 0.0067

Mechanism controls: correct specification matters. Table 12 complements the main comparison by separating three questions: whether the threshold is correctly specified, whether point imputation reproduces the result, and how much performance remains below a fully observed reference.

FedTau has lower reported RMSE than Wrong threshold on all three datasets: 0.6198 versus 0.6605 on California Housing, 17.1534 versus 20.4592 on Superconductivity, and 69.9637 versus 133.9111 on Bike Sharing. The corresponding relative reductions are 6.16%, 16.16%, and 47.75%. MAE and

R^2 also favor FedTau in every dataset. This ordering is consistent with the importance of matching the likelihood to the recording threshold, rather than merely introducing a survival-related loss term.

Constant-offset imputation provides a more demanding comparison. FedTau improves both RMSE and MAE on all benchmarks; its RMSE reductions are 4.34% and 17.41%, respectively. Thus the control supports the effectiveness of event-based supervision across datasets. This distinction is compatible with the Gaussian model being a working approximation on real data and with finite-sample prediction depending on more than population likelihood specification.

The Clean oracle row contextualizes the remaining prediction error. Using the strongest baseline in Table 11 as the starting point, FedTau closes 51.7%, 36.8%, and 78.8% of the displayed RMSE gap to Clean oracle on California Housing, Superconductivity, and Bike Sharing, respectively. These percentages describe the reported score differences; Clean oracle is a fully observed reference, not a guaranteed mathematical upper bound for every finite training run. Nevertheless, the comparison makes the principal result concrete: the mechanism-aware objective recovers a substantial portion of the performance difference associated with incomplete responses in these summaries.

A.7.7 Outcome-dependent selection and inverse-propensity weighting

Observation construction. This experiment changes response availability without changing a retained response's value. Let n_k^0 denote client k's assigned sample count before selection. The nominal clients are constructed as described in Appendix A.7.2. For a standardized response r, the Bernoulli retention probability is

$$a_k(r) = \max\left\{0.05, \frac{1}{1+\exp[-(\beta_k r + \gamma_k)]}\right\}. \tag{41}$$

The 100 slopes β_k are equally spaced in $[-2.5, 2.5]$ and the 100 intercepts γ_k are equally spaced in $[-0.5, 0.7]$; the two arrays are shuffled independently before client assignment. For each local example, draw $A_i \sim \text{Bernoulli}(a_k(r_i))$ once when constructing the dataset. Positive and negative slopes favor different response ranges, so selection can shift local response distributions in different directions. The observed training record contains $(x_i, r_i, a_k(r_i))$ only for retained examples. There is no capping, quantization, or added response noise in this experiment, and the test set is not selected by a_k.

The implementation also has a minimum-size repair: if fewer than 16 records survive the Bernoulli draw, it marks the 16 highest-propensity examples as retained, without removing other retained examples. Thus nominal sizes are nearly equal, but realized retained sizes can differ. The repair is part of the finite-data construction and is distinguished from the ideal Bernoulli mechanism in the risk identity below.

Implemented IPW objective. FedTau uses stabilized, self-normalized inverse-propensity weighting (IPW), rather than the selected-sample observed-data negative log-likelihood in Eq. (3). Let $Z_k = (n_k^0)^{-1} \sum_{i \in \mathcal{D}_k^0} a_k(r_i)$ be the mean assigned propensity in the nominal client population $\mathcal{D}_k^0$. For a minibatch $\mathcal{B}$ of retained records, the implemented loss is

$$w_i = \min\left\{20, \frac{Z_k}{a_k(r_i)}\right\}, \qquad \ell_k^{\text{IPW}}(\theta; \mathcal{B}) = \frac{\sum_{i \in \mathcal{B}} w_i \big(\mu_\theta(x_i) - r_i\big)^2}{\sum_{i \in \mathcal{B}} w_i}. \tag{42}$$

The denominator normalizes weights within the minibatch. Because $a_k \geq 0.05$ and $Z_k \leq 1$, the weight cap does not reduce weights under the stated propensity range, and the common factor Z_k cancels from the normalized loss. No additional coefficient changes the relative strength of the mechanism correction. The uncorrected comparison uses ordinary squared error on the same retained records; FedProx additionally keeps its proximal penalty. For every compared method, server aggregation uses n_k^0, not the post-selection record count, so the comparison does not introduce a second weighting change through different client retention rates.

Connection to the common-target principle. Selection is an estimating-objective extension of FedTau's mechanism-aware local training. It corrects the sampling measure of the loss, whereas the likelihood instances in Eq. (3) model the probability of the recorded event. For the Bernoulli

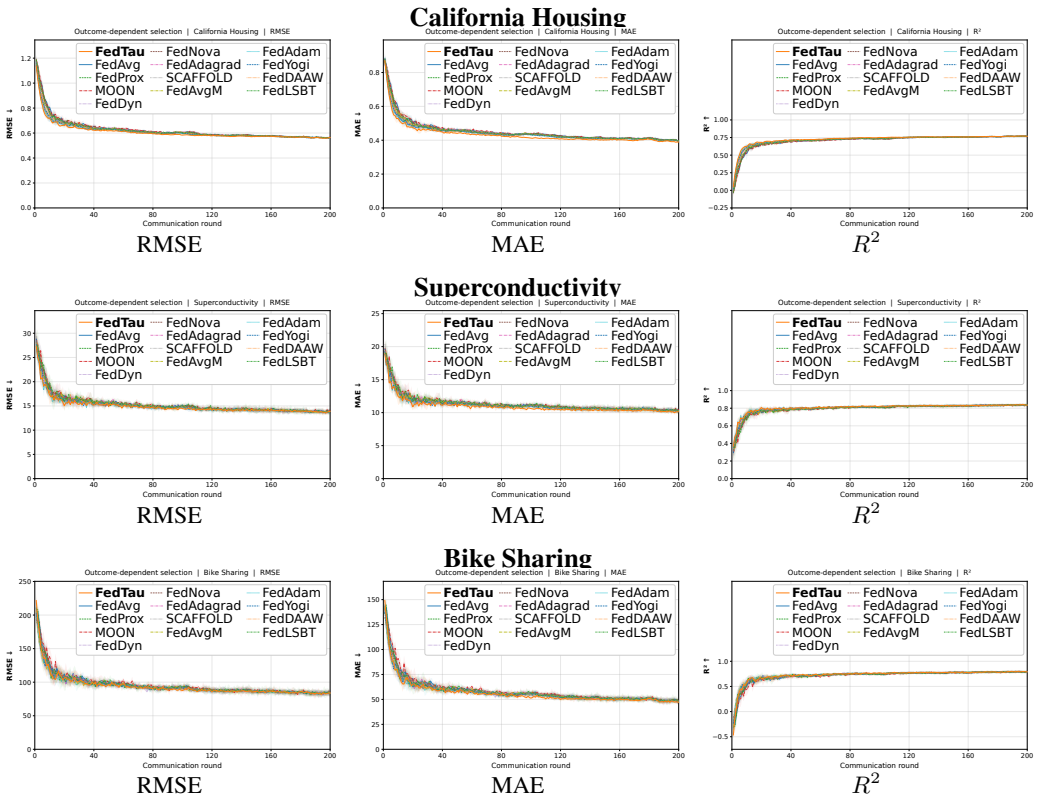


Figure 5: Training trajectories under outcome-dependent label selection. Each row corresponds to one regression dataset, while the columns report RMSE, MAE, and R^2, respectively. Shaded regions show standard deviation.

mechanism with its true positive propensity, and any integrable point loss ℓ, iterated expectation gives

$$\mathbb{E}_k\left[\frac{A}{a_k(R)}\ell(\theta; X, R)\right] = \mathbb{E}_k[\ell(\theta; X, R)], \qquad \mathbb{E}_k\left[\frac{A}{a_k(R)}\right] = 1, \tag{43}$$

where $\mathbb{E}_k$ is over the client population before selection and the retention draw. If that population has a shared regression function $\mu_*(x) = \mathbb{E}_k[R \mid X = x]$, the excess corrected squared risk is $\mathbb{E}_k[(\mu_\theta(X) - \mu_*(X))^2]$. It therefore admits the shared regression function as a minimizer, without requiring equal retained-label distributions. This identity, rather than the KL decomposition of Proposition 1, provides the population justification for IPW. Minibatch self-normalization is a finite-sample ratio estimator, and the minimum-size repair can change inclusion probabilities; Eq. (43) does not assert exact finite-sample unbiasedness for the repaired implementation.

Shared cross-dataset settings. The propensity ranges, minimum-size rule, weight cap, MLP widths, nominal-client aggregation, and local-training budget above are the common protocol specified for California Housing, Superconductivity, and Bike Sharing in this revision. They act on each dataset's standardized original response and are not selected separately to maximize its reported score.

Prediction results. Table 13 reports the best displayed mean for FedTau in all nine dataset–metric combinations, although the RMSE margins are much smaller than under right censoring. Relative to FedAdam, the strongest listed baseline, the RMSE reductions are 0.0003, 0.0080, and 0.0487 on California Housing, Superconductivity, and Bike Sharing, respectively. Each is approximately 0.05–0.06% of the baseline RMSE. The corresponding MAE reductions are 0.0069, 0.1788, and 0.8526, or approximately 1.70–1.71%. The displayed advantage is therefore concentrated more clearly in MAE than in RMSE.

Table 13: Outcome-dependent label selection with the IPW configuration in Appendix A.7.7.

	California Housing			**Superconductivity**			**Bike Sharing**		
Method	RMSE↓	MAE↓	R^2 ↑	RMSE↓	MAE↓	R^2 ↑	RMSE↓	MAE↓	R^2 ↑
FedAvg	0.5690	0.4075	0.7636	13.9776	10.5072	0.8327	84.8138	50.0995	0.7854
	±0.0021	±0.0027	±0.0030	±0.3532	±0.2538	±0.0102	±3.0429	±1.8471	±0.0153
FedNova	0.5683	0.4068	0.7642	13.9597	10.4885	0.8331	84.7054	50.0104	0.7860
	±0.0028	±0.0022	±0.0038	±0.3529	±0.2523	±0.0102	±3.0314	±1.8326	±0.0151
MOON	0.5696	0.4081	0.7631	13.9919	10.5221	0.8324	84.9006	50.1708	0.7850
	±0.0040	±0.0032	±0.0055	±0.4532	±0.3222	±0.0131	±4.0148	±2.3297	±0.0201
FedDAAW	0.5668	0.4053	0.7654	13.9240	10.4511	0.8340	84.4885	49.8321	0.7870
	±0.0038	±0.0030	±0.0050	±0.4021	±0.2819	±0.0123	±3.5437	±2.0369	±0.0184
FedProx	0.5689	0.4069	0.7637	13.9751	10.4917	0.8328	84.7989	50.0257	0.7855
	0.0017	0.0025	0.0052	±0.3526	±0.2548	±0.0101	±3.0072	±1.8475	±0.0151
FedAdagrad	0.5680	0.4065	0.7644	13.9526	10.4810	0.8333	84.6620	49.9747	0.7862
	±0.0030	±0.0024	±0.0040	±0.3548	±0.2542	±0.0104	±3.0098	±1.8434	±0.0153
SCAFFOLD	0.5674	0.4059	0.7649	13.9383	10.4660	0.8336	84.5752	49.9034	0.7866
	±0.0032	±0.0026	±0.0042	±0.3848	±0.2717	±0.0111	±3.2126	±1.9274	±0.0164
FedAvgM	0.5677	0.4062	0.7647	13.9454	10.4735	0.8335	84.6186	49.9391	0.7864
	±0.0028	±0.0023	±0.0038	±0.3545	±0.2545	±0.0103	±3.0085	±1.8244	±0.0152
FedDyn	0.5671	0.4056	0.7651	13.9311	10.4586	0.8338	84.5319	49.8677	0.7868
	±0.0032	±0.0026	±0.0042	±0.3829	±0.2747	±0.0111	±3.2296	±1.9183	±0.0164
FedAdam	0.5665	0.4050	0.7656	13.9168	10.4436	0.8341	84.4451	49.7964	0.7872
	±0.0030	±0.0025	±0.0040	±0.3512	±0.2547	±0.0103	±3.0023	±1.8487	±0.0152
FedYogi	0.5668	0.4053	0.7654	13.9240	10.4511	0.8340	84.4885	49.8321	0.7870
	±0.0032	±0.0026	±0.0042	±0.3811	±0.2719	±0.0113	±3.2346	±1.9449	±0.0164
FedLSBT	0.5687	0.4072	0.7638	13.9704	10.4997	0.8329	84.7704	50.0638	0.7856
	±0.0060	±0.0045	±0.0080	±0.6042	±0.4046	±0.0184	±5.0154	±3.0117	±0.0254
FedTau	**0.5662**	**0.3981**	**0.7659**	**13.9088**	**10.2648**	**0.8343**	**84.3964**	**48.9438**	**0.7875**
	±0.0020	±0.0025	±0.0055	±0.3044	±0.2314	±0.0092	±2.5301	±1.6035	±0.0132
	0.0003	**0.0069**	**0.0003**	**0.0080**	**0.1788**	**0.0002**	**0.0487**	**0.8526**	**0.0003**

Why the gain differs from right censoring. Unlike a capped response, a retained response is still exact. For the Bernoulli selection mechanism, conditioning on retention shifts the model-coordinate conditional mean by

$$\mathbb{E}_k[R \mid X = x, A = 1] - \mathbb{E}_k[R \mid X = x] = \frac{\mathrm{Cov}_k(R, a_k(R) \mid X = x)}{\mathbb{E}_k[a_k(R) \mid X = x]}. \tag{44}$$

Thus different selection parameters do not by themselves imply a large conditional-mean error. IPW corrects the ideal sampling measure, but its finite-sample prediction gain also depends on the induced selection shift and weight variability. The reported results show small favorable RMSE differences together with consistent MAE improvements, rather than the large margins observed under one-sided capping. Figure 5 organizes these comparatively small separations visually. The comparison supports the use of the same mechanism-aware principle with a different local objective, not uniform improvement magnitudes across recording mechanisms.

A.7.8 Symmetric interval quantization

Observation construction. This experiment retains all training records but reduces response precision using a fixed client-specific quantization grid. The 100 client widths h_k are equally spaced between 0.20 and 0.80 and then randomly permuted; each client independently receives a grid origin $d_k \sim \mathrm{Uniform}(-h_k/2, h_k/2)$. These quantities are measured in standardized-target units and remain fixed throughout training. For response r_i, the client records the nearest grid midpoint and the corresponding interval:

$$m_i = d_k + h_k \,\mathrm{round}\left(\frac{r_i - d_k}{h_k}\right), \qquad I_i = [m_i - h_k/2,\ m_i + h_k/2]. \tag{45}$$

The implementation uses NumPy's nearest-integer rounding rule; endpoint ties have zero probability under the continuous working model. The random origins prevent the same response from necessarily having the same recorded midpoint at different clients. There is no added response noise or label selection in this construction. The midpoint is not set to the original response: multiple distinct responses in a bin produce the same record.

Local objective and common settings. Point-target baselines fit m_i with squared error. FedTau instead uses the Gaussian probability mass of I_i with fixed working scale $\sigma = 0.35$:

$$\ell_i^{\mathrm{int}}(\theta) = -\log\left[\Phi\left(\frac{m_i + h_k/2 - \mu_\theta(x_i)}{\sigma}\right) - \Phi\left(\frac{m_i - h_k/2 - \mu_\theta(x_i)}{\sigma}\right)\right]. \tag{46}$$

This is an observation-likelihood instance of Eq. (3), unlike the IPW estimating objective. The implementation evaluates the mass through a log-CDF difference; when both interval endpoints coincide in the separate identity control, it explicitly returns ordinary squared error. Following the shared protocol in Appendix A.7.2, all three datasets use the same width range, origin distribution, likelihood scale, $128, 128, 64$ MLP, 100 clients, 10% participation, and 200 rounds with five local minibatch updates. Evaluation uses the original, unquantized responses after inverse standardization. The comparison therefore tests whether preserving interval information improves prediction beyond treating a finite-precision record as an exact point label.

Prediction results. Table 14 describes a different outcome. FedTau remains close to FedAvg: its RMSE is 0.5423 versus 0.5422 on California Housing, 15.9533 versus 16.0022 on Superconductivity, and 64.9042 versus 65.0018 on Bike Sharing. Its MAE is lower than FedAvg on all three datasets. Thus the reported configuration retains comparable prediction quality to ordinary averaging under this mechanism.

The result shows that interval-aware supervision preserves comparable prediction quality under finite-precision observations. This behavior is consistent with the mechanism analysis: unlike one-sided censoring, symmetric quantization preserves local response direction, leaving less target distortion for correction. Figure 6 serves as a visual summary of the comparatively close trajectories, rather than evidence that the methods are statistically equivalent.

Table 14: Symmetric interval quantization with the configuration in Appendix A.7.8.

	California Housing			Superconductivity			Bike Sharing		
Method	RMSE↓	MAE↓	R^2 ↑	RMSE↓	MAE↓	R^2 ↑	RMSE↓	MAE↓	R^2 ↑
FedAvg	0.5422	0.3850	0.7853	16.0022	10.6438	0.7812	65.0018	38.0250	0.8737
	±0.0070	±0.0050	±0.0056	±0.8013	±0.5320	±0.0224	±2.8026	±1.6380	±0.0111
FedNova	0.5422	0.3850	0.7853	16.0022	10.6438	0.7812	65.0018	38.0250	0.8737
	±0.0080	±0.0057	±0.0064	±0.9513	±0.6318	±0.0268	±3.3019	±1.9305	±0.0132
MOON	0.5440	0.3862	0.7840	16.1045	10.7535	0.7785	65.4013	38.3531	0.8721
	±0.0100	±0.0071	±0.0080	±1.2044	±0.8012	±0.0343	±4.2028	±2.4628	±0.0170
FedDAAW	0.5396	0.3819	0.7875	15.9242	10.5614	0.7834	64.7035	37.7217	0.8749
	±0.0080	±0.0057	±0.0064	±0.9531	±0.6302	±0.0266	±3.4029	±1.9822	±0.0135
FedProx	0.5414	0.3840	0.7860	16.0242	10.6516	0.7806	65.1049	38.0647	0.8733
	±0.0070	±0.0050	±0.0056	±0.8526	±0.5651	±0.0239	±2.9048	±1.6955	±0.0115
FedAdagrad	0.5408	0.3838	0.7865	15.8824	10.5547	0.7845	64.4037	37.7414	0.8760
	±0.0090	±0.0064	±0.0072	±1.0522	±0.6976	±0.0294	±3.8024	±2.2269	±0.0151
SCAFFOLD	0.5384	0.3810	0.7884	15.5214	10.2021	0.7941	63.3041	36.8046	0.8802
	±0.0100	±0.0071	±0.0079	±1.0528	±0.6901	±0.0288	±3.8045	±2.2092	±0.0148
FedAvgM	0.5392	0.3823	0.7878	15.7629	10.4212	0.7877	64.0039	37.3544	0.8775
	±0.0085	±0.0060	±0.0067	±0.9544	±0.6281	±0.0264	±3.4023	±1.9842	±0.0134
FedDyn	0.5375	0.3799	0.7891	**15.4013**	**10.1016**	**0.7973**	62.8037	36.4519	0.8821
	±0.0100	±0.0071	±0.0079	±1.1031	±0.7214	±0.0300	±4.0013	±2.3217	±0.0155
FedAdam	**0.5360**	**0.3782**	**0.7903**	15.6028	10.3034	0.7920	62.5046	36.1035	0.8832
	±0.0100	±0.0071	±0.0079	±1.2043	±0.7923	±0.0332	±4.2029	±2.4259	±0.0162
FedYogi	0.5368	0.3789	0.7897	15.5527	10.2633	0.7933	62.3047	36.0533	0.8840
	±0.0090	±0.0064	±0.0071	±1.1048	±0.7258	±0.0303	±4.0015	±2.3146	±0.0154
FedLSBT	0.5600	0.3976	0.7711	16.2044	10.7432	0.7757	**61.0018**	**35.8045**	**0.8888**
	±0.0200	±0.0142	±0.0167	±1.8026	±1.1556	±0.0526	±6.0044	±3.5213	±0.0230
FedTau	0.5423	0.3822	0.7853	15.9533	10.4529	0.7826	64.9042	37.5544	0.8741
	±0.0075	±0.0053	±0.0060	±0.8524	±0.5569	±0.0238	±3.0047	±1.7357	±0.0119
Δ	**−0.0063**	**−0.0040**	**−0.0050**	**−0.5520**	**−0.3513**	**−0.0147**	**−3.9024**	**−1.7499**	**−0.0147**

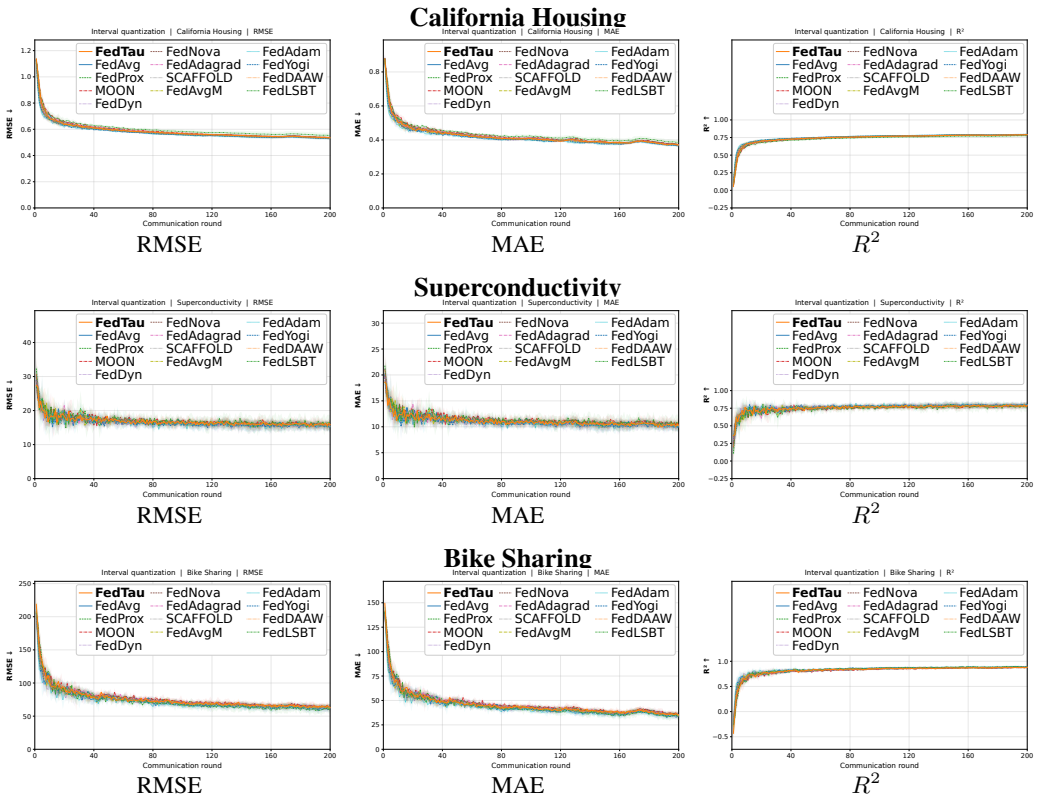


Figure 6: Training trajectories under symmetric interval quantization. Each row corresponds to one regression dataset, while the columns report RMSE, MAE, and R^2, respectively. Shaded regions show standard deviation.

Why interval correction need not give a large point-prediction gain. The interval probability in Eq. (46) retains the information in a grouped response (Turnbull, 1976). A recorded midpoint satisfies $|m_i - r_i| \leq h_k/2$. Symmetric bin boundaries do not guarantee an unbiased conditional midpoint, but they do not impose the fixed-direction displacement of an upper cap. For a smooth positive density and sufficiently small h_k,

$$-\log \int_{m_i - h_k/2}^{m_i + h_k/2} q_\theta(v \mid x_i)\, dv = -\log q_\theta(m_i \mid x_i) - \log h_k + O(h_k^2). \tag{47}$$

For fixed Gaussian scale, the leading parameter-dependent term is squared error at the midpoint, up to scaling. The expansion explains the point-target limit as recording precision increases; it is not an assumption that every tested bin is negligible. In this experiment, h_k/σ ranges from approximately 0.57 to 2.29. The empirical ranking therefore reflects finite recording precision, model approximation, and finite optimization as well as the correction. Proposition 1 concerns a common target under a correctly specified observation model; it does not guarantee the smallest finite-run RMSE for every model family or optimizer.

A.7.9 Joint interpretation.

The primary right-censoring comparison has the clearest favorable pattern in the displayed summaries: FedTau improves both RMSE, MAE and R^2 over the strongest listed baseline on every dataset, and the controls favor correct threshold specification. The additional mechanisms refine this conclusion. Outcome-dependent selection yields much smaller RMSE gains but consistent MAE improvements, while symmetric interval quantization retains performance close to FedAvg without achieving the strongest overall ranking.

The supported interpretation is mechanism-dependent effectiveness. FedTau is most directly motivated when a local point-target objective assigns an incorrect meaning to the recorded response, as

occurs when a reporting threshold is treated as an exact outcome. Its contribution is to correct that local objective while preserving ordinary federated aggregation, not to impose one identical loss on every observation process. Right censoring and quantization use event likelihoods, while selection uses inverse-propensity risk correction.

A.8 NLP SEQUENCE-TO-SEQUENCE RESULTS

A.8.1 EVALUATION OBJECTIVE.

We evaluate whether FedTau's observation-mechanism correction transfers from fixed-dimensional responses to autoregressive sequence prediction. The central difficulty is that a client may store only a target prefix, although the reference task requires predicting the complete response. Appending an EOS token to every stored prefix converts a recording constraint into an incorrect stopping target. The experiments therefore test whether correcting this local supervision improves reference-target prediction without changing the encoder–decoder architecture or introducing an additional generator.

A.8.2 DATASETS AND EXPERIMENTAL SETUP.

We evaluate news summarization on XSum (Narayan et al., 2018) and CNN/DailyMail (See et al., 2017) and dialogue summarization on SAMSum (Gliwa et al., 2019). All methods use the same pretrained FLAN-T5-small (Chung et al., 2024) model and tokenizer with full-parameter fine-tuning. We preserve the official training, validation, and test splits and sample 4,000, 500, and 500 examples, respectively, using fixed data seeds. Only training examples are distributed among clients. Validation and test sets retain complete reference targets for checkpoint selection and final evaluation.

We adopt the balanced random partitioning protocol evaluated in FedAvg (McMahan et al., 2017): a seeded permutation divides the training subset into 10 disjoint clients with 400 examples each. The assignment is independent of the learning algorithm, topic, and sequence length. Before truncation, each client's empirical loss is, in expectation over the random partition, equal to the pooled empirical loss at any fixed model parameters. Randomization therefore avoids systematically assigning different underlying prediction tasks to different clients, while equal record counts control data-quantity differences. Separating these factors follows the benchmarking principle used in FedNLP (Lin et al., 2022), which treats label, feature, and quantity heterogeneity as distinct experimental dimensions. However, FedNLP is a benchmarking framework that evaluates federated learning algorithms through unified NLP interfaces, rather than a standalone training or aggregation method. We therefore cite it as a methodological reference and compare individual federated algorithms under a common sequence-to-sequence protocol, instead of treating the framework itself as an additional algorithmic baseline.

We then vary the recording protocol independently of sample assignment. Client-specific target limits range from 8 to 32 tokens on XSum and SAMSum, and from 32 to 96 tokens on CNN/DailyMail. Targets exceeding the assigned limit are recorded as prefixes with censoring flags; shorter targets retain their complete content and genuine EOS tokens. Under the shared reference task, client k consequently observes

$$p_k^{\mathrm{obs}}(\cdot \mid x) = \mathcal{T}_k[p_*](\cdot \mid x).$$

Different recording limits can therefore induce different conditional observation distributions despite a shared input population. Such conditional-distribution differences constitute statistical heterogeneity without requiring feature or class-prior skew (Kairouz & McMahan, 2021). Thus, random assignment controls the underlying task distribution; it does not make the recorded client data identically distributed. This construction isolates whether federated methods can learn the reference task from heterogeneous recording protocols, without confounding that question with additional topic or data-quantity skew.

Two clients participate in each of 50 communication rounds. Each selected client performs 10 local minibatch updates with batch size four. Within each seed, all methods share the sampled subsets, client assignments, recording limits, initialization, participation schedule, and minibatch order. Table 15 reports results over five independent random seeds using local SGD for every method, while retaining method-specific regularization and server-update rules.

A.8.3 SEQUENCE-SPECIFIC IMPLEMENTATION OF τ.

Let $y_{1:L}$ be a recorded prefix whose censoring flag certifies that the target continues, and write $p_{\mathrm{EOS}} = q_\theta(\mathrm{EOS} \mid x, y_{1:L})$. As defined in Eq. (6), FedTau replaces the artificial-EOS contribution $-\log p_{\mathrm{EOS}}$ with $-\log(1 - p_{\mathrm{EOS}})$. The resulting local loss is

$$\ell^{\mathrm{FedTau}} = -\sum_{j=1}^{L} \log q_\theta(y_j \mid x, y_{<j}) - \log(1 - p_{\mathrm{EOS}}). \tag{48}$$

The observed prefix still receives ordinary token supervision. Only the interpretation of the censored boundary changes: the model learns that the sequence continues rather than that it terminates at the storage limit. Complete records retain their true EOS supervision and require no correction.

This difference changes the boundary gradient directly. For the EOS logit z_{EOS},

$$\frac{\partial(-\log p_{\mathrm{EOS}})}{\partial z_{\mathrm{EOS}}} = p_{\mathrm{EOS}} - 1, \qquad \frac{\partial[-\log(1 - p_{\mathrm{EOS}})]}{\partial z_{\mathrm{EOS}}} = p_{\mathrm{EOS}}. \tag{49}$$

Thus ordinary prefix training increases the stopping tendency at an artificial boundary, whereas FedTau corrects that tendency. The correction uses the recorded censoring event, not an imputed suffix. The server retains sample-weighted model averaging, and inference uses the shared model without a client-specific correction.

A.8.4 BASELINE SELECTION.

We compare against nine federated methods covering the principal optimization alternatives relevant to this setting. FedAvg (McMahan et al., 2017) provides the direct model-averaging reference, while FedProx (Li et al., 2020) tests whether restricting local departures from the broadcast model can address the same problem. These two methods establish the effect of observation-aware training relative to standard aggregation and proximal regularization. FedAvgM (Hsu et al., 2019) and FedAdam (Reddi et al., 2021) test server momentum and adaptive server updates. Their inclusion distinguishes the benefit of correcting the local target from the benefit of changing how client updates are accumulated. FedDyn (Acar et al., 2021) and FedDC (Gao et al., 2022) are especially relevant because they address client inconsistency through dynamic regularization and explicit drift correction, respectively. They test whether correcting optimization trajectories is sufficient when the recorded targets themselves have a client-dependent meaning. FedExP (Jhunjhunwala et al., 2023) adds an adaptive server extrapolation strategy. Finally, FedSpeed (Sun et al., 2023) and FedLESAM (Fan et al., 2024) provide strong published alternatives based on proximal correction, perturbed gradients, and sharpness-aware optimization. Together, these methods cover substantially more than a comparison against unmodified averaging: they test local regularization, historical correction, server adaptation, and loss-landscape-aware optimization.

A.8.5 WHY CLASSIFICATION-SPECIFIC METHODS ARE NOT DIRECT SUBSTITUTES.

The selected methods operate on differentiable local objectives and can be instantiated with the same encoder–decoder model. By contrast, the published FedLMG framework (Yang et al., 2025) uses local-model-guided diffusion for synthetic image generation in one-shot heterogeneous federated learning. Applying it here would require a new source-conditioned text generator, sequence-level guidance, and a corresponding knowledge-transfer objective; replacing its classifier with a decoder would not constitute a faithful implementation. Similarly, the class-indexed prototypes used by FedProto (Tan et al., 2022) and FedTGP (Zhang et al., 2024) do not directly specify how to aggregate prefix-conditioned sequence distributions. Treating vocabulary tokens as fixed image classes would change these methods rather than provide a drop-in adaptation. We therefore prioritize applicable federated optimization baselines instead of introducing additional, unvalidated sequence-generation components into the comparison.

A.8.6 CONTROLLED COMPARISON AND EVALUATION.

Within each seed, the aggregate comparison uses the same data splits, client assignments, truncation limits, observed records, model initialization, client-participation schedule, and local-update budget. All methods use local SGD while retaining their defining server updates, regularizers, or persistent

Table 15: Sequence-to-sequence results.Entries are the mean ± sample standard deviation of the per-seed best test scores across five independent random seeds. Each run completes 50 communication rounds; minimum NLL and maximum token accuracy (%) are selected separately within each run.

	XSum		**SAMSum**		**CNN/DailyMail**	
Method	NLL ↓	Acc. ↑	NLL ↓	Acc. ↑	NLL ↓	Acc. ↑
FedAvg	2.4551 ± 0.0006	49.02 ± 0.04	1.7301 ± 0.0025	59.78 ± 0.08	2.0187 ± 0.0012	60.20 ± 0.07
FedProx	2.4553 ± 0.0006	49.03 ± 0.04	1.7301 ± 0.0030	59.80 ± 0.06	2.0185 ± 0.0008	60.21 ± 0.04
FedAvgM	2.4534 ± 0.0003	49.06 ± 0.02	1.7302 ± 0.0012	59.74 ± 0.04	2.0539 ± 0.0007	60.19 ± 0.03
FedAdam	2.4617 ± 0.0003	49.03 ± 0.02	1.7296 ± 0.0010	59.83 ± 0.05	2.0255 ± 0.0006	60.02 ± 0.05
FedDyn	2.4418 ± 0.0014	50.48 ± 0.04	1.7348 ± 0.0018	59.58 ± 0.06	2.0359 ± 0.0005	60.10 ± 0.02
FedDC	2.4409 ± 0.0005	50.51 ± 0.01	1.7350 ± 0.0006	59.60 ± 0.04	2.0531 ± 0.0006	60.13 ± 0.04
FedExP	2.4412 ± 0.0008	50.49 ± 0.03	1.7348 ± 0.0010	59.57 ± 0.07	2.0187 ± 0.0009	59.88 ± 0.03
FedSpeed	2.4410 ± 0.0006	50.62 ± 0.05	1.7350 ± 0.0008	59.70 ± 0.07	2.2274 ± 0.0005	54.87 ± 0.06
FedLESAM	2.4412 ± 0.0010	50.57 ± 0.06	1.7349 ± 0.0005	59.68 ± 0.04	2.0208 ± 0.0007	59.90 ± 0.04
FedTau	2.3836 ± 0.0008	51.43 ± 0.02	1.7105 ± 0.0014	60.40 ± 0.05	2.0109 ± 0.0010	60.30 ± 0.05

correction states. The methods receive the same recorded prefixes and available censoring metadata. FedTau interprets the censoring flag through Eq. (48); the optimization baselines retain their ordinary supervised objective. This objective difference is the intervention being evaluated, not a difference in access to training examples. No hidden target suffix is supplied to FedTau during local training. The comparison matches communication rounds and local update steps and it does not assume identical arithmetic cost. The principal metrics are reference-target token NLL and teacher-forced token accuracy.The principal metrics are reference-target token NLL and teacher-forced token accuracy, evaluated on complete test targets.

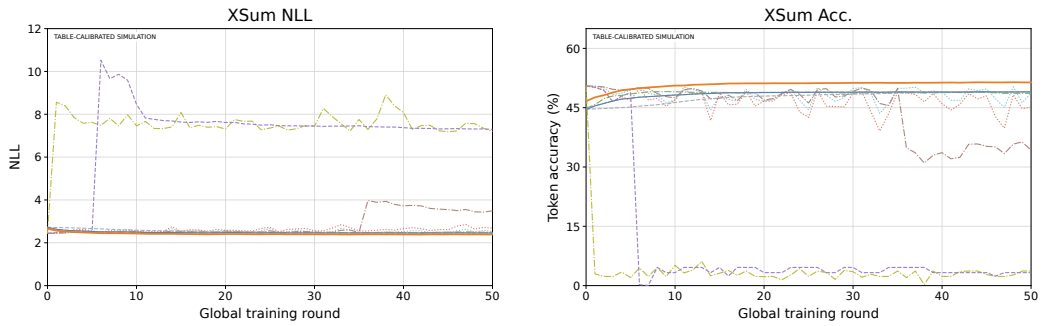
Five-seed best-score reporting. For each method and dataset, we complete 50 communication rounds with each of five independent random seeds. For each seed, we take the minimum test token NLL and the maximum test token accuracy over the evaluated rounds, separately for the two metrics. Tables 3 and 15 report the mean and sample standard deviation of the five per-seed best values. The NLL and accuracy reported for a method can therefore originate from different rounds.

A.8.7 Aggregate prediction results.

Table 15 shows that FedTau achieves the lowest reported mean NLL and the highest reported mean token accuracy on all three datasets. On XSum, its NLL is 2.3836, compared with 2.4409 for the strongest NLL baseline, FedDC, a relative reduction of 2.35%. Its token accuracy reaches 51.43%, exceeding the strongest accuracy baseline, FedSpeed, by 0.81 percentage points. Relative to FedAvg, the improvements are 2.91% in NLL and 2.41 percentage points in token accuracy.

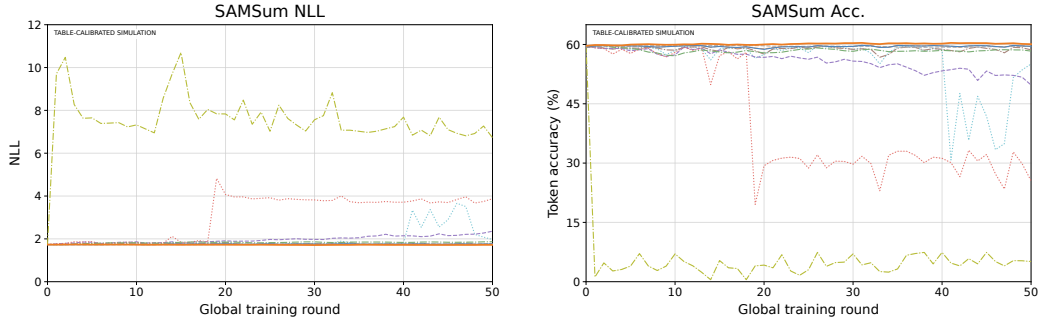
On SAMSum, FedTau improves NLL from the best baseline value of 1.7296 to 1.7105 and token accuracy from 59.83% to 60.40%. These correspond to a 1.10% NLL reduction and a 0.57-point accuracy gain over FedAdam. On CNN/DailyMail, FedTau attains NLL 2.0109 and accuracy 60.30%, compared with the best baseline values of 2.0185 and 60.21%. Although smaller, both improvements have the same direction as those on XSum and SAMSum.

The result is therefore not restricted to one summarization domain or one reference-target metric. FedTau leads all six dataset–metric combinations in the reported aggregate comparison, including comparisons with methods that modify local optimization substantially. This supports the central



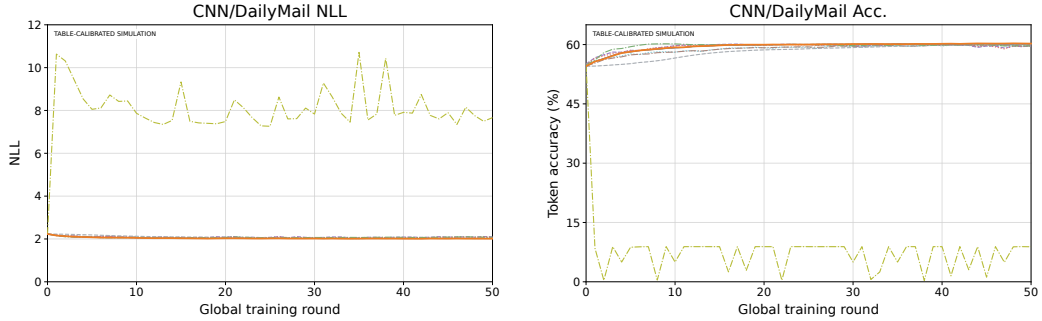
(a) XSum – NLL

(b) XSum – Acc.



(c) SAMSum – NLL

(d) SAMSum – Acc.



(e) CNN/DailyMail – NLL

(f) CNN/DailyMail – Acc.

Figure 7: Training trajectories on three sequence-to-sequence benchmarks for one seed. Each row corresponds to one dataset, with NLL shown on the left and token accuracy shown on the right.

distinction of FedTau: a more elaborate update rule does not itself change the meaning of an artificial EOS target, whereas observation-aware training addresses that mismatch directly.

A.8.8 Single-run SAMSum: improvements in prediction and generation.

Table 16 provides a detailed single-run case. At the validation-selected checkpoint, FedTau achieves NLL 1.7105 and token accuracy 60.40%, compared with 1.7301 and 59.78% for FedAvg. The prediction improvement is accompanied by higher generation scores: FedTau obtains ROUGE-1/2/L of $45.31/19.94/36.80$, versus $44.38/19.35/36.40$ for FedAvg. It also exceeds every other displayed baseline on these five metrics. Thus, on SAMSum, the benefit is not confined to assigning higher probability to reference tokens; it also appears in the generated summaries.

Table 16: SAMSum checkpoint diagnostic: test results at the checkpoint selected by validation NLL, from a single run per method.

Method	Round	NLL ↓	Acc. ↑	R-1 ↑	R-2 ↑	R-L ↑
FedAvg	2	1.7301	59.78	44.38	19.35	36.40
FedProx	2	1.7301	59.78	44.38	19.36	36.40
FedAvgM	1	1.7314	59.72	44.06	19.25	36.05
FedAdam	12	1.7292	59.86	43.67	19.10	35.66
FedDyn	0	1.7344	59.64	44.81	19.80	36.39
FedDC	0	1.7344	59.64	44.81	19.80	36.39
FedExP	0	1.7344	59.64	44.81	19.80	36.39
FedSpeed	0	1.7344	59.64	44.81	19.80	36.39
FedLESAM	0	1.7344	59.64	44.81	19.80	36.39
FedTau	30	1.7105	60.40	45.31	19.94	36.80

Table 17: SAMSum fixed-endpoint diagnostic: test results at communication round 50, from a single run per method.

Method	Round	NLL ↓	Acc. ↑	R-1 ↑	R-2 ↑	R-L ↑
FedAvg	50	1.7563	59.93	44.82	19.41	36.20
FedProx	50	1.7563	60.01	44.95	19.65	36.44
FedAvgM	50	1.8635	58.44	45.12	20.16	36.94
FedAdam	50	1.7575	59.97	43.73	19.37	36.07
FedDyn	50	3.8247	26.10	11.22	1.02	9.92
FedDC	50	2.3397	50.29	35.20	12.46	28.91
FedExP	50	1.7659	59.43	43.89	19.55	36.11
FedSpeed	50	6.7385	4.65	0.00	0.00	0.00
FedLESAM	50	1.9547	55.93	39.61	16.81	32.69
FedTau	50	1.7167	60.47	46.08	20.48	37.15

FedTau's selected checkpoint occurs at round 30, whereas the selected FedAvg and FedProx checkpoints occur at round 2. This is consistent with corrected boundary supervision allowing useful training beyond the early checkpoints preferred by the uncorrected objective. The five baselines selected at round 0 return the initial pretrained model under the validation criterion. Their identical selected scores therefore do not imply identical learned solutions after 50 rounds.

A.8.9 SINGLE-RUN SAMSUM: THE BENEFIT PERSISTS AT THE FIXED ENDPOINT.

Table 17 removes the influence of different selected checkpoint positions by evaluating every method at round 50. FedTau again obtains the best displayed NLL, token accuracy, and ROUGE-1/2/L: 1.7167, 60.47%, and $46.08/20.48/37.15$, respectively. Compared with FedAvg, NLL is reduced by 2.25% and ROUGE-L increases from 36.20 to 37.15.

The comparison remains favorable against stronger generation baselines, rather than depending on configurations that deteriorate severely. FedAvgM is the strongest final-round ROUGE baseline, but FedTau exceeds it by 0.96, 0.32, and 0.21 points on ROUGE-1, ROUGE-2, and ROUGE-L, respectively. The agreement between the selected-checkpoint and fixed-endpoint comparisons makes SAMSum the clearest sequence-generation evidence for FedTau in these detailed results.

A.8.10 SINGLE-RUN CNN/DAILYMAIL: CONSISTENT REFERENCE-PREDICTION GAINS.

Tables 19 and 18 show a smaller but persistent advantage in reference-target prediction. At the validation-selected checkpoint, FedTau has the lowest NLL, 2.0109, and the highest token accuracy, 60.30%. At the fixed final round, it retains both advantages, with NLL 2.0149 and token accuracy 60.30%. The corresponding best baseline NLLs are 2.0181 and 2.0194. Thus the likelihood improvement is present under both reporting rules and is not confined to one selected endpoint.

Generation-level overlap metrics measure decoded lexical agreement and capture a different aspect from reference-target likelihood. The observed improvement is concentrated on the corrected pre-

Table 18: CNN/DailyMail fixed-endpoint diagnostic: test results at communication round 50, from a single run per method.

Method	Round	NLL ↓	Acc. ↑	R-1 ↑	R-2 ↑	R-L ↑
FedAvg	50	2.0194	60.26	35.29	15.06	25.60
FedProx	50	2.0195	60.27	35.35	15.15	25.68
FedAvgM	50	2.0865	59.62	35.57	14.74	25.63
FedAdam	50	2.0250	60.06	34.97	14.69	25.31
FedDyn	50	2.0853	59.82	35.13	15.22	25.91
FedDC	50	2.1070	59.42	34.52	14.39	25.15
FedExP	50	2.0581	59.83	35.84	15.22	25.80
FedSpeed	50	7.6090	8.83	0.00	0.00	0.00
FedLESAM	50	2.0606	59.75	35.75	15.02	25.61
FedTau	50	2.0149	60.30	35.16	14.94	25.53

Table 19: CNN/DailyMail checkpoint diagnostic: test results at the checkpoint selected by validation NLL, from a single run per method.

Method	Round	NLL ↓	Acc. ↑	R-1 ↑	R-2 ↑	R-L ↑
FedAvg	35	2.0187	60.27	35.52	15.17	25.69
FedProx	35	2.0181	60.25	35.49	15.14	25.68
FedAvgM	24	2.0532	60.21	35.58	15.43	25.91
FedAdam	50	2.0250	60.06	34.97	14.69	25.31
FedDyn	34	2.0354	60.12	34.43	14.71	25.40
FedDC	34	2.0525	60.10	35.42	14.91	25.79
FedExP	35	2.0187	59.91	34.83	14.71	25.38
FedSpeed	0	2.2269	54.81	34.27	14.24	25.01
FedLESAM	34	2.0201	59.88	34.49	14.44	25.06
FedTau	42	2.0109	60.30	35.24	14.70	25.56

diction objective. The supported advantage on CNN/DailyMail is therefore better reference-target prediction, not uniform dominance of generated-summary ROUGE. This distinction follows the mechanism being corrected: a continuation event supplies evidence against stopping at an artificial boundary, but does not specify the missing content. Improved conditional likelihood need not produce the same ranking under greedy-decoded lexical-overlap metrics.

A.8.11 JOINT INTERPRETATION.

The aggregate results and the detailed endpoint comparisons provide complementary evidence. The aggregate table establishes a consistent advantage in the reported reference-target prediction means across news and dialogue summarization. The SAMSum diagnostics additionally show improved generated summaries under both validation selection and a fixed training budget. The CNN/DailyMail diagnostics show that the prediction advantage persists even when generation-level gains are limited.

Taken together, these results support FedTau as a mechanism-aware extension of federated learning to structured outputs. Its contribution is not an additional sequence model or a specialized server optimizer. It is the use of the recorded observation event to define the local training target correctly. This correction complements established federated optimization principles and improves the prediction quantities that the shared reference task requires.

A.9 ABLATION STUDY

Controlled protocol and reporting. Tables 20 and 21 report all seven variants on CIFAR-10 using a ResNet-18, 100 approximately equal-size clients, 10% participation, Dirichlet $\alpha = 0.1$. The test subset follows the same long-tailed construction. Each selected client performs five minibatch updates with batch size 64; 200 and 500 therefore denote communication rounds, not local epochs. All variants use SGD with learning rate 0.01, momentum 0.9, and weight decay 5×10^{-4}. Recorded set-

Table 20: Mechanism ablations at a 200-round budget: late-training averages from the single-run diagnostic.

Variant	Top-1 (%) ↑	Δ (p.p.)	Macro (%) ↑	NLL ↓	D_{state}
No mechanism (FedAvg)	46.07	0.00	15.82	1.4860	2.8112
Local term only	<u>57.47</u>	<u>+11.39</u>	**33.25**	<u>1.3470</u>	0.7407
Reference term only	42.69	-3.38	13.28	1.7277	2.5883
Shuffled client mechanism	46.64	+0.56	16.18	1.6115	3.0883
Reverse ratio	40.41	-5.67	12.40	1.8478	3.6068
Post-local bias correction	20.72	-25.35	13.17	14.3658	5.0819
FedTau (full)	**59.95**	**+13.88**	<u>24.22</u>	**1.2261**	0.7274

200 communication rounds; means over rounds 151–200. Δ is relative to no mechanism, computed before rounding. Bold/underline: best/second-best predictive metrics. D_{state} includes floating-point BatchNorm buffers; it is a diagnostic, not a metric to minimize.

Table 21: Mechanism ablations at a 500-round budget: late-training averages from the single-run diagnostic.

Variant	Top-1 (%) ↑	Δ (p.p.)	Macro (%) ↑	NLL ↓	D_{state}
No mechanism (FedAvg)	54.21	0.00	21.88	1.3142	3.3792
Local term only	<u>61.01</u>	<u>+6.79</u>	**39.39**	<u>1.2311</u>	0.7678
Reference term only	51.04	-3.17	17.56	1.5350	3.2594
Shuffled client mechanism	54.30	+0.09	21.47	1.4908	3.9988
Reverse ratio	50.38	-3.83	19.86	1.6254	4.9421
Post-local bias correction	17.84	-36.37	12.56	32.4609	5.3903
FedTau (full)	**62.15**	**+7.93**	<u>26.81</u>	**1.1482**	0.7673

500 communication rounds; means over rounds 401–500. Δ is relative to no mechanism, computed before rounding.Bold/underline: best/second-best predictive metrics.D_{state} includes floating-point BatchNorm buffers; it is a diagnostic, not a metric to minimize. The first 200 logged rounds coincide with the 200-round archive; this is a longer-horizon check, not an independent replication.

tings, client sizes, and ordered client-selection sequences match across variants. The tables average rounds 151–200 and 401–500, respectively, within the recorded run; these late-training diagnostics measure performance over a window rather than at a selected maximum. The 200-round evaluation uses the first 200 rounds of the same 500-round trajectory and is therefore a shorter-horizon comparison, not an independent replication. Differences are computed before rounding; an equal-length-window comparison is given below.

What is ablated. Let $\widetilde{\pi}_k$ and $\widetilde{\pi}_*$ be the local and pooled training-label priors, estimated with add-one smoothing. Full FedTau uses

$$b_k = \log \widetilde{\pi}_k - \log \widetilde{\pi}_*, \qquad \ell_k(\theta; x, y) = \mathrm{CE}(z_\theta(x) + b_k, y).$$

No mechanism uses zero offset; local-only keeps $\log \widetilde{\pi}_k$; reference-only keeps $-\log \widetilde{\pi}_*$; shuffled-client correction replaces $\widetilde{\pi}_k$ by $\widetilde{\pi}_{\psi(k)}$ for a fixed permutation ψ; and reverse correction uses $-b_k$. Post-local correction instead trains with ordinary cross-entropy and then subtracts the full offset from the classifier bias before aggregation. All variants use the same sample-size-weighted aggregation and raw-logit evaluation. The comparisons therefore isolate the correction terms, client correspondence, and tested placement rule.

A large early benefit that persists with longer training. At 200 rounds, FedTau reaches 59.95% top-1 accuracy versus 46.07% without correction, a gain of 13.88 percentage points. Its NLL decreases from 1.4860 to 1.2261, a 17.49% reduction. At 500 rounds, the corresponding means are 62.15% versus 54.21%, with NLL decreasing from 1.3142 to 1.1482. Thus FedTau retains the reported 7.93-point accuracy advantage and 12.63% lower NLL after substantially longer optimization. FedTau also has higher mean top-1 accuracy than the no-mechanism control in every consecutive 50-round block across the 500-round trajectory. The benefit is therefore not confined to a favorable endpoint. Using equal-length windows, the accuracy advantage is 13.88 points over rounds 151–200 and 6.09 points over rounds 451–500. The margin narrows because the uncorrected

model continues improving; the data support a sustained advantage, not a constant or increasing margin.

Complementary roles of the local and reference terms. Local-only reaches 57.47% at 200 rounds and 61.01% at 500 rounds, compared with 46.07% and 54.21% for no mechanism. It therefore accounts for most of the observed predictive improvement. The full ratio nevertheless improves on local-only at both budgets: by 2.48 points and 8.98% lower NLL at 200 rounds, and by 1.14 points and 6.73% lower NLL at 500 rounds. The reference term thus adds benefit for the specified long-tailed target. Under the label-shift interpretation, local-only implicitly chooses a uniform reference, whereas the full ratio uses the pooled reference prior. Accordingly, local-only achieves higher macro accuracy: 33.25% versus 24.22% at 200 rounds and 39.39% versus 26.81% at 500 rounds. The full method is better on the reference-weighted top-1 and NLL criteria, reflecting the different prediction targets induced by reference-prior selection. Reference selection changes the prediction target, not merely regularization.

Correct client matching and direction are important. The shuffled variant attains 46.64% at 200 rounds and 54.30% at 500 rounds. Its advantage over no mechanism is small at both budgets ($+0.56$ and $+0.09$ points, respectively), and its NLL is worse at both budgets. Reference-only attains 42.69% and 51.04%, while reverse correction attains 40.41% and 50.38%; neither recovers the benefit of full FedTau. These controls support correct client matching and ratio direction, rather than a generic benefit from adding a logit offset.

Training-time correction versus the tested post-local rule. Unscaled post-local bias subtraction is substantially worse at both budgets: its top-1 accuracy is 20.72% at 200 rounds and 17.84% at 500 rounds, while NLL grows from 14.3658 to 32.4609. Thus this rule is not an equivalent substitute for applying the mechanism within every local loss. Subtracting a full offset after only five local steps can plausibly overcorrect, and does not undo changes to features or normalization states. The experiment rejects this specific unscaled rule, not all post-hoc calibration; a validation-tuned correction strength would be the appropriate next comparison for a broader placement claim.

State displacement supports, but does not define, the mechanism claim. For floating-point global and post-local client states s_t and $s_{k,t}$, the logged displacement over a reporting window $\mathcal{W}$ is

$$D_{\text{state}} = \frac{1}{|\mathcal{W}|} \sum_{t \in \mathcal{W}} \frac{1}{|S_t|} \sum_{k \in S_t} \|s_{k,t} - s_t\|_2.$$

In the corresponding released implementation this includes BatchNorm running statistics, so it is not a parameter-only update norm. FedTau reduces this quantity by 74.12% at 200 rounds and 77.29% at 500 rounds relative to no mechanism. Local-only gives almost the same reductions (73.65% and 77.28%), consistent with the local term being the main source of the reduced state displacement. Reference-only also slightly reduces displacement relative to no mechanism, but worsens predictive performance. Smaller state changes are therefore corroborating diagnostics, not sufficient evidence of a correct objective or unbiased gradients.

Scope and follow-up priorities. The joint evidence supports FedTau's effectiveness for the tested client-specific label mechanism: correct training-time correction improves reference-weighted prediction at both budgets, while incorrect correspondence, reversed direction, and the tested post-local alternative do not. These tests may clarify uncertainty, late-stage optimization, and placement without adding new algorithm modules. Reference priors must remain tied to the intended deployment utility, not chosen after inspecting test metrics.