Constant Term Shrinkage for Federated Learning

Incheol Baek
Computer Science and Engineering
Korea University
Seoul 02841, Republic of Korea
inch307@korea.ac.kr

Minseo Kim
Computer Science and Engineering
Korea University
Seoul 02841, Republic of Korea
minss126@korea.ac.kr

Hyeonmin Kang
Computer Science and Engineering
Korea University
Seoul 02841, Republic of Korea
bnca0815@korea.ac.kr

Yon Dohn Chung*
Computer Science and Engineering
Korea University
Seoul 02841, Republic of Korea
ydchung@korea.ac.kr

Abstract

Federated Learning (FL) enables collaborative model training without centralizing private data, but its performance often degrades under data heterogeneity because local updates become biased toward client-specific data characteristics. In this paper, we propose **Constant Term Shrinkage for Federated Learning (FedCTS)**, a simple server-side aggregation method designed to mitigate such update drift. FedCTS decomposes each client update into a constant term and a residual term. For each convolutional filter or fully connected output unit, the constant term is formed by repeating the mean of the update values within that unit, while the residual term contains the remaining variation. Using this decomposition, we show that the constant term becomes more prominent and exhibits larger client drift as data heterogeneity increases. FedCTS therefore estimates the cross-client reliability of the constant term and shrinks it more strongly toward zero when the constant coefficients show greater disagreement across clients. The shrinkage strength is obtained in closed form from clients' update statistics, without requiring an additional tuning hyperparameter. Experiments across multiple datasets and FL algorithms demonstrate that FedCTS improves robustness under heterogeneous data with negligible additional computation, no extra communication overhead, and strong compatibility with existing FL techniques.

1 Introduction

Training data are increasingly distributed across devices and institutions that cannot, or will not, centralize them [8, 12]. Federated Learning (FL) takes a different approach of conventional training: the model is sent to the clients that hold the data, trained locally on private data, and only the resulting model updates are returned to the server. This paradigm is widely studied in settings where privacy [8, 2], computation [15, 16], or data ownership [21, 24] makes centralized training difficult [15].

A central challenge in FL is data heterogeneity [8]. FL usually trains a single global model for the aggregate data distribution across clients, but each client updates the model using only its own local distribution. Under heterogeneous data, such as label skew or feature shift, local updates can become biased toward client-specific distributions and point in conflicting directions. As a result, the averaged update may deviate from a clean global descent direction.

*Corresponding author.

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

Existing methods mitigate data heterogeneity by modifying local training [11, 13, 14, 20, 28] or server aggregation [6, 7, 22]. FedCTS keeps client training unchanged and takes a server-side view, examining the structure of each received client update before aggregation. Intuitively, when a client trains on locally skewed data, its update can be repeatedly pushed toward a client-specific direction. If this directional effect becomes pronounced, it may be reflected in the statistics of the updates observed at the server. Motivated by this intuition, we decompose each client update into two terms: a constant term and a residual term. We decompose each client update at the level of individual convolutional filters or fully connected output units. For each unit, we compute the mean of its update values. Repeating this mean over the same unit gives the constant term, while subtracting it leaves the residual term. This mean is computed within each client update, not across clients. The constant term captures the average direction of an update vector, while the residual term contains the remaining variation. Through this decomposition, we show that the constant term is strongly affected by data heterogeneity and can become unreliable under stronger heterogeneity.

The constant term, however, should not be removed unconditionally. Although it can contain client drift under heterogeneous data, it can also represent a useful part of the current update. Thus, the goal is not to discard the constant term, but to control its influence adaptively. Based on this idea, we propose **Constant Term Shrinkage for Federated Learning (FedCTS)**. FedCTS analyzes layer-wise statistics of the constant terms extracted from the received client updates, computes a layer-wise shrinkage factor, and applies it to the constant term of each client update in that layer. When the constant terms are consistent across clients, the shrinkage remains small; when they show large disagreement, FedCTS shrinks them more strongly toward zero. The residual term is left unchanged, and the server can aggregate the corrected updates.

Implemented entirely on the server after receiving client updates, FedCTS introduces no extra communication and does not modify client-side training. The additional computation is negligible, and the closed-form shrinkage coefficient requires no tuning hyperparameter. The correction can also be combined with existing FL methods that address other aspects of client drift. We validate this design through analysis and extensive experiments. Our contributions are summarized as follows:

- We decompose each client update into a constant term and a residual term for each convolutional filter or fully connected output unit, and show that the constant term is particularly affected by data heterogeneity.
- We propose FedCTS, a server-side aggregation method that adaptively shrinks the constant term using a closed-form shrinkage coefficient, without requiring an additional tuning hyperparameter.
- We empirically demonstrate that FedCTS improves FL performance under heterogeneous data with negligible computation, no extra communication or client-side overhead, and strong compatibility with existing FL methods.

2 Related Work

Existing approaches to mitigating client drift can be grouped by where the correction is applied. Client-side methods modify local training objectives, representations, or classifier outputs to reduce the bias induced by heterogeneous data. FedProx [13] constrains local optimization with a proximal term, while representation-level methods such as MOON [11] and FedRCL [20] encourage better alignment of client representations under non-IID data. Logit calibration methods such as FedRS [14] and FedLC [28] further address label distribution skew by modifying the classifier behavior during local training.

Some methods coordinate server-side information with client-side training. FedDyn [1] dynamically regularizes local objectives to better align local and global stationary points. FedACG [9] uses a server-provided lookahead gradient to guide local optimization and align client updates with the projected global update direction. These methods are not purely server-side aggregation rules, since their corrections also affect the local training process.

Server-side methods instead modify the global update after receiving client models. FedAvgM [6] and FedGM [22] incorporate server momentum to stabilize the global optimization trajectory, while FedExP [7] adaptively determines the server step size by treating aggregated client updates as pseudo-gradients and applying extrapolation.

FedCTS is also implemented on the server, but it targets a different structure of the client update. Rather than regularizing the entire local training process or uniformly modifying the global update, FedCTS decomposes each client update into a constant component and a zero-mean residual component, and adaptively shrinks only the constant component.

Gradient Centralization. The most closely related technique to FedCTS is Gradient Centralization (GC) [27], which projects gradients onto a zero-mean hyperplane. GC-Fed [19] extends this idea to federated learning by applying Local GC during client-side optimization and Global GC during server-side aggregation. Since Local GC removes the constant term at every local optimization step, GC-Fed can be viewed as a more aggressive suppression baseline than ours.

Although both GC-Fed and FedCTS reduce the constant term, their motivations and mechanisms differ. GC-Fed is motivated by the optimization benefit of constraining gradients to a zero-mean hyperplane. FedCTS instead starts from a structural analysis of federated client updates, identifying the constant component as a drift-prone component under heterogeneity. Accordingly, FedCTS leaves ordinary local training unchanged, estimates cross-client statistics, and adaptively shrinks the constant component at the server. Thus, their overlap is limited to reducing the constant component, while their motivation and correction rule are different.

3 Proposed Method

We first introduce the problem setup and notation. Then, we decompose each client update received by the server into a constant term and a residual term. Based on this decomposition, we show that the constant term becomes more prominent under stronger label skew and that disagreement in this term can account for a large part of client drift. Finally, we propose an adaptive shrinkage rule that mitigates the drift of the constant term and provide a theoretical analysis of FedCTS.

3.1 Problem Setup and Notation

We consider a standard FL setting with a global model w^t at communication round t. Let $\mathcal{S}_t$ be the set of participating clients at round t. Each client $i \in \mathcal{S}_t$ initializes its local model from w^t, performs ordinary local training, and returns the updated model w_i^t to the server. We denote the client update by

$$g_i^t = w_i^t - w^t. \tag{1}$$

The server aggregates these updates as

$$w^{t+1} = w^t + \sum_{i \in \mathcal{S}_t} p_i g_i^t, \tag{2}$$

where $p_i = n_i / \sum_{j \in S_t} n_j$ in all reported experiments, and n_i is the local data size of client i.

FedCTS applies a correction to selected weight tensors. For a convolutional layer, each output channel corresponds to one filter. For a fully connected layer, each output unit corresponds to one row of the weight matrix. We use ℓ to index a target layer. For layer ℓ, we reshape the update tensor of client i into a matrix

$$G_{i,\ell}^t = \begin{bmatrix} (g_{i,\ell,1}^t)^\top \\ \vdots \\ (g_{i,\ell,q_\ell}^t)^\top \end{bmatrix} \in \mathbb{R}^{q_\ell \times d_\ell}, \tag{3}$$

where q_ℓ is the number of filters or output rows, and d_ℓ is the number of parameters in each filter or row. For a convolutional layer, $q_\ell = C_{\text{out}}$ and $d_\ell = C_{\text{in}} k_h k_w$. For a fully connected layer, $q_\ell = C_{\text{out}}$ and $d_\ell = C_{\text{in}}$. The vector $g_{i,\ell,j}^t \in \mathbb{R}^{d_\ell}$ denotes the j-th row of $G_{i,\ell}^t$.

Heterogeneous data partition. We simulate data heterogeneity using a Dirichlet partition. For the full client set $\mathcal{N} = \{1, \dots, N\}$ and each class $c \in \{1, \dots, C\}$, we sample

$$\boldsymbol{\pi}_c = (\pi_{1,c}, \dots, \pi_{N,c}) \sim \text{Dir}(\gamma \mathbf{1}_N), \tag{4}$$

and assign approximately a $\pi_{i,c}$ fraction of class-c samples to client i. The local dataset of each client is obtained by combining the samples assigned from all classes. The concentration parameter γ controls the heterogeneity level: smaller γ yields stronger label skew across clients.

3.2 Constant–Residual View of Client Updates

To decompose each client update into constant and residual terms, let $\mathbf{1}_{d_\ell} \in \mathbb{R}^{d_\ell}$ be the all-one vector. In this reshaped matrix, each row corresponds to either a flattened convolutional filter or a row of a fully connected weight matrix. Thus, the row-wise mean is well-defined for both convolutional and fully connected layers.

For each update matrix $G_{i,\ell}^t$, FedCTS computes the row-wise mean vector

$$\mu_{i,\ell}^t = \frac{1}{d_\ell} G_{i,\ell}^t \mathbf{1}_{d_\ell} \in \mathbb{R}^{q_\ell}. \tag{5}$$

We refer to $\mu_{i,\ell}^t$ as the constant coefficient vector, because it parameterizes the row-wise constant term $\mu_{i,\ell}^t \mathbf{1}_{d_\ell}^\top$. The j-th entry of $\mu_{i,\ell}^t$ is the mean of the j-th row of $G_{i,\ell}^t$. Using this vector, we decompose the client update as

$$G_{i,\ell}^t = \underbrace{\mu_{i,\ell}^t \mathbf{1}_{d_\ell}^\top}_{\text{constant term}} + \underbrace{R_{i,\ell}^t}_{\text{residual term}} . \tag{6}$$

The first term is constant within each row and the residual term $R_{i,\ell}^t$ contains the remaining variation after the row mean is removed.

Update magnitude. Subtracting the row mean gives a zero-mean residual orthogonal to the row-wise constant term (see Appendix B.4):

$$R_{i,\ell}^t \mathbf{1}_{d_\ell} = 0, \qquad \left\langle \mu_{i,\ell}^t \mathbf{1}_{d_\ell}^\top, R_{i,\ell}^t \right\rangle_F = 0. \tag{7}$$

Therefore, the update magnitude can be represented as:

$$\left\| G_{i,\ell}^t \right\|_F^2 = \| \mu_{i,\ell}^t \mathbf{1}_{d_\ell}^\top + R_{i,\ell}^t \|_F^2 = d_\ell \left\| \mu_{i,\ell}^t \right\|_2^2 + \left\| R_{i,\ell}^t \right\|_F^2 . \tag{8}$$

Multiplying by p_i and summing over participating clients gives the layer-wise average squared update norm:

$$U_\ell^t = \sum_{i \in \mathcal{S}_t} p_i \left\| G_{i,\ell}^t \right\|_F^2 = \underbrace{\sum_{i \in \mathcal{S}_t} p_i d_\ell \left\| \mu_{i,\ell}^t \right\|_2^2}_{U_{\ell,\text{con}}^t} + \underbrace{\sum_{i \in \mathcal{S}_t} p_i \left\| R_{i,\ell}^t \right\|_F^2}_{U_{\ell,\text{res}}^t} . \tag{9}$$

We then define the constant update ratio and the residual update ratio as

$$\rho_{\ell,\text{con}}^t = \frac{U_{\ell,\text{con}}^t}{U_\ell^t}, \qquad \rho_{\ell,\text{res}}^t = 1 - \rho_{\ell,\text{con}}^t. \tag{10}$$

These ratios are useful for identifying where the change induced by stronger heterogeneity is relatively concentrated. If the change induced by stronger non-IID data is more concentrated in the constant term, $\rho_{\ell,\text{con}}^t$ increases. If it is more concentrated in the residual term, $\rho_{\ell,\text{res}}^t$ increases. Lemma B.1 in Appendix B.1 provides the algebraic condition.

Fig. 1 reports the constant update ratio under IID data and Dirichlet partitioning with $\gamma \in \{0.1, 0.3, 0.5\}$. Relative to IID, the constant term generally accounts for a larger share of the squared update magnitude under stronger heterogeneity. Additional results across datasets and layers are provided in Appendices A.1 and A.2, respectively.

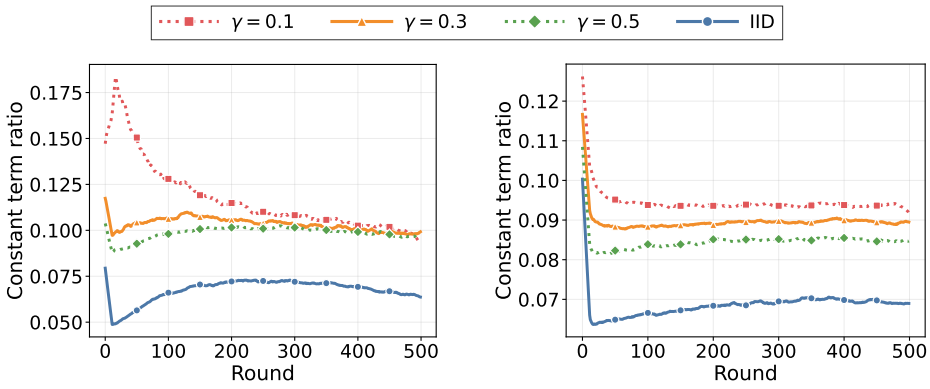
Client drift. We next examine the disagreement of the collected client updates. The weighted averages used in aggregation are defined as:

$$\bar{G}_\ell^t = \sum_{i \in \mathcal{S}_t} p_i G_{i,\ell}^t, \qquad \bar{\mu}_\ell^t = \sum_{i \in \mathcal{S}_t} p_i \mu_{i,\ell}^t, \qquad \bar{R}_\ell^t = \sum_{i \in \mathcal{S}_t} p_i R_{i,\ell}^t. \tag{11}$$

Here, $\bar{G}_\ell^t$, $\bar{\mu}_\ell^t$, and $\bar{R}_\ell^t$ denote the weighted averages of the update matrix, the constant coefficient vector, and the residual term.

Since $(R_{i,\ell}^t - \bar{R}_\ell^t)\mathbf{1}_{d_\ell} = 0$, the centered components remain orthogonal, giving the layer-wise client drift:

$$D_\ell^t = \sum_{i \in \mathcal{S}_t} p_i \left\| G_{i,\ell}^t - \bar{G}_\ell^t \right\|_F^2 = \underbrace{\sum_{i \in \mathcal{S}_t} p_i d_\ell \left\| \mu_{i,\ell}^t - \bar{\mu}_\ell^t \right\|_2^2}_{D_{\ell,\text{con}}^t} + \underbrace{\sum_{i \in \mathcal{S}_t} p_i \left\| R_{i,\ell}^t - \bar{R}_\ell^t \right\|_F^2}_{D_{\ell,\text{res}}^t} \tag{12}$$



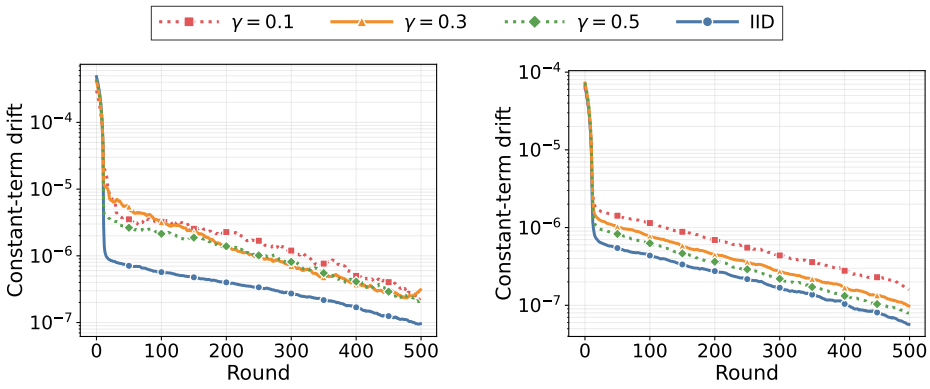
(a) CIFAR-10 (b) CIFAR-100

Figure 1: Constant update ratio $\rho^t_{\ell,\text{con}}$ under IID and Dirichlet partitions. A smaller γ indicates stronger data heterogeneity. The ratio increases as γ decreases, indicating that the constant term accounts for a larger portion of the update magnitude under stronger heterogeneity. The statistics are reported for the average over all layers.

Fig. 2 shows that constant-term client drift increases under stronger heterogeneity. A larger update ratio alone does not imply larger drift. Although the residual update ratio is larger in Fig. 1, magnitude alone does not determine the benefit of correction. Our motivation is based on changes relative to IID: the constant term accounts for a larger update share and exhibits greater client drift. Constant-term drift on additional datasets is reported in Appendix A.3. To further examine how client drift is distributed between the two terms, we measure the constant drift ratio on CIFAR-10 and CIFAR-100:

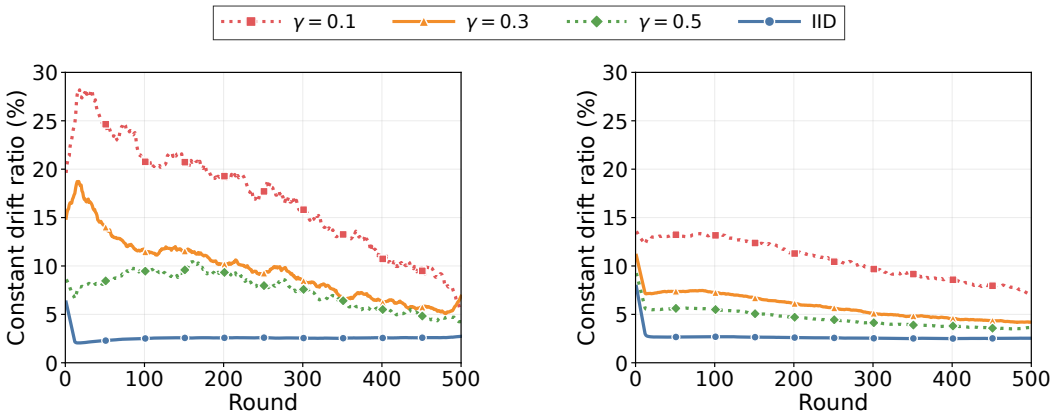
$$\rho^{t,\text{drift}}_{\text{con}} = \frac{\sum_\ell D^t_{\ell,\text{con}}}{\sum_\ell \left(D^t_{\ell,\text{con}} + D^t_{\ell,\text{res}} \right)}.$$

The residual drift ratio is $1 - \rho^{t,\text{drift}}_{\text{con}}$. As shown in Fig. 3, the constant term generally accounts for a larger fraction of client drift under non-IID partitions than under IID partitioning on both datasets. Although the residual term accounts for most of the total drift, the constant term exhibits greater client drift and generally accounts for a larger share of the total drift under non-IID partitioning than under IID partitioning. These observations motivate correcting the constant term separately to mitigate the effects of data heterogeneity.



(a) CIFAR-10 (b) CIFAR-100

Figure 2: Constant-term local drift $D^t_{\ell,\text{con}}$ under IID and Dirichlet partitions. A smaller γ indicates stronger data heterogeneity. The drift increases as γ decreases, indicating that stronger heterogeneity amplifies cross-client disagreement in the constant direction. The statistics are reported for the average over all layers.



(a) CIFAR-10 (b) CIFAR-100

Figure 3: Constant drift ratio under IID and Dirichlet partitions.

3.3 FedCTS: Adaptive Constant Term Shrinkage

Now, we propose FedCTS which decides how much of the constant term should be preserved. The key idea is to estimate how reliable the constant direction is. For each layer ℓ, FedCTS first computes the mean of the constant coefficient vectors:

$$\hat{\mu}_\ell^t = \frac{1}{|\mathcal{S}_t|} \sum_{i \in \mathcal{S}_t} \mu_{i,\ell}^t. \tag{13}$$

Note that we do not use a weighted average for $\hat{\mu}_\ell^t$. The reason is that this step estimates the reliability of the directions of the constant coefficient vectors, not their aggregation. The number of local samples does not necessarily indicate whether a client update is reliable: a client with many samples can still be highly biased under label skew, while a smaller client may have a more balanced local distribution.

To measure the reliability, FedCTS computes two layer-wise statistics:

$$M_\ell^t = \frac{1}{q_\ell} \left\| \hat{\mu}_\ell^t \right\|_2^2, \qquad V_\ell^t = \frac{1}{|\mathcal{S}_t| q_\ell} \sum_{i \in \mathcal{S}_t} \left\| \mu_{i,\ell}^t - \hat{\mu}_\ell^t \right\|_2^2. \tag{14}$$

Here, M_ℓ^t measures the average squared magnitude of the mean constant coefficient vector across the q_ℓ filters or output units, not agreement among clients. A small M_ℓ^t can reflect weak constant coefficients or cancellation in the mean. The statistic V_ℓ^t separately measures cross-client disagreement around $\hat{\mu}_\ell^t$, which is not assumed to be the true global direction.

FedCTS chooses a scalar shrinkage coefficient α to determine how much of the constant coefficient should be removed, resulting in $(1-\alpha)\mu_{i,\ell}^t$. After shrinkage, the constant coefficient can be represented with consensus and deviation:

$$(1-\alpha)\mu_{i,\ell}^t = (1-\alpha)\hat{\mu}_\ell^t + (1-\alpha)\left(\mu_{i,\ell}^t - \hat{\mu}_\ell^t\right). \tag{15}$$

Thus, increasing α suppresses the client-specific deviation, but it also attenuates the observed consensus. Our objective is to choose α by balancing these two effects:

$$\begin{aligned} \min_{\alpha \in [0,1]} \quad & \underbrace{\frac{1}{q_\ell} \left\| \hat{\mu}_\ell^t - (1-\alpha)\hat{\mu}_\ell^t \right\|_2^2}_{\text{consensus attenuation}} + \underbrace{\frac{1}{|\mathcal{S}_t| q_\ell} \sum_{i \in \mathcal{S}_t} \left\| (1-\alpha)\left(\mu_{i,\ell}^t - \hat{\mu}_\ell^t\right) \right\|_2^2}_{\text{remaining disagreement}} \\ = \min_{\alpha \in [0,1]} \quad & \alpha^2 M_\ell^t + (1-\alpha)^2 V_\ell^t. \end{aligned} \tag{16}$$

The two terms in Eq. (16) penalize mean attenuation and remaining disagreement, respectively. For $M_\ell^t + V_\ell^t > 0$, setting the derivative of this convex quadratic to zero gives

$$\alpha = \frac{V_\ell^t}{M_\ell^t + V_\ell^t}. \tag{17}$$

The solution lies in $[0, 1]$; when $M_\ell^t + V_\ell^t = 0$, all constant coefficients vanish and no correction is needed. If the clients agree on the constant coefficient vectors, then V_ℓ^t is small relative to M_ℓ^t, so α is close to zero and FedCTS preserves most of the constant term. If the clients strongly disagree, then V_ℓ^t is large relative to M_ℓ^t, so α becomes large and FedCTS suppresses the unreliable constant term.

The corrected update matrix is

$$\widetilde{G}_{i,\ell}^t = G_{i,\ell}^t - \alpha \mu_{i,\ell}^t \mathbf{1}_{d_\ell}^\top = (1-\alpha)\mu_{i,\ell}^t \mathbf{1}_{d_\ell}^\top + R_{i,\ell}^t. \tag{18}$$

Thus, FedCTS shrinks only the constant term of each target layer and leaves the residual component unchanged. We denote by $\widetilde{g}_i^t$ the full corrected client update obtained by reshaping each $\widetilde{G}_{i,\ell}^t$ back to its original tensor shape. The final aggregation is then performed using the corrected updates $\{\widetilde{g}_i^t\}_{i \in \mathcal{S}_t}$. Algorithm 1 summarizes the server-side correction and aggregation steps of FedCTS. Theoretical results on client drift reduction and the properties of α are provided in Appendix B.

Algorithm 1 FedCTS

Require: Global model w^t, clients $\mathcal{S}_t$, target layers $\mathcal{L}$
Ensure: Updated global model w^{t+1}
1: Each client $i \in \mathcal{S}_t$ trains from w^t and returns w_i^t
2: Compute client updates $g_i^t = w_i^t - w^t$
3: **for** each target layer $\ell \in \mathcal{L}$ **do**
4: Compute $\mu_{i,\ell}^t = d_\ell^{-1} G_{i,\ell}^t \mathbf{1}_{d_\ell}$ for all i
5: Compute $\hat{\mu}_\ell^t$, M_ℓ^t, and V_ℓ^t
6: Set $\alpha = V_\ell^t / (V_\ell^t + M_\ell^t)$
7: Correct $\widetilde{G}_{i,\ell}^t = G_{i,\ell}^t - \alpha \mu_{i,\ell}^t \mathbf{1}_{d_\ell}^\top$
8: **end for**
9: Aggregate $w^{t+1} = w^t + \sum_{i \in \mathcal{S}_t} p_i \widetilde{g}_i^t$

Overhead. FedCTS is implemented entirely on the server after receiving the usual client updates. Therefore, it requires no client-side modification and adds no communication overhead. The additional computation is negligible, consisting only of row-wise means, layer-wise squared norms, and constant-term subtraction for the layers.

4 Experiments

We empirically evaluate FedCTS through accuracy comparisons across diverse FL algorithms and datasets, and ablation studies on the adaptive shrinkage rule.

Experimental setup. We simulate a cross-device FL setting with 100 clients and randomly sample 10% of clients at each communication round. All methods use ResNet-18 [5] as the backbone model. We evaluate on ten image classification datasets: BloodMNIST, OrganAMNIST, OrganCMNIST, OrganSMNIST, PathMNIST [26], Fashion-MNIST [23], CIFAR-10, CIFAR-100 [10], ImageNet-100 [17], and Tiny-ImageNet [3]. To simulate heterogeneous client data, we use a Dirichlet partition with concentration parameter $\gamma = 0.3$, where a smaller γ induces stronger heterogeneity. For local training, we use SGD with momentum 0.9, an initial learning rate of 0.01, weight decay 0.001, batch size 64, local epochs 5, and a learning-rate decay factor of 0.998 per communication round. We conduct 500 communication rounds for CIFAR-10 and CIFAR-100, and 100 rounds for the remaining datasets. The final results are reported as the average over the last 10% of the rounds.

Statistical noise correction. When computing the adaptive shrinkage coefficient, we apply a method-of-moments noise correction to account for the estimated noise contribution to the observed cross-client disagreement V_ℓ^t. Let $K = |S_t|$ and define the mean squared residual as

$$s_{\ell,\mathrm{res}}^t = \frac{1}{K q_\ell d_\ell} \sum_{i \in S_t} \|R_{i,\ell}^t\|_F^2.$$

We adjust the disagreement used for coefficient computation as

$$V_\ell^t \leftarrow \max\left\{0,\, V_\ell^t - \left(1 - \frac{1}{K}\right) \frac{s_{\ell,\mathrm{res}}^t}{d_\ell}\right\}.$$

Table 1: Test accuracy (%) averaged over three seeds. CTS denotes the proposed add-on. Dataset abbreviations: Blood = BloodMNIST, Org-A/C/S = OrganAMNIST/OrganCMNIST/OrganSMNIST, Path = PathMNIST, FMNIST = Fashion-MNIST, C10/C100 = CIFAR-10/100, IN-100 = ImageNet-100, and Tiny = Tiny-ImageNet.

Method	**Blood**	**Org-A**	**Org-C**	**Org-S**	**Path**	**FMNIST**	**C10**	**C100**	**IN-100**	**Tiny**
FedAvg [15]	77.26	86.86	76.49	62.22	67.94	84.72	80.82	60.55	53.08	40.00
+ CTS	**83.91**	**89.16**	**81.29**	**66.12**	**72.32**	**87.97**	**82.64**	**60.93**	**54.24**	**41.50**
FedLC [28]	85.44	89.27	81.32	67.37	81.56	88.17	84.88	58.87	54.10	41.08
+ CTS	**86.31**	**89.83**	**82.38**	**69.44**	**83.81**	**88.99**	**85.33**	**60.43**	**55.18**	**41.89**
FedRS [14]	84.67	89.65	80.74	67.32	80.83	88.39	84.77	59.50	53.99	41.23
+ CTS	**86.67**	**90.10**	**82.04**	**69.76**	**82.11**	**89.22**	**85.45**	**60.79**	**54.95**	**41.77**
FedGM [22]	80.62	88.42	79.37	63.97	70.35	85.89	82.17	60.59	53.22	40.49
+ CTS	**84.36**	**89.95**	**82.54**	**67.77**	**74.72**	**88.58**	**83.05**	**61.26**	**54.38**	**41.66**
FedAvgM [6]	76.25	88.05	77.04	63.10	68.90	85.17	81.09	61.21	53.95	40.94
+ CTS	**82.67**	**89.65**	**81.37**	**66.60**	**72.82**	**88.22**	**82.78**	**62.42**	**55.00**	**42.32**
FedExP [7]	73.30	86.47	77.09	61.13	69.33	84.73	81.06	60.36	52.75	40.06
+ CTS	**82.32**	**87.80**	**81.37**	**65.54**	**74.01**	**88.08**	**82.48**	**61.09**	**54.22**	**41.45**
FedACG [9]	72.88	86.89	62.08	49.48	75.36	84.76	87.86	70.65	60.46	48.39
+ CTS	**82.99**	**89.15**	**73.52**	**57.74**	**79.75**	**87.55**	**89.23**	**70.66**	**61.58**	**49.21**
FedDyn [1]	77.93	80.84	77.15	58.48	69.61	**51.68**	86.01	65.99	60.69	46.40
+ CTS	**84.43**	**83.46**	**81.39**	**65.71**	**72.82**	51.34	**86.32**	**66.30**	**61.35**	**46.73**
FedRCL [20]	71.93	85.16	74.90	59.33	68.83	84.22	79.62	59.87	52.32	38.42
+ CTS	**78.46**	**87.57**	**79.59**	**64.78**	**73.83**	**86.51**	**81.77**	**60.22**	**53.07**	**39.84**
FedYoYo [25]	63.90	85.72	63.56	49.98	85.24	82.95	88.34	68.77	46.58	**37.58**
+ CTS	**74.44**	**87.60**	**72.64**	**56.86**	**85.95**	**85.17**	**88.81**	**68.93**	**46.91**	37.54
FedProx [13]	77.25	87.33	75.47	61.17	67.81	85.09	81.01	59.96	52.57	40.01
+ CTS	**82.82**	**88.74**	**80.50**	**66.31**	**73.55**	**88.26**	**82.54**	**61.35**	**53.97**	**41.15**
MOON [11]	76.07	87.08	75.89	61.92	70.03	84.92	81.48	60.20	53.11	40.24
+ CTS	**82.84**	**89.03**	**81.55**	**66.57**	**72.83**	**87.83**	**82.90**	**61.39**	**54.37**	**41.42**

4.1 Accuracy comparison

Table 1 reports the main test accuracy results, and Table 2 summarizes the overall trends. FedCTS is applied to each base FL method as an add-on server-side correction.

Across all evaluated base algorithms, FedCTS yields a positive average gain for every dataset group and every algorithm group, with an overall average gain of $+2.74$ percentage points. Among the 120 method–dataset pairs, 118 improve after applying FedCTS; the largest gain is $+11.44$ percentage points, and the worst degradation is only -0.34 percentage points. These results support the intended role of FedCTS as an add-on correction: it provides meaningful improvements when the constant term is harmful, while adaptive shrinkage limits unnecessary modification when the correction is less useful. Runtime results in Appendix C show negligible computational overhead.

Data heterogeneity. We additionally evaluate FedCTS under a stronger non-IID split with $\gamma = 0.1$ and under IID data partitioning. Table 3 summarizes the results for $\gamma = 0.1$. FedCTS consistently improves the average accuracy across dataset groups and base algorithms, showing that the benefit of constant-term shrinkage is maintained under stronger data heterogeneity. This result is consistent with our observation that stronger heterogeneity amplifies client drift in the constant term. Under IID data partitioning, FedCTS also provides a small positive average gain, which is provided in Appendix D.4. Full accuracy results for both settings and additional experiments with different model architectures, numbers of clients, and local epochs are provided in Appendix D.

Table 2: Aggregated accuracy gains from CTS under the Dirichlet partition with $\gamma = 0.3$. *Avg. gain* is reported as mean $\pm$ SEM (Standard Error of the Mean) over three seeds. For each group, *Min. gain* and *Max. gain* denote the minimum and maximum gain observed across runs. In the final row, the *Min. gain* and *Max. gain* columns report the averages of group-wise minima and maxima.

By dataset

Dataset	Avg. gain $\pm$ SEM	Min. gain	Max. gain
Blood	$+\mathbf{6.23} \pm 0.90$	$+0.87$	$+10.54$
Org-A	$+\mathbf{1.69} \pm 0.19$	$+0.45$	$+2.61$
Org-C	$+\mathbf{4.92} \pm 0.97$	$+1.07$	$+11.44$
Org-S	$+\mathbf{4.81} \pm 0.51$	$+2.07$	$+8.26$
Path	$+\mathbf{3.56} \pm 0.52$	$+0.70$	$+5.74$
FMNIST	$+\mathbf{2.25} \pm 0.28$	-0.34	$+3.35$
C10	$+\mathbf{1.18} \pm 0.14$	$+0.31$	$+2.15$
C100	$+\mathbf{0.77} \pm 0.10$	$+0.02$	$+1.55$
IN-100	$+\mathbf{1.04} \pm 0.07$	$+0.34$	$+1.47$
Tiny	$+\mathbf{0.97} \pm 0.09$	-0.04	$+1.51$
Avg.	$+\mathbf{2.74} \pm 0.18$	$+0.55$	$+4.86$

By algorithm

Algorithm	Avg. gain $\pm$ SEM	Min. gain	Max. gain
FedAvg	$+\mathbf{3.02} \pm 0.52$	$+0.38$	$+6.65$
FedLC	$+\mathbf{1.15} \pm 0.14$	$+0.45$	$+2.25$
FedRS	$+\mathbf{1.18} \pm 0.22$	$+0.45$	$+2.44$
FedGM	$+\mathbf{2.32} \pm 0.43$	$+0.67$	$+4.38$
FedAvgM	$+\mathbf{2.81} \pm 0.50$	$+1.05$	$+6.42$
FedExP	$+\mathbf{3.21} \pm 0.76$	$+0.74$	$+9.03$
FedACG	$+\mathbf{4.26} \pm 1.06$	$+0.02$	$+11.44$
FedDyn	$+\mathbf{2.51} \pm 0.62$	-0.34	$+7.24$
FedRCL	$+\mathbf{3.10} \pm 0.63$	$+0.35$	$+6.53$
FedYoYo	$+\mathbf{3.22} \pm 0.90$	-0.04	$+10.54$
FedProx	$+\mathbf{3.15} \pm 0.51$	$+1.14$	$+5.74$
MOON	$+\mathbf{2.98} \pm 0.59$	$+1.18$	$+6.78$
Avg.	$+\mathbf{2.74} \pm 0.18$	$+0.50$	$+6.62$

Table 3: Aggregated accuracy gains from CTS under the Dirichlet partition with $\gamma = 0.1$. Summary statistics follow the definitions in Table 2.

By dataset

Dataset	n	Avg. gain $\pm$ SEM	Min. gain	Max. gain
Blood	9	$+\mathbf{21.74} \pm 3.79$	$+4.11$	$+36.40$
Org-A	9	$+\mathbf{9.29} \pm 3.05$	$+0.21$	$+28.84$
Org-C	9	$+\mathbf{16.09} \pm 3.68$	$+0.90$	$+30.67$
Org-S	9	$+\mathbf{6.44} \pm 1.74$	-2.27	$+13.83$
Path	9	$+\mathbf{6.57} \pm 2.39$	-1.14	$+22.31$
FMNIST	9	$+\mathbf{7.82} \pm 0.90$	$+3.24$	$+10.51$
C10	9	$+\mathbf{9.89} \pm 5.94$	$+0.85$	$+57.26$
C100	9	$+\mathbf{1.41} \pm 0.20$	$+0.75$	$+2.42$
IN-100	9	$+\mathbf{2.22} \pm 0.27$	$+0.99$	$+3.51$
Tiny	9	$+\mathbf{2.13} \pm 0.25$	$+0.79$	$+2.96$
Overall	90	$+\mathbf{8.36} \pm 1.08$	$+0.84$	$+20.87$

By algorithm

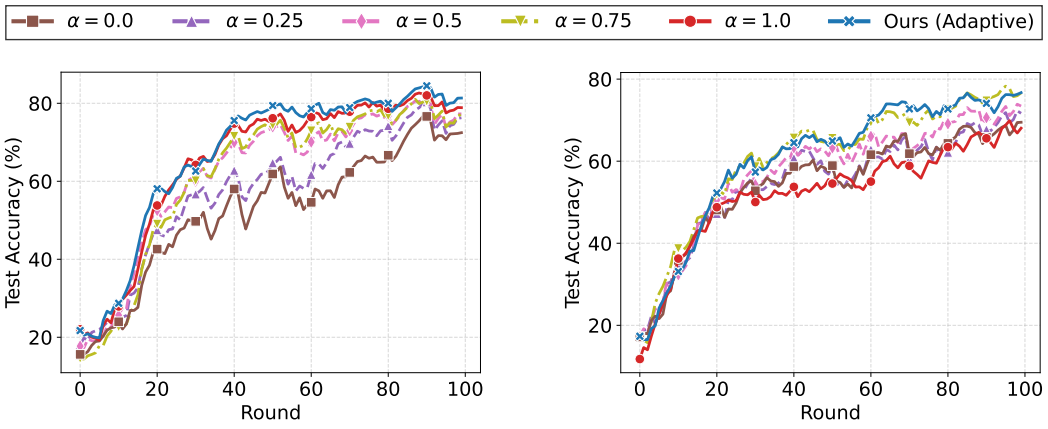
Algorithm	n	Avg. gain $\pm$ SEM	Min. gain	Max. gain
FedAvg	10	$+\mathbf{8.49} \pm 3.23$	$+2.07$	$+36.40$
FedLC	10	$+\mathbf{2.75} \pm 1.19$	$+0.21$	$+12.79$
FedRS	10	$+\mathbf{3.12} \pm 1.29$	-1.14	$+13.83$
FedGM	10	$+\mathbf{7.68} \pm 2.25$	$+1.11$	$+21.07$
FedExP	10	$+\mathbf{9.15} \pm 2.60$	$+2.28$	$+22.32$
FedACG	10	$+\mathbf{17.02} \pm 5.33$	$+0.94$	$+57.26$
FedRCL	10	$+\mathbf{7.08} \pm 3.28$	-2.27	$+31.71$
FedProx	10	$+\mathbf{10.03} \pm 3.60$	$+0.87$	$+31.50$
MOON	10	$+\mathbf{9.92} \pm 3.15$	$+1.95$	$+30.15$
Overall	90	$+\mathbf{8.36} \pm 1.08$	$+0.67$	$+28.56$

4.2 Ablation on constant term

Shrinkage coefficient. We study the effect of the shrinkage strength by replacing α with a fixed value in $\{0, 0.25, 0.5, 0.75, 1.0\}$, where $\alpha = 0$ corresponds to FedAvg and $\alpha = 1$ completely removes the constant term. Fig. 4 shows two representative results. On BloodMNIST, strong suppression performs best, whereas on PathMNIST, both no suppression and complete removal perform poorly. These results show that the appropriate shrinkage strength depends on the dataset and training dynamics, motivating FedCTS to adaptively compute the shrinkage coefficient rather than using a fixed global value. Additional results on additional datasets, together with analyses of layer-wise behavior and the extent of update correction, are provided in Appendix E.

Comparison with GC-Fed. We include GC-Fed because it also suppresses the row-wise constant term through zero-mean projection. Unlike fixed $\alpha = 1$, which removes the constant term once at the server, GC-Fed applies gradient centralization at every local optimization step on each client.

Table 4 shows that FedCTS achieves the best performance in most cases, while GC-Fed performs best in only a few cases with ResNet18. These results suggest that FedCTS adapts the correction across architectures and datasets, whereas unconditional zero-mean projection can degrade performance, especially on MobileNetV2 and CCT4. These results support our main design choice: the constant term should not be uniformly eliminated, but should be shrunk according to its cross-client reliability.



(a) BloodMNIST (b) PathMNIST

Figure 4: Ablation study on the shrinkage coefficient α. Fixed shrinkage values behave differently across datasets, while FedCTS adaptively selects the shrinkage strength.

Table 4: Ablation on constant-term suppression across models and datasets. FedCTS adaptively shrinks the constant term, while GC-Fed enforces zero-mean updates more aggressively.

Model	Method	Blood	Org-A	Org-C	Org-S	Path	FMNIST	C10	C100	IN-100	Tiny
ResNet18	FedAvg	77.26	86.86	76.49	62.22	67.94	84.72	80.82	60.55	53.08	40.00
	+CTS	**83.91**	**89.16**	**81.29**	**66.12**	72.32	**87.97**	**82.64**	60.93	54.24	**41.50**
	GC-Fed	77.42	87.95	80.76	63.40	**72.88**	85.70	81.61	**62.08**	**54.65**	40.94
MobileNetV2	FedAvg	50.27	80.13	73.43	48.84	68.39	79.37	66.48	41.26	41.04	31.90
	+CTS	**60.02**	**83.22**	**76.00**	**53.91**	**72.93**	**82.58**	**67.76**	**42.25**	**42.47**	**33.92**
	GC-Fed	44.03	75.16	57.52	33.40	70.14	80.26	66.34	39.19	37.25	24.53
CCT4	FedAvg	77.41	76.22	63.43	48.35	84.15	81.02	72.11	49.00	39.63	23.80
	+CTS	**78.82**	**78.61**	**69.92**	**54.03**	**84.48**	**81.78**	**73.65**	**49.19**	**40.08**	**23.93**
	GC-Fed	75.08	68.13	56.81	47.50	77.58	70.72	57.76	46.15	33.40	20.41

5 Conclusion

In this paper, we introduced **FedCTS (Constant Term Shrinkage for Federated Learning)**, a server-side aggregation method for mitigating update drift under heterogeneous data. FedCTS decomposes each client update into a constant term and a residual term, evaluates the cross-client reliability of the constant terms, and shrinks the unreliable constant term while leaving the residual term unchanged.

Our analysis showed that the constant term is strongly affected by data heterogeneity. Under stronger heterogeneity, the constant term becomes more prominent in update magnitude, and client drift in the constant component also increases. These observations justify correcting the constant term separately. FedCTS implements this correction through a closed-form shrinkage coefficient. As a result, the correction naturally becomes stronger when the constant coefficients disagree across clients and remains small when the constant term is reliable. Empirically, FedCTS achieved positive average gains across all evaluated base algorithms and improved the overall average accuracy across the reported datasets and methods.

Most importantly, FedCTS introduces no additional communication, no client-side training overhead, and no tuning hyperparameter for the shrinkage. Its additional computation is limited to simple server-side statistics and constant-term subtraction. These properties make FedCTS practical for FL, showing that a small correction of a specific update component can provide robust gains without increasing system complexity.

Acknowledgments and Disclosure of Funding

This research was supported in part by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grants funded by the Korea government (MSIT) (IITP-2026-RS-2020-II201819 and RS-2026-25522620), in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (RS-2026-25616569 and RS-2026-25494942), and in part by NRF grants funded by the Ministry of Education (RS-2021-NR060143).

References

[1] Durmus Alp Emre Acar, Yue Zhao, Ramon Matas Navarro, Matthew Mattina, Paul N Whatmough, and Venkatesh Saligrama. Federated learning based on dynamic regularization. *arXiv preprint arXiv:2111.04263*, 2021.

[2] 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*, pages 1175–1191, 2017.

[3] Stanford CS231n Course. Tiny imagenet challenge, 2015.

[4] Ali Hassani, Steven Walton, Nikhil Shah, Abulikemu Abuduweili, Jiachen Li, and Humphrey Shi. Escaping the big data paradigm with compact transformers. *arXiv preprint arXiv:2104.05704*, 2021.

[5] 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.

[6] 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.

[7] Divyansh Jhunjhunwala, Shiqiang Wang, and Gauri Joshi. Fedexp: Speeding up federated averaging via extrapolation. *arXiv preprint arXiv:2301.09604*, 2023.

[8] Peter Kairouz, H. Brendan McMahan, et al. Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 14(1–2):1–210, 2021.

[9] Geeho Kim, Jinkyu Kim, and Bohyung Han. Communication-efficient federated learning with accelerated client gradient. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12385–12394, 2024.

[10] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[11] Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10713–10722, 2021.

[12] Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3):50–60, 2020.

[13] 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.

[14] Xin-Chun Li and De-Chuan Zhan. Fedrs: Federated learning with restricted softmax for label distribution non-iid data. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pages 995–1005, 2021.

[15] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

[16] Takayuki Nishio and Ryo Yonetani. Client selection for federated learning with heterogeneous resources in mobile edge. In *Proceedings of the IEEE International Conference on Communications*, pages 1–7, 2019.

[17] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[18] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[19] Jungwon Seo, Ferhat Ozgur Catak, Chunming Rong, Kibeom Hong, and Minhoe Kim. Gc-fed: Gradient centralized federated learning with partial client participation. *Information Fusion*, page 104148, 2026.

[20] Seonguk Seo, Jinkyu Kim, Geeho Kim, and Bohyung Han. Relaxed contrastive learning for federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12279–12288, 2024.

[21] Micah J. Sheller, Brandon Edwards, G. Anthony Reina, Jason Martin, Sarthak Pati, Aikaterini Kotrotsou, Mikhail Milchenko, Weilin Xu, Daniel Marcus, Rivka R. Colen, and Spyridon Bakas. Federated learning in medicine: Facilitating multi-institutional collaborations without sharing patient data. *Scientific Reports*, 10(1):12598, 2020.

[22] Jianhui Sun, Xidong Wu, Heng Huang, and Aidong Zhang. On the role of server momentum in federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 15164–15172, 2024.

[23] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

[24] Jie Xu, Benjamin S. Glicksberg, Chang Su, Peter B. Walker, Jiang Bian, and Fei Wang. Federated learning for healthcare informatics. *Journal of Healthcare Informatics Research*, 5(1):1–19, 2021.

[25] Shanshan Yan, Zexi Li, Chao Wu, Meng Pang, Yang Lu, Yan Yan, and Hanzi Wang. You are your own best teacher: Achieving centralized-level performance in federated learning under heterogeneous and long-tailed data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2750–2759, 2025.

[26] Jiancheng Yang, Rui Shi, and Bingbing Ni. Medmnist classification decathlon: A lightweight automl benchmark for medical image analysis. In *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pages 191–195. IEEE, 2021.

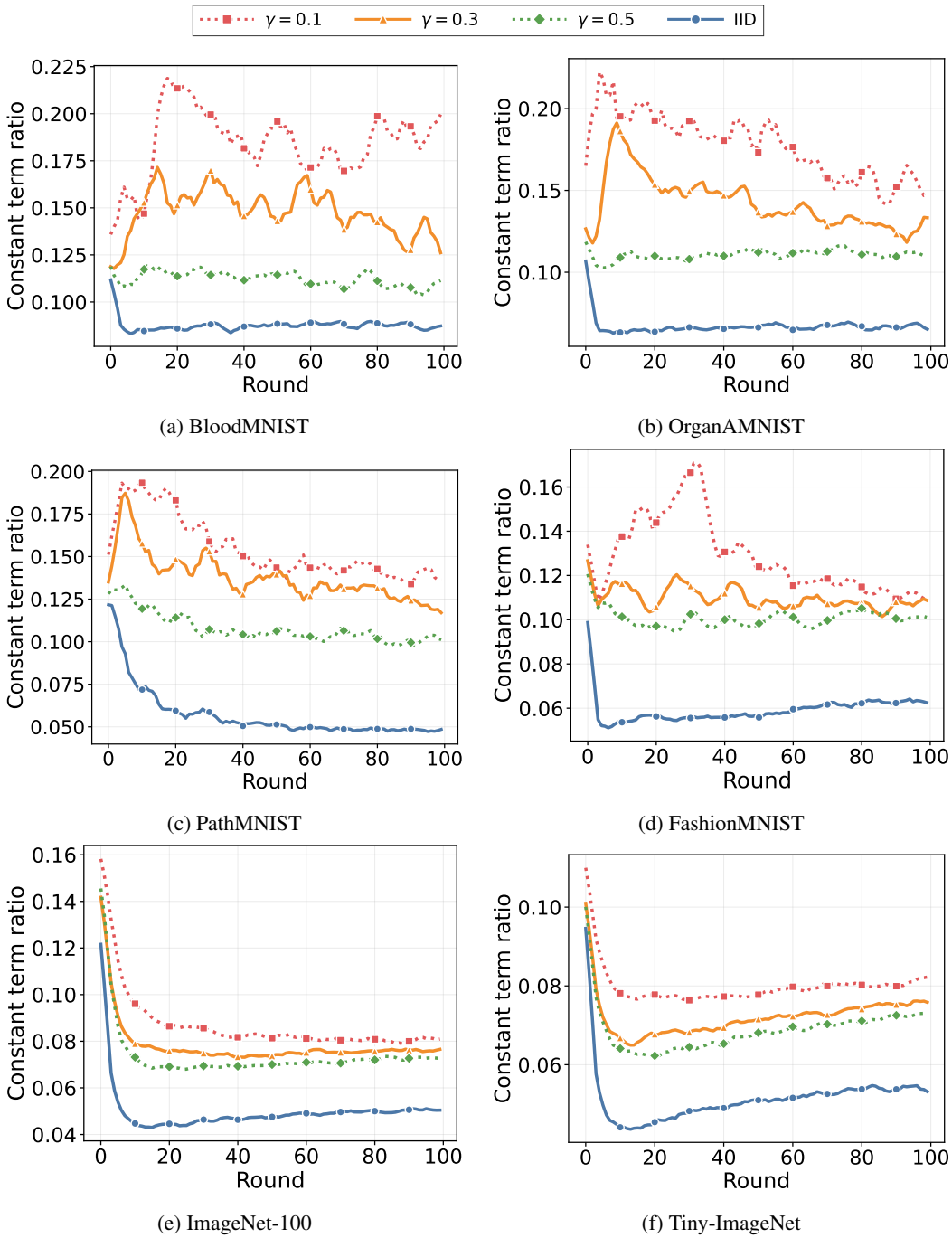
[27] Hongwei Yong, Jianqiang Huang, Xiansheng Hua, and Lei Zhang. Gradient centralization: A new optimization technique for deep neural networks. In *European Conference on Computer Vision*, pages 635–652. Springer, 2020.

[28] 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 *International Conference on Machine Learning*, pages 26311–26329. PMLR, 2022.

A Additional statistics

A.1 Constant update ratio

We additionally measure the constant update ratio on six additional datasets. Across datasets, the ratio generally increases as γ decreases, indicating that stronger heterogeneity makes the constant term account for a larger portion of the update magnitude.



(a) BloodMNIST (b) OrganAMNIST

(c) PathMNIST (d) FashionMNIST

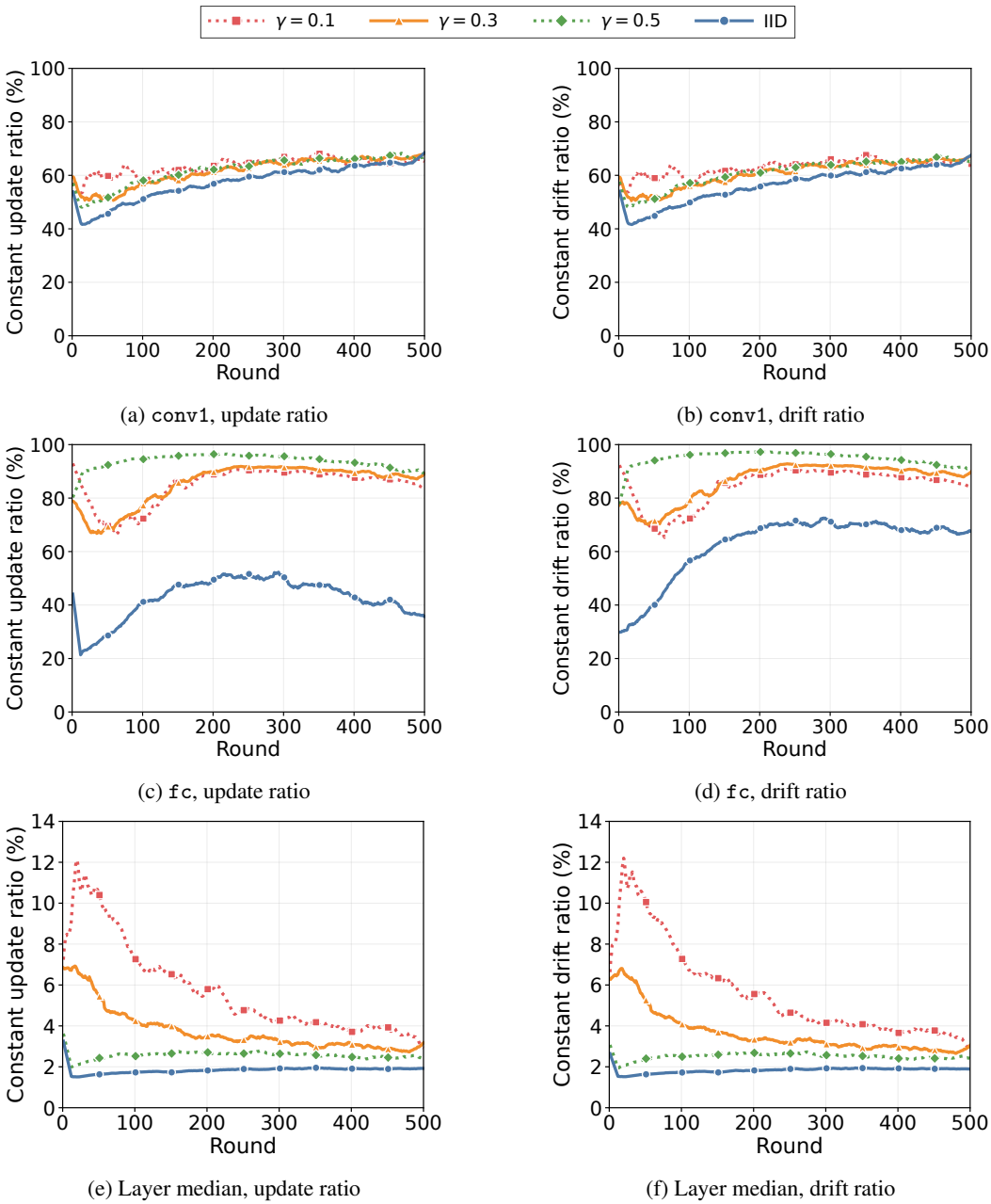
(e) ImageNet-100 (f) Tiny-ImageNet

Figure 5: Constant update ratio $\rho^t_{\ell,\mathrm{con}}$ under IID and Dirichlet partitions. A smaller γ indicates stronger data heterogeneity. The ratio increases as γ decreases, indicating that the constant term accounts for a larger portion of the update magnitude under stronger heterogeneity.

A.2 Layer-wise constant update and drift ratios

We additionally examine how the constant term varies across layers. For each layer, we measure the constant update ratio $U^t_{\ell,\mathrm{con}}/U^t_\ell$ and the constant drift ratio $D^t_{\ell,\mathrm{con}}/D^t_\ell$. Figs. 6 and 7 show the first convolutional layer (`conv1`), the final classifier (`fc`), and the median over all 21 weight layers. The median is computed from the layer-wise ratios at each round.

The first convolutional layer and the final classifier have substantially larger constant update and drift ratios than the layer median. Thus, an average over layers can hide differences in the contribution of the constant term. The layer median also generally increases under stronger heterogeneity, although the degree of change differs across datasets and rounds. These results support computing the shrinkage coefficient separately for each layer rather than applying the same correction to all layers.



(a) `conv1`, update ratio

(b) `conv1`, drift ratio

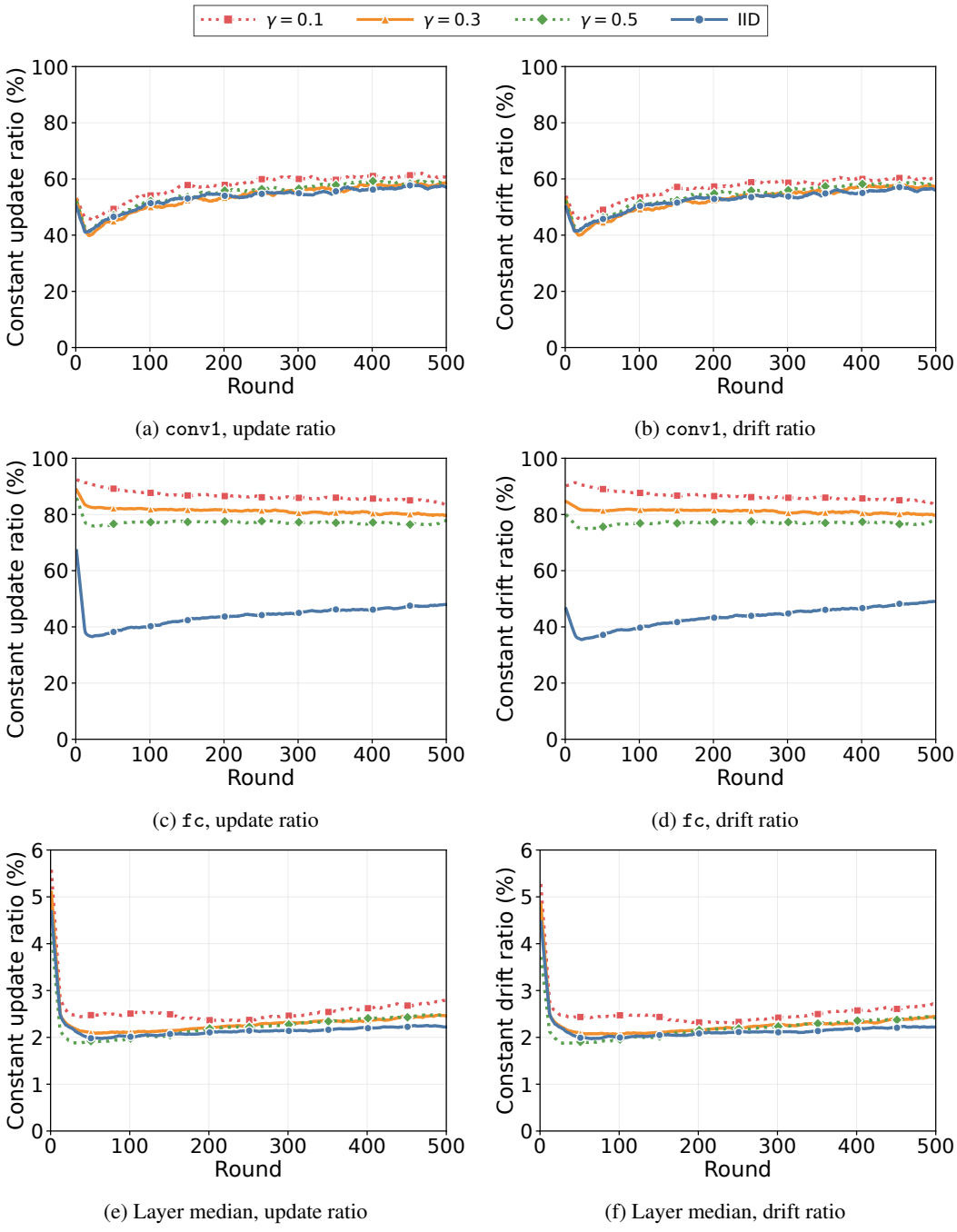
(c) `fc`, update ratio

(d) `fc`, drift ratio

(e) Layer median, update ratio

(f) Layer median, drift ratio

Figure 6: Layer-wise constant update and drift ratios on CIFAR-10. The first convolutional layer and the final classifier have larger ratios than the layer median.



(a) `conv1`, update ratio

(b) `conv1`, drift ratio

(c) `fc`, update ratio

(d) `fc`, drift ratio

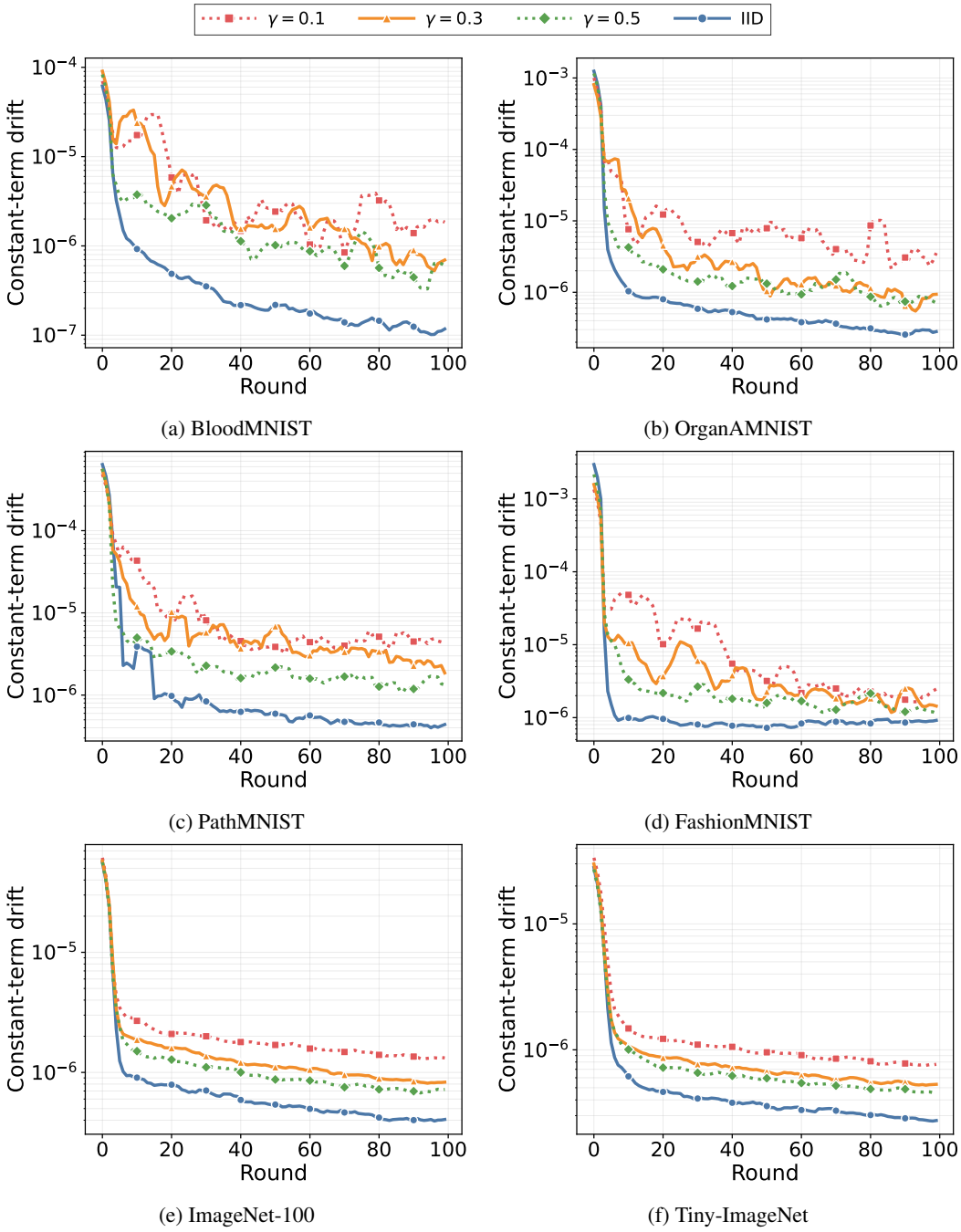
(e) Layer median, update ratio

(f) Layer median, drift ratio

Figure 7: Layer-wise constant update and drift ratios on CIFAR-100. The first convolutional layer and the final classifier have larger ratios than the layer median.

A.3 Constant-term client drift

We additionally measure constant-term client drift on six additional datasets. Across datasets, the drift generally increases as γ decreases, indicating that stronger heterogeneity amplifies cross-client disagreement in the constant term.



(a) BloodMNIST (b) OrganAMNIST

(c) PathMNIST (d) FashionMNIST

(e) ImageNet-100 (f) Tiny-ImageNet

Figure 8: Constant-term client drift $D^t_{\ell,\mathrm{con}}$ under IID and Dirichlet partitions. A smaller γ indicates stronger data heterogeneity. The drift increases as γ decreases, indicating that stronger heterogeneity amplifies cross-client disagreement in the constant direction.

B Theoretical Analysis

B.1 Update ratio

Lemma B.1. *Let $S_{\mathrm{con}}, S_{\mathrm{res}} > 0$ be the constant and residual magnitudes under a less heterogeneous setting. Suppose stronger heterogeneity adds magnitudes $T_{\mathrm{con}}, T_{\mathrm{res}} \geq 0$ to the components. Define*

$$\rho_{\mathrm{con}} = \frac{S_{\mathrm{con}}}{S_{\mathrm{con}} + S_{\mathrm{res}}}, \qquad \rho'_{\mathrm{con}} = \frac{S_{\mathrm{con}} + T_{\mathrm{con}}}{S_{\mathrm{con}} + S_{\mathrm{res}} + T_{\mathrm{con}} + T_{\mathrm{res}}}. \tag{19}$$

Then

$$\rho'_{\mathrm{con}} \geq \rho_{\mathrm{con}} \quad \Longleftrightarrow \quad T_{\mathrm{con}} S_{\mathrm{res}} \geq S_{\mathrm{con}} T_{\mathrm{res}}. \tag{20}$$

If $S_{\mathrm{con}} > 0$ and $S_{\mathrm{res}} > 0$, this is equivalent to

$$\frac{T_{\mathrm{con}}}{S_{\mathrm{con}}} \geq \frac{T_{\mathrm{res}}}{S_{\mathrm{res}}}. \tag{21}$$

Thus, the constant ratio increases when the heterogeneity-induced increase is relatively more concentrated in the constant term. Conversely, the residual ratio increases when the increase is relatively more concentrated in the residual term.

B.2 Reduction of client drift

The following lemma shows that FedCTS reduces the constant component of the cross-client disagreement among the updates used in aggregation.

Lemma B.2. *(Reduction of Layer-wise Client-Update Drift) For a target layer ℓ, define*

$$D_\ell^t = \sum_{i \in \mathcal{S}_t} p_i \left\| G_{i,\ell}^t - \bar{G}_\ell^t \right\|_F^2 = D_{\ell,\mathrm{con}}^t + D_{\ell,\mathrm{res}}^t,$$

where

$$D_{\ell,\mathrm{con}}^t = \sum_{i \in \mathcal{S}_t} p_i d_\ell \left\| \mu_{i,\ell}^t - \bar{\mu}_\ell^t \right\|_2^2, \qquad D_{\ell,\mathrm{res}}^t = \sum_{i \in \mathcal{S}_t} p_i \left\| R_{i,\ell}^t - \bar{R}_\ell^t \right\|_F^2.$$

Let

$$\widetilde{D}_\ell^t = \sum_{i \in \mathcal{S}_t} p_i \left\| \widetilde{G}_{i,\ell}^t - \bar{\widetilde{G}}_\ell^{\,t} \right\|_F^2$$

be the corresponding disagreement after FedCTS correction. Then

$$\widetilde{D}_\ell^t = (1 - \alpha_\ell^t)^2 D_{\ell,\mathrm{con}}^t + D_{\ell,\mathrm{res}}^t.$$

Consequently,

$$D_\ell^t - \widetilde{D}_\ell^t = \alpha_\ell^t (2 - \alpha_\ell^t) D_{\ell,\mathrm{con}}^t \geq 0.$$

Proof. For a target layer ℓ, the corrected update is

$$\widetilde{G}_{i,\ell}^t = (1 - \alpha_\ell^t) \mu_{i,\ell}^t \mathbf{1}_{d_\ell}^\top + R_{i,\ell}^t.$$

Taking the weighted average over clients gives

$$\bar{\widetilde{G}}_\ell^{\,t} = (1 - \alpha_\ell^t) \bar{\mu}_\ell^t \mathbf{1}_{d_\ell}^\top + \bar{R}_\ell^t.$$

Therefore,

$$\widetilde{G}_{i,\ell}^t - \bar{\widetilde{G}}_\ell^{\,t} = (1 - \alpha_\ell^t)(\mu_{i,\ell}^t - \bar{\mu}_\ell^t) \mathbf{1}_{d_\ell}^\top + (R_{i,\ell}^t - \bar{R}_\ell^t).$$

The row-wise constant component and the residual component are orthogonal, because $R_{i,\ell}^t \mathbf{1}_{d_\ell} = 0$ and $\bar{R}_\ell^t \mathbf{1}_{d_\ell} = 0$. Hence,

$$\left\| \widetilde{G}_{i,\ell}^t - \bar{\widetilde{G}}_\ell^{\,t} \right\|_F^2 = (1 - \alpha_\ell^t)^2 d_\ell \left\| \mu_{i,\ell}^t - \bar{\mu}_\ell^t \right\|_2^2 + \left\| R_{i,\ell}^t - \bar{R}_\ell^t \right\|_F^2.$$

Multiplying by p_i and summing over clients gives

$$\widetilde{D}_\ell^t = (1 - \alpha_\ell^t)^2 D_{\ell,\mathrm{con}}^t + D_{\ell,\mathrm{res}}^t.$$

Since $0 \leq \alpha_\ell^t \leq 1$,

$$D_\ell^t - \widetilde{D}_\ell^t = \left[1 - (1 - \alpha_\ell^t)^2 \right] D_{\ell,\mathrm{con}}^t = \alpha_\ell^t (2 - \alpha_\ell^t) D_{\ell,\mathrm{con}}^t \geq 0.$$

□

The reduction is proportional to the constant-component drift $D_{\ell,\mathrm{con}}^t$ and to the shrinkage factor $\alpha_\ell^t(2 - \alpha_\ell^t)$. Thus, FedCTS has a stronger effect when the client disagreement is concentrated in the constant term and when the adaptive coefficient chooses stronger shrinkage.

B.3 Monotonicity of the adaptive coefficient

The following lemma explains how FedCTS automatically chooses the shrinkage strength.

Lemma B.3. *For $M, V \geq 0$ and $M + V > 0$, let*

$$\alpha = \frac{V}{V + M}. \tag{22}$$

Then $0 \leq \alpha \leq 1$. Moreover,

$$\frac{\partial \alpha}{\partial V} = \frac{M}{(V + M)^2} \geq 0, \qquad \frac{\partial \alpha}{\partial M} = -\frac{V}{(V + M)^2} \leq 0. \tag{23}$$

Thus, the shrinkage strength increases with cross-client disagreement and decreases with the observed consensus magnitude. If $M + V = 0$, then all constant coefficients are zero, and FedCTS is not needed.

Proof. The bound $0 \leq \alpha \leq 1$ follows directly from $M, V \geq 0$ and $M + V > 0$. The monotonicity follows by direct differentiation. When $M + V = 0$, we have $M = 0$ and $V = 0$, so the constant coefficients are all zero and the correction has no effect. □

The statistic M measures the magnitude of the observed consensus in the constant coefficient, whereas V measures disagreement across clients. When V is large relative to M, the client constant terms are inconsistent, so α becomes large and FedCTS suppresses the constant term more strongly. When M is large relative to V, the clients show a stable common constant direction, so α becomes small and FedCTS preserves most of that term. Therefore, the adaptive coefficient balances two effects: preserving observed consensus and reducing cross-client disagreement.

B.4 Derivations of the decomposition and shrinkage rule

Orthogonal decomposition. By the definitions of $\mu_{i,\ell}^t$ and $R_{i,\ell}^t$,

$$\begin{aligned} R_{i,\ell}^t \mathbf{1}_{d_\ell} &= \left(G_{i,\ell}^t - \mu_{i,\ell}^t \mathbf{1}_{d_\ell}^\top \right) \mathbf{1}_{d_\ell} \\ &= G_{i,\ell}^t \mathbf{1}_{d_\ell} - d_\ell \mu_{i,\ell}^t = 0. \end{aligned}$$

Hence,

$$\left\langle \mu_{i,\ell}^t \mathbf{1}_{d_\ell}^\top, R_{i,\ell}^t \right\rangle_F = (\mu_{i,\ell}^t)^\top R_{i,\ell}^t \mathbf{1}_{d_\ell} = 0.$$

Since $\|\mu_{i,\ell}^t \mathbf{1}_{d_\ell}^\top\|_F^2 = d_\ell \|\mu_{i,\ell}^t\|_2^2$, the Pythagorean identity gives Eq. (8). Multiplying by p_i and summing over clients gives Eq. (9). Moreover, $\bar{R}_\ell^t \mathbf{1}_{d_\ell} = \sum_{i \in S_t} p_i R_{i,\ell}^t \mathbf{1}_{d_\ell} = 0$. Thus, the centered residual remains orthogonal to the centered constant term, and the same identity gives Eq. (12).

Shrinkage coefficient. The objective in Eq. (16) can be written as

$$\begin{aligned} J(\alpha) &= \frac{1}{q_\ell} \|\alpha \hat{\mu}_\ell^t\|_2^2 + \frac{1}{|S_t| q_\ell} \sum_{i \in S_t} \left\| (1 - \alpha)(\mu_{i,\ell}^t - \hat{\mu}_\ell^t) \right\|_2^2 \\ &= \alpha^2 M_\ell^t + (1 - \alpha)^2 V_\ell^t. \end{aligned}$$

Differentiation yields

$$J'(\alpha) = 2\alpha(M_\ell^t + V_\ell^t) - 2V_\ell^t, \qquad J''(\alpha) = 2(M_\ell^t + V_\ell^t) \geq 0.$$

For $M_\ell^t + V_\ell^t > 0$, the unique minimizer is $\alpha = V_\ell^t / (M_\ell^t + V_\ell^t)$, which lies in $[0, 1]$. If $M_\ell^t + V_\ell^t = 0$, all constant coefficients are zero, so the correction has no effect for any α.

C Runtime for baselines and FedCTS

Table 5 reports the runtime results. Since FedCTS is applied only once on the server and requires only row-wise means, layer-wise squared norms, and constant-term subtraction, it adds negligible computational overhead. FedCTS also requires no extra communication or client-side computation.

The experiments are conducted on a V100 GPU.

Table 5: Wall-clock time comparison across all methods. Each entry reports the cumulative runtime in hours over three seeds. The total runtimes of each base method and its +CTS variant are comparable, although individual entries vary.

Method	**Blood**	**Org-A**	**Org-C**	**Org-S**	**Path**	**FMNIST**	**C10**	**C100**	**IN-100**	**Tiny**	**Total**
FedAvg	0.48	1.41	0.54	0.57	2.91	2.22	8.49	8.94	8.16	6.03	39.75
+ CTS	0.48	1.41	0.57	0.63	3.18	1.89	8.49	9.15	7.47	5.52	38.79
GC-Fed	0.48	1.44	0.57	0.60	3.27	2.01	9.03	9.21	7.68	5.67	39.96
FedLC	0.42	1.38	0.54	0.57	2.88	1.89	8.64	9.63	7.56	5.55	39.06
+ CTS	0.42	1.41	0.54	0.60	2.88	1.89	8.64	9.57	7.56	5.58	39.09
FedRS	0.42	1.38	0.57	0.60	2.91	1.89	8.64	8.61	7.56	5.58	38.16
+ CTS	0.42	1.41	0.57	0.60	2.88	1.89	8.61	8.64	7.56	5.58	38.16
FedGM	0.42	1.38	0.57	0.57	2.88	1.86	8.64	8.91	7.53	5.52	38.28
+ CTS	0.42	1.38	0.54	0.57	2.88	1.89	8.67	8.79	7.53	5.52	38.19
FedAvgM	0.42	1.38	0.54	0.60	2.88	1.89	8.52	9.09	7.53	5.52	38.37
+ CTS	0.42	1.38	0.54	0.57	2.91	1.89	8.61	9.00	7.53	5.52	38.37
FedExP	0.42	1.41	0.54	0.60	3.03	2.01	8.64	8.52	7.56	5.67	38.40
+ CTS	0.45	1.41	0.57	0.60	3.00	1.98	8.67	8.64	7.59	5.79	38.70
FedACG	0.60	1.92	0.72	0.78	4.38	2.82	12.42	12.48	9.12	6.75	51.99
+ CTS	0.60	1.95	0.72	0.81	4.29	2.82	12.66	12.72	9.21	6.87	52.65
FedDyn	0.45	1.47	0.57	0.63	3.12	2.04	9.27	9.15	7.86	5.82	40.38
+ CTS	0.48	1.50	0.60	0.63	3.06	2.04	9.33	9.21	7.89	5.91	40.65
FedRCL	0.42	1.44	0.60	0.60	3.12	2.04	9.18	9.15	7.86	5.76	40.17
+ CTS	0.45	1.47	0.57	0.63	3.09	2.04	9.18	9.12	7.80	5.94	40.29
FedYoYo	0.96	2.91	1.08	1.14	7.47	4.59	20.10	20.79	18.12	13.53	90.69
+ CTS	0.96	2.94	1.11	1.17	7.53	4.62	18.09	20.76	18.09	13.44	88.71
FedProx	0.66	2.01	0.78	0.81	4.71	2.97	13.23	13.59	9.36	6.87	54.99
+ CTS	0.63	2.01	0.78	0.81	4.56	2.97	13.11	13.62	9.36	6.90	54.75
MOON	0.51	1.59	0.63	0.69	3.51	2.28	10.35	10.68	10.29	7.74	48.27
+ CTS	0.54	1.62	0.63	0.66	3.48	2.28	10.26	10.68	10.29	7.74	48.18

D Additional Experimental Results

D.1 Model: MobileNetV2

This experiment uses MobileNetV2 [18] as a lightweight convolutional backbone. Unlike ResNet18, which relies on standard convolution, MobileNetV2 uses inverted residual blocks with linear bottlenecks and depthwise separable convolutions, combining depthwise spatial filtering with pointwise 1×1 channel mixing. The table reports Base and +CTS results to examine whether CTS remains effective under this efficient CNN design.

Table 6: Test accuracy (%) for Base and +CTS. Dataset abbreviations: Blood = BloodMNIST, Org-A/C/S = OrganAMNIST/OrganCMNIST/OrganSMNIST, Path = PathMNIST, FMNIST = Fashion-MNIST, C10/C100 = CIFAR-10/100, IN-100 = ImageNet-100, and Tiny = Tiny-ImageNet.

Method	Blood	Org-A	Org-C	Org-S	Path	FMNIST	C10	C100	IN-100	Tiny
FedAvg [15]	50.27	80.13	73.43	48.84	68.39	79.37	66.48	41.26	41.04	31.90
+ CTS	**60.02**	**83.22**	**76.00**	**53.91**	**72.93**	**82.58**	**67.76**	**42.25**	**42.47**	**33.92**
FedLC [28]	70.09	83.90	72.60	55.68	75.11	84.29	**73.42**	40.02	42.55	32.89
+ CTS	**76.60**	**84.58**	**74.99**	**58.38**	**76.01**	**86.33**	72.58	**42.13**	**43.08**	**33.84**
FedRS [14]	70.91	84.46	71.88	54.12	77.27	84.56	68.51	40.55	**42.85**	**33.73**
+ CTS	**75.37**	**85.81**	**75.46**	**58.76**	**78.19**	**86.56**	**69.64**	**42.22**	42.31	33.33
FedGM [22]	50.99	83.72	73.02	51.30	72.15	82.36	**67.68**	42.28	41.47	31.87
+ CTS	**60.98**	**84.48**	**75.37**	**56.24**	**72.54**	**84.04**	66.92	**43.12**	**42.43**	**33.34**
FedExP [7]	49.86	81.78	71.64	50.73	69.77	79.91	66.34	40.73	40.83	32.31
+ CTS	**55.19**	**84.36**	**75.06**	**54.18**	**72.36**	**82.69**	**66.97**	**42.54**	**42.15**	**33.23**
FedACG [9]	31.28	74.56	74.87	49.06	65.85	63.54	76.32	**61.09**	50.52	41.21
+ CTS	**51.67**	**82.05**	**79.39**	**55.12**	**69.33**	**67.59**	**81.04**	59.61	**52.32**	**41.99**
FedRCL [20]	**58.37**	79.81	71.15	49.35	62.44	79.33	62.50	42.15	41.50	32.25
+ CTS	57.28	**80.27**	**73.46**	**55.51**	**71.18**	**80.59**	**65.34**	**44.61**	**43.59**	**32.77**
FedProx [13]	56.31	82.97	71.36	51.49	70.65	81.55	66.10	41.82	40.21	31.88
+ CTS	**60.91**	**83.81**	**74.81**	**56.20**	**75.42**	**83.95**	**69.31**	**42.33**	**41.66**	**33.38**
MOON [11]	56.74	82.81	72.24	49.31	67.89	80.78	**66.07**	38.95	41.15	32.30
+ CTS	**57.71**	**84.45**	**74.84**	**55.82**	**73.39**	**83.09**	65.74	**42.20**	**41.54**	**32.98**

Table 7: Aggregated gains in percentage points using only the nine algorithms available in the CTS results. *Avg. gain* is reported as mean ± SEM. For each group, *Min. gain* and *Max. gain* denote the minimum and maximum gain observed across the corresponding entries. In the final row, the *Min. gain* and *Max. gain* columns report the averages of group-wise minima and maxima.

By dataset

Dataset	n	Avg. gain ± SEM	Min. gain	Max. gain
Blood	9	$+\mathbf{6.77} \pm 2.08$	-1.10	$+20.39$
Org-A	9	$+\mathbf{2.10} \pm 0.74$	$+0.46$	$+7.49$
Org-C	9	$+\mathbf{3.02} \pm 0.25$	$+2.31$	$+4.52$
Org-S	9	$+\mathbf{4.92} \pm 0.42$	$+2.70$	$+6.52$
Path	9	$+\mathbf{3.54} \pm 0.90$	$+0.39$	$+8.74$
FMNIST	9	$+\mathbf{2.42} \pm 0.28$	$+1.27$	$+4.05$
C10	9	$+\mathbf{1.32} \pm 0.64$	-0.84	$+4.72$
C100	9	$+\mathbf{1.35} \pm 0.45$	-1.48	$+3.25$
IN-100	9	$+\mathbf{1.05} \pm 0.27$	-0.54	$+2.09$
Tiny	9	$+\mathbf{0.94} \pm 0.23$	-0.40	$+2.02$
Avg.	90	$+\mathbf{2.74} \pm 0.31$	$+0.28$	$+6.38$

By algorithm

Algorithm	n	Avg. gain ± SEM	Min. gain	Max. gain
FedAvg	10	$+\mathbf{3.39} \pm 0.83$	$+0.99$	$+9.75$
FedLC	10	$+\mathbf{1.80} \pm 0.62$	-0.84	$+6.51$
FedRS	10	$+\mathbf{1.88} \pm 0.58$	-0.54	$+4.64$
FedGM	10	$+\mathbf{2.26} \pm 0.98$	-0.76	$+9.99$
FedExP	10	$+\mathbf{2.48} \pm 0.44$	$+0.63$	$+5.33$
FedACG	10	$+\mathbf{5.18} \pm 1.88$	-1.48	$+20.39$
FedRCL	10	$+\mathbf{2.57} \pm 0.91$	-1.10	$+8.74$
FedProx	10	$+\mathbf{2.74} \pm 0.52$	$+0.52$	$+4.76$
MOON	10	$+\mathbf{2.35} \pm 0.70$	-0.34	$+6.52$
Avg.	90	$+\mathbf{2.74} \pm 0.31$	-0.32	$+8.51$

D.2 Model: CCT4

This experiment replaces the backbone with CCT4 [4] and evaluates whether CTS generalizes to a compact transformer architecture. Across datasets and algorithms, CTS improves the Base variants by $+2.23 \pm 0.41$ percentage points on average, indicating that the effect is not limited to a single model family.

Table 8: Test accuracy (%) for Base and +CTS in CCT4 [4]. Dataset abbreviations: Blood = BloodMNIST, Org-A/C/S = OrganAMNIST/OrganCMNIST/OrganSMNIST, Path = PathMNIST, FMNIST = Fashion-MNIST, C10/C100 = CIFAR-10/100, IN-100 = ImageNet-100, and Tiny = Tiny-ImageNet.

Method	Blood	Org-A	Org-C	Org-S	Path	FMNIST	C10	C100	IN-100	Tiny
FedAvg [15]	77.41	76.22	63.43	48.35	84.15	81.02	72.11	49.00	39.63	23.80
+ CTS	**78.82**	**78.61**	**69.92**	**54.03**	**84.48**	**81.78**	**73.65**	**49.19**	**40.08**	**23.93**
FedLC [28]	76.35	65.54	50.98	43.97	**81.75**	76.15	60.57	45.83	**35.51**	21.27
+ CTS	**76.75**	**67.63**	**53.88**	**46.82**	80.28	**77.07**	**61.03**	**46.09**	35.21	**21.32**
FedRS [14]	76.78	64.27	51.87	40.89	**81.32**	76.12	60.32	45.86	35.34	**21.40**
+ CTS	**77.15**	**67.38**	**54.68**	**44.17**	80.56	**77.05**	**60.85**	**46.21**	**35.46**	21.37
FedGM [22]	75.97	62.44	52.38	43.05	**79.89**	72.35	54.97	44.93	33.83	19.97
+ CTS	**76.88**	**65.71**	**58.19**	**47.32**	78.06	**74.77**	**58.59**	**45.51**	**33.97**	**20.16**
FedExP [7]	73.72	52.87	44.61	36.57	67.66	62.58	40.45	38.63	20.40	10.87
+ CTS	**74.36**	**55.62**	**51.21**	**37.66**	**68.64**	**63.78**	**40.57**	**40.61**	**21.36**	**12.91**
FedACG [9]	**22.19**	45.77	57.59	**17.68**	74.23	67.84	20.63	52.96	**41.83**	26.36
+ CTS	17.77	**62.67**	**67.34**	**17.68**	**81.71**	**72.64**	**48.69**	**54.62**	41.32	**26.95**
FedRCL [20]	75.17	62.64	54.58	44.50	75.10	71.11	59.31	44.33	32.28	**18.37**
+ CTS	**75.48**	**65.12**	**57.46**	**47.97**	**76.27**	**72.80**	**61.68**	**44.91**	**32.46**	18.29
FedProx [13]	74.76	60.73	51.15	40.74	**79.46**	70.59	53.38	44.50	33.38	19.66
+ CTS	**75.57**	**64.81**	**57.24**	**46.99**	77.81	**73.66**	**57.95**	**45.23**	**33.69**	**19.84**
MOON [11]	75.18	61.88	51.94	42.34	**78.87**	70.73	53.35	44.58	33.90	20.16
+ CTS	**75.90**	**65.79**	**57.56**	**47.09**	77.62	**73.52**	**57.07**	**45.32**	**33.95**	**20.20**

Table 9: Aggregated gains in percentage points using only the nine algorithms available in the CTS results. *Avg. gain* is reported as mean ± SEM. For each group, *Min. gain* and *Max. gain* denote the minimum and maximum gain observed across the corresponding entries. In the final row, the *Min. gain* and *Max. gain* columns report the averages of group-wise minima and maxima.

By dataset

Dataset	n	Avg. gain ± SEM	Min. gain	Max. gain
Blood	9	$+\mathbf{0.13} \pm 0.58$	-4.42	$+1.41$
Org-A	9	$+\mathbf{4.56} \pm 1.56$	$+2.09$	$+16.90$
Org-C	9	$+\mathbf{5.44} \pm 0.76$	$+2.81$	$+9.75$
Org-S	9	$+\mathbf{3.52} \pm 0.68$	$+0.00$	$+6.25$
Path	9	$+\mathbf{0.33} \pm 0.97$	-1.83	$+7.47$
FMNIST	9	$+\mathbf{2.06} \pm 0.45$	$+0.76$	$+4.80$
C10	9	$+\mathbf{5.00} \pm 2.93$	$+0.12$	$+28.06$
C100	9	$+\mathbf{0.79} \pm 0.21$	$+0.19$	$+1.98$
IN-100	9	$+\mathbf{0.16} \pm 0.14$	-0.51	$+0.96$
Tiny	9	$+\mathbf{0.35} \pm 0.22$	-0.08	$+2.04$
Overall	90	$+\mathbf{2.23} \pm 0.41$	-0.09	$+7.96$

By algorithm

Algorithm	n	Avg. gain ± SEM	Min. gain	Max. gain
FedAvg	10	$+\mathbf{1.94} \pm 0.73$	$+0.13$	$+6.49$
FedLC	10	$+\mathbf{0.82} \pm 0.44$	-1.47	$+2.91$
FedRS	10	$+\mathbf{1.07} \pm 0.46$	-0.75	$+3.28$
FedGM	10	$+\mathbf{1.94} \pm 0.74$	-1.83	$+5.81$
FedExP	10	$+\mathbf{1.84} \pm 0.58$	$+0.12$	$+6.60$
FedACG	10	$+\mathbf{6.43} \pm 3.08$	-4.42	$+28.06$
FedRCL	10	$+\mathbf{1.51} \pm 0.40$	-0.08	$+3.48$
FedProx	10	$+\mathbf{2.44} \pm 0.87$	-1.65	$+6.25$
MOON	10	$+\mathbf{2.11} \pm 0.74$	-1.26	$+5.61$
Overall	90	$+\mathbf{2.23} \pm 0.41$	-1.25	$+7.61$

D.3 Heterogeneity: $\gamma = 0.1$

This setting uses a stronger non-IID split with $\gamma = 0.1$, where client-side label distributions are more skewed. CTS achieves a large average gain of $+8.36 \pm 1.08$ percentage points, showing that it is particularly beneficial under severe client heterogeneity.

Table 10: Test accuracy (%) for Base and +CTS under the heterogeneity setting $\gamma = 0.1$. Results are averaged over all seeds. Dataset abbreviations: Blood = BloodMNIST, Org-A/C/S = OrganAMNIST/OrganCMNIST/OrganSMNIST, Path = PathMNIST, FMNIST = Fashion-MNIST, C10/C100 = CIFAR-10/100, IN-100 = ImageNet-100, and Tiny = Tiny-ImageNet.

Method	Blood	Org-A	Org-C	Org-S	Path	FMNIST	C10	C100	IN-100	Tiny
FedAvg [15]	31.52	51.25	32.41	34.15	43.63	67.55	64.48	57.87	45.37	35.98
+ CTS	**67.92**	**61.93**	**39.98**	**39.27**	**47.77**	**75.94**	**70.04**	**59.94**	**47.92**	**38.45**
FedLC [28]	72.66	85.88	72.07	48.56	65.86	80.80	73.48	56.37	49.42	39.04
+ CTS	**77.17**	**86.09**	**72.97**	**61.34**	**66.69**	**84.03**	**75.70**	**57.35**	**50.41**	**39.83**
FedRS [14]	71.38	82.54	71.44	47.65	**65.84**	80.87	75.65	56.56	48.94	38.88
+ CTS	**75.49**	**86.31**	**74.11**	**61.47**	64.70	**84.36**	**76.50**	**58.15**	**50.03**	**39.82**
FedGM [22]	54.30	56.50	35.65	39.29	47.92	69.36	68.39	59.26	46.17	36.53
+ CTS	**73.44**	**65.92**	**56.72**	**44.54**	**51.85**	**78.22**	**72.04**	**60.37**	**48.35**	**38.72**
FedExP [7]	47.42	44.07	29.06	34.39	42.98	67.62	64.08	57.93	45.60	35.80
+ CTS	**69.74**	**62.01**	**50.50**	**38.74**	**46.11**	**76.95**	**69.45**	**60.35**	**47.87**	**38.76**
FedACG [9]	24.71	29.28	19.56	26.34	23.71	49.27	11.13	67.39	54.17	44.44
+ CTS	**45.99**	**58.12**	**31.23**	**37.76**	**46.02**	**59.79**	**68.39**	**68.33**	**57.68**	**46.89**
FedRCL [20]	33.32	62.88	35.80	**38.29**	46.45	71.32	57.51	57.48	44.24	33.99
+ CTS	**65.03**	**63.56**	**54.49**	36.03	**51.55**	**78.46**	**62.28**	**58.24**	**46.33**	**36.15**
FedProx [13]	35.56	55.66	33.55	32.94	44.22	69.34	65.06	59.03	45.95	35.82
+ CTS	**67.06**	**62.08**	**64.22**	**37.00**	**52.01**	**78.67**	**69.77**	**59.89**	**48.35**	**38.35**
MOON [11]	45.73	55.49	35.31	35.45	37.58	66.07	63.55	58.05	45.48	35.95
+ CTS	**70.45**	**61.10**	**65.46**	**38.90**	**50.62**	**76.11**	**68.19**	**60.00**	**48.37**	**38.67**

D.4 Heterogeneity: IID

This experiment evaluates the IID setting, where the statistical mismatch across clients is minimized. CTS still provides a small positive average gain of $+0.30 \pm 0.07$ percentage points, suggesting that it does not harm performance even in the easier homogeneous regime.

Table 11: Test accuracy (%) for Base and +CTS under IID data partitioning. Results are averaged over all seeds. Dataset abbreviations: Blood = BloodMNIST, Org-A/C/S = OrganAMNIST/OrganCMNIST/OrganSMNIST, Path = PathMNIST, FMNIST = Fashion-MNIST, C10/C100 = CIFAR-10/100, IN-100 = ImageNet-100, and Tiny = Tiny-ImageNet.

Method	Blood	Org-A	Org-C	Org-S	Path	FMNIST	C10	C100	IN-100	Tiny
FedAvg [15]	90.87	**92.29**	86.36	74.21	84.03	**91.97**	**90.16**	61.51	57.00	**43.24**
+ CTS	**91.72**	92.16	**87.25**	**74.24**	**86.51**	91.95	89.63	**62.08**	**57.01**	43.17
FedLC [28]	91.69	91.98	86.46	73.93	85.16	91.65	89.64	61.43	**57.14**	42.67
+ CTS	**92.28**	**92.04**	**87.26**	**74.34**	**85.93**	**91.88**	**89.88**	**62.02**	56.81	**42.80**
FedRS [14]	90.87	**92.29**	86.36	74.21	84.03	**91.97**	**90.16**	61.55	57.00	42.81
+ CTS	**91.72**	92.16	**87.25**	**74.24**	**86.51**	91.95	89.63	**62.39**	**57.01**	**42.96**
FedGM [22]	**91.93**	**92.01**	**87.11**	**74.33**	83.75	91.96	**90.02**	61.72	57.00	**43.41**
+ CTS	91.81	91.99	86.99	74.15	**86.00**	**92.00**	89.90	**61.86**	**57.36**	43.34
FedExP [7]	91.59	91.90	86.29	**74.52**	82.81	91.98	**90.01**	61.39	**56.67**	**43.09**
+ CTS	**92.19**	**91.99**	**87.02**	74.10	**83.67**	**92.00**	89.42	**61.82**	56.51	43.06
FedACG [9]	90.92	92.49	87.90	74.72	90.57	93.26	93.26	**72.43**	69.25	**55.30**
+ CTS	**92.46**	**92.77**	**87.96**	**76.00**	**90.80**	**93.28**	**93.48**	72.16	**69.52**	55.04
FedRCL [20]	**91.64**	**92.25**	86.31	**74.27**	85.70	91.84	**89.27**	61.38	56.24	42.16
+ CTS	91.51	92.15	**86.32**	73.90	**86.46**	**91.85**	89.18	**61.52**	**56.47**	**42.49**
FedProx [13]	90.91	92.12	**87.01**	**74.47**	84.08	**92.08**	89.38	61.14	**56.85**	**43.25**
+ CTS	**91.68**	**92.45**	86.95	73.78	**85.21**	91.91	**89.79**	**62.09**	56.40	42.98
MOON [11]	90.81	91.99	**86.80**	73.46	81.88	91.90	**89.99**	61.49	57.14	43.20
+ CTS	**91.96**	**92.09**	86.21	**74.01**	**84.60**	**92.03**	89.81	**62.58**	**57.42**	**43.46**

Table 12: Aggregated CTS gains in percentage points for IID data partitioning. Gains are computed as +CTS minus Base. *Avg. gain* is reported as mean $\pm$ SEM. For each group, *Min. gain* and *Max. gain* denote the minimum and maximum gain observed across the corresponding entries. In the final row, the *Min. gain* and *Max. gain* columns report the averages of group-wise minima and maxima.

By dataset

Dataset	n	Avg. gain $\pm$ SEM	Min. gain	Max. gain
Blood	9	$+\mathbf{0.68} \pm 0.18$	-0.13	$+1.54$
Org-A	9	$+\mathbf{0.05} \pm 0.06$	-0.13	$+0.33$
Org-C	9	$+\mathbf{0.29} \pm 0.18$	-0.58	$+0.89$
Org-S	9	$+\mathbf{0.07} \pm 0.20$	-0.69	$+1.29$
Path	9	$+\mathbf{1.52} \pm 0.32$	$+0.23$	$+2.72$
FMNIST	9	$+\mathbf{0.03} \pm 0.04$	-0.17	$+0.23$
C10	9	$-\mathbf{0.13} \pm 0.12$	-0.59	$+0.41$
C100	9	$+\mathbf{0.50} \pm 0.15$	-0.27	$+1.09$
IN-100	9	$+\mathbf{0.02} \pm 0.10$	-0.45	$+0.36$
Tiny	9	$+\mathbf{0.02} \pm 0.07$	-0.28	$+0.33$
Overall	90	$+\mathbf{0.30} \pm 0.07$	-0.31	$+0.92$

By algorithm

Algorithm	n	Avg. gain $\pm$ SEM	Min. gain	Max. gain
FedAvg	10	$+\mathbf{0.41} \pm 0.27$	-0.53	$+2.48$
FedLC	10	$+\mathbf{0.35} \pm 0.11$	-0.33	$+0.80$
FedRS	10	$+\mathbf{0.46} \pm 0.27$	-0.53	$+2.48$
FedGM	10	$+\mathbf{0.21} \pm 0.23$	-0.18	$+2.25$
FedExP	10	$+\mathbf{0.16} \pm 0.15$	-0.59	$+0.86$
FedACG	10	$+\mathbf{0.34} \pm 0.19$	-0.27	$+1.54$
FedRCL	10	$+\mathbf{0.08} \pm 0.10$	-0.37	$+0.75$
FedProx	10	$+\mathbf{0.19} \pm 0.20$	-0.69	$+1.14$
MOON	10	$+\mathbf{0.55} \pm 0.29$	-0.58	$+2.72$
Overall	90	$+\mathbf{0.30} \pm 0.07$	-0.45	$+1.67$

D.5 Local Epochs: 3

This setting uses three local epochs per communication round to test CTS when each client performs relatively short local updates. CTS improves average accuracy by $+3.26 \pm 0.48$ percentage points, showing that the method remains effective under limited local training.

Table 13: Test accuracy (%) for Base and +CTS with 3 local epochs. Results are averaged over all seeds. Dataset abbreviations: Blood = BloodMNIST, Org-A/C/S = OrganAMNIST/OrganCMNIST/OrganSMNIST, Path = PathMNIST, FMNIST = Fashion-MNIST, C10/C100 = CIFAR-10/100, IN-100 = ImageNet-100, and Tiny = Tiny-ImageNet.

Method	Blood	Org-A	Org-C	Org-S	Path	FMNIST	C10	C100	IN-100	Tiny
FedAvg [15]	68.08	82.63	81.65	60.66	67.60	81.60	79.99	58.85	48.34	36.98
+ CTS	**77.51**	**87.61**	**82.84**	**65.24**	**76.13**	**85.57**	**81.30**	**59.19**	**49.78**	**38.86**
FedLC [28]	78.01	88.09	**81.30**	**64.46**	77.60	86.70	81.04	57.15	49.31	38.50
+ CTS	**81.01**	**88.82**	80.80	63.89	**79.83**	**88.11**	**82.22**	**58.21**	**50.69**	**39.31**
FedRS [14]	79.24	87.15	79.75	63.21	75.24	86.97	81.60	56.83	49.95	38.55
+ CTS	**80.04**	**89.08**	**80.28**	**65.20**	**79.79**	**88.19**	**81.99**	**58.46**	**50.59**	**39.63**
FedGM [22]	72.74	85.19	82.50	64.14	69.12	82.31	80.75	58.50	48.54	37.73
+ CTS	**78.18**	**88.54**	**82.74**	**65.15**	**77.06**	**86.25**	**81.55**	**59.87**	**50.12**	**39.22**
FedExP [7]	71.44	80.35	81.05	63.56	67.87	81.42	79.83	58.16	48.08	36.96
+ CTS	**75.55**	**86.94**	**82.55**	**64.96**	**68.46**	**85.17**	**80.94**	**59.80**	**49.64**	**38.90**
FedACG [9]	53.09	67.02	75.18	30.91	62.98	79.32	87.51	71.07	62.76	50.51
+ CTS	**60.50**	**81.74**	**83.69**	**64.91**	**72.23**	**86.27**	**88.66**	**71.28**	**64.66**	**51.47**
FedRCL [20]	70.55	80.54	79.77	60.64	**63.70**	79.24	78.35	57.69	45.82	34.30
+ CTS	**76.00**	**85.72**	**81.00**	**64.61**	59.11	**84.17**	**79.43**	**58.84**	**47.99**	**36.01**
FedProx [13]	62.08	80.75	**82.76**	62.90	69.32	81.71	80.51	58.93	47.76	37.03
+ CTS	**78.42**	**87.64**	82.60	**65.49**	**75.58**	**85.62**	**81.23**	**59.47**	**49.47**	**38.64**
MOON [11]	71.32	81.53	**82.45**	62.25	69.92	80.87	80.03	59.01	48.53	36.90
+ CTS	**77.18**	**88.15**	82.00	**64.24**	**76.00**	**85.91**	**81.63**	**59.47**	**49.93**	**39.15**

Table 14: Aggregated CTS gains in percentage points for the 3-local-epoch setting. Gains are computed as +CTS minus Base. *Avg. gain* is reported as mean $\pm$ SEM. For each group, *Min. gain* and *Max. gain* denote the minimum and maximum gain observed across the corresponding entries. In the final row, the *Min. gain* and *Max. gain* columns report the averages of group-wise minima and maxima.

By dataset

Dataset	n	Avg. gain $\pm$ SEM	Min. gain	Max. gain
Blood	9	$+\mathbf{6.43} \pm 1.49$	$+0.80$	$+16.35$
Org-A	9	$+\mathbf{5.66} \pm 1.34$	$+0.73$	$+14.71$
Org-C	9	$+\mathbf{1.34} \pm 0.93$	-0.50	$+8.51$
Org-S	9	$+\mathbf{5.66} \pm 3.58$	-0.57	$+34.00$
Path	9	$+\mathbf{4.54} \pm 1.49$	-4.60	$+9.25$
FMNIST	9	$+\mathbf{3.90} \pm 0.59$	$+1.22$	$+6.94$
C10	9	$+\mathbf{1.04} \pm 0.12$	$+0.39$	$+1.61$
C100	9	$+\mathbf{0.93} \pm 0.19$	$+0.21$	$+1.63$
IN-100	9	$+\mathbf{1.53} \pm 0.14$	$+0.64$	$+2.17$
Tiny	9	$+\mathbf{1.52} \pm 0.16$	$+0.81$	$+2.25$
Overall	90	$+\mathbf{3.26} \pm 0.48$	-0.09	$+9.74$

By algorithm

Algorithm	n	Avg. gain $\pm$ SEM	Min. gain	Max. gain
FedAvg	10	$+\mathbf{3.77} \pm 1.00$	$+0.34$	$+9.43$
FedLC	10	$+\mathbf{1.07} \pm 0.34$	-0.57	$+3.00$
FedRS	10	$+\mathbf{1.48} \pm 0.39$	$+0.39$	$+4.55$
FedGM	10	$+\mathbf{2.72} \pm 0.77$	$+0.24$	$+7.95$
FedExP	10	$+\mathbf{2.42} \pm 0.58$	$+0.59$	$+6.60$
FedACG	10	$+\mathbf{8.50} \pm 3.19$	$+0.21$	$+34.00$
FedRCL	10	$+\mathbf{2.23} \pm 0.94$	-4.60	$+5.45$
FedProx	10	$+\mathbf{4.04} \pm 1.56$	-0.16	$+16.35$
MOON	10	$+\mathbf{3.09} \pm 0.81$	-0.45	$+6.61$
Overall	90	$+\mathbf{3.26} \pm 0.48$	-0.45	$+10.44$

D.6 Local Epochs: 10

This setting increases local training to ten epochs, allowing client models to move farther before aggregation. CTS still improves average accuracy by $+2.00 \pm 0.28$ percentage points over the available algorithms, indicating stable behavior under longer local updates.

Table 15: Test accuracy (%) for Base and +CTS with 10 local epochs. Results are averaged over all seeds. Dataset abbreviations: Blood = BloodMNIST, Org-A/C/S = OrganAMNIST/OrganCMNIST/OrganSMNIST, Path = PathMNIST, FMNIST = Fashion-MNIST, C10/C100 = CIFAR-10/100, IN-100 = ImageNet-100, and Tiny = Tiny-ImageNet.

Method	Blood	Org-A	Org-C	Org-S	Path	FMNIST	C10	C100	IN-100	Tiny
FedAvg [15]	76.33	89.30	84.57	65.08	79.98	84.47	83.15	64.37	54.85	40.96
+ CTS	**82.60**	**90.62**	**85.69**	**72.51**	**80.92**	**87.88**	**83.63**	**64.89**	**56.94**	**42.54**
FedLC [28]	84.52	91.15	84.39	71.93	86.61	89.39	88.08	63.20	56.07	42.18
+ CTS	**86.99**	**91.34**	**85.37**	**72.54**	**86.67**	**90.20**	**88.43**	**64.79**	**57.51**	**42.93**
FedRS [14]	85.41	91.22	85.02	71.74	**85.19**	89.11	88.25	63.90	56.41	42.40
+ CTS	**87.75**	**91.71**	**85.41**	**72.26**	84.69	**89.82**	**88.76**	**65.38**	**57.49**	**42.88**
FedGM [22]	81.72	90.52	84.37	70.72	81.70	86.13	84.67	64.89	54.97	41.32
+ CTS	**83.77**	**91.12**	**85.38**	**73.07**	**81.90**	**88.85**	**85.76**	**65.79**	**57.20**	**42.77**
FedExP [7]	65.78	88.81	83.65	66.32	**80.31**	84.72	82.85	64.57	55.07	41.04
+ CTS	**82.52**	**90.50**	**85.75**	**72.11**	78.92	**88.11**	**84.36**	**65.20**	**56.35**	**42.21**
FedRCL [20]	70.97	88.10	80.95	62.55	81.88	85.96	83.84	62.16	54.34	39.80
+ CTS	**76.23**	**89.24**	**83.88**	**71.70**	**84.66**	**89.24**	**84.45**	**62.84**	**55.56**	**41.14**
FedProx [13]	76.96	89.51	84.83	69.70	80.89	85.73	83.01	63.96	54.51	40.39
+ CTS	**82.92**	**90.79**	**85.56**	**72.10**	**81.15**	**88.63**	**84.62**	**64.84**	**55.92**	**42.00**
MOON [11]	76.78	89.12	85.00	66.40	**80.51**	84.80	83.37	64.05	55.12	41.26
+ CTS	**83.24**	**90.73**	**85.72**	**72.37**	80.24	**88.26**	**84.53**	**65.34**	**56.17**	**42.67**

Table 16: Aggregated CTS gains in percentage points for the 10-local-epoch setting, using the eight algorithms. Gains are computed as +CTS minus Base. *Avg. gain* is reported as mean ± SEM. For each group, *Min. gain* and *Max. gain* denote the minimum and maximum gain observed across the corresponding entries. In the final row, the *Min. gain* and *Max. gain* columns report the averages of group-wise minima and maxima.

By dataset

Dataset	n	Avg. gain ± SEM	Min. gain	Max. gain
Blood	8	$+\mathbf{5.94} \pm 1.68$	$+2.04$	$+16.74$
Org-A	8	$+\mathbf{1.04} \pm 0.19$	$+0.19$	$+1.68$
Org-C	8	$+\mathbf{1.25} \pm 0.30$	$+0.39$	$+2.93$
Org-S	8	$+\mathbf{4.28} \pm 1.15$	$+0.52$	$+9.15$
Path	8	$+\mathbf{0.26} \pm 0.43$	-1.40	$+2.79$
FMNIST	8	$+\mathbf{2.59} \pm 0.41$	$+0.71$	$+3.45$
C10	8	$+\mathbf{0.91} \pm 0.17$	$+0.35$	$+1.61$
C100	8	$+\mathbf{1.00} \pm 0.14$	$+0.52$	$+1.58$
IN-100	8	$+\mathbf{1.47} \pm 0.16$	$+1.05$	$+2.22$
Tiny	8	$+\mathbf{1.22} \pm 0.14$	$+0.48$	$+1.61$
Overall	80	$+\mathbf{2.00} \pm 0.28$	$+0.49$	$+4.38$

By algorithm

Algorithm	n	Avg. gain ± SEM	Min. gain	Max. gain
FedAvg	10	$+\mathbf{2.52} \pm 0.77$	$+0.49$	$+7.43$
FedLC	10	$+\mathbf{0.92} \pm 0.23$	$+0.05$	$+2.47$
FedRS	10	$+\mathbf{0.75} \pm 0.24$	-0.51	$+2.34$
FedGM	10	$+\mathbf{1.46} \pm 0.26$	$+0.19$	$+2.72$
FedExP	10	$+\mathbf{3.29} \pm 1.60$	-1.40	$+16.74$
FedRCL	10	$+\mathbf{2.84} \pm 0.84$	$+0.61$	$+9.15$
FedProx	10	$+\mathbf{1.91} \pm 0.51$	$+0.26$	$+5.95$
MOON	10	$+\mathbf{2.28} \pm 0.72$	-0.27	$+6.46$
Overall	80	$+\mathbf{2.00} \pm 0.28$	-0.07	$+6.66$

D.7 Total Clients: 50

This experiment reduces the total client population to 50 to evaluate CTS in a smaller federation. CTS obtains an average gain of $+3.14 \pm 0.37$ percentage points, showing that the proposed strategy remains useful when fewer clients participate in the overall system.

Table 17: Test accuracy (%) for Base and +CTS with 50 total clients. Results are averaged over all seeds. Dataset abbreviations: Blood = BloodMNIST, Org-A/C/S = OrganAMNIST/OrganCMNIST/OrganSMNIST, Path = PathMNIST, FMNIST = Fashion-MNIST, C10/C100 = CIFAR-10/100, IN-100 = ImageNet-100, and Tiny = Tiny-ImageNet.

Method	Blood	Org-A	Org-C	Org-S	Path	FMNIST	C10	C100	IN-100	Tiny
FedAvg [15]	65.86	85.99	76.66	52.48	69.25	84.59	79.17	66.55	58.06	45.09
+ CTS	**76.29**	**89.28**	**83.67**	**64.59**	**71.85**	**86.56**	**81.53**	**67.90**	**59.79**	**47.44**
FedLC [28]	84.47	89.38	82.98	65.40	**81.20**	89.00	86.84	67.21	60.44	47.59
+ CTS	**87.46**	**90.16**	**84.41**	**68.86**	81.15	**89.66**	**87.62**	**67.47**	**61.44**	**48.37**
FedRS [14]	84.33	89.57	80.21	66.07	**81.97**	88.54	86.93	66.61	60.30	47.42
+ CTS	**86.85**	**90.28**	**82.56**	**67.66**	80.33	**89.50**	**88.02**	**67.65**	**61.71**	**48.54**
FedGM [22]	69.64	88.22	82.05	54.53	**71.97**	86.41	81.43	67.41	58.95	45.94
+ CTS	**78.50**	**89.57**	**83.63**	**65.69**	71.65	**86.88**	**82.76**	**68.06**	**60.48**	**47.41**
FedExP [7]	67.98	86.05	75.31	53.95	70.80	84.68	79.53	66.71	57.84	44.92
+ CTS	**75.11**	**89.57**	**83.30**	**64.70**	**72.52**	**86.57**	**81.12**	**67.54**	**60.52**	**46.83**
FedACG [9]	55.02	77.52	60.20	42.51	69.84	80.30	71.41	69.28	62.40	49.41
+ CTS	**71.11**	**86.20**	**75.34**	**47.94**	**74.24**	**85.51**	**74.94**	**69.66**	**63.54**	**50.86**
FedRCL [20]	71.10	85.04	75.98	54.01	**73.21**	86.91	79.98	65.77	57.88	44.69
+ CTS	**75.83**	**88.78**	**81.17**	**54.14**	71.90	**87.52**	**83.62**	**66.05**	**59.39**	**45.56**
FedProx [13]	70.63	87.12	79.27	50.64	71.33	85.74	79.83	66.78	58.06	45.14
+ CTS	**74.61**	**89.60**	**84.06**	**62.67**	**73.56**	**87.37**	**80.63**	**67.76**	**60.10**	**47.12**
MOON [11]	**74.66**	87.28	75.32	53.76	68.62	84.94	78.56	67.13	58.71	45.79
+ CTS	74.14	**89.79**	**83.42**	**61.46**	**74.23**	**87.36**	**81.26**	**67.90**	**60.28**	**47.26**

Table 18: Aggregated CTS gains in percentage points for the 50-total-client setting. Gains are computed as +CTS minus Base. *Avg. gain* is reported as mean $\pm$ SEM. For each group, *Min. gain* and *Max. gain* denote the minimum and maximum gain observed across the corresponding entries. In the final row, the *Min. gain* and *Max. gain* columns report the averages of group-wise minima and maxima.

By dataset

Dataset	n	Avg. gain $\pm$ SEM	Min. gain	Max. gain
Blood	9	$+\mathbf{6.24} \pm 1.67$	-0.53	$+16.09$
Org-A	9	$+\mathbf{3.00} \pm 0.80$	$+0.70$	$+8.67$
Org-C	9	$+\mathbf{5.95} \pm 1.44$	$+1.44$	$+15.14$
Org-S	9	$+\mathbf{7.15} \pm 1.56$	$+0.13$	$+12.11$
Path	9	$+\mathbf{1.47} \pm 0.84$	-1.64	$+5.62$
FMNIST	9	$+\mathbf{1.76} \pm 0.49$	$+0.47$	$+5.21$
C10	9	$+\mathbf{1.98} \pm 0.37$	$+0.78$	$+3.64$
C100	9	$+\mathbf{0.73} \pm 0.12$	$+0.26$	$+1.35$
IN-100	9	$+\mathbf{1.62} \pm 0.17$	$+1.00$	$+2.69$
Tiny	9	$+\mathbf{1.49} \pm 0.17$	$+0.79$	$+2.35$
Overall	90	$+\mathbf{3.14} \pm 0.37$	$+0.34$	$+7.29$

By algorithm

Algorithm	n	Avg. gain $\pm$ SEM	Min. gain	Max. gain
FedAvg	10	$+\mathbf{4.52} \pm 1.24$	$+1.35$	$+12.11$
FedLC	10	$+\mathbf{1.21} \pm 0.36$	-0.05	$+3.46$
FedRS	10	$+\mathbf{1.11} \pm 0.36$	-1.64	$+2.53$
FedGM	10	$+\mathbf{2.81} \pm 1.23$	-0.32	$+11.16$
FedExP	10	$+\mathbf{4.00} \pm 1.07$	$+0.84$	$+10.74$
FedACG	10	$+\mathbf{6.14} \pm 1.76$	$+0.39$	$+16.09$
FedRCL	10	$+\mathbf{1.94} \pm 0.70$	-1.31	$+5.19$
FedProx	10	$+\mathbf{3.29} \pm 1.05$	$+0.80$	$+12.03$
MOON	10	$+\mathbf{3.23} \pm 0.92$	-0.53	$+8.10$
Overall	90	$+\mathbf{3.14} \pm 0.37$	-0.05	$+9.04$

D.8 Total Clients: 150

This experiment increases the total client population to 150 to test whether CTS scales to a larger federation. CTS achieves a positive average gain of $+1.80 \pm 0.26$ percentage points, indicating that its benefit remains consistent as the client pool grows.

Table 19: Test accuracy (%) for Base and +CTS with 150 total clients. Results are averaged over all seeds. Dataset abbreviations: Blood = BloodMNIST, Org-A/C/S = OrganAMNIST/OrganCMNIST/OrganSMNIST, Path = PathMNIST, FMNIST = Fashion-MNIST, C10/C100 = CIFAR-10/100, IN-100 = ImageNet-100, and Tiny = Tiny-ImageNet.

Method	Blood	Org-A	Org-C	Org-S	Path	FMNIST	C10	C100	IN-100	Tiny
FedAvg [15]	79.39	89.06	**83.71**	66.46	62.59	85.67	80.12	55.82	47.72	35.68
+ CTS	**86.94**	**90.18**	82.16	**67.39**	**68.56**	**87.43**	**81.02**	**57.08**	**49.19**	**37.05**
FedLC [28]	88.17	**89.41**	80.74	64.97	**81.05**	87.88	81.70	54.05	48.53	36.13
+ CTS	**89.08**	89.39	**80.85**	**66.77**	80.07	**88.70**	**82.97**	**55.16**	**49.49**	**36.38**
FedRS [14]	87.55	**89.49**	80.35	**66.78**	79.83	87.88	81.47	54.81	48.92	36.45
+ CTS	**88.68**	89.17	**81.57**	66.51	**81.92**	**88.63**	**83.11**	**55.16**	**49.40**	**37.13**
FedGM [22]	85.97	89.37	81.96	67.24	65.08	85.96	81.07	55.26	48.08	36.40
+ CTS	**86.30**	**90.40**	**83.32**	**68.37**	**70.92**	**87.82**	**81.79**	**56.54**	**49.46**	**37.04**
FedExP [7]	80.73	88.58	81.57	65.87	58.64	85.53	78.80	55.04	47.48	36.00
+ CTS	**83.18**	**90.02**	**82.27**	**67.16**	**63.58**	**87.44**	**80.60**	**56.89**	**49.13**	**36.80**
FedACG [9]	82.43	83.50	68.81	51.03	73.25	84.06	88.47	70.04	63.24	50.24
+ CTS	**88.30**	**90.29**	**83.38**	**64.36**	**75.23**	**87.37**	**90.36**	**70.60**	**64.95**	**50.77**
FedRCL [20]	79.20	86.39	**81.00**	66.38	65.87	85.09	79.11	54.95	46.90	34.35
+ CTS	**80.98**	**88.93**	79.22	**67.07**	**67.37**	**86.98**	**80.06**	**55.13**	**47.24**	**35.65**
FedProx [13]	82.45	89.08	81.98	66.49	64.96	85.79	79.67	54.73	47.50	36.12
+ CTS	**85.36**	**90.05**	**83.12**	**67.91**	**70.51**	**87.83**	**81.02**	**56.45**	**49.16**	**36.86**
MOON [11]	82.27	89.13	82.69	66.25	64.49	85.76	80.01	55.16	48.12	35.73
+ CTS	**84.29**	**89.99**	**82.75**	**68.52**	**66.69**	**87.29**	**80.54**	**56.73**	**48.90**	**37.03**

Table 20: Aggregated CTS gains in percentage points for the 150-total-client setting. Gains are computed as +CTS minus Base. *Avg. gain* is reported as mean ± SEM. For each group, *Min. gain* and *Max. gain* denote the minimum and maximum gain observed across the corresponding entries. In the final row, the *Min. gain* and *Max. gain* columns report the averages of group-wise minima and maxima.

By dataset

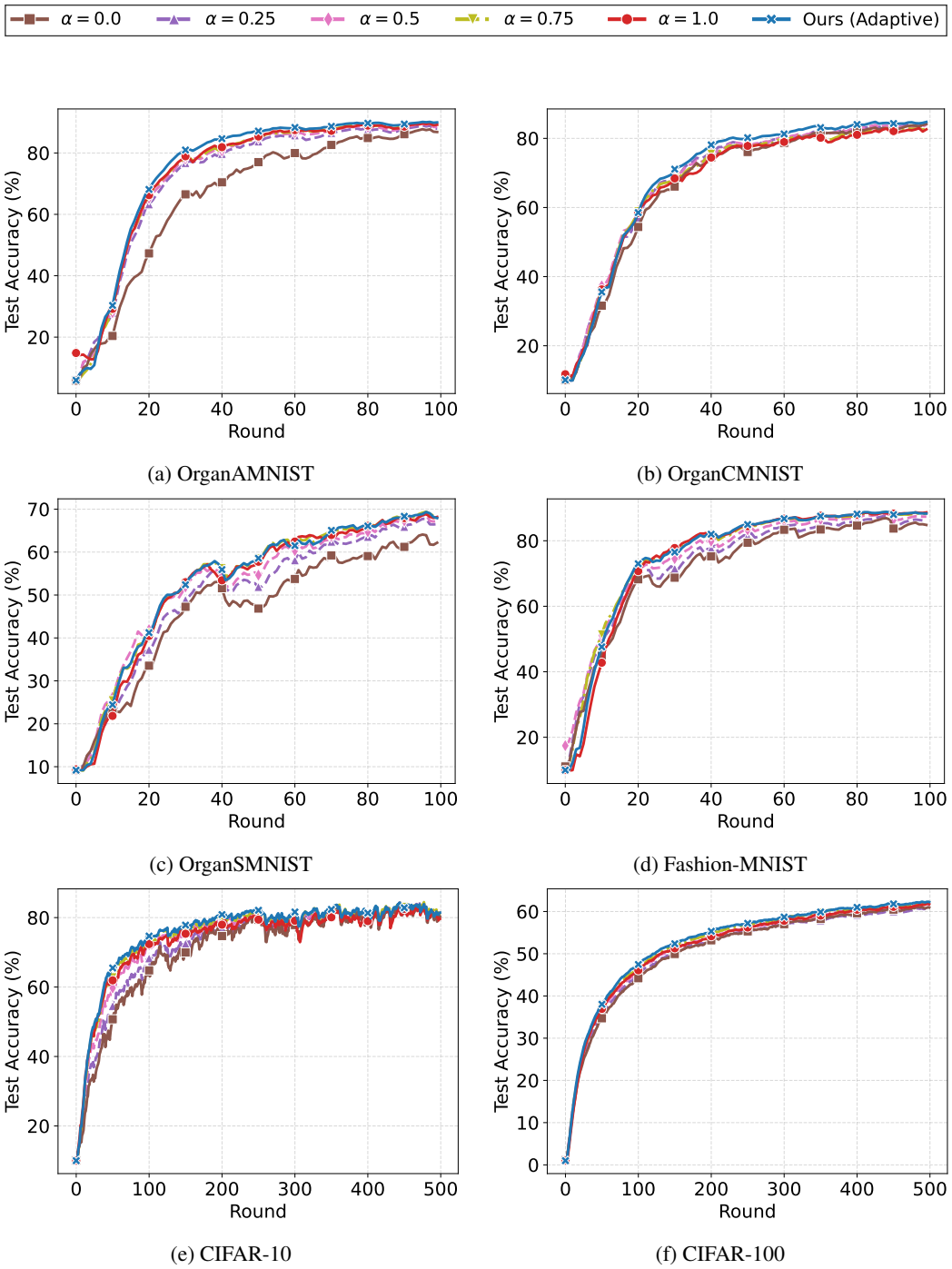
Dataset	n	Avg. gain ± SEM	Min. gain	Max. gain
Blood	9	$+\mathbf{2.77} \pm 0.80$	$+0.33$	$+7.55$
Org-A	9	$+\mathbf{1.60} \pm 0.70$	-0.32	$+6.78$
Org-C	9	$+\mathbf{1.76} \pm 1.65$	-1.77	$+14.57$
Org-S	9	$+\mathbf{2.51} \pm 1.37$	-0.27	$+13.34$
Path	9	$+\mathbf{3.23} \pm 0.81$	-0.98	$+5.97$
FMNIST	9	$+\mathbf{1.76} \pm 0.25$	$+0.75$	$+3.31$
C10	9	$+\mathbf{1.23} \pm 0.16$	$+0.53$	$+1.89$
C100	9	$+\mathbf{1.10} \pm 0.20$	$+0.18$	$+1.86$
IN-100	9	$+\mathbf{1.16} \pm 0.18$	$+0.34$	$+1.71$
Tiny	9	$+\mathbf{0.85} \pm 0.13$	$+0.25$	$+1.37$
Overall	90	$+\mathbf{1.80} \pm 0.26$	-0.10	$+5.83$

By algorithm

Algorithm	n	Avg. gain ± SEM	Min. gain	Max. gain
FedAvg	10	$+\mathbf{2.08} \pm 0.84$	-1.55	$+7.55$
FedLC	10	$+\mathbf{0.62} \pm 0.25$	-0.98	$+1.80$
FedRS	10	$+\mathbf{0.78} \pm 0.24$	-0.32	$+2.09$
FedGM	10	$+\mathbf{1.56} \pm 0.49$	$+0.33$	$+5.83$
FedExP	10	$+\mathbf{1.88} \pm 0.38$	$+0.69$	$+4.94$
FedACG	10	$+\mathbf{5.05} \pm 1.62$	$+0.53$	$+14.57$
FedRCL	10	$+\mathbf{0.94} \pm 0.38$	-1.77	$+2.54$
FedProx	10	$+\mathbf{1.95} \pm 0.44$	$+0.73$	$+5.55$
MOON	10	$+\mathbf{1.31} \pm 0.24$	$+0.06$	$+2.26$
Overall	90	$+\mathbf{1.80} \pm 0.26$	-0.25	$+5.24$

E Additional results for ablation study

We provide additional fixed-shrinkage ablation results on six additional datasets. The best fixed α varies across datasets, showing that the constant term should not be uniformly preserved or removed. These results further support the adaptive shrinkage rule of FedCTS.



(a) OrganAMNIST

(b) OrganCMNIST

(c) OrganSMNIST

(d) Fashion-MNIST

(e) CIFAR-10

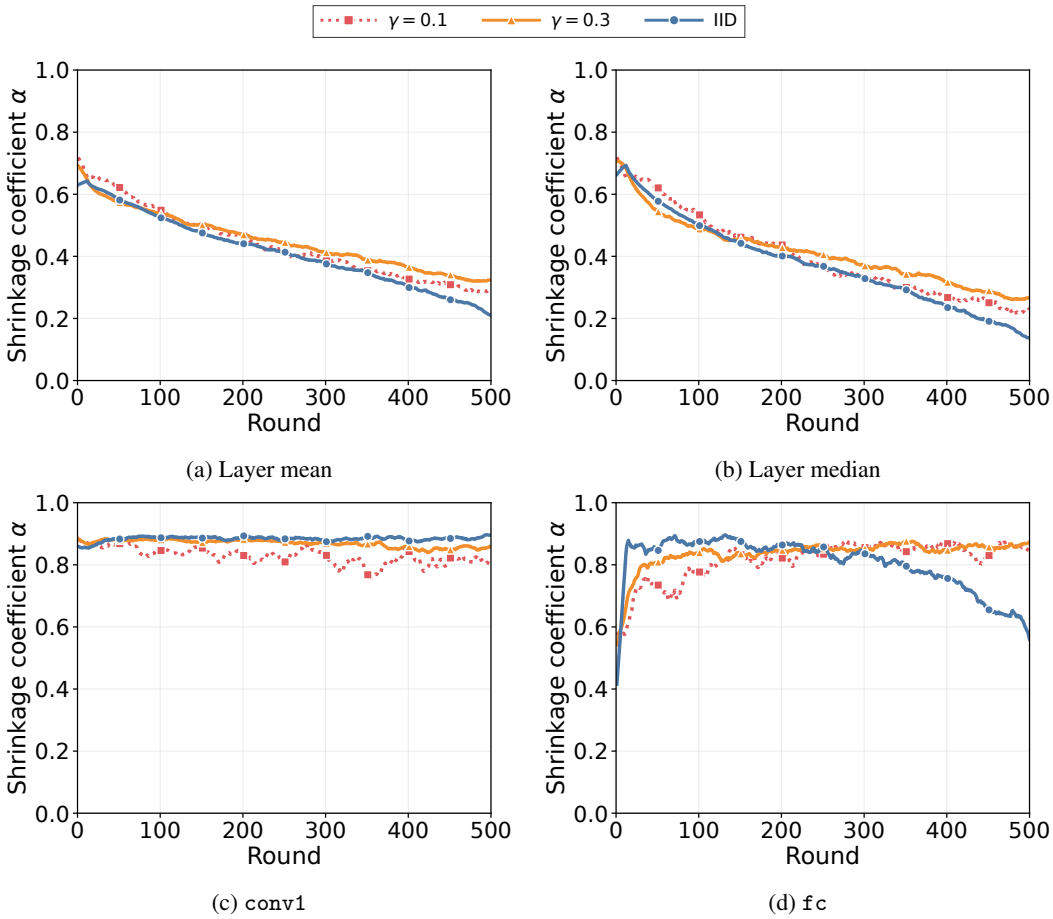
(f) CIFAR-100

Figure 9: Additional ablation results for fixed shrinkage coefficients. The optimal fixed α varies across datasets, supporting the need for adaptive shrinkage in FedCTS.

E.1 Adaptive shrinkage coefficients

We report the shrinkage coefficients recorded during FedCTS training to examine how the correction changes across layers and communication rounds. Figs. 10 and 11 show the layer mean, layer median, first convolutional layer, and final classifier. For $\gamma = 0.3$, we first average the coefficient of each layer over three seeds and then compute the layer mean and median. The IID and $\gamma = 0.1$ results use one seed. The coefficients are computed with the noise correction described in the experimental setup.

The layer mean and median generally decrease as training proceeds, while `conv1` and `fc` often retain larger coefficients. This shows that the required shrinkage differs across layers and changes during training. A large coefficient in one layer does not by itself imply a large change to the full update, since the correction also depends on the magnitude of the constant term in that layer.



(a) Layer mean

(b) Layer median

(c) `conv1`

(d) `fc`

Figure 10: Adaptive shrinkage coefficients during FedCTS training on CIFAR-10. The coefficients vary across layers and communication rounds.

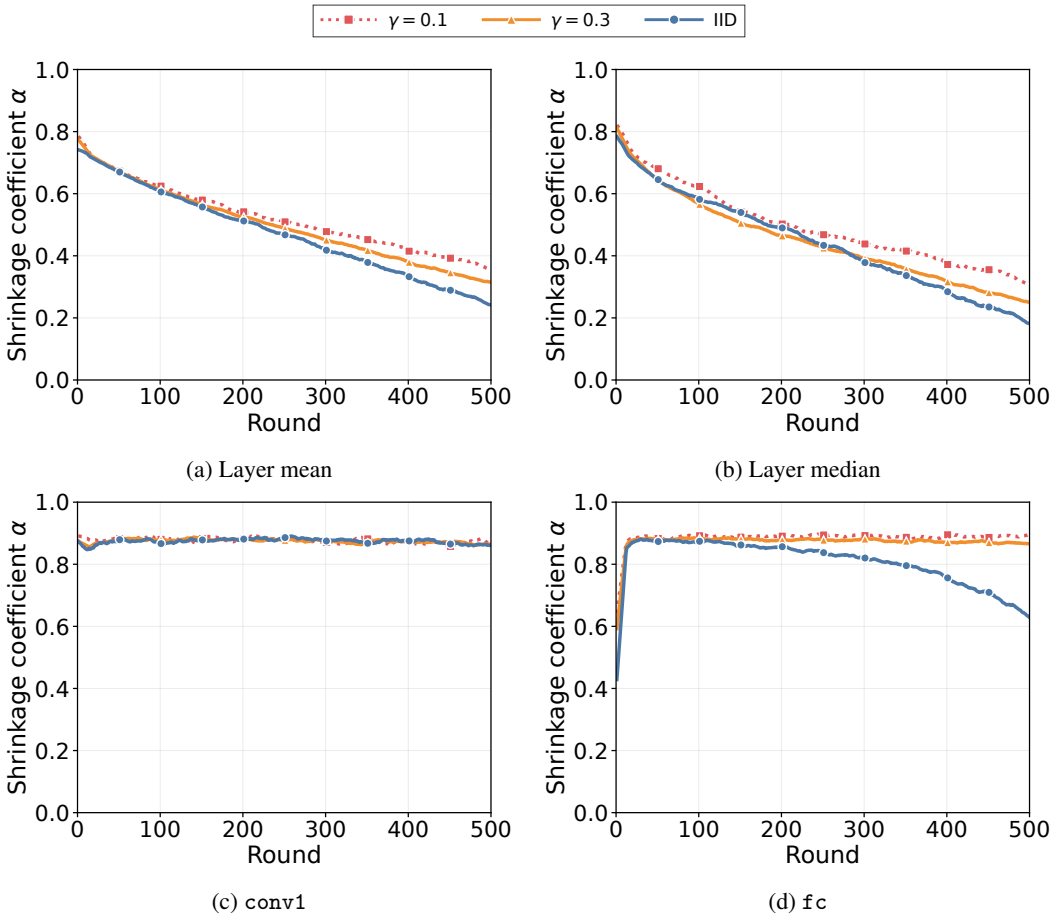
E.2 Relative correction magnitude

We additionally measure the relative squared magnitude of the correction:

$$C^t = \frac{\sum_\ell (\alpha_\ell^t)^2 U_{\ell,\mathrm{con}}^t}{\sum_\ell U_\ell^t},$$

Here, α_ℓ^t is computed from the received FedAvg updates using Eq. (17) with the noise correction described in the experimental setup. Thus, this analysis measures the correction that FedCTS would make to those updates, rather than the correction along a FedCTS training trajectory. The sums include the same 21 weight layers as the drift analysis.

Fig. 12 shows that the correction is not fixed across communication rounds or data partitions. It remains smaller under IID data, whereas stronger heterogeneity generally leads to a larger correction. The constant update ratio



(a) Layer mean

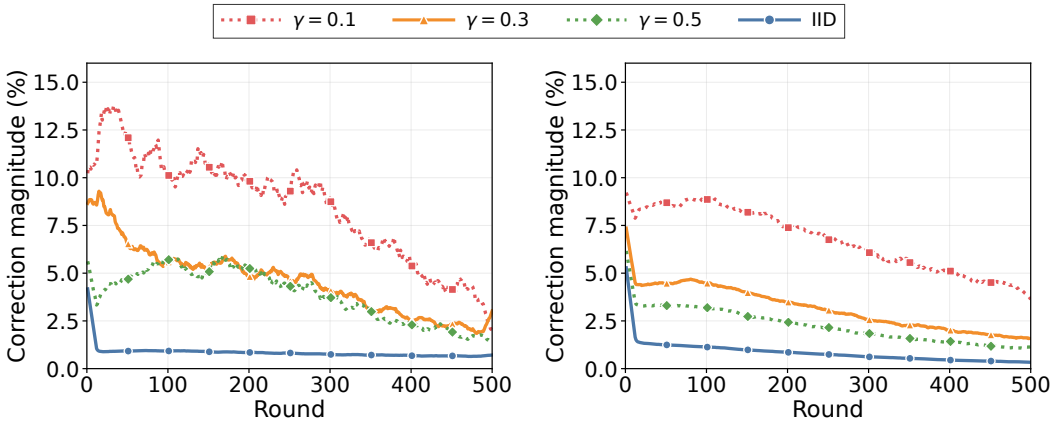
(b) Layer median

(c) `conv1`

(d) `fc`

Figure 11: Adaptive shrinkage coefficients during FedCTS training on CIFAR-100. The coefficients vary across layers and communication rounds.

therefore does not directly specify how much of the update is changed. FedCTS adjusts the correction according to both the shrinkage coefficient and the magnitude of the constant term.



(a) CIFAR-10

(b) CIFAR-100

Figure 12: Relative squared correction magnitude evaluated on FedAvg updates. The correction varies across communication rounds and is generally larger under stronger data heterogeneity.

F Limitations

FedCTS is evaluated primarily on image classification benchmarks under Dirichlet label skew. Its behavior under feature shift, temporal drift, and language tasks remains to be studied.

Non-uniform client participation. Our experiments use uniform client sampling. With persistent differences in availability, the global model may under-represent rarely participating clients. FedCTS computes $\hat{\mu}_\ell^t$, M_ℓ^t, and V_ℓ^t from S_t alone and does not use participation history. Thus, apparent agreement within S_t need not represent the full client population, and the correction does not compensate for unequal participation frequencies. Incorporating participation frequency into aggregation weights or reliability estimation remains a direction for future work.

Broader Impact

FedCTS is a general server-side aggregation method for federated learning. Its potential positive impact is to improve the robustness and efficiency of collaborative training when data cannot be centralized, which may benefit privacy-sensitive and resource-constrained applications. However, FedCTS itself does not provide formal privacy, fairness, or security guarantees. If applied to sensitive domains, it should be combined with appropriate privacy-preserving protocols, fairness evaluation, and deployment-specific safeguards. Since this work uses standard image classification benchmarks and does not release high-risk data or models, we do not identify additional misuse risks beyond those of general federated learning systems.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The claims match the theoretical and experimental results. The abstract, introduction, and conclusion accurately reflect the contributions and scope of the work.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: We discuss the limitations of the work.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

   Answer: [Yes]

   Justification: We provide the full set of assumptions and proofs in the paper, especially in the technical appendices.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.

- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: We provide full experimental hyperparameters, environments, code, and data to support reproducibility.

   Guidelines:

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

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

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

   Answer: [Yes]

   Justification: We provide the dataset information (publicly available) and code in supplemental materials.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).

- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: We provide the training and test details in the experimental settings section and appendices.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [Yes]

   Justification: We report mean $\pm$ SEM for the main aggregated gains. However, we skip it in the ablation study due to the training time.

   Guidelines:

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: We provide the computer resources needed to reproduce the experiments in appendices.

   Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: We follow the Code of ethics.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: We discuss the potential positive societal impacts and there may not be negative societal impacts.

    Guidelines:

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

11. **Safeguards**

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

    Answer: [N/A]

    Justification: We have not used data or models that have a high risk for misuse.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.

- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: We properly cited following the license of datasets, codes, and papers.

    Guidelines:

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

13. **New assets**

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

    Answer: [N/A]

    Justification: We do not introduce new assets in this paper.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

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

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

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

Answer: [N/A]

Justification: This paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

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

16. **Declaration of LLM usage**

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

Answer: [N/A]

Justification: There is no need to be described about LLM usage following the NeurIPS handbook.

Guidelines:

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