Who to Trust? Aggregating Client Predictions in Federated Distillation

Viktor Kovalchuk[1] **Denis Son**[2] **Arman Bolatov**[1] **Mohsen Guizani**[1] **Samuel Horváth**[1] **Maxim Panov**[1]
Martin Takáč[1] **Eduard Gorbunov**[1] **Nikita Kotelevskii**[1]

[1]Mohamed bin Zayed University of Artificial Intelligence (MBZUAI), Abu Dhabi, UAE
[2]Independent Researcher

Abstract

Federated Distillation enables distributed learning for clients with heterogeneous model architectures. In this paradigm, the server and clients exchange predictions on a shared unlabeled public dataset, rather than model parameters or gradients. However, under data heterogeneity (e.g., *class mismatch*), clients produce unreliable predictions for instances from unfamiliar classes. An equally weighted combination of such predictions corrupts the teacher signal used for distillation. In this paper, we theoretically analyze Federated Distillation and show that aggregating client predictions on a shared public dataset converges to a neighborhood of the optimum, with the neighborhood size controlled by the aggregation quality. We propose two uncertainty-aware aggregation methods, **UWA** and **sUWA**, that use density-based estimates to down-weight unreliable client predictions. Experiments on image and text classification datasets confirm that our methods are most effective under high data heterogeneity, while matching standard averaging when heterogeneity is low. Code is available at `https://github.com/kovalchuk026/fd_aggregators`.

1 INTRODUCTION

Federated Learning (FL) is a distributed learning paradigm that allows multiple clients to collaboratively train their models without sharing private data [McMahan et al., 2017, Konečnỳ et al., 2016, Kairouz et al., 2021, Wang et al., 2021]. However, most FL algorithms require sharing model parameters or gradients, which can be prohibitively expensive for modern deep networks.

To reduce communication costs, several works explore an alternative paradigm called Federated Distillation (FD; Li and Wang, 2019, Shao et al., 2024, Wei and Li, 2022). In FD, clients share model outputs (e.g., logits or predicted probabilities) on a public dataset, rather than model parameters. This setup allows for a significant reduction in communication and naturally supports heterogeneous client architectures, since FD only requires the output spaces of client models to match. As a result, the core algorithmic challenge shifts from *parameter aggregation* to *logits or probabilities aggregation*, which are used for distillation.

However, standard FD approaches [Li and Wang, 2019, Lin et al., 2020] treat all client predictions as equally reliable, which becomes problematic when the data are heterogeneous. In this work, we focus on a practically important form of such heterogeneity, a label shift that leads to *class mismatch*. In this setting, each client observes only a subset of the global label space, while the public dataset contains samples from all classes. Consequently, some public samples belong to classes absent from a given client's training data. For such samples, the client effectively predicts out-of-distribution (OoD) inputs, and the resulting predictions can be misleading. Naive averaging of those predictions, therefore, may severely degrade the quality of the teacher signal and hinder knowledge transfer.

In this paper, we theoretically analyze the benefits of aggregation in FD and propose a particular method that improves model training under data heterogeneity. Our **contributions** are summarized below.

1. We introduce a family of uncertainty-aware approaches to aggregating predictions from different clients; see Section 3.
2. Under mild assumptions, we provide convergence guarantees for the two-stage training procedure: a non-convex stationarity rate and a linear rate to a neighborhood determined by teacher quality; see Section 4.
3. We validate the proposed methods and baselines [Li and Wang, 2019, Lin et al., 2020] on image and textual data under distributed heterogeneous scenarios; see Section 5.

Proceedings of the 42nd Conference on Uncertainty in Artificial Intelligence (UAI 2026), PMLR 337:3162–3181.

2 RELATED WORK

In this section, we discuss two major paradigms of distributed learning, Federated Learning and Federated Distillation, and then review uncertainty quantification methods that motivate our approach.

Federated Learning. Federated Learning (FL) was introduced by McMahan et al. [2017] with the FedAvg algorithm. In FL, distributed clients aim to train a global (shared) model collaboratively, but they are restricted from sharing their data. Instead, the clients are allowed to share gradients or model parameters with a server, which aggregates them across communication rounds. While FedAvg performs well when client data are homogeneous, its performance degrades significantly under data heterogeneity [Li et al., 2020, Karimireddy et al., 2020]. Several methods address this through variance reduction, such as SCAFFOLD [Karimireddy et al., 2020] and S-Local-SVRG [Gorbunov et al., 2021], or through personalization [Collins et al., 2021, Kotelevskii et al., 2022b, Li et al., 2021] and partial personalization [Raghu et al., 2020, Pillutla et al., 2022, Mishchenko et al., 2025]. However, standard FL techniques require all clients to share the same model architecture, which can be restrictive in practice and creates significant communication overhead for large models.

Federated Distillation. Federated Distillation (FD) was proposed to reduce communication costs and relax the shared-architecture constraint [Li and Wang, 2019, Seo et al., 2022, Li et al., 2024, Sattler et al., 2022]. FD draws on knowledge distillation [Hinton et al., 2015, Gou et al., 2021], in which a student model learns to match the teacher's predictions. In the federated setting, the teacher signal is formed by aggregating predictions from multiple clients on a shared public dataset. The change from exchanging weights to predictions reduces communication overhead and naturally supports heterogeneous client architectures, and led to different methods [Shao et al., 2024, Wei and Li, 2022, Lin et al., 2020].

However, constructing a high-quality teacher signal from heterogeneous client data remains a key challenge for FD. Under *class mismatch*, an aggregation that averages predictions of different clients [Li and Wang, 2019, Lin et al., 2020] treats all client predictions equally, regardless of whether a given client has seen the relevant classes during training. This can result in out-of-distribution predictions being weighted equally with informed ones, leading to an unreliable teacher signal.

Uncertainty Quantification. Uncertainty quantification provides tools for assessing the reliability of model predictions [Hüllermeier and Waegeman, 2021, Gawlikowski et al., 2023]. A variety of approaches exist to quantify predictive uncertainty and improve the reliability of outputs. These include ensemble-based methods [Lakshminarayanan et al., 2017], Bayesian approaches [Blundell et al., 2015], and post-hoc calibration techniques [Guo et al., 2017]. In our setting, density-based methods are particularly relevant [Van Amersfoort et al., 2020, Mukhoti et al., 2023, Kotelevskii et al., 2022a], as they can estimate epistemic uncertainty, which reflects a model's awareness of the data it predicts on, for a single deterministic model without requiring an ensemble.

We build on these density-based methods and incorporate each client's uncertainty estimate into the server-side aggregation step. This allows the server to down-weight unreliable client predictions when constructing the teacher signal.

3 SETTING AND METHOD

We start this section by formalizing our setup and introducing the necessary notation. A glossary of the notation used throughout the paper is provided in Appendix C.1. We consider M distributed clients, collaboratively solving a classification problem with C classes. Each client $i \in \mathcal{S} = \{1, \dots, M\}$ holds a private labeled dataset $\mathcal{D}_i \sim \mathcal{P}_i$, where labels come from a subset of all possible classes $\mathcal{C}_i \subseteq \{1, \dots, C\}$.

Following standard restrictions of the federated setup, we assume that clients cannot share their *private* data. We also assume that all clients have access to a *shared* public unlabeled dataset $\mathcal{D}_{\text{pub}} \sim \mathcal{P}^X_{\text{pub}}$, containing covariates from all classes, where $\mathcal{P}^X_{\text{pub}}$ is a marginal distribution of $\mathcal{P}_{\text{pub}}$ w.r.t. the input x without label. Below, all guarantees are stated with respect to $x \sim \mathcal{P}^X_{\text{pub}}$.

We assume that client i has a local model parameterized by $\theta^{(i)} \in \mathbb{R}^d$ and aims to minimize its expected loss over $\mathcal{P}_i$:

$$F_i(\theta) \coloneqq \mathbb{E}_{\zeta \sim \mathcal{P}_i}[L(\theta, \zeta)],$$

where $\zeta = (x, y) \sim \mathcal{P}_i$ is a labeled example, with covariate x and label $y \in \{1, \dots, C\}$, and $L(\theta, \zeta) = \ell\big(f(x, \theta), y\big)$ is a proper point-wise loss. For the cross-entropy instance used throughout, $L(\theta, (x, y)) = -\log f(x, \theta)_y = \mathrm{CE}\big(e_y \,\|\, f(x, \theta)\big)$, where e_y denotes the one-hot encoding of y. The global objective is to minimize the average loss over clients:

$$\min_{\theta \in \mathbb{R}^d} F(\theta) \coloneqq \frac{1}{M} \sum_{i=1}^{M} F_i(\theta), \quad \theta^* \in \arg\min_{\theta \in \mathbb{R}^d} F(\theta).$$

Let $f(x, \theta) \in \Delta^{C-1}$ denote the predicted probability vector over C classes and let $p^*(x) \coloneqq f(x, \theta^*)$ be the reference predictor.

In what follows, our theoretical analysis assumes all clients share the same model architecture, i.e., a common parameter space $\mathbb{R}^d$, so that the averaged objective $F(\theta)$ and its minimizer $\theta^* \in \mathbb{R}^d$ are well-defined over a single parameter vector. In practice, however, clients can use different

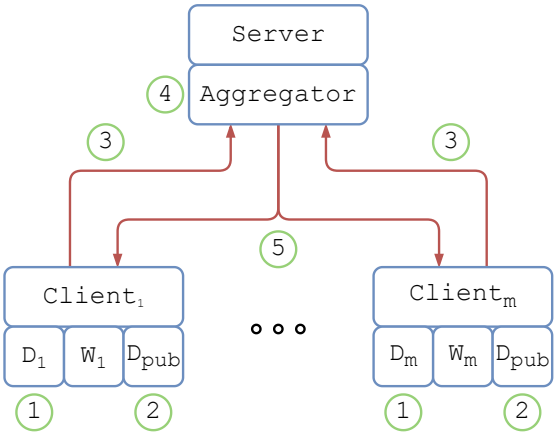

Figure 1: Our two-stage training loop works as follows. **Stage one.** (1) Clients train on their private labeled data. (2) Clients compute predictions on the public unlabeled dataset. (3) Clients send predictions to the server. **Stage two.** (4) The server aggregates predictions into soft labels. (5) The soft labels are sent to clients for refinement.

architectures, since logit-based methods only exchange predictions rather than model parameters. We do not cover this heterogeneous-architecture case in the theory, but we study it empirically in Section 5.4.

3.1 TWO-STAGE TRAINING

A single training round consists of two stages (see Figure 1).

In **Stage 1**, each client trains on its local data. It then computes predictions $q_i(x) = \mathrm{softmax}(f_i(x;\theta_i)) \in \Delta^{C-1}$ for every $x \in \mathcal{D}_{\text{pub}}$ and sends them to the server.

In **Stage 2**, the server forms a teacher $p_{\text{agg}}(x) \in \Delta^{C-1}$ by aggregating the received predictions. It then sends the aggregated vectors back to the clients, which refine their models on $\mathcal{D}_{\text{pub}}$.

Definition 3.1. We say that the aggregation rule is probability mixing if there exist weights $w_i(x) \geq 0$, $\sum_{i\in\mathcal{S}} w_i(x) = 1$ such that

$$p_{\text{agg}}(x) = \sum_{i\in\mathcal{S}} w_i(x) q_i(x).$$

In practice, one may aggregate logits and then apply softmax. Our analysis is *aggregation-agnostic* at the level of $p_{\text{agg}}(x)$. All rates depend on the teacher error $\mathbb{E}_x\|p_{\text{agg}}(x) - p^*(x)\|^2$. Definition 3.1 is used only when we want explicit M-dependence (e.g. $1/M$) from moment assumptions. In the following sections, we discuss different aggregation strategies for the server to combine client predictions. We illustrate the setup in Figure 1 and summarize the procedure in Algorithm 1.

Algorithm 1 Uncertainty-Aware Federated Distillation

Input: Private datasets $\{\mathcal{D}_i\}_{i=1}^M$, public dataset $\mathcal{D}_{\text{pub}}$, rounds R, learning rates η, η_{pub}, private epochs E_{priv}, public epochs E_{pub}, aggregation rule Agg, calibration datasets $\{\mathcal{D}_i^{cal}\}_{i=1}^M$
Output: Client models $\{f_i(\cdot;\theta_i)\}_{i=1}^M$

Initialize $\{\theta_i^0\}_{i=1}^M$
for round $r = 1$ to R **do**
  for each client $i \in [M]$ **in parallel do**
    Private training: update θ_i on $\mathcal{D}_i$ for E_{priv} epochs using cross-entropy
    Public inference: $q_i(x) \leftarrow \mathrm{softmax}(f_i(x;\theta_i))$ for all $x \in \mathcal{D}_{\text{pub}}$
    Fit GMM G_i on $\mathcal{D}_i^{cal}$
    Upload G_i and $\{q_i(x)\}_{x\in\mathcal{D}_{\text{pub}}}$ to the server
  end for
  for each $x \in \mathcal{D}_{\text{pub}}$ **do**
    Soft targets $p_{\text{agg}}(x) \leftarrow \mathrm{Agg}\big(\{(q_i(x), G_i)\}_{i=1}^M\big)$
  end for
  Broadcast $\{p_{\text{agg}}(x)\}_{x\in\mathcal{D}_{\text{pub}}}$ to all clients
  for each client $i \in [M]$ **in parallel do**
    Public refinement: update θ_i on $\mathcal{D}_{\text{pub}}$ for E_{pub} epochs using $\mathrm{CE}\big(p_{\text{agg}}(x)\|\mathrm{softmax}(f_i(x;\theta_i))\big)$
  end for
end for

4 THEORETICAL ANALYSIS

We provide a theoretical analysis of logit aggregation that justifies why averaging client predictions on a public dataset can recover an optimal predictor, and how aggregation of logits followed by finetuning on public data yields a provable improvement in the supervised objective.

4.1 TEACHER AGGREGATION DECOMPOSITION

Assumption 4.1 (Bias-variance-correlation for weighted aggregation)**.** Fix $x \sim \mathcal{P}_{\text{pub}}^{\mathcal{X}}$ and let $p^*(x) = f(x,\theta^*)$. Define client prediction errors

$$\xi_i(x) := q_i(x) - p^*(x), \qquad i \in \mathcal{S}.$$

Let $w(x) = (w_i(x))_{i\in\mathcal{S}}$ be the (possibly data-dependent) aggregation weights with $w_i(x) \geq 0$ and $\sum_{i\in\mathcal{S}} w_i(x) = 1$. Define the aggregated error $\bar{\xi}(x) := \sum_{i\in\mathcal{S}} w_i(x)\xi_i(x)$.

(Bias term.) Define the squared bias functional

$$B_w^2 := \mathbb{E}_x\Big[\big\|\mathbb{E}[\bar{\xi}(x) \mid x, w(x)]\big\|_2^2\Big].$$

(Variance and correlation terms.) Define centered errors $\tilde{\xi}_i(x) := \xi_i(x) - \mathbb{E}[\xi_i(x) \mid x, w(x)]$ and assume there exist constants $\sigma \geq 0$ and $\rho_c \in [0,1]$ such that uniformly in

$(x, w(x))$:

$$\mathbb{E}\big[\|\tilde{\xi}_i(x)\|_2^2 \mid x, w(x)\big] \le \sigma^2, \tag{1}$$

$$\big|\mathbb{E}\big[\langle \tilde{\xi}_i(x), \tilde{\xi}_j(x)\rangle \mid x, w(x)\big]\big| \le \rho_c \sigma^2, \qquad i \neq j. \tag{2}$$

Assumption 4.1 is a bias-variance-correlation control for the *weighted* teacher error $\bar{\xi}(x) = \sum_{i\in\mathcal{S}} w_i(x)\xi_i(x)$. The term B_w^2 is the (squared) *weighted bias* of the aggregated teacher error on public inputs, and is the main place where a *data-dependent* rule (e.g. UWA) can improve over uniform averaging by down-weighting systematically unreliable clients. The parameters σ^2 and ρ_c bound conditional second moments and correlations of centered client errors; $\rho_c = 0$ corresponds to conditionally uncorrelated client errors (maximal averaging benefit), while $\rho_c = 1$ yields no variance reduction. Finally, the quantity $\chi(x) = \sum_i w_i(x)^2$ measures weight concentration.

Lemma 4.2 (Teacher MSE under bias-variance-correlation)**.** *Consider probability-mixing aggregation $p_{\mathrm{agg}}(x) = \sum_{i\in\mathcal{S}} w_i(x) q_i(x)$ and assume the boundedness of its bias by B_w as well as the boundedness of variance and correlation of $q_i(x) - p^*(x)$ by σ^2 and $\rho_c\sigma^2$ respectively (see Assumption 4.1). Let*

$$\chi(x) := \sum_{i\in\mathcal{S}} w_i(x)^2, \qquad \chi := \mathbb{E}_x[\chi(x)].$$

Then, the teacher MSE satisfies:

$$\begin{aligned}\varepsilon_{\mathrm{teach}} &:= \mathbb{E}_x\|p_{\mathrm{agg}}(x) - p^*(x)\|_2^2 \\ &\le B_w^2 \;+\; \sigma^2\Big(\rho_c + (1-\rho_c)\chi\Big).\end{aligned} \tag{3}$$

Lemma 4.2 decomposes the teacher MSE into a *weighted bias* term B_w^2 and a *variance/correlation* term controlled by the weight concentration $\chi = \mathbb{E}_x \sum_i w_i(x)^2$. This makes the effect of the aggregation rule explicit.

4.2 AVERAGING AGGREGATION

The simplest existing approach is to average predictions across clients [Li and Wang, 2019, Lin et al., 2020]:

$$p_{\mathrm{agg}}^{\mathrm{AVG}}(x) = \frac{1}{M}\sum_{i=1}^{M} q_i(x).$$

Lemma 4.2 implies that $w_i(x) \equiv 1/M$ for all $i \in \mathcal{S}$ and all x, hence $\chi_{\mathrm{AVG}} = 1/M$. Thus, averaging yields the strongest possible variance reduction among all convex weights (it minimizes χ subject to $\sum_i w_i = 1$).

Recall that the client prediction error is

$$\xi_i(x) := q_i(x) - p^*(x),$$

so that $p_{\mathrm{agg}}(x) - p^*(x) = \sum_{i\in\mathcal{S}} w_i(x)\xi_i(x)$.

Then, the bias term becomes the bias of an *unconditional mixture* of client predictors:

$$B_{w,\mathrm{AVG}}^2 = \mathbb{E}_x\big\|\mathbb{E}[\tfrac{1}{M}\sum_{i\in\mathcal{S}} \xi_i(x) \mid x, w]\big\|^2,$$

so clients that are systematically unreliable on a given x still contribute equally to the teacher.

While straightforward, this treats all clients equally informed and reliable, regardless of how informative they are. A client that is simply guessing will have the same weight as one producing meaningful predictions. Under the class-mismatch setup we consider, some clients have never seen covariates from certain classes in the public dataset, so their predictions on these objects can be misleading.

4.3 CONVERGENCE RATES

Let $\mathcal{P}_{\mathrm{pub}}^X$ denote the input marginal of the public distribution. The public distillation objective for any model parameters θ is:

$$J_{\mathrm{agg}}(\theta) := \mathbb{E}_{x\sim\mathcal{P}_{\mathrm{pub}}^X}\Big[\mathrm{KL}\big(p_{\mathrm{agg}}(x)\|f(x,\theta)\big)\Big].$$

Stage 2 runs SGD on J_{agg}. At iteration t, client i draws $\{x_{tj}\}_{j=1}^b \overset{i.i.d.}{\sim} \mathcal{P}_{\mathrm{pub}}^X$ and sets:

$$\begin{aligned} g_t^{(i)} &:= \frac{1}{b}\sum_{j=1}^{b} \nabla_\theta \mathrm{KL}\big(p_{\mathrm{agg}}(x_{tj})\|f(x_{tj},\theta_t^{(i)})\big), \\ \theta_{t+1}^{(i)} &= \theta_t^{(i)} - \gamma g_t^{(i)}, \quad \mathbb{E}[g_t^{(i)} \mid \theta_t^{(i)}] = \nabla J_{\mathrm{agg}}(\theta_t^{(i)}). \end{aligned} \tag{4}$$

Since our analysis of Stage 2 applies to any client i, we omit the superscript (i) in the formulas and results that follow.

Assumption 4.3 (SGD noise for Stage 2)**.** There exists $\nu^2 \ge 0$ such that for all t:

$$\mathbb{E}[g_t \mid \theta_t] = \nabla J_{\mathrm{agg}}(\theta_t), \tag{5}$$

$$\mathbb{E}\big[\|g_t - \nabla J_{\mathrm{agg}}(\theta_t)\|_2^2 \big| \theta_t\big] \le \nu^2. \tag{6}$$

Assumption 4.3 is a standard unbiasedness and bounded variance condition for stochastic gradient estimators used in stochastic approximation / SGD analyses [Nemirovski et al., 2009, Ghadimi et al., 2016].

Assumption 4.4 (Smoothness (and optionally PL later))**.** F is L_F-smooth:

$$\|\nabla F(\theta) - \nabla F(\theta')\|_2 \le L_F\|\theta - \theta'\|_2, \quad \forall \theta, \theta' \in \mathbb{R}^d. \tag{7}$$

Optionally, for a stronger result, we assume Polyak-Łojasiewicz condition [Polyak, 1963, Lojasiewicz, 1963]:

$$\begin{aligned}\|\nabla F(\theta)\|_2^2 &\ge 2\mu_F\big(F(\theta) - F^*\big), \ \forall \theta \in \mathbb{R}^d, \\ F^* &:= F(\theta^*).\end{aligned} \tag{8}$$

Assumption 4.4 uses L_F-smoothness, a standard regularity condition for first-order optimization methods [Nesterov, 2018]. When we additionally invoke the Polyak-Łojasiewicz (PL) inequality (second line of Assumption 4.4), we follow the classical condition analyzed by Polyak [1963].

Assumption 4.5 (Gradient alignment between F and ideal KD)**.** Define the *ideal-teacher* objective

$$J_*(\theta) := \mathbb{E}_{x\sim\mathcal{D}_{\mathrm{pub},x}}\Big[\mathrm{KL}(p^*(x)\|f(x,\theta))\Big]. \tag{9}$$

There exist constants $c \in (0,1)$ and $b \geq 0$ such that the following inequality holds for all t:

$$\mathbb{E}\left[\|\nabla F(\theta_t) - \nabla J_*(\theta_t)\|^2\right] \leq c\mathbb{E}\left[\|\nabla F(\theta_t)\|^2\right] + b^2 \tag{10}$$

Remark 4.6. By the definition of $p^*(x)$ and θ^*, we have $\nabla J_*(\theta^*) = 0$ and $\nabla F(\theta^*) = 0$. Hence, the above assumption is natural near θ^*. In practice, as training approaches its final phase, we expect θ_t to be close to θ^*, implying that b decreases toward the end of Stage 2. For the sake of analytical simplicity, however, we treat the constants c and b as fixed.

Assumption 4.7 (Score-energy bound)**.** There exists a constant $G_{\mathrm{sc}} \geq 0$ such that for all t:

$$\mathbb{E}\Big[\mathbb{E}_{x\sim\mathcal{D}_{\mathrm{pub},x}}\Big[\sum_{c=1}^{C}\|\nabla_\theta \log f_c(x,\theta_t)\|_2^2\Big]\Big] \leq G_{\mathrm{sc}}^2. \tag{11}$$

Assumption 4.7 upper bounds the Frobenius norm of the *score matrix* $S(x,\theta) \in \mathbb{R}^{C\times d}$ with rows $\nabla_\theta \log f_c(x,\theta)^\top$. In softmax models, $S(x,\theta)$ is controlled by the Jacobian of logits (and feature norms), so empirically G_{sc}^2 can be supported by measuring average score/Jacobian norms along training. In many softmax models, $\|S(x,\theta)\|_F^2$ can scale like $O(C)$ times a feature-norm term; thus S^2 may grow with C unless features/parameters are controlled (e.g., by regularization or by operating in a bounded region).

Biased gradient oracle model. Although Stage 2 optimizes J_{agg}, we evaluate progress in terms of the objective function F. Therefore, if we define

$$\beta(\theta) := \nabla J_{\mathrm{agg}}(\theta) - \nabla F(\theta), \quad n_t := g_t - \nabla J_{\mathrm{agg}}(\theta_t),$$

then we can interpret the procedure described in (4) as biased SGD $\theta_{t+1} = \theta_t - \gamma g_t$ applied to the minimization of function F with biased stochastic gradient:

$$g_t = \nabla F(\theta_t) + \beta(\theta_t) + n_t, \quad \mathbb{E}[n_t \mid \theta_t] = 0.$$

Under our aggregation and regularity assumptions, we show in Proposition 4.8 that this bias satisfies an *expected biased-oracle* inequality of the form $\mathbb{E}\|\beta(\theta_t)\|^2 \leq m\,\mathbb{E}\|\nabla F(\theta_t)\|^2 + \zeta^2$. This is a weaker version of the pointwise oracle used by Ajalloeian and Stich [2021], and we therefore provide an expected-oracle adaptation of their descent argument in Appendix C.6.

Proposition 4.8 (Expected-oracle conditions adaptation)**.** *Assume the boundedness of aggregation bias by B_w as well as the boundedness of variance and correlation of $q_i(x) - p^*(x)$ by σ^2 and $\rho_c\sigma^2$ respectively (see Assumption 4.1); $c \in (0,1)$ and $b \geq 0$ quantify the alignment between $\nabla F(\theta_t)$ and the ideal distillation objective $\nabla J_*(\theta_t)$ (Assumption 4.5); G_{sc}^2 bounds the expected squared Frobenius norm of the score matrix (Assumption 4.7); and ν^2 bounds the variance of the stochastic gradient g_t (Assumption 4.3). Fix $\alpha > \frac{c}{1-c}$ and define*

$$m := \Big(1 + \frac{1}{\alpha}\Big)c,$$
$$\zeta^2 := (1+\alpha)G_{\mathrm{sc}}^2\,\varepsilon_{\mathrm{teach}} + \Big(1 + \frac{1}{\alpha}\Big)b^2. \tag{12}$$

Then the bias satisfies

$$\mathbb{E}\big[\|\beta(\theta_t)\|_2^2\big] \leq m\,\mathbb{E}\big[\|\nabla F(\theta_t)\|_2^2\big] + \zeta^2, \tag{13}$$

and the noise satisfies

$$\mathbb{E}\big[\|n_t\|_2^2 \mid \theta_t\big] \leq \nu^2. \tag{14}$$

Therefore, the Stage 2 update (4) *fits the biased SGD oracle model of [Ajalloeian and Stich, 2021] (with $M_{Aj} = 0$ and $\sigma_{Aj}^2 = \nu^2$ in their Assumption 3).*

Below, we state the resulting non-convex stationarity and PL convergence bounds, highlighting how the constants (m, ζ^2, ν^2) depend on the aggregation quality through $\varepsilon_{\mathrm{teach}}$ (and, via Lemma 4.2, on the weight concentration χ and weighted bias B_w^2). We also translate these bounds into iteration complexities.

Corollary 4.9 (Big-$\mathcal{O}$ complexity for non-convex stationarity)**.** *Assume F is L_F-smooth and let the conditions of Proposition 4.8 hold. Define m, ζ, ν^2 as in that proposition. Let $\epsilon_{\min} := \zeta^2/(1-m)$ and fix any $\epsilon > \epsilon_{\min}$. Define $\bar{\epsilon} := \epsilon - \epsilon_{\min}$. Choosing $\gamma := \min\{1/L_F,\ \bar{\epsilon}(1-m)/(2L_F\nu^2)\}$, it suffices to take*[1]

$$T = \mathcal{O}\left(\frac{L_F\nu^2\,(F(\theta_0) - F_{\inf})}{(1-m)^2\,\bar{\epsilon}^2}\right)$$

to ensure $\mathbb{E}\|\nabla F(\theta_{\mathrm{out}})\|^2 \leq \epsilon$.

Corollary 4.10 (Big-$\mathcal{O}$ complexity under PL)**.** *Assume F is L_F-smooth and satisfies the Polyak–Łojasiewicz (PL) condition: $\|\nabla F(\theta)\|^2 \geq 2\mu_F(F(\theta) - F^*)$, $\mu_F > 0$. Let the conditions of Proposition 4.8 hold and define m, ζ, ν^2 as in that proposition. Define the floor*

$$\epsilon_{\mathrm{floor}}(\gamma) := \frac{\zeta^2 + L_F\gamma\nu^2}{2\mu_F(1-m)}.$$

[1] We neglect $1/\bar{\epsilon}$ term since it is dominated by the $1/\bar{\epsilon}^2$ term for sufficiently small $\bar{\epsilon}$.

For any $\epsilon > \epsilon_{\text{floor}}(\gamma)$, $\gamma \le 1/L_F$, it suffices to take

$$T = \mathcal{O}\Big(\frac{1}{\gamma\mu_F(1-m)} \log \frac{F(\theta_0) - F^*}{\epsilon - \epsilon_{\text{floor}}(\gamma)}\Big)$$

to ensure $\mathbb{E}[F(\theta_T) - F^*] \le \epsilon$.

Proof sketch (non-convex stationarity). Stage 2 updates satisfy the biased-gradient decomposition $g_t = \nabla F(\theta_t) + \beta(\theta_t) + n_t$. Proposition 4.8 (Appendix) bounds the bias in expected-oracle form $\mathbb{E}\|\beta(\theta_t)\|^2 \le m\,\mathbb{E}\|\nabla F(\theta_t)\|^2 + \zeta^2$ and controls the noise by $\mathbb{E}[\|n_t\|^2 \mid \theta_t] \le \nu^2$. Applying L_F-smoothness to one SGD step and taking the conditional expectation yields a descent inequality with a negative $\mathbb{E}\|\nabla F(\theta_t)\|^2$ term plus bias/noise corrections. Summing over t and sampling θ_{out} uniformly from $\{\theta_0, \dots, \theta_{T-1}\}$ gives the stated stationarity bound. Full details are in Appendix C.

Proof sketch (PL linear rate). Starting from the same one-step descent inequality, the PL condition converts the gradient-norm term into a contraction in $\mathbb{E}[F(\theta_t) - F^*]$, yielding a linear recursion with rate $1 - \gamma\mu_F(1-m)$ and an additive floor proportional to $(\zeta^2 + L_F\gamma\nu^2)/(\mu_F(1-m))$. Unrolling the recursion gives the claimed linear convergence to a neighborhood. Full details are in Appendix C.

On the error floor. Both results establish convergence only up to an irreducible optimization error. In the general non-convex setting, this error is given by $\epsilon_{\min} := \zeta^2/(1-m)$, which is consistent with existing results for biased SGD [Ajalloeian and Stich, 2021]. However, as shown in Proposition 4.8, the error floor can be reduced through improved aggregation. Specifically, the first term in the definition of ζ^2 in (12) is proportional to $\varepsilon_{\text{teach}} := \mathbb{E}_x\|p_{\text{agg}}(x) - p^*(x)\|_2^2$, which quantifies the quality of the aggregation rule. In the case of (near-)ideal aggregation, we have $\varepsilon_{\text{teach}} \approx 0$, and consequently $\zeta \sim b$, where b (together with c) characterizes the alignment between $\nabla F(\theta_t)$ and the ideal distillation objective $\nabla J_*(\theta_t)$ (Assumption 4.5). As discussed in the appendix (see Remark 4.6), it is natural to expect b to be small toward the end of training. Therefore, our results imply convergence to a small and practically meaningful optimization error, provided that the aggregation rule is sufficiently accurate.

Regarding single-round scope. Theorems in Section 4.3 analyze a single Stage 2 distillation phase with a fixed teacher p_{agg} (computed once from Stage 1 predictors). This isolates the effect of aggregation quality through $\varepsilon_{\text{teach}}$ (and χ) on the optimization floor.

In a multi-round procedure, each round recomputes a teacher $p_{\text{agg}}^{(r)}$ and runs Stage 2 for T_2 steps. The single-round analysis applies within each round with round-dependent quantities $\varepsilon_{\text{teach},r}$, χ_r, and the induced oracle parameters (m_r, ζ_r^2). Without additional restrictions on Stage 1, monotone improvement across rounds cannot be guaranteed in the worst case. However, if teacher-quality proxies (e.g., $\varepsilon_{\text{teach},r}$ or χ_r) improve over rounds, the theory predicts a shrinking neighborhood floor accordingly.

Proofs and further details. All missing technical details and complete proofs are provided in Appendix C.

4.4 UWA: UNCERTAINTY-WEIGHTED AVERAGING

To address issues raised in Section 4.2, we propose weighting each client's prediction by its confidence. We use a density-based approach that links predictive uncertainty to how typical the model's outputs look [Van Amersfoort et al., 2020, Mukhoti et al., 2023, Kotelevskii et al., 2022a].

For each client i, we keep a separate labeled calibration split $\mathcal{D}_i^{cal}$ (20% of private data in our experiments) that is not used for training, but is used to fit a density model. There are multiple options for selecting a density model, depending on the level of flexibility required. One of the reasonable choices is a Gaussian Mixture Model (GMM) with $|\mathcal{C}_i|$ components (one per class $k \in \mathcal{C}_i$), equal mixture weights, and diagonal covariance. For simplicity, we fit the GMM in a gaussian-per-class manner, to the logits produced on the calibration set $\mathcal{D}_i^{cal}$: $z_i(x) = f_i(x;\theta_i) \in \mathbb{R}^C$. We refit it after each training phase on a local (private) dataset.

Let us now describe how we use it for objects coming from the shared public dataset. Denote by $\mu_{i,k,d}$ and $\sigma_{i,k,d}$ the mean and standard deviation of the d-th logit component for class k on client i. To evaluate the reliability of a client, we compute the log-likelihood under the fitted GMM for each x from the shared public dataset:

$$\begin{aligned}\ell_i(x) &:= \log p(x \mid i) \\ &= \log\Big(\frac{1}{|\mathcal{C}_i|}\sum_{k\in\mathcal{C}_i} \mathcal{N}\big(z_i(x); \mu_{i,k}, \sigma_{i,k}^2\big)\Big),\end{aligned}$$

where $\mathcal{N}(\cdot;\mu,\Sigma)$ denotes a Gaussian with mean μ and covariance Σ. The fully expanded per-component expression for $\ell_i(x)$ is given in Appendix A.4.

The value of $\ell_i(x)$ tells us how reliable the prediction of client i is for input x. When $\ell_i(x)$ is large, the logits that the client predicts look similar to what it saw during training, so it is likely producing informed predictions. When $\ell_i(x)$ is small, the logits are unusual for this client, which typically happens for classes outside $\mathcal{C}_i$. Therefore, the predictions are likely not well-informed, and the client is instead guessing. We empirically confirm that these per-client GMMs separate a client's own-class inputs from out-of-class inputs in Appendix B.7.

We then produce weights from these log-likelihood values

by applying a softmax over all clients for a fixed input x:

$$w_i(x) = \frac{\exp(\ell_i(x))}{\sum_{j=1}^{M} \exp(\ell_j(x))} = \frac{p(x \mid i)}{\sum_j p(x \mid j)}.$$

Note that if $p(i) = p(j)$ for all i, j, Bayes' theorem gives $w_i(x) = p(i \mid x)$.

The soft target labels are then a weighted average of client predictions:

$$p_{\text{agg}}^{UWA}(x) = \sum_{i=1}^{M} w_i(x) q_i(x) \approx \mathbb{E}_{w_i(x)} q_i(x). \tag{15}$$

For UWA, the weights $w_i(x)$ depend on the input x, so $\chi(x)$ is generally *larger* than $1/M$ when the weights concentrate on a subset of clients (less averaging):

$$\chi_{UWA} = \mathbb{E}_x \sum_{i \in \mathcal{S}} w_i(x)^2 \in \Big[\frac{1}{M}, 1\Big].$$

This increases the variance term in Lemma 4.2 relative to AVG whenever weights are concentrated. However, a data-dependent rule, such as UWA, may reduce B_w^2 by down-weighting clients whose conditional mean error is large, i.e., those that are out-of-distribution for the given input. While logit-density scores have been used for out-of-distribution detection [Mukhoti et al., 2023], we adapt them here to federated aggregation under client heterogeneity.

More explicitly, let $\mu_i(x, w) := \mathbb{E}[\xi_i(x) \mid x, w(x)]$. Then

$$\mathbb{E}[\bar{\xi}(x) \mid x, w] = \sum_{i \in \mathcal{S}} w_i(x) \mu_i(x, w),$$
$$\big\|\mathbb{E}[\bar{\xi}(x) \mid x, w]\big\|_2^2 \le \sum_{i \in \mathcal{S}} w_i(x) \, \|\mu_i(x, w)\|_2^2,$$

so the bias contribution can decrease when weights are concentrated on clients with smaller $\|\mu_i(x, w)\|_2$.

Hence, UWA trades off less variance reduction (larger χ) for potentially smaller bias (smaller B_w^2). In regimes with strong client heterogeneity (large systematic bias), UWA can reduce B_w^2 substantially and outperform averaging. In more homogeneous regimes (small bias), uniform averaging tends to be preferable because it minimizes χ.

Smoothed UWA. In our experiments, we observed that UWA can produce highly peaked weight distributions, especially in early rounds of training. We hypothesize that this is due to unreliable logits learned by clients, which result in flawed density models fitted on them. This effect can arise from overfitting of clients' models to their local data.

In what follows, we present a practical modification of UWA that empirically improves performance. We refer to it as *smoothed UWA (sUWA)*.

The idea is to scale the log-likelihoods produced by local density models in a temperature scaling-like manner [Guo et al., 2017]. Specifically, we introduce a scalar temperature $\tau > 0$ and produce sUWA weights as follows:

$$w_i^{\text{sUWA}}(x) = \frac{\exp(\tau \ell_i(x))}{\sum_{j \in \mathcal{S}} \exp(\tau \ell_j(x))}.$$

The method interpolates between three extremes:

- $\tau = 1$ recovers UWA;
- $\tau \to \infty$ selects the most confident model;
- $\tau = 0$ recovers AVG.

In our experiments, we set $\tau < 1$ to reduce the concentration of aggregation weight. Specifically, we fix $\tau = 0.25$ across all experiments.

From the perspective of Lemma 4.2, the temperature controls the bias-variance tradeoff: lower τ decreases weight concentration χ (better variance reduction) at the cost of higher bias B_w^2.

5 EXPERIMENTS

In this section, we empirically study different aggregation approaches. We consider classification in two domains: (i) image classification on CIFAR-10 [Krizhevsky, 2009] trained from scratch, (ii) image classification on CIFAR-100 using transfer learning [Krizhevsky, 2009], and (iii) text classification on Yahoo Answers [Chang et al., 2008] with transfer learning. In all settings, we use 20 clients and vary the degree of heterogeneity by changing the number of classes k assigned to each client.

We compare three aggregation methods, namely **AVG** (uniform averaging, baseline), **UWA** (uncertainty-weighted averaging), and **sUWA** (smoothed UWA with temperature $\tau = 0.25$); their precise definitions are collected in Appendix A.3.

5.1 EXPERIMENTAL SETUP

Depending on the dataset, we consider different schemes of class distributions over clients:

- For **CIFAR-10**, each client has $k \in \{2, 3, 4, 5, 7, 9\}$ classes. For the classification, we use ResNet-18 [He et al., 2016] trained from scratch.
- For **CIFAR-100**, $k \in \{20, 30, 40, 50, 70, 90\}$ classes per client, and as a model we use ResNet-18, pretrained on ImageNet [Deng et al., 2009].
- For **Yahoo Answers** [Zhang et al., 2015], a topic classification benchmark with 10 classes, each client is assigned $k \in \{2, 3, 4, 5, 7, 9\}$ classes. We use BERT-tiny [Turc et al., 2019], pretrained on Wikipedia and BookCorpus, as the client model.

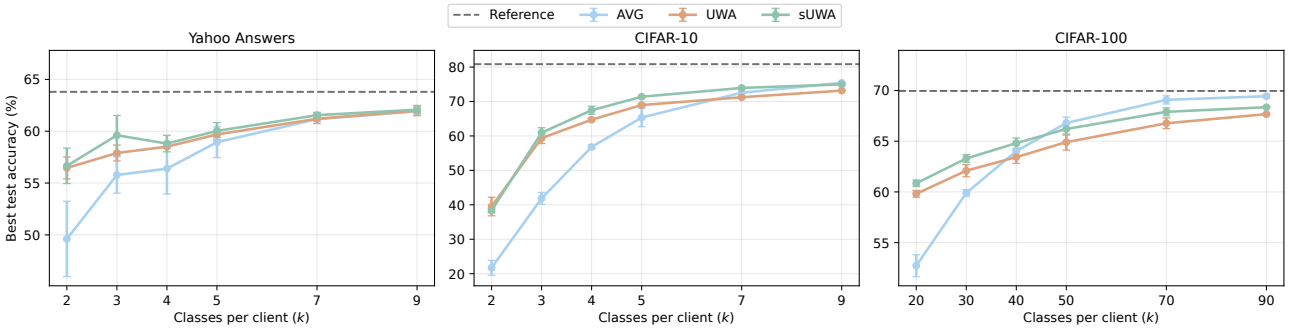

Figure 2: Best test accuracy vs. classes per client (k). Dashed line: fully-informed reference model trained on all-classes dataset with size $|\mathcal{D}_{pub}| + |\mathcal{D}_i|$.

Full dataset statistics and preprocessing details are provided in Appendix A.2, and we analyze how the public and private dataset sizes affect performance in Appendix B.5.

5.2 HOMOGENEOUS SETTING

Before studying heterogeneous settings, we verify our theoretical analysis in the homogeneous case. We train $M = 20$ clients on CIFAR-10, each observing all 10 classes with 100 samples per class (1,000 private training samples per client). The shared public dataset contains 500 samples per class (5,000 total).

We observe that communication between clients helps significantly. Without it, local training achieves only $42.40\% \pm 0.4\%$ test accuracy. A single round of distillation already raises accuracy to $58.04\% \pm 0.06\%$. This confirms that aggregating clients' predictions improves convergence (see Section 4), and unlabeled public data provides an effective channel for knowledge transfer.

5.3 HETEROGENEOUS SETTING

In this section, we consider a practical scenario of distributed heterogeneous clients.

We present average client test accuracy across all methods and heterogeneity levels. The results are summarized in Table 2. The density-weighted methods provide the most significant improvement at low k (i.e., high heterogeneity), where most public samples are out-of-distribution for a given client. As heterogeneity decreases, clients become equally aware of all classes, and therefore UWA and sUWA converge to AVG. Without heterogeneity, all methods achieve nearly the same test accuracy. Smoothed UWA consistently outperforms UWA, especially at moderate heterogeneity.

In Figure 2, we display the same results visually. We see that uncertainty-aware approaches (both UWA and sUWA) are most effective in high-data-heterogeneity settings. On CIFAR-10, these approaches reach almost 40% accuracy when clients observe only 20% of classes. As k increases, client predictions become more reliable, the weights in uncertainty-aware methods approach uniformity, reducing these methods to simple averaging. Full per-round training curves (global and local test accuracy) for CIFAR-10 across all heterogeneity levels are shown in Figure 5 in the appendix.

Smoothed UWA tends to outperform both AVG and UWA across a broad range of k. We have the following explanation for this observation. Early in training, local models can overfit to their private data, leading to unreliable density estimates. This issue makes the uncertainty weights flawed and overly concentrated on a few clients. The temperature parameter flattens the weight distribution, preventing any single client from dominating the aggregation (in practice, $\tau = 0.25$). In section B.6 we present an ablation study on the temperature parameter.

5.4 HETEROGENEOUS CLIENT ARCHITECTURES

Section 3 noted that federated distillation supports clients with different model architectures, a case our theory does not cover. We verify it empirically on CIFAR-10 by assigning the 20 clients four distinct backbones in round-robin (five clients each): ResNet-18, a VGG-style CNN, DenseNet-121, and ResNet-34 (ranging from $\sim$0.6M to $\sim$21M parameters). Each client still observes only k of the 10 classes, and all clients share the same C-dimensional output space, so the aggregation step is unchanged. Figure 4 shows test accuracy over rounds for uniform averaging (AVG) and sUWA. Density-based weighting remains effective with mixed architectures: sUWA improves over AVG at every tested k, by roughly 3 to 10 percentage points. This indicates that the benefit of down-weighting unreliable clients does not depend on a shared architecture.

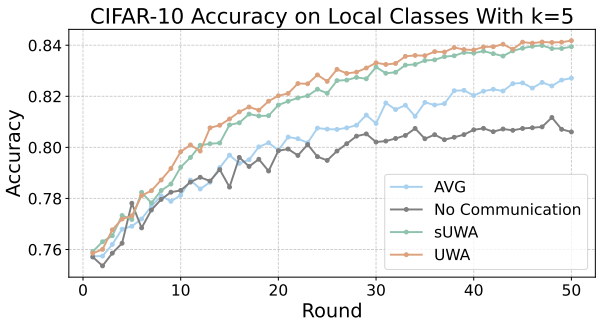

Figure 3: Mean accuracy on local classes over communication rounds.

5.5 COMMUNICATION COST

We compare the cost of our logit-based methods against SCAFFOLD [Karimireddy et al., 2020], a gradient-based federated learning method that uses control variates to correct client drift.

The two paradigms use a comparable number of communication rounds to converge, so we compare the bytes transferred per round, which are far lower for logit-based approaches. Each SCAFFOLD round transmits the full model together with its control variate, whereas a distillation round transmits only soft labels on the public set. On Yahoo Answers (BERT-tiny, $\sim$4.3M parameters), this is about 34MB per client per round for SCAFFOLD, versus about 0.08MB for AVG and UWA (2,000 public samples $\times$ 10 classes $\times$ 4 bytes), a $\sim 430\times$ reduction. On CIFAR-100 (ResNet-18, $\sim$11.7M parameters, $C = 100$, 5,000 public samples), it is about 94MB per round for SCAFFOLD versus 2MB for the logit methods, a $\sim 47\times$ reduction.

5.6 COMPARISON TO GRADIENT-BASED BASELINES

We also compare accuracy directly against gradient-based federated methods on CIFAR-10: FedAvg [McMahan et al., 2017], SCAFFOLD [Karimireddy et al., 2020], and a private-data adaptation of FedAux [Sattler et al., 2022]. All methods share the same federation setup (20 clients, 50 rounds, 3 seeds). The gradient-based baselines train on private data only, whereas the aggregation methods additionally use the unlabeled public set for distillation; SCAFFOLD uses SGD with a tuned learning rate (see Appendix A.1). Figure 6 reports the best test accuracy over the 50 rounds. The outcome is heterogeneity-dependent. At the most severe class mismatch ($k = 2$), SCAFFOLD's drift correction makes it competitive with the uncertainty-aware aggregators, with UWA, sUWA, and SCAFFOLD all within a standard deviation of one another. At moderate heterogeneity ($k = 3$), the uncertainty-aware aggregators lead (sUWA ahead of UWA); SCAFFOLD is the strongest gradient baseline but trails them, while FedAvg and FedAux fall well behind. At $k = 4$, sUWA is best, while SCAFFOLD, FedAvg, FedAux, and UWA cluster closely below it. FedAvg and FedAux lag under strong heterogeneity and only catch up as k grows. Crucially, the distillation methods reach this accuracy while transmitting orders of magnitude less data (Section 5.5).

5.7 LOCAL CLASS IMPROVEMENT

Over communication rounds, the test accuracy on local classes (measured after the local stage) improves significantly compared to training without communication (see Figure 3). The overlap between local class sets across clients can explain this improvement. Our methods perform significantly better in this scenario as well.

5.8 LIMITATIONS

Our approach relies on a shared, unlabeled public dataset accessible to all clients and the server. While such data is available in many practical settings (e.g., publicly crawled text or images), it may not always be available in specialized domains, such as medical or financial data. Additionally, the communication advantage over gradient-based methods decreases as the number of classes grows. Each client sends a $(C-1)$-dimensional probability vector per public sample, so for datasets with many classes and large public sets, the per-round cost increases accordingly, though it remains below gradient-based costs.

6 CONCLUSION

In this paper, we studied the problem of aggregating client predictions in Federated Distillation under data heterogeneity. We provided a theoretical analysis showing that aggregating predictions from different clients on a shared public dataset converges to a neighborhood of the optimum, with the neighborhood size controlled by the quality of the aggregating strategy.

As specific aggregation strategies, we proposed UWA and sUWA, uncertainty-aware methods that down-weight unreliable client predictions using the density of predicted logits. In our experiments on image (CIFAR-10, CIFAR-100) and text (Yahoo Answers) datasets, we demonstrated that these methods are most effective in high-data-heterogeneity scenarios. At the same time, they match the standard averaging procedure in low- or no-heterogeneity settings.

References

Ahmad Ajalloeian and Sebastian U. Stich. On the convergence of sgd with biased gradients, 2021. URL

https://arxiv.org/abs/2008.00051.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In *International conference on machine learning*, pages 1613–1622. PMLR, 2015.

Ming-Wei Chang, Lev-Arie Ratinov, Dan Roth, and Vivek Srikumar. Importance of semantic representation: Dataless classification. In *AAAI*, volume 2, pages 830–835, 2008.

Liam Collins, Hamed Hassani, Aryan Mokhtari, and Sanjay Shakkottai. Exploiting shared representations for personalized federated learning. In *International conference on machine learning*, pages 2089–2099. PMLR, 2021.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

Jakob Gawlikowski, Cedrique Rovile Njieutcheu Tassi, Mohsin Ali, Jongseok Lee, Matthias Humt, Jianxiang Feng, Anna Kruspe, Rudolph Triebel, Peter Jung, Ribana Roscher, et al. A survey of uncertainty in deep neural networks. *Artificial intelligence review*, 56(Suppl 1): 1513–1589, 2023.

Saeed Ghadimi, Guanghui Lan, and Hongchao Zhang. Minibatch stochastic approximation methods for nonconvex stochastic composite optimization. *Mathematical Programming*, 155(1):267–305, 2016.

Eduard Gorbunov, Filip Hanzely, and Peter Richtárik. Local SGD: Unified theory and new efficient methods. In *International Conference on Artificial Intelligence and Statistics*, pages 3556–3564. PMLR, 2021.

Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International journal of computer vision*, 129(6):1789–1819, 2021.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR, 2017.

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*, pages 770–778, 2016.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Eyke Hüllermeier and Willem Waegeman. Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine learning*, 110(3): 457–506, 2021.

Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. 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*, pages 5132–5143. PMLR, 2020.

Jakub Konečnỳ, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

Nikita Kotelevskii, Aleksandr Artemenkov, Kirill Fedyanin, Fedor Noskov, Alexander Fishkov, Artem Shelmanov, Artem Vazhentsev, Aleksandr Petiushko, and Maxim Panov. Nonparametric uncertainty quantification for single deterministic neural network. *Advances in Neural Information Processing Systems*, 35:36308–36323, 2022a.

Nikita Kotelevskii, Maxime Vono, Alain Durmus, and Eric Moulines. Fedpop: A bayesian approach for personalised federated learning. *Advances in Neural Information Processing Systems*, 35:8687–8701, 2022b.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

Daoyuan Li and Junpu Wang. Fedmd: Heterogeneous federated learning via model distillation. *arXiv preprint arXiv:1910.03581*, 2019.

Lin Li, Jianping Gou, Baosheng Yu, Lan Du, and Zhang Yiand Dacheng Tao. Federated distillation: A survey. *arXiv preprint arXiv:2404.08564*, 2024.

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.

Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Ditto: Fair and robust federated learning through personalization. In *International conference on machine learning*, pages 6357–6368. PMLR, 2021.

Tao Lin, Lingjing Kong, Sebastian U Stich, and Martin Jaggi. Ensemble distillation for robust model fusion in federated learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Stanislaw Lojasiewicz. A topological property of real analytic subsets. *Coll. du CNRS, Les équations aux dérivées partielles*, 117(87-89):2, 1963.

H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 1273–1282, 2017.

Konstantin Mishchenko, Rustem Islamov, Eduard Gorbunov, and Samuel Horváth. Partially personalized federated learning: Breaking the curse of data heterogeneity. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=8tMMCf4YYn.

Jishnu Mukhoti, Andreas Kirsch, Joost Van Amersfoort, Philip HS Torr, and Yarin Gal. Deep deterministic uncertainty: A new simple baseline. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24384–24394, 2023.

Arkadi Nemirovski, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on optimization*, 19(4):1574–1609, 2009.

Yurii Nesterov. *Lectures on convex optimization*, volume 137. Springer, 2018.

Krishna Pillutla, Kshitiz Malik, Abdel-Rahman Mohamed, Mike Rabbat, Maziar Sanjabi, and Lin Xiao. Federated learning with partial model personalization. In *International Conference on Machine Learning*, pages 17716–17758. PMLR, 2022.

Boris T Polyak. Gradient methods for the minimisation of functionals. *USSR Computational Mathematics and Mathematical Physics*, 3(4):864–878, 1963.

Aniruddh Raghu, Maithra Raghu, Samy Bengio, and Oriol Vinyals. Rapid learning or feature reuse? towards understanding the effectiveness of maml. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=rkgMkCEtPB.

Felix Sattler, Arturo Marban, Roman Rischke, and Wojciech Samek. Cfd: Communication-efficient federated distillation via soft-label quantization and delta coding. *IEEE Transactions on Network Science and Engineering*, 9(4): 2025–2038, 2022. doi: 10.1109/TNSE.2021.3081748.

Hyowoon Seo, Jihong Park, Seungeun Oh, Mehdi Bennis, and Seong-Lyun Kim. Federated knowledge distillation. *Machine Learning and Wireless Communications*, 457, 2022.

Jiawei Shao, Fangzhao Wu, and Jun Zhang. Selective knowledge sharing for privacy-preserving federated distillation without a good teacher. *Nature Communications*, 15(1): 349, 2024.

Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Well-read students learn better: On the importance of pre-training compact models. *arXiv preprint arXiv:1908.08962*, 2019.

Joost Van Amersfoort, Lewis Smith, Yee Whye Teh, and Yarin Gal. Uncertainty estimation using a single deep deterministic neural network. In *International conference on machine learning*, pages 9690–9700. PMLR, 2020.

Jianyu Wang, Zachary Charles, Zheng Xu, Gauri Joshi, H Brendan McMahan, Maruan Al-Shedivat, Galen Andrew, Salman Avestimehr, Katharine Daly, Deepesh Data, et al. A field guide to federated optimization. *arXiv preprint arXiv:2107.06917*, 2021.

Guoyizhe Wei and Xiu Li. Knowledge lock: Overcoming catastrophic forgetting in federated learning. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 601–612. Springer, 2022.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, volume 28, pages 649–657, 2015.

Supplementary Material

Viktor Kovalchuk[1] **Denis Son**[2] **Arman Bolatov**[1] **Mohsen Guizani**[1] **Samuel Horváth**[1] **Maxim Panov**[1]
Martin Takáč[1] **Eduard Gorbunov**[1] **Nikita Kotelevskii**[1]

[1]Mohamed bin Zayed University of Artificial Intelligence (MBZUAI), Abu Dhabi, UAE
[2]Independent Researcher

A EXPERIMENTAL DETAILS

A.1 HYPERPARAMETERS

Table 1 summarizes the hyperparameters used across all experiments.

Table 1: Hyperparameters for all experiments. All experiments use $M{=}20$ clients, 3 random seeds, and sUWA temperature $\tau{=}0.25$. For UWA and sUWA GMMs are fitted locally using the same private dataset, but augmented.

Parameter	**Yahoo Answers**	**CIFAR-10**	**CIFAR-100**
Federation Setup			
Classes per client (k)	{2,3,4,5,7,9}	{2,3,4,5,7,9}	{20,30,40,50,70,90}
Private samples/client	1,500	5,000	5,000
Public dataset size	2,000	5,000	5,000
Communication rounds	50	50	50
Model Architecture			
Backbone	BERT-tiny	ResNet-18	ResNet-18
Pretrained	Yes (Wikipedia + BookCorpus)	No	Yes (ImageNet-1K)
Training			
Optimizer	AdamW	Adam	Adam
Learning rate	$2{\times}10^{-5}$	$1{\times}10^{-3}$	$1{\times}10^{-4}$
Batch size	32	128	128
First round epochs	3	20	1
Subsequent epochs	1	2	1

A.2 DATASET DETAILS

Yahoo Answers. Yahoo Answers [Zhang et al., 2015] is a topic classification dataset with 10 classes (e.g., Science & Mathematics, Sports, Health). It contains 1,400,000 training samples and 60,000 test samples. Each sample consists of a question title, question content, and best answer concatenated together.

CIFAR-10 and CIFAR-100. CIFAR-10 contains 60,000 32×32 color images across 10 classes (50,000 train, 10,000 test). CIFAR-100 contains the same number of images but with 100 fine-grained classes grouped into 20 superclasses.

A.3 AGGREGATION METHOD DETAILS

Simple Averaging (AVG). Computes the element-wise mean of client softmax probabilities:

$$\bar{q}(x) = \frac{1}{M} \sum_{i=1}^{M} q_i(x), \quad q_i(x) = \text{softmax}(f_i(x)).$$

UWA (Uncertainty-Weighted Averaging). Each client fits a Gaussian mixture model over its logit distribution using private training data (one component per local class). At aggregation time, the confidence score for client i on input x is:

$$\ell_i(x) = \log p_{\text{GMM}_i}(f_i(x)), \quad w_i(x) = \frac{\exp(\ell_i(x))}{\sum_j \exp(\ell_j(x))}.$$

sUWA (Smoothed UWA). Uses a temperature parameter $\tau = 0.25$ to smooth the UWA weights:

$$w_i^{\text{sUWA}}(x) = \frac{\exp(\tau \cdot \ell_i(x))}{\sum_j \exp(\tau \cdot \ell_j(x))}.$$

As $\tau \to 0$, this reduces to uniform averaging (AVG).

A.4 UWA CONFIDENCE SCORE FORMULA

For the Uncertainty-Weighted Averaging (UWA) method, the confidence score for client i on input x is computed as:

$$\ell_i(x) \coloneqq \log \left(\frac{1}{|\mathcal{C}_i|} \sum_{k \in \mathcal{C}_i} (2\pi)^{-C/2} \prod_{d=1}^{C} \sigma_{i,k,d}^{-1} \exp \left[-\tfrac{1}{2} \sum_{d=1}^{C} \frac{\left(f_{i,d}(x) - \mu_{i,k,d}\right)^2}{\sigma_{i,k,d}^2} \right] \right), \tag{16}$$

where we assume a mean-field approximation to each component and uniform component weights. The parameters $\mu_{i,k,d}$ and $\sigma_{i,k,d}$ are the mean and standard deviation of the d-th logit dimension for the k-th class component in client i's Gaussian mixture model.

B ADDITIONAL RESULTS

B.1 EXPERIMENTAL RESULTS

Table 2: Best test accuracy (%) across datasets and heterogeneity levels (k classes per client). Mean ± std over 3 seeds. Best result per column in **bold**. $Ref.$ is fully informed reference trained on all-classes dataset with size $|\mathcal{D}_{pub}| + |\mathcal{D}_i|$

Dataset	Method	k=2	k=3	k=4	k=5	k=7	k=9	$Ref.$
CIFAR-10	AVG	21.70±2.15	41.88±1.73	56.79±0.61	65.33±2.65	72.55±0.96	**75.42**±0.05	80.84
	UWA	**39.51**±2.69	59.37±1.57	64.71±0.49	68.96±0.40	71.23±0.15	73.16±0.13	
	sUWA	38.33±0.63	**60.96**±1.42	**67.45**±1.17	**71.38**±0.27	**73.94**±0.21	74.98±0.21	
Yahoo	AVG	49.61±3.63	55.78±1.76	56.39±2.45	58.94±1.48	61.17±0.23	61.92±0.40	63.80
	UWA	56.46±1.06	57.90±0.77	58.51±0.16	59.69±0.27	61.20±0.45	61.97±0.47	
	sUWA	**56.66**±1.71	**59.61**±1.91	**58.81**±0.80	**60.03**±0.81	**61.55**±0.27	**62.08**±0.41	
		k=20	k=30	k=40	k=50	k=70	k=90	
CIFAR-100	AVG	52.73±1.08	59.89±0.34	64.02±0.59	**66.79**±0.59	**69.07**±0.40	**69.43**±0.18	69.95
	UWA	59.82±0.33	62.09±0.60	63.43±0.62	64.91±0.79	66.76±0.53	67.67±0.09	
	sUWA	**60.85**±0.31	**63.29**±0.39	**64.81**±0.50	66.20±0.60	67.90±0.39	68.35±0.03	

B.2 HETEROGENEOUS MODEL ARCHITECTURES

Figure 4 empirically shows that it is possible to use different architectures during federated distillation.

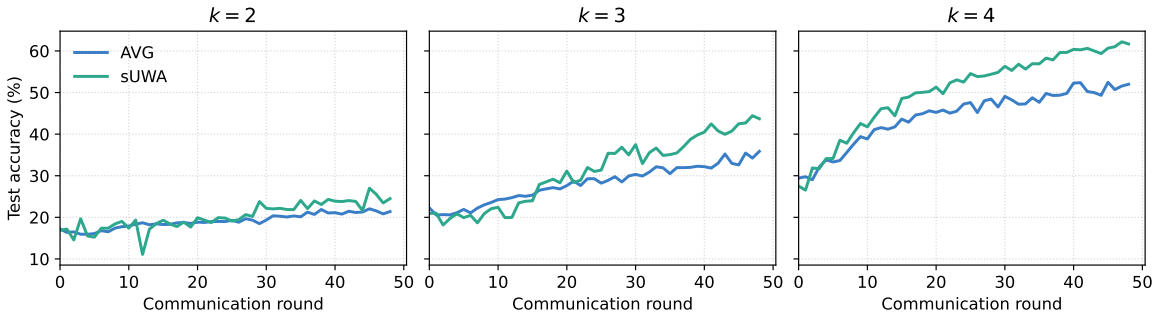

Figure 4: CIFAR-10 with heterogeneous client architectures: test accuracy across communication rounds for AVG and sUWA at $k \in \{2, 3, 4\}$. The 20 clients are split across ResNet-18, a VGG-style CNN, DenseNet-121, and ResNet-34.

B.3 ACCURACY CURVES

Figure 5 shows the global and local test accuracy across communication rounds for all datasets and heterogeneity levels. The top row shows global test accuracy (evaluated on all classes) and the bottom row shows local test accuracy (evaluated only on each client's assigned classes). Lines show the mean over 3 seeds; shaded regions indicate ± 1 standard deviation.

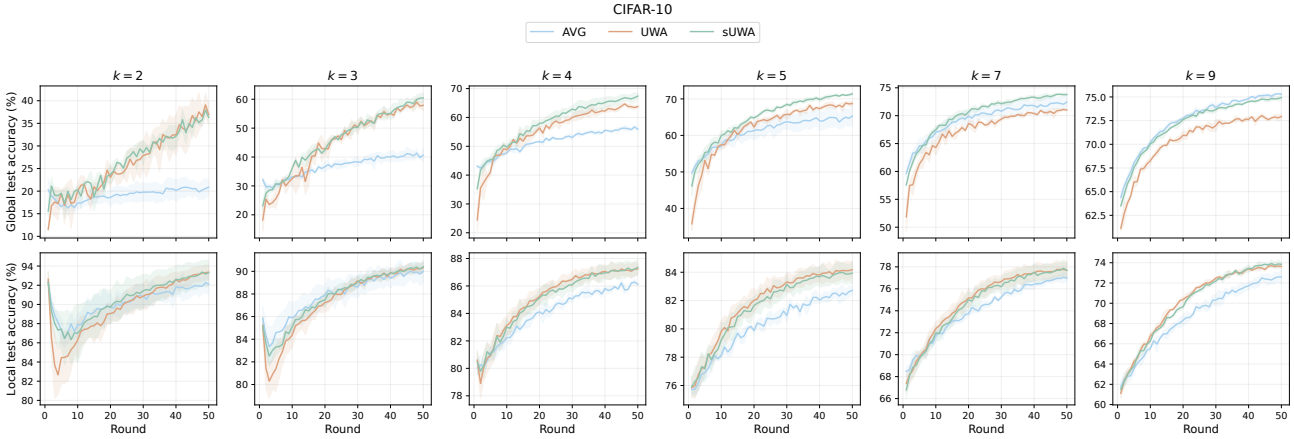

Figure 5: CIFAR-10: global (top) and local (bottom) test accuracy across communication rounds for $k \in \{2, 3, 4, 5, 7, 9\}$.

B.4 FEDERATED LEARNING BASELINES COMPARISON

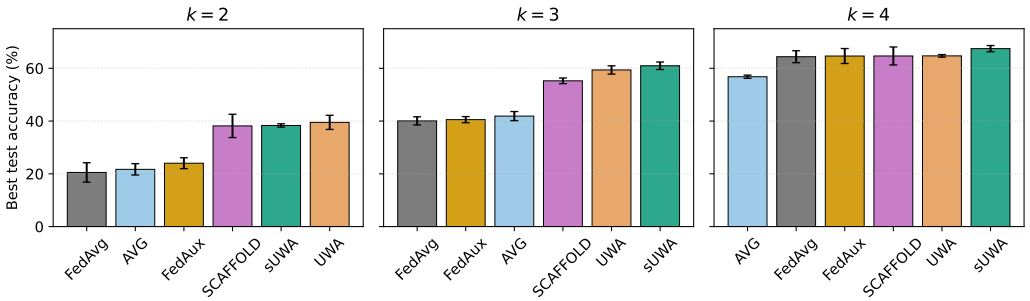

Figure 6: CIFAR-10: best test accuracy (mean over 3 seeds, error bars show ± 1 std) of the aggregation methods (AVG, UWA, sUWA) and gradient-based baselines (`FedAvg`, `SCAFFOLD`, `FedAux`) at three heterogeneity levels. Bars are sorted in increasing order within each panel.

B.5 ABLATION: PUBLIC VS. PRIVATE DATASET SIZE

To understand how the sizes of the public and private datasets affect performance, we run an ablation study on Yahoo Answers using `sUWA` with k=3 and 30 communication rounds. We vary the public dataset size over $\{500, 1{,}000, 2{,}000, 5{,}000\}$ and private samples per client over $\{500, 1{,}000, 1{,}500, 3{,}000\}$.

Figure 7 shows the results as a heatmap. Increasing the public dataset size has the largest impact on accuracy: moving from 500 to 2,000 public samples yields a $\sim$17 percentage-point improvement regardless of private data size. Beyond 2,000 public samples, gains saturate unless private data also increases. Private data size has a smaller but consistent effect, especially when public data is abundant. This suggests that under a fixed data budget, prioritizing the collection of shared public data yields larger returns than expanding per-client private data, since the public set mediates all inter-client knowledge transfer in the distillation framework.

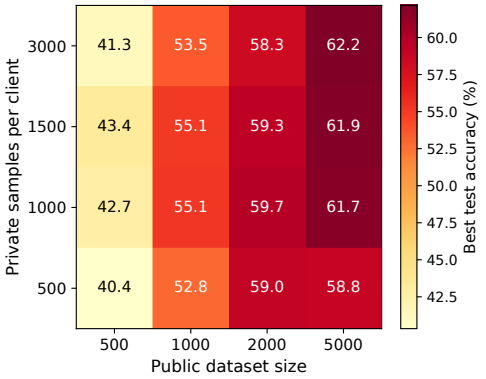

Figure 7: Ablation on Yahoo Answers (`sUWA`, k=3): best test accuracy (%) as a function of public dataset size and private samples per client.

B.6 SENSITIVITY TO THE TEMPERATURE τ

We fix the sUWA temperature to $\tau = 0.25$ in all main experiments. Figure 8 examines sensitivity to this choice on CIFAR-10 (seed 0) for $\tau \in \{0.125, 0.25, 0.5, 0.75\}$. For $k \geq 3$, $\tau = 0.25$ gives the best accuracy, and performance changes gradually on either side, so the method is not sensitive to the exact value within this range. At the most extreme heterogeneity ($k = 2$), a larger temperature ($\tau = 0.75$) is marginally better: when almost all clients are uninformed for a given input, concentrating weight less aggressively helps. We keep $\tau = 0.25$ throughout, as it is the best or near-best choice across the range of k.

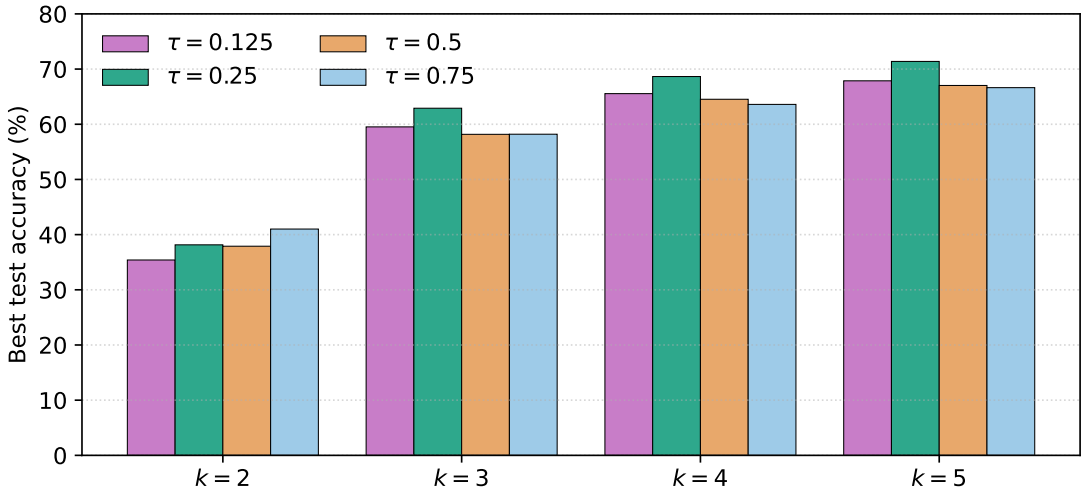

Figure 8: CIFAR-10 (seed 0): best test accuracy of sUWA for temperatures $\tau \in \{0.125, 0.25, 0.5, 0.75\}$ at four heterogeneity levels.

B.7 GMM SEPARATION QUALITY

We evaluate how well each client-specific GMM, fitted on local logits and used for UWA/sUWA weighting, separates inputs from the client's own classes from inputs belonging to other classes. We report the average client AUROC on CIFAR-10 in Table 3.

Table 3: Average client AUROC on CIFAR-10 for GMM-based separation of own-class inputs from other-class inputs.

k	AUROC
2	0.6878
3	0.7418
4	0.7727
5	0.7808

C THEORETICAL ANALYSIS (LEMMAS AND PROOFS)

In this section, we provide the missing details on the theoretical convergence analysis sketched in Section 4.

C.1 NOTATION TABLE

Symbol	Meaning
$\mathcal{S}, M$	participating client set and its size ($\lvert\mathcal{S}\rvert = M$)
C	number of classes
$\theta \in \mathbb{R}^d$	model parameters
$f(x,\theta) \in \Delta^{C-1}$	model predictive probability vector
$F(\theta) = \frac{1}{M}\sum_i F_i(\theta)$	global supervised objective
$F^* = \inf_\theta F(\theta)$	optimal value of F
θ^*	a minimizer of F
$p^*(x) = f(x,\theta^*)$	reference predictor at optimum
$\mathcal{D}_{\mathrm{pub},x}$	public input distribution
$q_i(x) = f(x,\theta_i)$	client predictor after Stage 1
$p_{\mathrm{agg}}(x)$	aggregated teacher distribution
$w_i(x)$	aggregation weights ($w_i \geq 0$, $\sum_i w_i = 1$)
$J_{\mathrm{agg}}(\theta)$	Stage 2 distillation objective w.r.t. p_{agg}
$J_*(\theta)$	ideal KD objective w.r.t. p^*
g_t	stochastic gradient estimator for $\nabla J_{\mathrm{agg}}(\theta_t)$
ν^2	bound on Stage 2 gradient noise variance (Assumption 4.3)
(B, σ, ρ_c)	aggregation bias/variance/correlation constants (Assumption 4.1)
c, b	alignment constants (Assumption 4.5)
G_{sc}^2	score-energy bound (Assumption 4.7)
$\varepsilon_{\mathrm{teach}}$	teacher MSE $\mathbb{E}_x\lVert p_{\mathrm{agg}}(x) - p^*(x)\rVert^2$
$\beta(\theta)$	Stage 2 bias $\nabla J_{\mathrm{agg}}(\theta) - \nabla F(\theta)$
n_t	zero-mean noise $g_t - \nabla J_{\mathrm{agg}}(\theta_t)$
m, ζ^2	biased-oracle constants (Prop. 4.8)

C.2 A HILBERT-SPACE IDENTITY USED IN AGGREGATION MSE PROOFS

Lemma C.1 (Conditional identity)**.** *Let $Z \in \mathbb{R}^k$ be square-integrable and X any random variable. Then*

$$\mathbb{E}\|Z\|_2^2 = \mathbb{E}\|\mathbb{E}[Z \mid X]\|_2^2 + \mathbb{E}\|Z - \mathbb{E}[Z \mid X]\|_2^2. \tag{17}$$

Proof. Write

$$Z = \mathbb{E}[Z \mid X] + \big(Z - \mathbb{E}[Z \mid X]\big).$$

Then

$$\|Z\|^2 = \|\mathbb{E}[Z \mid X]\|^2 + \|Z - \mathbb{E}[Z \mid X]\|^2 + 2\langle \mathbb{E}[Z \mid X], Z - \mathbb{E}[Z \mid X]\rangle.$$

Taking expectation, the cross term vanishes because

$$\mathbb{E}\left[\langle \mathbb{E}[Z \mid X], Z - \mathbb{E}[Z \mid X]\rangle\right] = \mathbb{E}\left[\mathbb{E}\left[\langle \mathbb{E}[Z \mid X], Z - \mathbb{E}[Z \mid X]\rangle \mid X\right]\right] = \mathbb{E}\left[\langle \mathbb{E}[Z \mid X], \mathbb{E}[Z - \mathbb{E}[Z \mid X] \mid X]\rangle\right] = 0.$$

Thus (17) holds. □

C.3 PROOF OF LEMMA 4.2 (TEACHER MSE)

Proof. By definition, $p_{\mathrm{agg}}(x) - p^*(x) = \bar{\xi}(x)$. Fix x and condition on $(x, w(x))$. Using the conditional variance identity,

$$\mathbb{E}\big[\|\bar{\xi}(x)\|_2^2 \mid x, w\big] = \big\|\mathbb{E}[\bar{\xi}(x) \mid x, w]\big\|_2^2 + \mathbb{E}\big[\|\bar{\xi}(x) - \mathbb{E}[\bar{\xi}(x) \mid x, w]\|_2^2 \mid x, w\big].$$

By definition of $\tilde{\xi}_i(x) := \xi_i(x) - \mathbb{E}[\xi_i(x) \mid x, w]$, we have

$$\bar{\xi}(x) - \mathbb{E}[\bar{\xi}(x) \mid x, w] = \sum_{i \in \mathcal{S}} w_i(x)\tilde{\xi}_i(x).$$

Expanding the quadratic form gives

$$\begin{aligned}\mathbb{E}\Big[\Big\|\sum_{i\in\mathcal{S}} w_i\tilde{\xi}_i\Big\|_2^2\Big|x, w\Big] &= \sum_i w_i^2\mathbb{E}[\|\tilde{\xi}_i\|_2^2 \mid x, w] + 2\sum_{i<j} w_i w_j\mathbb{E}[\langle\tilde{\xi}_i, \tilde{\xi}_j\rangle \mid x, w] \\ &\le \sigma^2\sum_i w_i^2 + 2\rho_c\sigma^2\sum_{i<j} w_i w_j,\end{aligned}$$

where we used (1)-(2). Since $\sum_i w_i = 1$, we have $2\sum_{i<j} w_i w_j = 1 - \sum_i w_i^2 = 1 - \chi(x)$, hence

$$\mathbb{E}\Big[\Big\|\sum_i w_i\tilde{\xi}_i\Big\|_2^2\Big|x, w\Big] \le \sigma^2\chi(x) + \rho_c\sigma^2(1 - \chi(x)) = \sigma^2\big(\rho_c + (1 - \rho_c)\chi(x)\big).$$

Therefore,

$$\mathbb{E}\big[\|\bar{\xi}(x)\|_2^2 \mid x, w\big] \le \big\|\mathbb{E}[\bar{\xi}(x) \mid x, w]\big\|_2^2 + \sigma^2\big(\rho_c + (1 - \rho_c)\chi(x)\big).$$

Taking expectation over x (and any randomness in w) yields

$$\varepsilon_{\mathrm{teach}} \le \mathbb{E}_x\big\|\mathbb{E}[\bar{\xi}(x) \mid x, w(x)]\big\|_2^2 + \sigma^2\big(\rho_c + (1 - \rho_c)\mathbb{E}_x\chi(x)\big) = B_w^2 + \sigma^2\big(\rho_c + (1 - \rho_c)\chi\big),$$

which is (3). □

C.4 GRADIENT DISCREPANCY BOUND (AGGREGATION ERROR ⇒ GRADIENT ERROR)

Define the score matrix $S(x, \theta) \in \mathbb{R}^{C\times d}$ with rows $[S(x, \theta)]_{c,:} = \nabla_\theta \log f_c(x, \theta)^\top$.

Lemma C.2 (Aggregation-induced gradient discrepancy)**.** *Let*

$$\Delta_{\mathrm{agg}}(\theta) := \nabla J_{\mathrm{agg}}(\theta) - \nabla J_*(\theta).$$

Under Assumption 4.7 and for Stage 2 iterates θ_t,

$$\mathbb{E}\|\Delta_{\mathrm{agg}}(\theta)\|_2^2 \le G_{\mathrm{sc}}^2 \cdot \mathbb{E}_{x\sim\mathcal{D}_{\mathrm{pub},x}}\|p_{\mathrm{agg}}(x) - p^*(x)\|_2^2 = G_{\mathrm{sc}}^2\varepsilon_{\mathrm{teach}}. \tag{18}$$

Proof. Fix θ. For any $p(x) \in \Delta^{C-1}$,

$$\nabla_\theta \mathrm{KL}(p(x)\|f(x, \theta)) = -\sum_{c=1}^{C} p_c(x)\nabla_\theta \log f_c(x, \theta).$$

Hence

$$\begin{aligned}\Delta_{\mathrm{agg}}(\theta) &= \nabla J_{\mathrm{agg}}(\theta) - \nabla J_*(\theta) \\ &= \mathbb{E}_x\left[\sum_{c=1}^{C}\big({p^*}_c(x) - {p_{\mathrm{agg}}}_c(x)\big)\nabla_\theta \log f_c(x, \theta)\right] \\ &= -\mathbb{E}_x\big[S(x, \theta)^\top\bar{\xi}(x)\big].\end{aligned}$$

Now bound step-by-step:

$$\begin{aligned}\|\Delta_{\mathrm{agg}}(\theta)\| &= \left\|\mathbb{E}_x\left[S(x,\theta)^\top \bar{\xi}(x)\right]\right\| \le \mathbb{E}_x\left\|S(x,\theta)^\top \bar{\xi}(x)\right\| \quad \text{(Jensen's inequality)} \\ &\le \mathbb{E}_x\left[\|S(x,\theta)\|_F \|\bar{\xi}(x)\|\right] \quad \text{(since } \|A^\top v\| \le \|A\|_F\|v\|\text{)} \\ &\le \sqrt{\mathbb{E}_x\|S(x,\theta)\|_F^2}\cdot\sqrt{\mathbb{E}_x\|\bar{\xi}(x)\|^2} \quad \text{(Cauchy-Schwarz).}\end{aligned}$$

Square both sides:

$$\|\Delta_{\mathrm{agg}}(\theta)\|^2 \le \left(\mathbb{E}_x\|S(x,\theta)\|_F^2\right)\left(\mathbb{E}_x\|\bar{\xi}(x)\|^2\right).$$

Take expectation over Stage 2 randomness (if θ is random) and use Assumption 4.7:

$$\mathbb{E}\|\Delta_{\mathrm{agg}}(\theta_t)\|^2 \le G_{\mathrm{sc}}^2 \cdot \mathbb{E}_x\|\bar{\xi}(x)\|^2 = G_{\mathrm{sc}}^2\varepsilon_{\mathrm{teach}}.$$

□

C.5 PROOF OF PROPOSITION 4.8

Proof of Proposition 4.8. We must show the two inequalities in (13)-(14).

Noise. By definition $n_t = g_t - \nabla J_{\mathrm{agg}}(\theta_t)$. Assumption 4.3 yields $\mathbb{E}[\|n_t\|^2 \mid \theta_t] \le \nu^2$, proving (14).

Bias. Write

$$\beta(\theta_t) = \nabla J_{\mathrm{agg}}(\theta_t) - \nabla F(\theta_t) = \underbrace{\left(\nabla J_{\mathrm{agg}}(\theta_t) - \nabla J_*(\theta_t)\right)}_{\Delta_{\mathrm{agg}}(\theta_t)} + \underbrace{\left(\nabla J_*(\theta_t) - \nabla F(\theta_t)\right)}_{e(\theta_t)}.$$

Use the inequality $\|u+v\|^2 \le (1+\alpha)\|u\|^2 + (1+1/\alpha)\|v\|^2$:

$$\|\beta(\theta_t)\|^2 \le (1+\alpha)\|\Delta_{\mathrm{agg}}(\theta_t)\|^2 + \left(1+\frac{1}{\alpha}\right)\|e(\theta_t)\|^2. \tag{19}$$

Take expectation.

Step 1: bound $\mathbb{E}\|\Delta_{\mathrm{agg}}(\theta_t)\|^2$. By Lemma C.2,

$$\mathbb{E}\|\Delta_{\mathrm{agg}}(\theta_t)\|^2 \le G_{\mathrm{sc}}^2\varepsilon_{\mathrm{teach}}.$$

Step 2: bound $\mathbb{E}\|e(\theta_t)\|^2$ *using alignment.* Take expectation and apply Assumption 4.5:

$$\mathbb{E}\|e(\theta_t)\|^2 = \mathbb{E}\left[\|\nabla J_*(\theta_t) - \nabla F(\theta_t)\|^2\right] \le c\mathbb{E}\|\nabla F(\theta_t)\|^2 + b^2.$$

Step 3: combine. Insert the bounds into (19) and group terms:

$$\mathbb{E}\|\beta(\theta_t)\|^2 \le \left(1+\frac{1}{\alpha}\right)c\mathbb{E}\|\nabla F(\theta_t)\|^2 + (1+\alpha)G_{\mathrm{sc}}^2\varepsilon_{\mathrm{teach}} + \left(1+\frac{1}{\alpha}\right)b^2.$$

This matches (13) with m, ζ^2 as in (12). □

C.6 CONVERGENCE RATES

Theorem C.3 (Nonconvex stationarity under an *expected* biased oracle)**.** *Assume F is L_F-smooth and Stage 2 updates satisfy $g_t = \nabla F(\theta_t) + \beta(\theta_t) + n_t$ with $\mathbb{E}[n_t \mid \theta_t] = 0$ and $\mathbb{E}[\|n_t\|^2 \mid \theta_t] \le \nu^2$. Assume there exist $m \in [0,1)$ and $\zeta \ge 0$ such that for all t,*

$$\mathbb{E}\|\beta(\theta_t)\|^2 \le m\mathbb{E}\|\nabla F(\theta_t)\|^2 + \zeta^2.$$

If $\gamma \le 1/L_F$ and θ_{out} is chosen uniformly at random from $\{\theta_0, \dots, \theta_{T-1}\}$, then

$$\mathbb{E}\|\nabla F(\theta_{\mathrm{out}})\|^2 \le \frac{2(F(\theta_0) - F_{\mathrm{inf}})}{T\gamma(1-m)} + \frac{L_F\gamma\nu^2}{1-m} + \frac{\zeta^2}{1-m}, \tag{20}$$

where $F_{\mathrm{inf}} := \inf_\theta F(\theta)$.

Proof. The proof follows the descent template of Ajalloeian and Stich [2021] (with expectations moved outside). By L_F-smoothness,

$$F(\theta_{t+1}) \le F(\theta_t) - \gamma\langle \nabla F(\theta_t), g_t\rangle + \frac{L_F\gamma^2}{2}\|g_t\|^2.$$

Condition on θ_t. Since $\mathbb{E}[n_t \mid \theta_t] = 0$,

$$\mathbb{E}[\langle \nabla F(\theta_t), g_t\rangle \mid \theta_t] = \langle \nabla F(\theta_t), \nabla F(\theta_t) + \beta(\theta_t)\rangle.$$

Also expanding the square and using $\mathbb{E}[n_t \mid \theta_t] = 0$ gives

$$\mathbb{E}[\|g_t\|^2 \mid \theta_t] = \|\nabla F(\theta_t) + \beta(\theta_t)\|^2 + \mathbb{E}[\|n_t\|^2 \mid \theta_t].$$

Using $\gamma \le 1/L_F$ and combining the inner-product and squared-norm terms yields

$$\mathbb{E}[F(\theta_{t+1})] \le \mathbb{E}[F(\theta_t)] - \frac{\gamma}{2}\mathbb{E}\|\nabla F(\theta_t)\|^2 + \frac{\gamma}{2}\mathbb{E}\|\beta(\theta_t)\|^2 + \frac{L_F\gamma^2}{2}\nu^2.$$

Plug $\mathbb{E}\|\beta(\theta_t)\|^2 \le m\mathbb{E}\|\nabla F(\theta_t)\|^2 + \zeta^2$ and rearrange:

$$\frac{\gamma}{2}(1-m)\mathbb{E}\|\nabla F(\theta_t)\|^2 \le \mathbb{E}[F(\theta_t)] - \mathbb{E}[F(\theta_{t+1})] + \frac{\gamma}{2}\zeta^2 + \frac{L_F\gamma^2}{2}\nu^2.$$

Sum over $t = 0, \dots, T-1$ (telescoping) and use $\mathbb{E}[F(\theta_T)] \ge F_{\inf}$. Dividing by T and using the random uniformly distributed output iterate identity yields (20). □

Theorem C.4 (PL linear rate under an *expected* biased oracle)**.** *Assume the conditions of Theorem C.3 and additionally PL:* $\|\nabla F(\theta)\|^2 \ge 2\mu_F(F(\theta) - F^*)$. *Then for any* $\gamma \le 1/L_F$,

$$\mathbb{E}[F(\theta_T) - F^*] \le \big(1 - \gamma\mu_F(1-m)\big)^T (F(\theta_0) - F^*) + \frac{\zeta^2 + L_F\gamma\nu^2}{2\mu_F(1-m)}. \tag{21}$$

Proof. From the one-step inequality inside the proof of Theorem C.3,

$$\mathbb{E}[F(\theta_{t+1}) - F^*] \le \mathbb{E}[F(\theta_t) - F^*] - \frac{\gamma}{2}(1-m)\mathbb{E}\|\nabla F(\theta_t)\|^2 + \frac{\gamma}{2}\zeta^2 + \frac{L_F\gamma^2}{2}\nu^2.$$

Apply PL: $\mathbb{E}\|\nabla F(\theta_t)\|^2 \ge 2\mu_F\mathbb{E}[F(\theta_t) - F^*]$, yielding a linear recursion. Unrolling it gives (21). □

C.7 ITERATION COMPLEXITIES

Proof of Corollary 4.9. Start from the exact nonconvex stationarity bound (20):

$$\mathbb{E}\|\nabla F(\theta_{\text{out}})\|^2 \le \frac{2(F(\theta_0) - F_{\inf})}{T\gamma(1-m)} + \frac{L_F\gamma\nu^2}{1-m} + \frac{\zeta^2}{1-m}.$$

By definition, $\epsilon_{\min} = \zeta^2/(1-m)$, so it suffices to enforce

$$\frac{2(F(\theta_0) - F_{\inf})}{T\gamma(1-m)} + \frac{L_F\gamma\nu^2}{1-m} \le \bar{\epsilon}. \tag{22}$$

We choose γ so that the variance term is at most $\bar{\epsilon}/2$:

$$\frac{L_F\gamma\nu^2}{1-m} \le \frac{\bar{\epsilon}}{2} \quad\Longleftrightarrow\quad \gamma \le \frac{\bar{\epsilon}(1-m)}{2L_F\nu^2}.$$

Together with the stepsize restriction $\gamma \le 1/L_F$ from C.3, this yields the choice

$$\gamma := \min\left\{\frac{1}{L_F},\ \frac{\bar{\epsilon}(1-m)}{2L_F\nu^2}\right\}.$$

Under this choice, (22) is implied by requiring the remaining term to be at most $\bar{\epsilon}/2$:

$$\frac{2(F(\theta_0) - F_{\inf})}{T\gamma(1-m)} \leq \frac{\bar{\epsilon}}{2} \quad \Longleftrightarrow \quad T \geq \frac{4(F(\theta_0) - F_{\inf})}{\gamma(1-m)\bar{\epsilon}}.$$

Now substitute the concrete upper bound $\gamma \leq \frac{\bar{\epsilon}(1-m)}{2L_F\nu^2}$ (which holds whenever that branch is active):

$$T \geq \frac{4(F(\theta_0) - F_{\inf})}{(1-m)\bar{\epsilon}} \cdot \frac{2L_F\nu^2}{\bar{\epsilon}(1-m)} = \frac{8L_F\nu^2(F(\theta_0) - F_{\inf})}{(1-m)^2\bar{\epsilon}^2}.$$

Thus it suffices to take

$$T = \mathcal{O}\left(\frac{L_F\nu^2(F(\theta_0) - F_{\inf})}{(1-m)^2\bar{\epsilon}^2}\right),$$

which proves the claim. □

Proof of Corollary 4.10. Start from the PL bound (21):

$$\mathbb{E}[F(\theta_T) - F^*] \leq \big(1 - \gamma\mu_F(1-m)\big)^T (F(\theta_0) - F^*) + \epsilon_{\text{floor}}(\gamma).$$

Fix a target $\epsilon > \epsilon_{\text{floor}}(\gamma)$ and define $\bar{\epsilon} := \epsilon - \epsilon_{\text{floor}}(\gamma) > 0$. It suffices to ensure

$$\big(1 - \gamma\mu_F(1-m)\big)^T (F(\theta_0) - F^*) \leq \bar{\epsilon}. \tag{23}$$

Since $\gamma \leq 1/L_F$ and $\mu_F(1-m) > 0$, we have $\gamma\mu_F(1-m) \in (0,1)$ for any standard stepsize choice. Using the inequality $1 - x \leq e^{-x}$ for $x \in (0,1)$ gives

$$\big(1 - \gamma\mu_F(1-m)\big)^T \leq \exp\big(-\gamma\mu_F(1-m)T\big).$$

Therefore, (23) is implied by

$$\exp\big(-\gamma\mu_F(1-m)T\big)(F(\theta_0) - F^*) \leq \bar{\epsilon},$$

which is equivalent (taking logarithms) to

$$T \geq \frac{1}{\gamma\mu_F(1-m)} \log\left(\frac{F(\theta_0) - F^*}{\bar{\epsilon}}\right) = \frac{1}{\gamma\mu_F(1-m)} \log\left(\frac{F(\theta_0) - F^*}{\epsilon - \epsilon_{\text{floor}}(\gamma)}\right).$$

This yields the stated Big-$\mathcal{O}$ complexity. □