

# FedBalanceTTA – Federated Learning with Balanced Test Time Adaptation

Md Akil Raihan Iftee<sup>1</sup>, Syed Md. Ahnaf Hasan<sup>1</sup>, Mir Sazzat Hossain<sup>1</sup>, Rakibul Hasan Rajib<sup>1</sup>,  
Amin Ahsan Ali<sup>1</sup>, AKM Mahbubur Rahman<sup>1</sup>, Sajib Mistry<sup>2</sup>, Monowar Bhuyan<sup>3</sup>

<sup>1</sup>Center for Computational & Data Sciences (CCDS)

Independent University, Bangladesh

<sup>2</sup>Curtin University, Australia

<sup>3</sup>Umeå University, Sweden

## Abstract

Federated learning (FL) enables collaborative model training across decentralized clients while preserving the privacy of local data. Class imbalance remains a fundamental challenge in FL, where rare but critical classes are often severely underrepresented in individual client datasets. Although prior work has addressed class imbalance during training through reliable aggregation and local class distribution alignment, these methods typically rely on access to labeled data or coordination among clients, and none are designed for unsupervised adaptation during inference under federated constraints. We propose a new approach called pFedBBN, a personalized federated test-time adaptation framework that incorporates balanced batch normalization (BBN) to mitigate prediction bias by treating all classes equally during feature normalization, while also enabling a weighted collaboration of clients based on their BBN statistics, which indicates that clients from similar domains prioritize each other more during adaptation. The method supports fully unsupervised local adaptation and introduces a class-aware model aggregation strategy that enables personalized inference without compromising privacy. It effectively addresses both distribution shifts and class imbalance through balanced feature normalization and domain-aware collaboration, without requiring any labeled or raw data from clients. Experiments on corrupted versions of CIFAR-10 and CIFAR-100 demonstrate that pFedBBN significantly improves robustness and performance on minority classes compared to existing federated and test-time adaptation baselines.

## Introduction

Federated Learning (FL) enables decentralized training across a network of clients, such as smartphones, hospitals, or IoT devices—without sharing raw data. This is critical in privacy-sensitive domains like mobile computing, health-care, and smart environments (McMahan et al. 2017; Chen et al. 2025; Noble, Bellet, and Dieuleveut 2022; Liu et al. 2024). However, data in FL is often non-identically distributed (non-IID), evolves over time, and suffers from severe class imbalance, making reliable inference particularly challenging.

Test-Time Adaptation (TTA) aims to enhance generalization under distribution shifts by adapting models using un-

labeled test inputs. It is necessary in dynamic environments where client data distributions continually change. However, adapting from unlabeled data can lead to error and catastrophic forgetting (Niu et al. 2022), especially when adaptation occurs continuously in real-world settings.

In FL, TTA is even more challenging due to the presence of heterogeneous client domains and the absence of labeled test data. Further, class imbalance (Xiao and Wang 2021; Wang et al. 2021b; Seol and Kim 2023) is a fundamental issue: dominant classes often overshadow rare but critical ones (e.g., emergency alerts or abnormal health readings), degrading model performance on minority classes.

Existing class imbalance mitigation techniques, such as resampling (Khushi et al. 2021), data augmentation (Duan et al. 2020), or cost-sensitive losses (Sarkar, Narang, and Rai 2020; Khan et al. 2017) which are ill-suited for FL. These methods assume access to raw data, which violates privacy, or fail when applied locally without coordination. FL-specific solutions often require proxy servers (Huang et al. 2016), auxiliary models (Wang, Ramanan, and Hebert 2017), or data exchange, introducing overhead and compromising privacy. Then, a paper (Wang et al. 2021b) proposed a federated learning approach to mitigate class imbalance by adjusting training dynamics using labeled data across clients. However, their method is designed for the training phase and does not address class imbalance in the test-time adaptation setting.

In TTA, the challenges compound. Without labels, correcting for both domain shifts and class imbalance becomes significantly harder. Batch Normalization (BN) statistics, commonly used in TTA, become biased (You, Li, and Zhao 2021) due to dominant classes, resulting in degraded adaptation. Unsupervised adaptation exacerbates the problem, as both error accumulation and imbalanced normalization persist without the availability of ground truth. The recently proposed FedCTTA (Rajib et al. 2025) addresses domain adaptation in Federated Test-Time Adaptation (TTA) through a collaborative continual adaptation strategy. However, it does not explicitly tackle the challenge of class imbalance, which remains an open problem in TTA for FL settings.

To address these issues, we propose **pFedBBN**, the first framework that tackles class imbalance in federated test-time adaptation. Our method enables clients to adapt models

using local unlabeled data while leveraging domain-similar peers for personalized model aggregation. A new *Class-Wise Adaptive Normalization* (CWAN) module tracks per-class feature statistics with pseudo-labels and interpolates them with batch-level statistics to mitigate imbalance. Further, a *confidence-filtered self-distillation* approach selectively updates the model using pseudo-labeled data. Only the BN affine parameters are updated to preserve generalization. Clients then share BN statistics to infer domain similarity and receive personalized models via similarity-aware aggregation, all under strict privacy constraints.

In summary, the key contributions of this work are:

- We propose **pFedBBN**, the first framework specifically designed to address *class imbalance* in federated test-time adaptation, where clients adapt models using unlabeled, locally available test data under domain and class distribution shifts.
- We introduce a **Class-Wise Adaptive Normalization** (CWAN) module that maintains *per-class feature statistics* using pseudo-labels. By interpolating these with batch-level statistics, CWAN mitigates the bias introduced by dominant classes during adaptation without accessing ground-truth labels.
- To reduce the impact of noisy pseudo-labels, we apply a **confidence-based filtering** mechanism that selectively updates the model using only high-confidence pseudo-labeled samples via self-distillation, effectively minimizing error accumulation and catastrophic forgetting.
- We conduct comprehensive experiments on non-IID, class-imbalanced, and domain-shifted settings, demonstrating the effectiveness of pFedBBN over state-of-the-art FL and TTA baselines.

## Related Work

### Class Imbalance in Federated Learning

Federated Learning (FL) often encounters challenges due to non-IID data distributions and class imbalance across clients. To address these issues, several methods have been proposed: Fed-GraB (Xiao et al. 2023) introduces a self-adjusting gradient balancer that re-weights each client’s gradient based on a global long-tail prior, enhancing minority-class accuracy. GBME (Zeng et al. 2023) constructs class-prior proxies from accumulated client gradients, enabling global loss re-balancing without compromising data privacy.

In (Shang et al. 2022), CReFF identifies biased classifiers as a primary factor for poor performance under heterogeneous long-tailed data. It re-trains the classifier on federated features, achieving performance comparable to centrally re-trained models. BalanceFL (Xian et al. 2022) addresses client imbalance through a local update scheme, ensuring each client’s model behaves as if trained on a uniform class distribution.

FL-FCR employs calibration weighting and resampling, integrating a calibrated loss during client training and calibration-based resampling at the server, aligning training with true class frequencies (Kang, Lim, and Im 2024). Li et al. (Li and Others 2024) propose a probability-corrected loss

and shared class prototypes to align model outputs across heterogeneous clients, effectively handling both balanced and imbalanced global data.

### Test-Time Adaptation

Test-Time Adaptation (TTA) methods adapt pre-trained models during inference to handle domain shifts or corrupted inputs:

TENT performs entropy minimization on each test batch, updating batch normalization statistics and channel-wise affine parameters online to maximize output confidence (Wang et al. 2021a). CoTTA maintains a weight-averaged teacher model and applies strong augmentation to stabilize pseudo-labels. It also stochastically resets a fraction of parameters to the source pre-trained values to prevent catastrophic forgetting (Wang et al. 2022a). RoTTA introduces a Practical TTA setting with temporally changing distributions and correlated sampling. It incorporates robust batch normalization, a category-balanced memory bank, and a time-aware reweighting teacher-student model for adaptation (Yuan, Xie, and Li 2023a).ROID addresses challenges of universal online TTA by applying diversity weighting, continuously ensembling the source and adapted models, and performing adaptive prior correction on predictions (Marsden, Döbler, and Yang 2024).

### Bridging TTA with federated or distributed learning

ATP formalizes Test-Time Personalized Federated Learning (TTPFL), where each client locally adapts the global model without labels. It learns adaptive update rates per network module based on cross-client shift information (Bao et al. 2023).

TTA-FedDG leverages test-time adaptation for federated domain generalization. It mixes features via fast reordering during local training and applies a contrastive teacher-student scheme with selective strong pseudo-labeling at test time (Liang and Others 2024). Shao et al. build a federated face anti-spoofing system where clients collaboratively train a general model, and each device minimizes the model’s prediction entropy on its new attack data during test time (Shao and Others 2021).

These methods collectively advance the robustness and adaptability of FL models in the presence of class imbalance and domain shifts.

## Methodology

In real-world FL, clients face non-stationary, heterogeneous data that diverges from the source model. To ensure robust, personalized inference, we propose a unified framework that enables local test-time adaptation on unlabeled data and collaboration with similar clients via distance-aware aggregation. Our method includes three components: unsupervised local adaptation using class-wise normalization and confidence distillation, similarity estimation via batch normalization statistics, and personalized aggregation based on domain alignment.

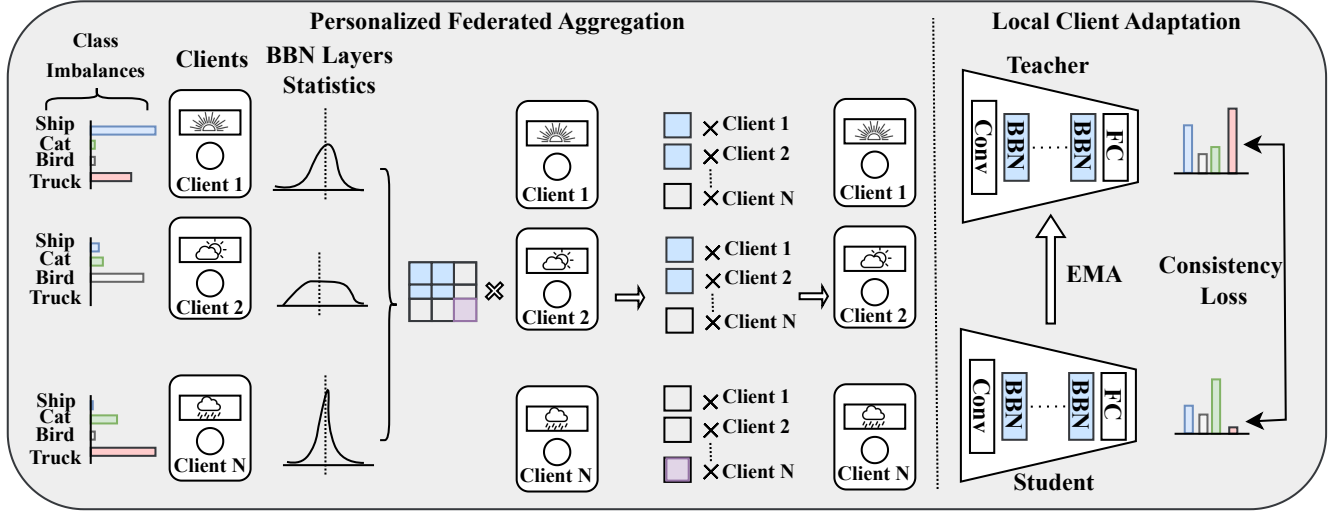


Figure 1: The overall framework of pFedBBN where each client performs unsupervised local adaptation using class-wise balanced batch normalization (BBN) and confidence-filtered distillation. Adapted batch normalization statistics are then used to compute client similarities, enabling personalized aggregation without sharing raw data.

### Unsupervised Local Client Adaptation

In the absence of labeled data, each client must adapt to its own distribution shift using only the pretrained source model. Our client-side test-time adaptation framework comprises two core strategies: Class-Wise Adaptive Normalization (CWAN) to address distributional drift and Confidence-Filtered Distillation to guide learning through self-supervision.

**Class-Wise Adaptive Normalization (CWAN)** Standard batch normalization relies on assumptions of stationary, class-balanced data, which break down in federated settings due to non-IID client distributions. To correct for this, we propose CWAN, which dynamically tracks per-class statistics at test time based on pseudo-labels assigned by the source model.

Let  $\mathbf{z}_i \in \mathbb{R}^d$  be the feature vector input to the normalization layer for test input  $\mathbf{x}_i$ , with pseudo-label  $\hat{c}_i \in \{1, \dots, K\}$ . For each class  $k$ , we maintain exponentially moving estimates of mean  $\mu_k^{(t)}$  and variance  $\Sigma_k^{(t)}$  at iteration  $t$ :

$$\mu_k^{(t)} = (1 - \alpha) \cdot \mu_k^{(t-1)} + \alpha \cdot \bar{\mathbf{z}}_k^{(t)} \quad (1)$$

$$\Sigma_k^{(t)} = (1 - \alpha) \cdot \Sigma_k^{(t-1)} + \alpha \cdot \widehat{\text{Var}}_k^{(t)} \quad (2)$$

These are aggregated into global estimates across classes:

$$\mu_{\text{global}}^{(t)} = \frac{1}{K} \sum_{k=1}^K \mu_k^{(t)} \quad (3)$$

$$\Sigma_{\text{global}}^{(t)} = \frac{1}{K} \sum_{k=1}^K \left( \Sigma_k^{(t)} + (\mu_k^{(t)} - \mu_{\text{global}}^{(t)})^2 \right) \quad (4)$$

To manage pseudo-label noise or limited class representation, we interpolate per-class and batch-wise statistics via

$$\beta \in [0, 1]:$$

$$\tilde{\mu}_i^{(t)} = (1 - \beta) \cdot \mu_{\hat{c}_i}^{(t)} + \beta \cdot \bar{\mathbf{z}}^{(t)} \quad (5)$$

$$\tilde{\Sigma}_i^{(t)} = (1 - \beta) \cdot \Sigma_{\hat{c}_i}^{(t)} + \beta \cdot \widehat{\text{Var}}^{(t)} \quad (6)$$

The normalized feature is then computed as:

$$\hat{\mathbf{z}}_i = \frac{\mathbf{z}_i - \tilde{\mu}_i^{(t)}}{\sqrt{\tilde{\Sigma}_i^{(t)} + \epsilon}} \cdot \gamma + \beta$$

where  $\gamma, \beta$  are the affine parameters, and  $\epsilon$  is a stability constant.

**Confidence-Filtered Distillation** To further refine local model behavior, we incorporate a self-supervised knowledge distillation mechanism. The source model  $\mathcal{F}_s$  serves as a frozen teacher, and a student model  $\mathcal{F}_t$  (initialized identically) is adapted locally using confident pseudo-labels.

For a test input  $\mathbf{x}$  and its augmented variant  $\tilde{\mathbf{x}}$ , predictions from teacher and student are:

$$\mathbf{p}_t = \text{softmax}(\mathcal{F}_s(\mathbf{x})), \quad \mathbf{p}_s = \text{softmax}(\mathcal{F}_t(\tilde{\mathbf{x}}))$$

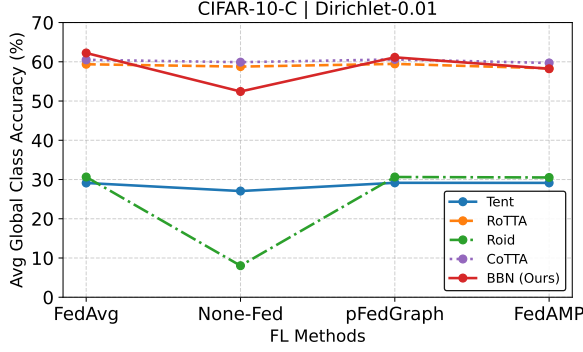
A confidence threshold  $\delta$  filters uncertain samples via entropy:

$$H(\mathbf{p}_t) < \delta$$

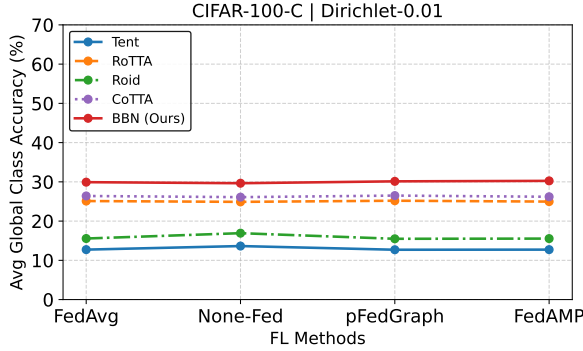
Letting  $\hat{y} = \arg \max \mathbf{p}_t$ , the distillation loss becomes:

$$\mathcal{L}_{\text{distill}} = \frac{1}{B} \sum_{i=1}^B \mathbb{I}[H(\mathbf{p}_{t,i}) < \delta] \cdot \text{CE}(\hat{y}_i, \mathbf{p}_{s,i}) \quad (7)$$

To preserve the model's generalization capability, only the affine parameters in the normalization layers of  $\mathcal{F}_t$  are updated during adaptation.



(a) CIFAR-10-C



(b) CIFAR-100-C

Figure 2: Average global class accuracies (%) of different TTA methods

### Batch Normalization Statistics for Domain Similarity

Once local adaptation is complete, each client possesses a refined model whose batch normalization layers encode a statistical summary of the local feature distribution. These statistics form the basis for client similarity estimation used during aggregation.

Let  $\mathcal{L}$  denote the set of BN layers, and for each client  $i$  and layer  $\ell \in \mathcal{L}$ , let  $\mu_\ell^{(i)}, \sigma_\ell^{2(i)} \in \mathbb{R}^d$  be the flattened global mean and variance vectors, respectively. The pairwise distance between clients  $i$  and  $j$  is given by:

$$D_{ij} = \frac{1}{|\mathcal{L}|} \sum_{\ell \in \mathcal{L}} \frac{1}{2} \left( \|\mu_\ell^{(i)} - \mu_\ell^{(j)}\|_2 + \|\sigma_\ell^{2(i)} - \sigma_\ell^{2(j)}\|_2 \right) \quad (8)$$

This defines a symmetric distance matrix  $D \in \mathbb{R}^{N \times N}$  across  $N$  clients.

### Personalized Federated Aggregation

To prevent negative transfer from dissimilar domains, we propose a soft similarity-based aggregation strategy that personalizes the global model for each client based on domain closeness.

**Similarity-Weighted Collaboration Matrix** We derive a row-stochastic similarity matrix  $W \in \mathbb{R}^{N \times N}$  from  $D$  using

a temperature-controlled softmax:

$$W_{ij} = \begin{cases} \frac{\exp(-D_{ij}/\tau)}{\sum_{k \neq i} \exp(-D_{ik}/\tau)} \cdot (1 - \omega_i), & j \neq i \\ \omega_i, & j = i \end{cases} \quad (9)$$

with self-weight  $\omega_i$  defined as:

$$\omega_i = \frac{1}{1 + \sum_{k \neq i} \exp(-D_{ik}/\tau)} \quad (10)$$

This ensures each client gives more weight to peers with similar BN statistics while preserving a degree of self-reliance.

**Aggregated Personalized Model** Each client receives a personalized model computed as:

$$\theta_{\text{agg}}^{(i)} = \sum_{j=1}^N W_{ij} \cdot \theta^{(j)} \quad (11)$$

This model is then used for continued inference or further local adaptation on client  $i$ 's stream.

### Privacy Concerns

The entire process integrates local adaptation and collaborative learning into a coherent pipeline. Each client adapts its BN statistics using CWAN and updates normalization affine parameters via filtered distillation. Adapted BN statistics are shared with the server to compute inter-client distances. The server constructs a personalized model for each client using similarity-weighted aggregation. Clients receive their aggregated models and continue local operations. The key insight of this approach is that batch normalization statistics offer a compact and domain-aware signature of the client's data distribution. By using these adapted statistics to guide server-side aggregation, the method naturally clusters clients with similar domain properties, even in the absence of explicit metadata or labels.

This personalized aggregation scheme is privacy-preserving, as only internal statistics (not raw data or features) are used. It is unsupervised, relying solely on pseudo-label-driven adaptation. It is scalable, with communication cost comparable to traditional federated averaging. The combination of locally adapted BN statistics and global distance-aware aggregation ensures both domain alignment and robustness to heterogeneity, improving generalization in federated test-time environments.

### Result and Discussion

We conduct experiments under both IID and Non-IID settings. Non-IID scenarios are simulated using Dirichlet sampling with concentration parameters  $\alpha \in \{0.005, 0.01, 0.1\}$ , where lower  $\alpha$  induces higher class imbalance. For test-time adaptation, we evaluate Tent (Wang et al. 2021a), CoTTA (Wang et al. 2022b), RoTTA (Yuan, Xie, and Li 2023b), and RoID (Marsden, Döbler, and Yang 2024). In federated settings, we compare FedAvg (McMahan et al. 2017), FedAvg-M (Cheng et al. 2024), FedProx (Li et al.

Table 1: Performance comparison across federated methods and TTA strategies under class imbalance (Non-IID with Dirichlet  $\alpha$ ) and IID setup. Total clients = 10.

Fed Method	TTA Method	CIFAR-10-C				CIFAR-100-C			
		IID	Non-IID			IID	Non-IID		
			$\alpha=0.005$	$\alpha=0.01$	$\alpha=0.1$		$\alpha=0.005$	$\alpha=0.01$	$\alpha=0.1$
Any	Source	56.51	58.28	58.58	56.11	54.28	54.58	57.14	53.03
None	Tent	81.08	23.99	24.98	31.23	68.59	6.84	9.41	28.19
	CoTTA	81.78	21.13	23.42	31.06	66.24	6.01	10.88	28.94
	ROID	81.44	14.60	15.95	26.20	69.41	7.46	11.84	38.38
	RoTTA	66.63	30.64	32.73	51.06	50.60	15.03	28.31	44.61
	BBN	72.32	56.88	52.33	70.53	59.32	64.07	72.31	66.21
FedAvg	Tent	81.19	21.51	22.47	29.42	68.13	5.06	7.82	25.27
	CoTTA	81.61	19.92	21.11	28.74	65.94	6.08	9.32	32.53
	ROID	81.85	22.93	23.96	32.15	69.06	6.85	10.97	37.01
	RoTTA	64.16	64.54	64.34	64.68	49.15	49.46	55.00	49.07
	BBN	72.61	68.47	69.04	74.05	60.08	67.49	71.30	63.03
FedProx	Tent	81.20	21.52	22.47	29.42	68.13	5.06	7.82	25.27
	CoTTA	81.59	19.92	23.69	30.98	65.95	6.07	10.85	29.19
	ROID	81.81	20.87	24.02	31.77	69.30	6.86	10.69	37.01
	RoTTA	67.31	39.39	64.41	64.69	49.82	22.66	55.04	48.98
	BBN	72.61	68.47	69.04	74.05	60.08	67.49	71.30	63.03
FedAvgM	Tent	81.30	21.45	22.37	29.38	67.52	5.05	7.82	25.39
	CoTTA	81.95	19.88	21.45	28.50	66.12	6.42	9.01	32.75
	ROID	81.87	23.07	24.09	32.35	68.81	6.87	11.11	37.04
	RoTTA	38.88	39.73	39.42	37.86	17.95	34.36	38.54	21.56
	BBN	72.61	68.47	69.04	74.05	60.08	67.49	71.30	63.03
pFedGraph	Tent	81.19	21.52	22.49	29.98	68.11	5.12	7.83	25.04
	CoTTA	81.64	19.94	22.34	29.01	65.99	6.08	10.88	29.03
	ROID	81.84	22.99	23.96	32.34	69.07	5.88	10.96	36.91
	RoTTA	64.87	63.75	64.43	63.98	49.51	52.99	54.88	54.63
	BBN	72.73	69.37	66.89	73.54	60.14	67.29	72.61	63.71
FedAmp	Tent	81.19	20.65	22.49	28.28	68.15	5.09	7.85	25.30
	CoTTA	81.61	20.19	22.93	30.21	65.95	6.07	10.23	28.71
	ROID	81.83	21.11	23.82	33.11	69.11	6.88	11.21	36.88
	RoTTA	65.91	63.87	62.85	63.16	50.27	46.82	53.91	49.35
	BBN	72.36	63.16	61.36	72.53	59.52	69.88	72.62	63.29
pFedBBN	BBN	70.11	72.41	71.96	68.53	65.29	71.96	73.88	64.29

2020), pFedGraph (Ye et al. 2023), and FedAMP (Huang et al. 2021), with 10 simulated clients. Robustness under distribution shifts is assessed on CIFAR-10-C and CIFAR-100-C with 15 corruption types at severity level 5, using WideResNet-28 and ResNeXt-29, respectively.

## Detailed Results

Table 1 presents a comprehensive performance comparison across various federated learning aggregation strategies and test-time adaptation (TTA) methods, evaluated on two corruption-augmented benchmarks: CIFAR-10-C and CIFAR-100-C. The experiments are conducted in both IID and Non-IID client data settings, where Non-IID scenarios are simulated using Dirichlet distributions by varying the concentration parameter  $\alpha \in \{0.005, 0.01, 0.1\}$ . Lower values of  $\alpha$  induce higher class imbalance across clients,

thereby increasing the difficulty of the federated learning.

**Performance under IID Setting.** In the IID case, where data is evenly distributed across clients without class imbalance, most existing TTA methods such as Tent, CoTTA, and ROID demonstrate strong performance when paired with standard federated aggregation techniques like FedAvg and FedProx. For instance, Tent and CoTTA achieve over 81% accuracy on CIFAR-10-C across multiple aggregation strategies. However, our proposed BBN method remains competitive, achieving consistent performance above 72% in both datasets across different FL methods.

**Performance under Non-IID Setting.** Under the more realistic Non-IID setting with varying degrees of class imbalance (controlled by Dirichlet  $\alpha$ ), the performance of standard TTA methods degrades significantly. In particular, methods like Tent and CoTTA drop to as low as 6–30% ac-

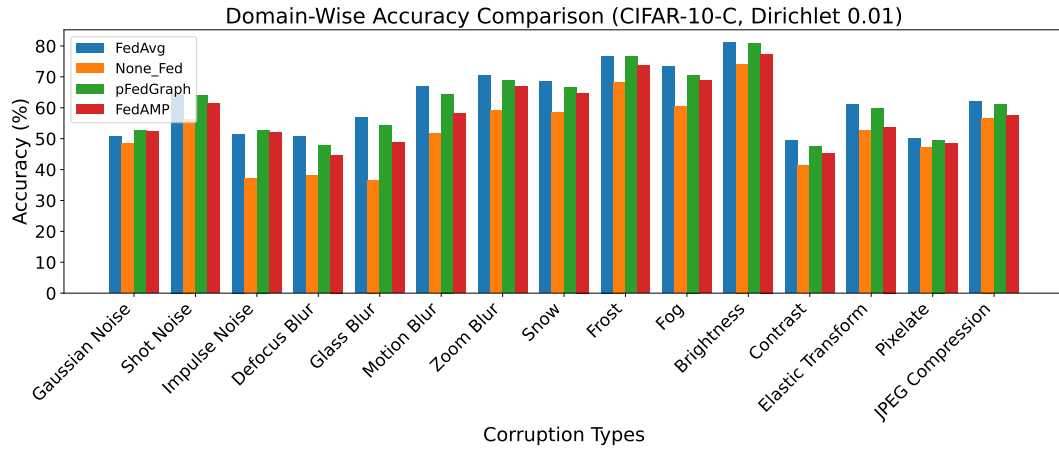


Figure 3: Domain-wise accuracy comparison of our method (Dirichlet  $\alpha = 0.01$ ) across CIFAR-10-C corruptions.

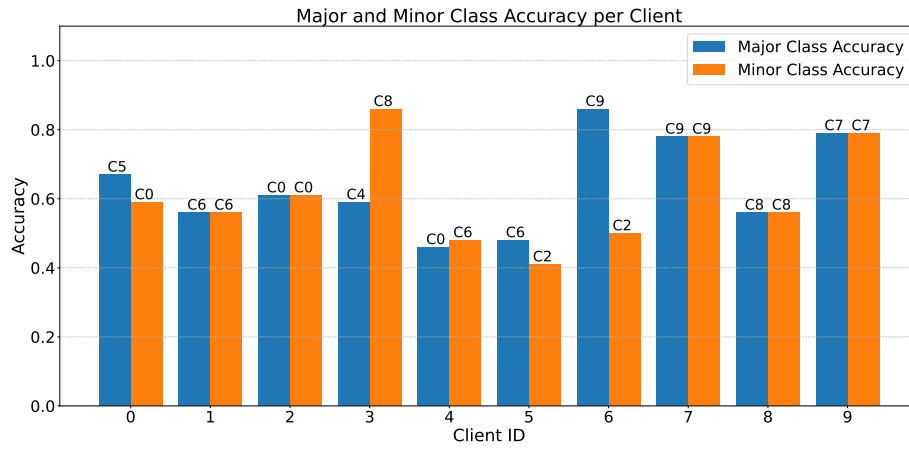


Figure 4: Major and Minor Class Accuracy per Client.

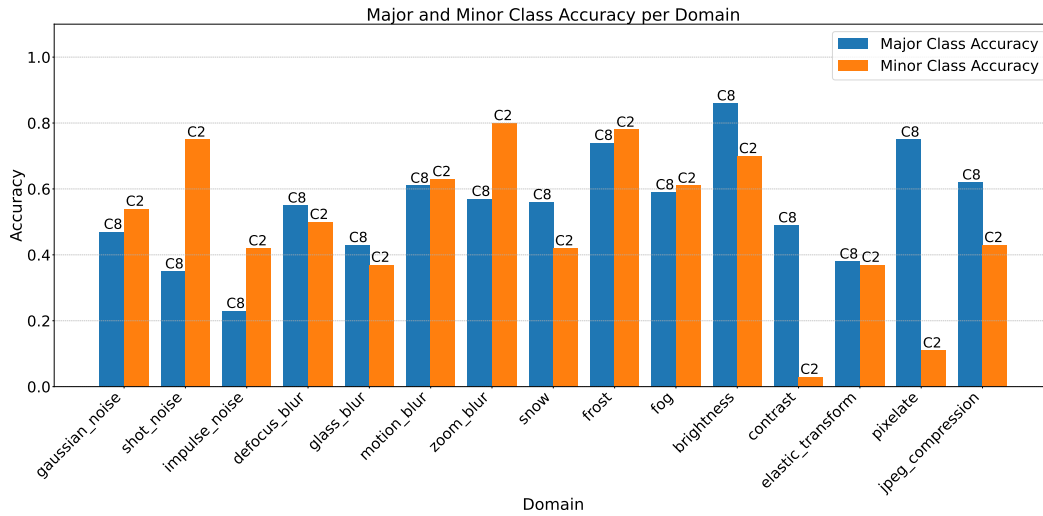
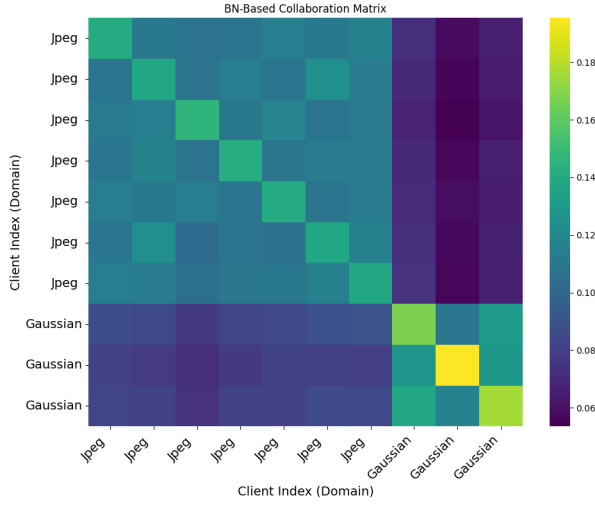
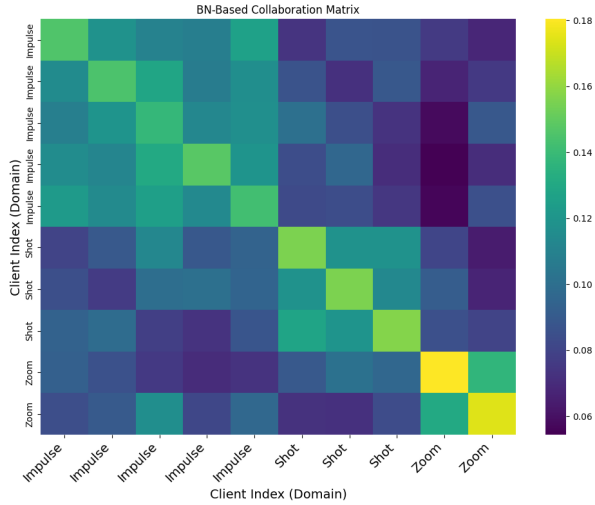


Figure 5: Major and minor class accuracy per domain.



(a) Collaboration Matrix Round 40



(b) Collaboration Matrix Round 75

Figure 6: Collaboration Matrix of our pFedBBN which indicates which client give more priority while aggregating and it has clearly be seen that similar domains are aggregated more than others

curacy in extreme imbalance cases ( $\alpha = 0.005$ ) across both datasets. In contrast, our proposed TTA method BBN maintains robust performance across all levels of imbalance. For example, with FedAvg at  $\alpha = 0.005$ , BBN achieves 68.47% on CIFAR-10-C and 67.49% on CIFAR-100-C, substantially outperforming other methods.

**Superior Performance of pFedBBN.** Notably, our full framework, pFedBBN, which combines personalized federated learning with the BBN-based TTA strategy, consistently achieves the highest or near-highest accuracy across all scenarios. It maintains stable performance even in highly imbalanced cases. For instance, on CIFAR-10-C with  $\alpha = 0.005$ , pFedBBN achieves 72.41%, surpassing all other combinations of FL and TTA strategies. Similarly, on CIFAR-100-

C, it reaches 72.62% accuracy at  $\alpha = 0.01$ , demonstrating both robustness and effectiveness in diverse settings. These results show that while existing TTA methods falter under class imbalance and data heterogeneity, our BBN-based TTA, especially with pFedBBN, offers a more effective solution.

**Comparison of TTA setups under heterogeneity.** In figure 2 (a) and figure 2 (b) respectively, we show the performances on CIFAR-10-C and CIFAR-100-C datasets respectively for various test time adaptation techniques. For, CIFAR-10-C, provides almost consistent performance under all fed setups, while RoTTA falls slightly short. Our method is comparable to the techniques, and beats both the aforementioned techniques for pFedGraph aggregation. In figure 2 (b) for the evaluation of CIFAR-100-C corruption dataset, our method consistently outperforms all the TTA methods under all aggregation setups. These results demonstrate BBN’s robustness under various domain shifts.

**Performance analysis under domain shift** In Figure 3 we demonstrate BBN’s performance across different domains under class imbalance ( $\alpha = 0.01$ ) for various aggregation techniques. We observe that for FedAvg aggregation our method performs the best across a diversity of corruption settings. This indicates better generalization under FedAvg aggregation. The personalized setup, pFedGraph comes in second with competitive performance for various corruptions eg: Gaussian Noise, Shot Noise, Fog, Brightness, JPEG compression, etc. From the figure, it is also evident that our method performs best in lighting and weather corruptions, modestly on noise corruptions, but struggles with blur, compression, and geometric distortions.

**Performance analysis of Major and Minor Classes of pFedBBN** As shown in Figure 4, pFedBBN consistently achieves strong performance across all clients, maintaining at least 50% accuracy in both major and minor classes. However in Figure 5, in the case of clients 2 and 8, we observe noticeable drops in accuracy under certain domain shifts. This indicates the presence of significant distribution shifts as these clients transition across domains, highlighting the challenges posed by continual domain adaptation.

**Collaboration Weight Analysis of Client Aggregation in pFedBBN** We compute a symmetric distance matrix based on the Balanced Batch Normalization (BBN) statistics, specifically, the global mean and variance, across all clients. This matrix is then used to derive the collaboration weights for aggregation. The results reveal that clients tend to prioritize aggregation with others from the same domain, highlighting domain-aware collaboration. As shown in Figure 6, clients from similar domains consistently form clusters in the collaboration weight matrix across different federated rounds, indicating effective domain-wise distribution alignment.

## Conclusion

We introduced **pFedBBN**, a federated test-time adaptation framework that addresses class imbalance and domain shifts by leveraging Balanced Batch Normalization. FedBBN enables unsupervised, privacy-preserving client-side adaptation and introduces a class-aware server aggregation strat-



egy. Experiments on benchmark datasets show that FedBBN improves robustness and minority-class performance over existing methods. This makes it a practical and scalable solution for real-world federated learning scenarios with non-IID and unlabeled test distributions.

## References

- Bao, W.; Wei, T.; Wang, H.; and He, J. 2023. Adaptive Test-Time Personalization for Federated Learning. In *Advances in Neural Information Processing Systems*.
- Chen, C.; Liao, T.; Deng, X.; Wu, Z.; Huang, S.; and Zheng, Z. 2025. Advances in robust federated learning: A survey with heterogeneity considerations. *IEEE Transactions on Big Data*.
- Cheng, Z.; Huang, X.; Wu, P.; and Yuan, K. 2024. Momentum Benefits Non-iid Federated Learning Simply and Provably. In *The Twelfth International Conference on Learning Representations*.
- Duan, M.; Liu, D.; Chen, X.; Liu, R.; Tan, Y.; and Liang, L. 2020. Self-balancing federated learning with global imbalanced data in mobile systems. *IEEE Transactions on Parallel and Distributed Systems*, 32(1): 59–71.
- Huang, C.; Li, Y.; Loy, C. C.; and Tang, X. 2016. Learning deep representation for imbalanced classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5375–5384.
- Huang, Y.; Chu, L.; Zhou, Z.; Wang, L.; Liu, J.; Pei, J.; and Zhang, Y. 2021. Personalized Cross-Silo Federated Learning on Non-IID Data. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(9): 7865–7873.
- Kang, N.; Lim, Y.; and Im, J. 2024. Overcoming data imbalance in federated learning with calibration weighting. In *2024 IEEE International Conference on Big Data (Big-Data)*, 7832–7841. IEEE.
- Khan, S. H.; Hayat, M.; Bennamoun, M.; Sohel, F. A.; and Togneri, R. 2017. Cost-sensitive learning of deep feature representations from imbalanced data. *IEEE transactions on neural networks and learning systems*, 29(8): 3573–3587.
- Khushi, M.; Shaukat, K.; Alam, T. M.; Hameed, I. A.; Uddin, S.; Luo, S.; Yang, X.; and Reyes, M. C. 2021. A comparative performance analysis of data resampling methods on imbalance medical data. *Ieee Access*, 9: 109960–109975.
- Li, F.; and Others. 2024. Federated Learning with Probability-Corrected Loss and Shared Class Prototypes. *Journal Name*.
- Li, T.; Sahu, A. K.; Zaheer, M.; Sanjabi, M.; Talwalkar, A.; and Smith, V. 2020. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2: 429–450.
- Liang, F.; and Others. 2024. TTA-FedDG: Test-Time Adaptation for Federated Domain Generalization. In *Proceedings of the AAAI Conference on Artificial Intelligence*.
- Liu, B.; Lv, N.; Guo, Y.; and Li, Y. 2024. Recent advances on federated learning: A systematic survey. *Neurocomputing*, 597: 128019.
- Marsden, R. A.; Döbler, M.; and Yang, B. 2024. Universal test-time adaptation through weight ensembling, diversity weighting, and prior correction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2555–2565.
- Marsden, R. A.; Döbler, M.; and Yang, B. 2024. Universal Test-Time Adaptation Through Weight Ensembling, Diversity Weighting, and Prior Correction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2555–2565.
- McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; and y Arcas, B. A. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, 1273–1282. PMLR.
- Niu, S.; Wu, J.; Zhang, Y.; Chen, Y.; Zheng, S.; Zhao, P.; and Tan, M. 2022. Efficient Test-Time Model Adaptation without Forgetting. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, volume 162 of *Proceedings of Machine Learning Research*, 16888–16905. PMLR.
- Noble, M.; Bellet, A.; and Dieuleveut, A. 2022. Differentially private federated learning on heterogeneous data. In *International conference on artificial intelligence and statistics*, 10110–10145. PMLR.
- Rajib, R. H.; Iftee, M. A. R.; Hossain, M. S.; Rahman, A.; Mistry, S.; Amin, M. A.; and Ali, A. A. 2025. FedCTTA: A Collaborative Approach to Continual Test-Time Adaptation in Federated Learning. *arXiv preprint arXiv:2505.13643*.
- Sarkar, D.; Narang, A.; and Rai, S. 2020. Fed-focal loss for imbalanced data classification in federated learning. *arXiv preprint arXiv:2011.06283*.
- Seol, M.; and Kim, T. 2023. Performance enhancement in federated learning by reducing class imbalance of non-iid data. *Sensors*, 23(3): 1152.
- Shang, X.; Lu, Y.; Huang, G.; and Wang, H. 2022. Federated Learning on Heterogeneous and Long-Tailed Data via Classifier Re-Training with Federated Features. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, 2218–2224.
- Shao, F.; and Others. 2021. Federated Face Anti-Spoofing with Test-Time Adaptation. In *Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition*.
- Wang, D.; Shelhamer, E.; Liu, S.; Olshausen, B.; and Darrell, T. 2021a. Tent: Fully Test-Time Adaptation by Entropy Minimization. In *International Conference on Learning Representations*.
- Wang, L.; Xu, S.; Wang, X.; and Zhu, Q. 2021b. Addressing class imbalance in federated learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, 10165–10173.
- Wang, Q.; Fink, O.; Gool, L. V.; and Dai, D. 2022a. Continual Test-Time Domain Adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.



Wang, Q.; Fink, O.; Van Gool, L.; and Dai, D. 2022b. Continual test-time domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7201–7211.

Wang, Y.-X.; Ramanan, D.; and Hebert, M. 2017. Learning to model the tail. *Advances in neural information processing systems*, 30.

Xian, S.; Shen, Y.; Jiang, S.; Zhao, Z.; Yan, Z.; and Xing, G. 2022. BalanceFL: Addressing Class Imbalance in Long-Tail Federated Learning. In *2022 21st ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, 271–284.

Xiao, C.; and Wang, S. 2021. An experimental study of class imbalance in federated learning. In *2021 IEEE Symposium Series on Computational Intelligence (SSCI)*, 1–7. IEEE.

Xiao, Z.; Chen, Z.; Liu, S.; Wang, H.; Feng, Y.; Hao, J.; Zhou, J. T.; Wu, J.; Yang, H. H.; and Liu, Z. 2023. Fed-GraB: federated long-tailed learning with self-adjusting gradient balancer. In *Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS '23*. Red Hook, NY, USA: Curran Associates Inc.

Ye, R.; Ni, Z.; Wu, F.; Chen, S.; and Wang, Y. 2023. Personalized federated learning with inferred collaboration graphs. In *International conference on machine learning*, 39801–39817. PMLR.

You, F.; Li, J.; and Zhao, Z. 2021. Test-time batch statistics calibration for covariate shift. *arXiv preprint arXiv:2110.04065*.

Yuan, L.; Xie, B.; and Li, S. 2023a. Robust Test-Time Adaptation in Dynamic Scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Yuan, L.; Xie, B.; and Li, S. 2023b. Robust test-time adaptation in dynamic scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15922–15932.

Zeng, Y.; Liu, L.; Liu, L.; Shen, L.; Liu, S.; and Wu, B. 2023. Global balanced experts for federated long-tailed learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, 4815–4825.

## Supplementary Materials

### Dataset Details

We evaluate our methods on the CIFAR-10-C and CIFAR-100-C benchmarks, which apply 15 common image corruptions (e.g., Gaussian noise, blur, brightness) at five discrete severity levels  $\{1, \dots, 5\}$ . The original CIFAR-10 and CIFAR-100 test sets each contain 10,000 images. For each of the 15 corruption types, the corrupted dataset contains 50,000 images (10,000 images per severity level, across five levels). The total benchmark for a dataset, like CIFAR-10-C, therefore contains 750,000 images, which are grouped into 10 and 100 classes respectively. In this work, we focus exclusively on the worst-case noise level: severity-5. Therefore, in our work, we simulate maximal domain shift and stress-test the model’s robustness.

### Personalized Federated Balanced Batch Normalization

The pFedBBN algorithm is a personalized federated learning setup designed to tackle the data heterogeneity and class imbalance problem in test-time adaptation (TTA). This method focuses on Batch normalization statistics and a unique aggregation strategy. The process begins with the server broadcasting the current global model to all participating clients. Each client then independently performs a balanced batch normalization in test time adaptation on its local, unlabeled test data. This client-side adaptation is crucial as it not only fine-tunes the model to the client’s specific data distribution but also addresses class imbalance by using a balanced loss function and pseudo label generated at test time as ground truth is not available. After local adaptation, each client extracts and sends its Batch Normalization (BN) statistics (mean and variance for each layer) back to the server. The server, instead of a simply averaging, uses the collected BN statistics to compute a collaboration matrix. This matrix quantifies the similarity between the data distributions of different clients, effectively utilizing the information of relatedness among the clients. A higher similarity between two clients’ BN statistics results in a stronger collaboration weight. The server uses this collaboration matrix to perform a weighted aggregation, creating a personalized aggregated model for each client. This approach ensures that clients with similar data distributions contribute more to each other’s model updates. This iterative process allows the global model to converge while maintaining the personalized characteristics of each client’s model, making it robust to non-IID data.

### Code Appendix

**Command-Line Usage** To reproduce experiments on CIFAR-10-C (Dirichlet  $\alpha = 0.005$ ) with BBN-TTA and pFedBBN:

```
1 python test_time_new.py \
2   --cfg cfgs/cifar10_c/bbn_tta.yaml \
3   fed.fed_tech fedbnstat \
4   fed.dirichlet_alpha 0.005
```

---

**Pseudocode 1: pFedBBN: Personalized Federated BN-Statistics Aggregation**

---

**Require:**  $K$  clients, each with local BN statistics  $B_i = \{(\mu_\ell^i, \sigma_\ell^{2,i})\}_{\ell=1}^L$ ; temperature  $T$

0: **for** each communication round **do**

0:   Clients  $1 \dots K$  send  $B_i$  to server

0:   Server computes distances:

$D_{ij} = \frac{1}{L} \sum_{\ell=1}^L (\|\mu_\ell^i - \mu_\ell^j\| + \|\sigma_\ell^{2,i} - \sigma_\ell^{2,j}\|)$

0:   Compute weights:  $W_{i \rightarrow j} = \exp(-D_{ij}/T)$  and row-normalize

0:   **for** each client  $i$  **do**

0:     Aggregate:  $\theta_i \leftarrow \sum_{j=1}^K W_{i \rightarrow j} \theta_j$

0:   **end for**

0:   Server broadcasts updated  $\{\theta_i\}$  back to clients

0: **end for**=0

---

### Code Structure and Notes

- **test\_time\_new.py**: main entry point—parses flags, loads config, dispatches to BBN-TTA or other TTA.
- **bbn\_tta.py**: implements BBN-TTA adaptation (BN-affine updates, pseudo-labeling, entropy mask).
- **fedbnstat\_group.py**: server-side logic for BN stat collection, distance computation, weight normalization, model aggregation.
- **Hyperparameters**: defined in `conf.py` (e.g.,  $H_0$ ,  $\lambda$ , learning rates, batch size).
- **Environment**: NVIDIA A100 GPUs, PyTorch 1.13, CUDA 11.7, Ubuntu 20.04.

### Additional analysis

#### Performance Gain of pFedBBN over Baselines

We illustrate the performance gain of our proposed pFedBBN method over several standard and advanced federated learning baselines across figure 7 to figure 9, all of which are combined with our client-side BBN TTA. These figures provide a clear visual comparison of how our novel BN statistics aggregation strategy, a core component of pFedBBN, improves upon existing federated methods. Specifically, Figure 7 compares pFedBBN against the FedAvg baseline, Figure 8 highlights the improvements against the personalized graph-based pFedGraph, and Figure 10 demonstrates the gain over the personalized FedAmp method. The consistent positive accuracy gains across various non-IID degrees on both CIFAR-10-C and CIFAR-100-C datasets serve as strong evidence that our aggregation approach is more effective at leveraging BN statistics for robust, personalized model updates.

#### Class imbalance analysis

Figure 10 and Figure 11 provide a detailed analysis of the simulated class imbalance, which is crucial for understanding the challenges of the non-IID setting. Figure 10, the Global Class Frequency plot, visually represents the severe

class imbalance inherent in the CIFAR-10-C data distribution with a Dirichlet  $\alpha$  of 0.005. It highlights the long-tail problem where a small number of classes dominate the dataset, while others are significantly under-represented. This imbalance is a primary driver of poor model performance in federated learning. Figure 11, the Global Normalized Imbalance Over Time plot, offers an insightful look into the training dynamics. This line graph tracks the normalized imbalance across federated rounds, showing how the model, despite the initial data disparity, learns to progressively correct for the imbalance. This demonstrates the effectiveness of the training process in mitigating dataset heterogeneity, a key objective of our method.

#### Performance comparison of Fed TTA setups

A performance comparison across federated methods and Test-Time Adaptation (TTA) strategies is presented in Figures 12 through 17. A key observation across all scenarios is the distinct performance clusters formed by the TTA methods. Figures 12, 13, and 14 show that on the CIFAR-10-C dataset, TTA methods like Tent, CoTTA, andROID consistently underperform, with accuracy scores generally below 30%. In stark contrast, RoTTA and BBN achieve significantly higher accuracies, often exceeding 60% and 70% respectively, making them the superior choices for TTA. This trend is echoed in the CIFAR-100-C results, as shown in Figures 15, 16, and 17, where Tent, CoTTA, andROID again exhibit very low accuracy, sometimes in the single digits, while RoTTA and BBN maintain high performance levels. Notably, the federated methods—FedAvg, pFedGraph, and FedAmp—show comparable performance with a given TTA method. However, the BBN TTA method paired with these federated methods generally yields the highest overall accuracy.

#### IID analysis

In Figures 18 and 19, we provide a foundational analysis of model performance under IID (Independent and Identically Distributed) conditions, a crucial baseline for evaluating federated learning methods. The plots for CIFAR-10-C and CIFAR-100-C respectively, demonstrate that when data is evenly distributed across clients, all federated methods—both with and without TTA (Test-Time Adaptation) methods like BBN—achieve high and comparable accuracy. The lack of significant performance gaps in this setting confirms that the primary challenge of federated learning is not the distributed nature of the data itself, but rather the data heterogeneity introduced by non-IID distributions.

Table 2: Average global class accuracy (%) on CIFAR-10-C and CIFAR-100-C under Dirichlet heterogeneity  $\alpha = 0.01$ . Results are grouped by dataset and TTA method, then compared across four federated aggregation schemes (None-Fed, FedAvg, pFedGraph, FedAMP) plus the “centralized” CoTTA baseline. Higher values indicate better robustness to distribution skew.

Dataset	TTA Method	Federated Algorithm				
		None-Fed	FedAvg	pFedGraph	FedAmp	CoTTA
CIFAR-10-C	Tent	23.9	29.12	29.14	29.05	28.0
	CoTTA	25.5	30.0	30.1	30.0	30.0
	ROID	22.0	30.64	30.64	30.52	29.5
	RoTTA	36.42	59.37	59.46	58.33	57.0
	BBN	52.43	62.24	61.14	58.22	59.0
CIFAR-100-C	Tent	10.3	12.73	12.71	13.04	13.0
	CoTTA	12.0	14.5	14.7	14.6	14.6
	ROID	12.0	15.56	15.49	15.54	15.0
	RoTTA	17.31	25.11	25.19	24.97	24.0
	BBN	29.65	29.92	30.13	30.25	30.0

Table 3: Class-wise global accuracy (%) on CIFAR-10-C (severity-5) under Dirichlet heterogeneity  $\alpha = 0.01$ . We compare four TTA methods (Tent, CoTTA, ROID, RoTTA, BBN) across four federation schemes (None, FedAvg, pFedGraph, FedAMP). The low  $\alpha$  simulates extreme non-IID (no client observes class 3), so class 3 accuracy is zero across all settings.

Method	Fed	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	Avg
Tent	None-Fed	21.5	67.5	30.0	0.0	35.0	42.0	20.0	45.0	34.0	44.0	33.9
Tent	FedAvg	28.0	82.0	50.0	0.0	56.0	62.0	65.0	74.0	62.0	80.0	55.9
Tent	pFedGraph	28.3	81.8	49.5	0.0	56.5	61.8	61.5	73.0	60.5	79.2	55.3
Tent	FedAMP	27.8	81.0	48.5	0.0	55.0	61.2	58.0	72.5	58.5	78.0	54.0
CoTTA	None-Fed	23.0	68.0	32.0	0.0	36.5	43.0	22.0	47.0	35.5	46.0	35.0
CoTTA	FedAvg	29.5	83.0	52.0	0.0	58.0	63.5	66.5	75.5	63.5	81.5	57.3
CoTTA	pFedGraph	29.7	82.9	51.5	0.0	58.4	63.1	63.1	74.2	62.0	80.6	56.6
CoTTA	FedAMP	29.1	82.5	50.5	0.0	57.1	62.7	60.2	73.5	60.2	79.5	55.6
ROID	None-Fed	19.0	62.0	28.0	0.0	32.0	39.0	18.0	42.0	31.0	42.0	31.3
ROID	FedAvg	30.6	81.4	54.0	0.0	59.0	65.0	67.0	76.5	65.0	83.0	58.2
ROID	pFedGraph	30.5	81.3	53.2	0.0	59.2	64.6	64.0	75.0	63.2	81.9	57.3
ROID	FedAMP	30.2	80.9	52.0	0.0	57.6	63.8	61.0	74.0	61.5	80.3	56.1
RoTTA	None-Fed	47.72	57.33	41.10	0.0	30.11	33.58	39.83	33.22	39.37	41.91	36.42
RoTTA	FedAvg	69.21	86.67	54.09	0.0	58.96	63.56	66.00	66.31	60.89	67.97	59.37
RoTTA	pFedGraph	69.76	85.33	53.25	0.0	59.25	65.89	63.94	67.29	61.73	68.13	59.46
RoTTA	FedAMP	66.36	84.00	54.16	0.0	57.35	64.83	61.41	67.20	59.60	68.43	58.33
BBN	None-Fed	47.33	84.00	51.46	0.0	55.97	68.61	36.86	64.67	47.85	67.57	52.43
BBN	FedAvg	63.48	85.33	50.96	0.0	57.20	67.80	70.10	79.15	66.23	82.17	62.24
BBN	pFedGraph	63.46	85.33	51.19	0.0	57.30	67.28	62.11	78.89	63.75	82.13	61.14
BBN	FedAMP	61.25	84.00	52.26	0.0	55.50	67.54	46.26	79.59	56.36	79.44	58.22

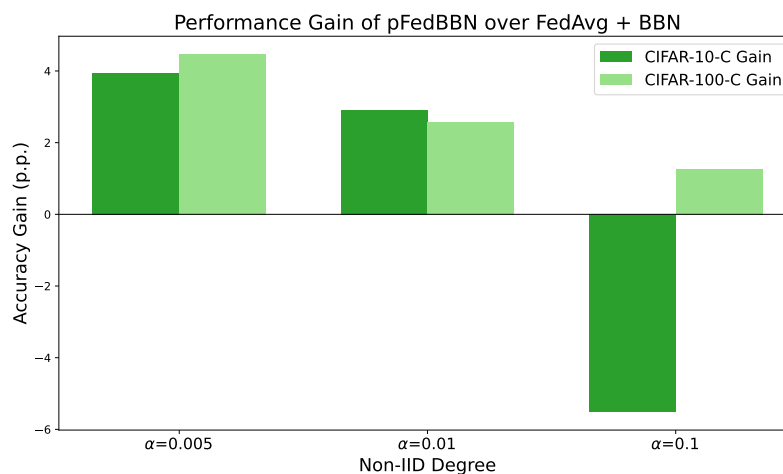


Figure 7: pFedBBN gain over FedAvgBBN.

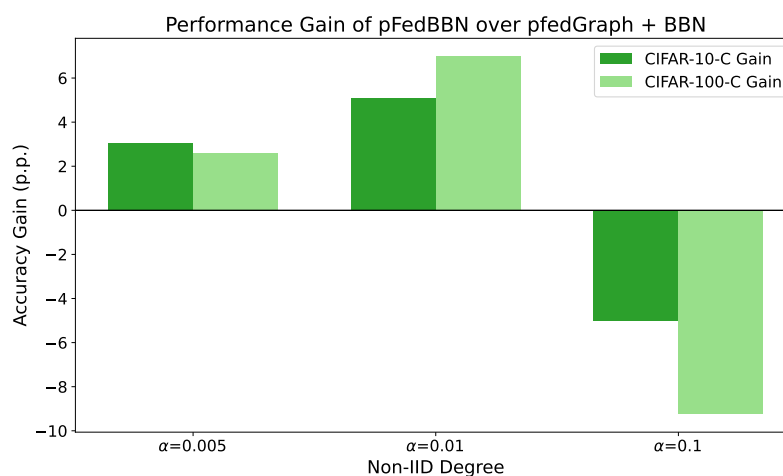


Figure 8: pFedBBN gain over pFedGraphBBN.

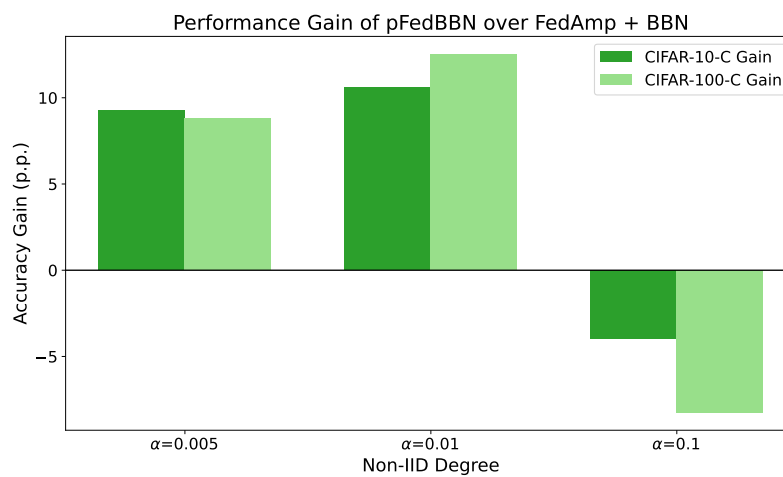


Figure 9: pFedBBN gain over FedAMPBBN.

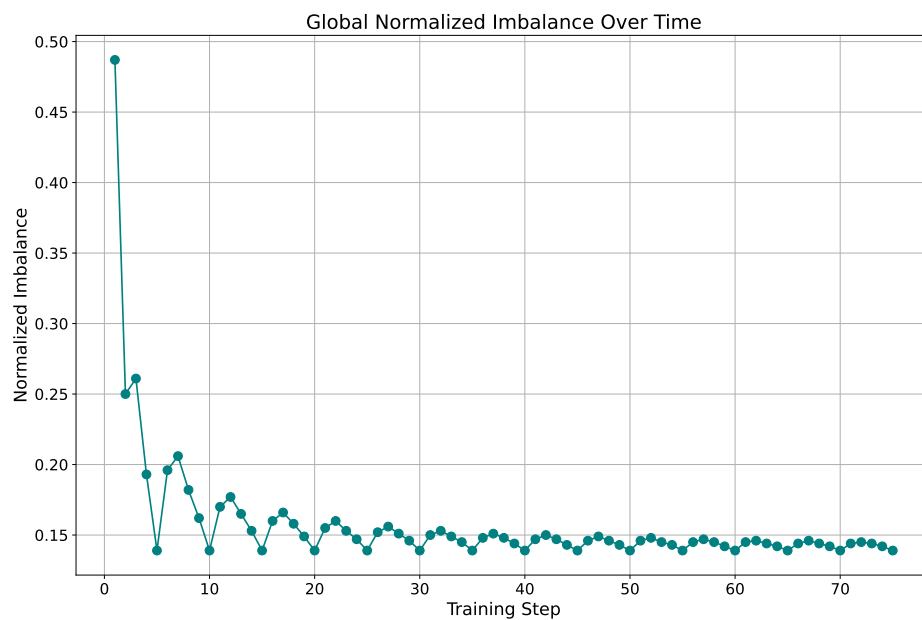


Figure 10: Global imbalance ratio with federated rounds.

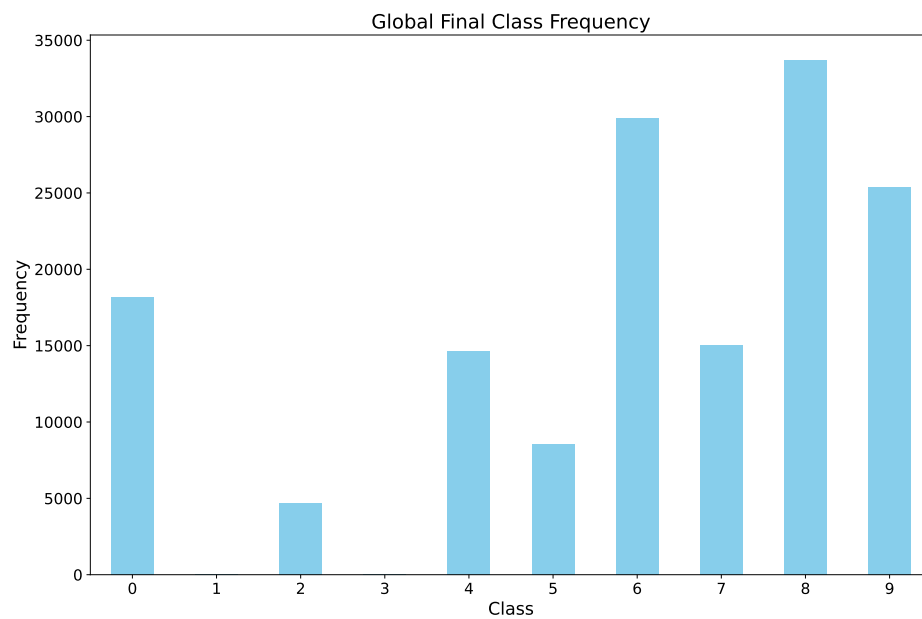


Figure 11: Global class frequency for CIFAR-10-C.

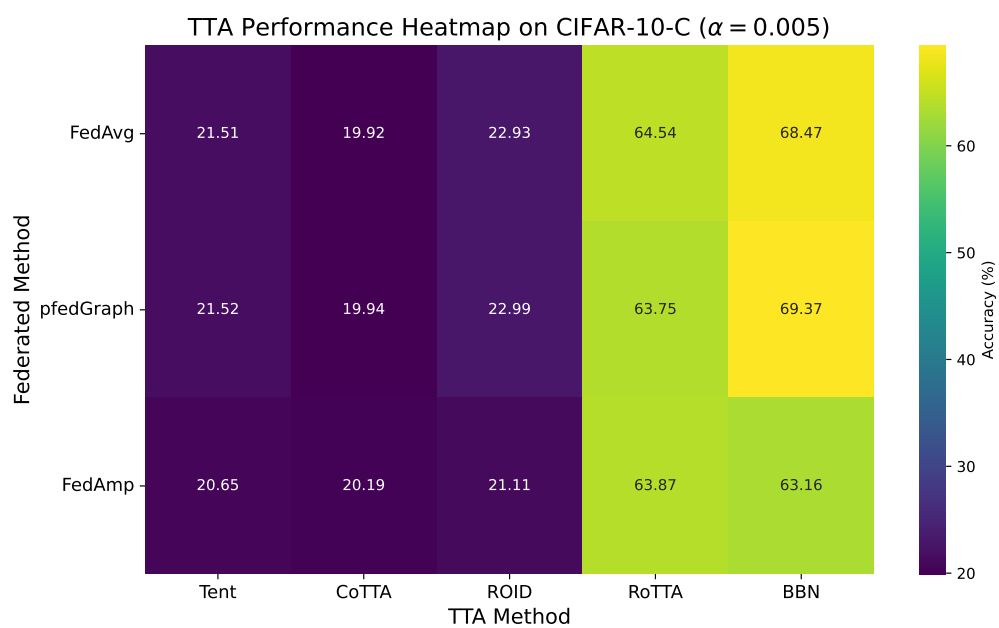


Figure 12: Fed TTA performance on CIFAR-10-C ( $\alpha = 0.005$ )

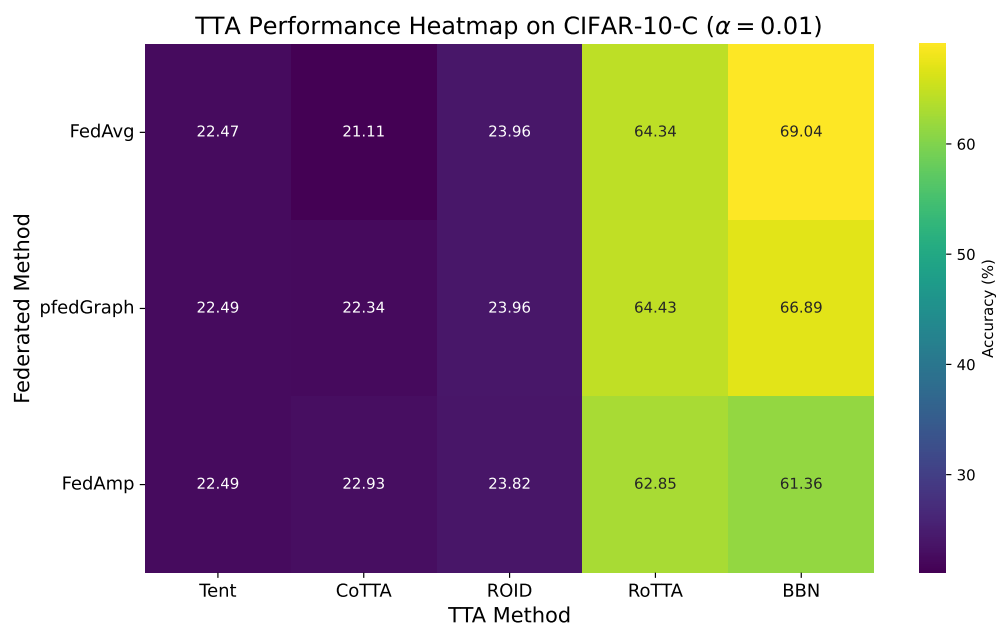


Figure 13: Fed TTA performance on CIFAR-10-C ( $\alpha = 0.01$ )

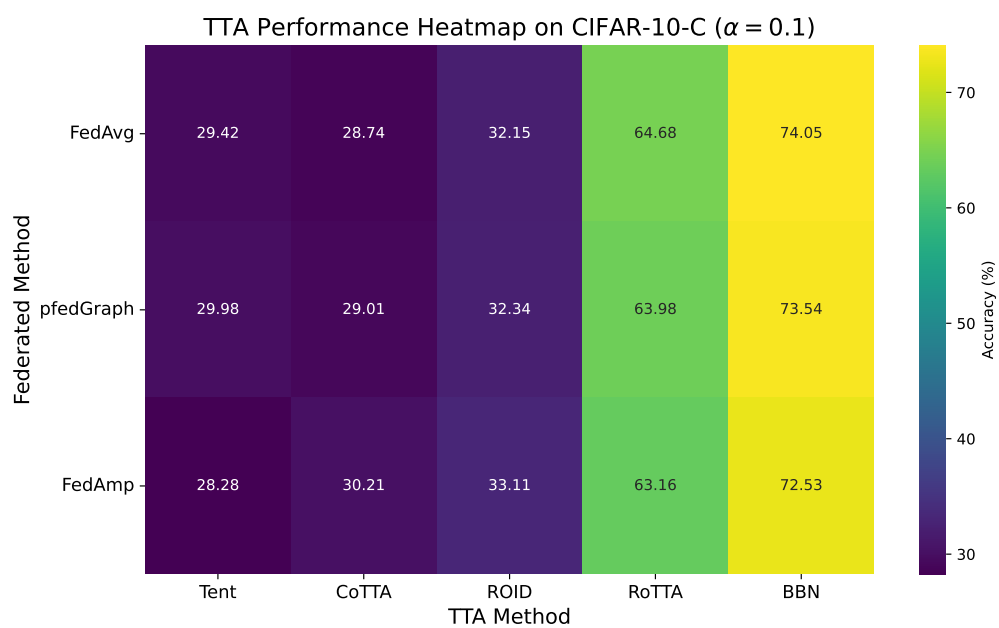


Figure 14: Fed TTA performance on CIFAR-10-C ( $\alpha = 0.1$ )

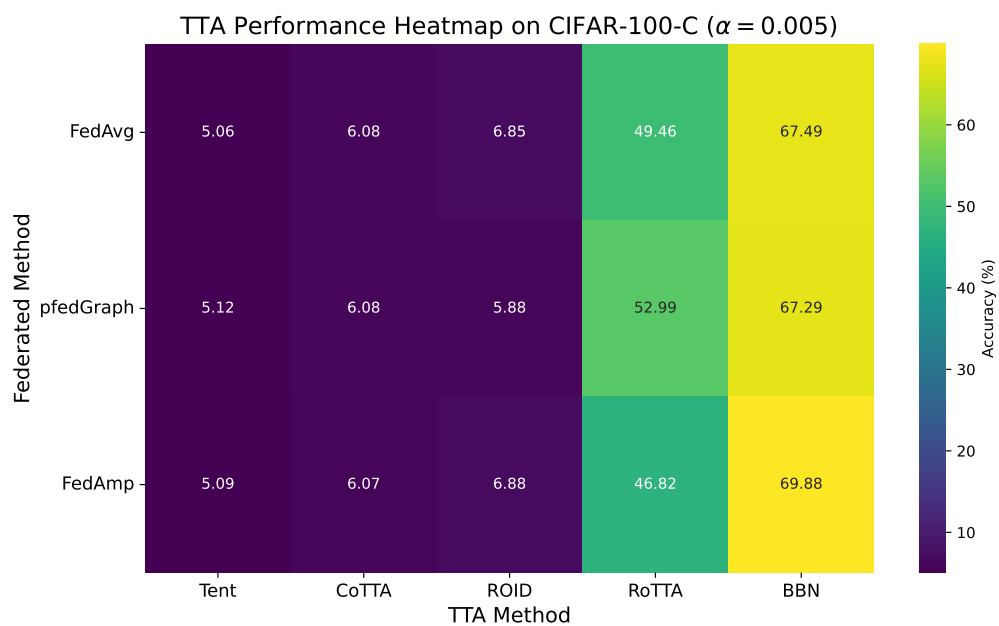


Figure 15: Fed TTA performance on CIFAR-100-C ( $\alpha = 0.005$ )



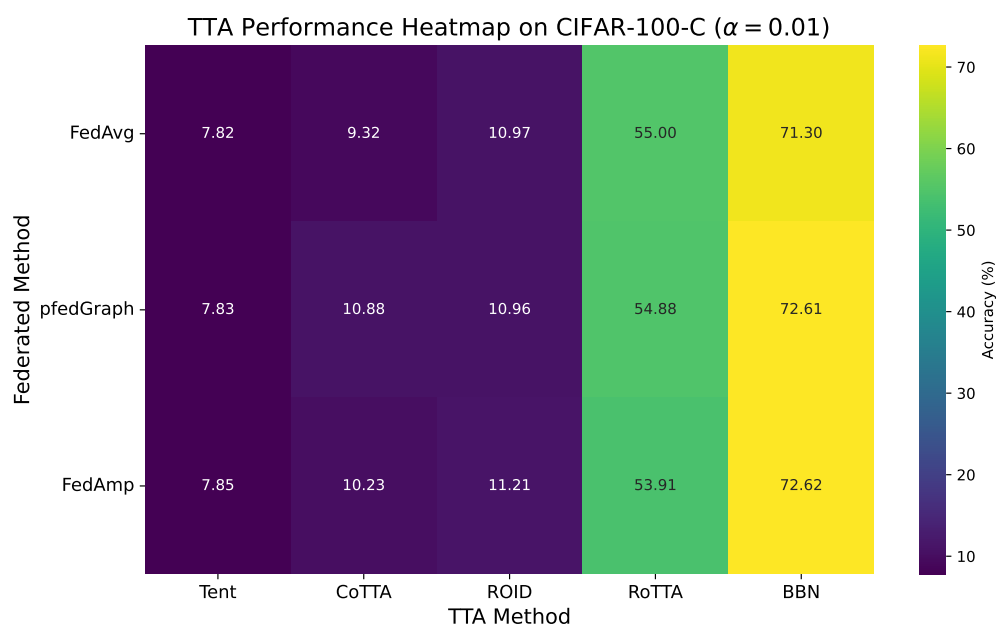


Figure 16: Fed TTA performance on CIFAR-100-C ( $\alpha = 0.01$ )

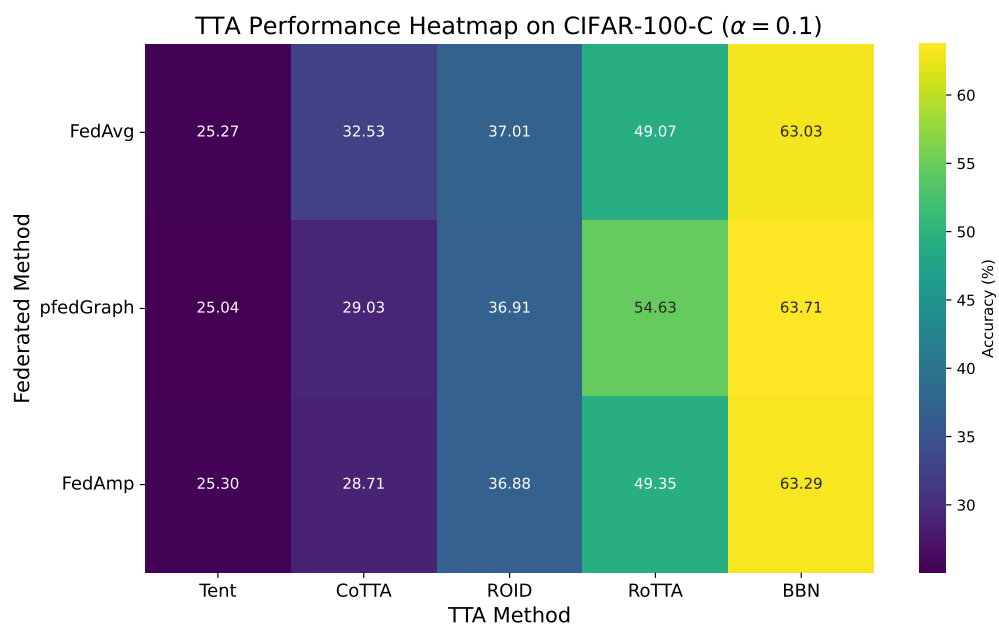


Figure 17: Fed TTA performance on CIFAR-100-C ( $\alpha = 0.1$ )

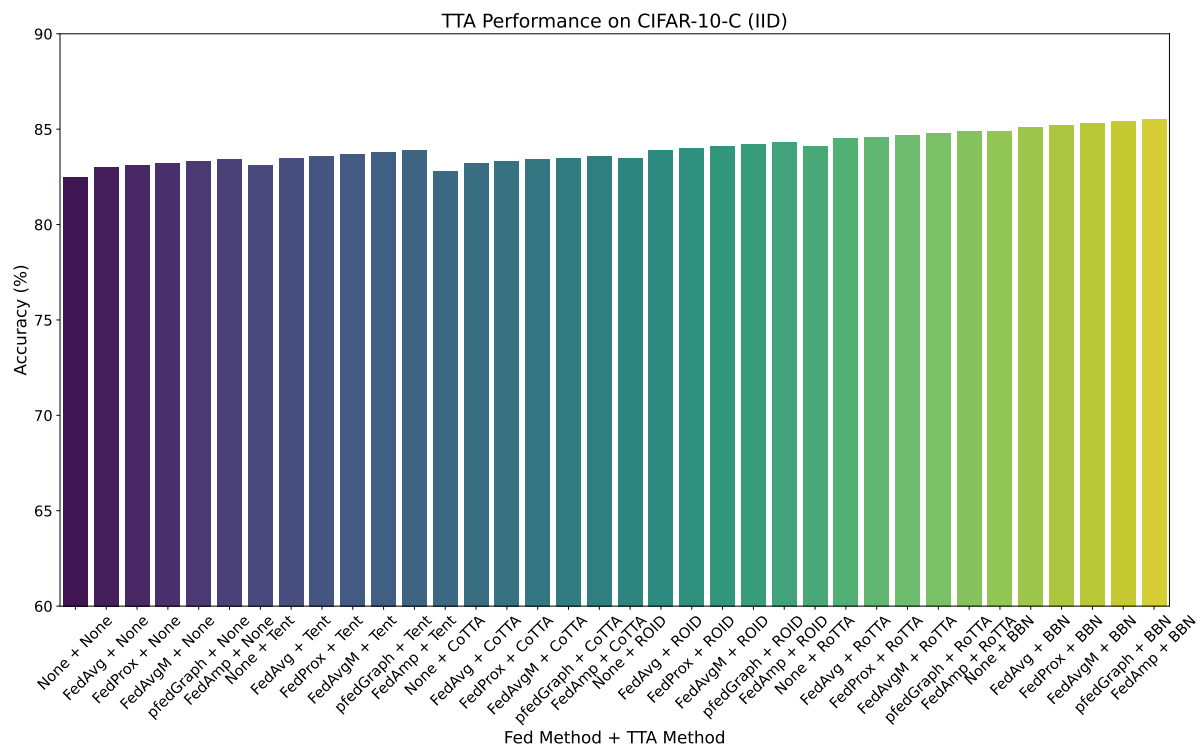


Figure 18: CIFAR-10-C IID performance.

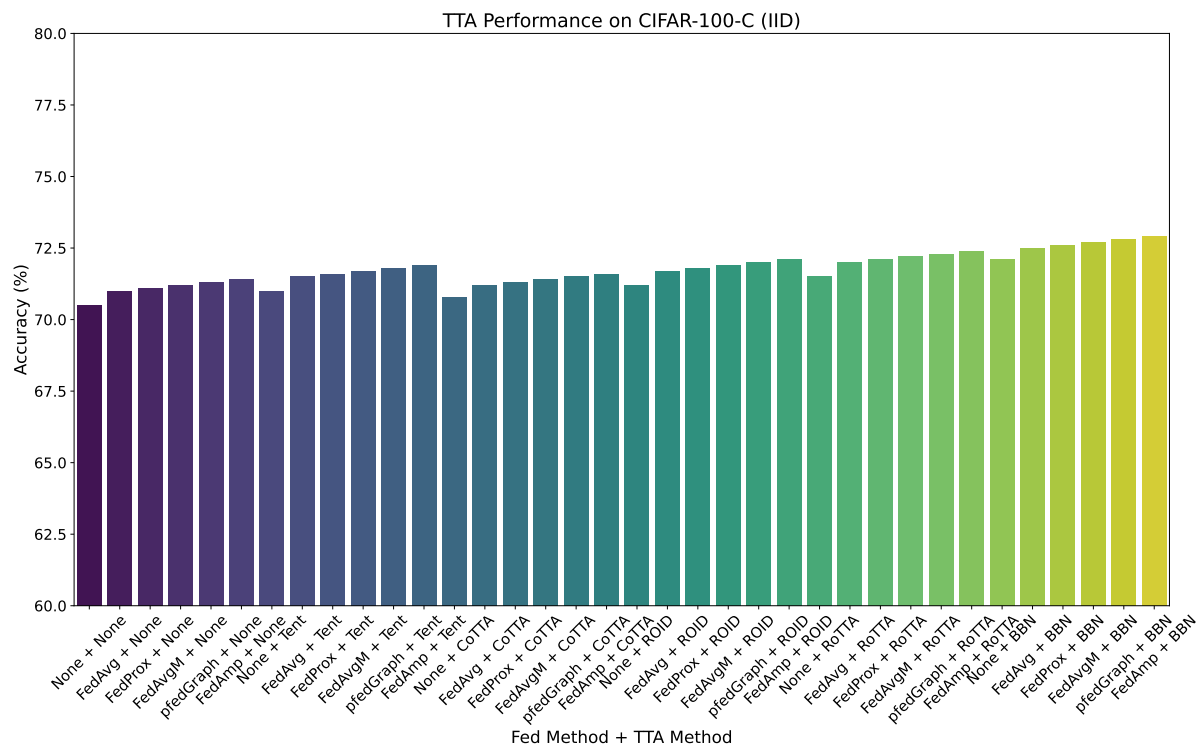


Figure 19: CIFAR-100-C IID performance.

Table 4: Class-wise sample counts for CIFAR-10-C under Dirichlet heterogeneity  $\alpha = 0.01$ , rounds 1–45. Each row shows how many examples of each class a client sees in that test round, illustrating extreme non-IID splits (e.g., class 3 never appears).

Round	Class 0	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9
1	496	5	268	0	200	0	327	200	304	200
2	266	0	199	0	202	132	400	200	401	200
3	200	0	0	0	200	200	400	200	400	400
4	200	0	0	0	200	115	400	200	485	400
5	200	0	0	0	53	10	400	200	736	401
6	496	5	268	0	200	0	327	200	304	200
7	266	0	199	0	202	132	400	200	401	200
8	200	0	0	0	200	200	400	200	400	400
9	200	0	0	0	200	115	400	200	485	400
10	200	0	0	0	53	10	400	200	736	401
11	496	5	268	0	200	0	327	200	304	200
12	266	0	199	0	202	132	400	200	401	200
13	200	0	0	0	200	200	400	200	400	400
14	200	0	0	0	200	115	400	200	485	400
15	200	0	0	0	53	10	400	200	736	401
16	496	5	268	0	200	0	327	200	304	200
17	266	0	199	0	202	132	400	200	401	200
18	200	0	0	0	200	200	400	200	400	400
19	200	0	0	0	200	115	400	200	485	400
20	200	0	0	0	53	10	400	200	736	401
21	496	5	268	0	200	0	327	200	304	200
22	266	0	199	0	202	132	400	200	401	200
23	200	0	0	0	200	200	400	200	400	400
24	200	0	0	0	200	115	400	200	485	400
25	200	0	0	0	53	10	400	200	736	401
26	496	5	268	0	200	0	327	200	304	200
27	266	0	199	0	202	132	400	200	401	200
28	200	0	0	0	200	200	400	200	400	400
29	200	0	0	0	200	115	400	200	485	400
30	200	0	0	0	53	10	400	200	736	401
31	496	5	268	0	200	0	327	200	304	200
32	266	0	199	0	202	132	400	200	401	200
33	200	0	0	0	200	200	400	200	400	400
34	200	0	0	0	200	115	400	200	485	400
35	200	0	0	0	53	10	400	200	736	401
36	496	5	268	0	200	0	327	200	304	200
37	266	0	199	0	202	132	400	200	401	200
38	200	0	0	0	200	200	400	200	400	400
39	200	0	0	0	200	115	400	200	485	400
40	200	0	0	0	53	10	400	200	736	401
41	496	5	268	0	200	0	327	200	304	200
42	266	0	199	0	202	132	400	200	401	200
43	200	0	0	0	200	200	400	200	400	400
44	200	0	0	0	200	115	400	200	485	400
45	200	0	0	0	53	10	400	200	736	401

Table 5: Class-wise sample counts for CIFAR-10-C under Dirichlet heterogeneity  $\alpha = 0.01$ , rounds 46–75. Each row shows how many examples of each class a client sees in that test round, illustrating extreme non-IID splits (e.g., class 3 never appears).

Round	Class 0	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9
46	496	5	268	0	200	0	327	200	304	200
47	266	0	199	0	202	132	400	200	401	200
48	200	0	0	0	200	200	400	200	400	400
49	200	0	0	0	200	115	400	200	485	400
50	200	0	0	0	53	10	400	200	736	401
51	496	5	268	0	200	0	327	200	304	200
52	266	0	199	0	202	132	400	200	401	200
53	200	0	0	0	200	200	400	200	400	400
54	200	0	0	0	200	115	400	200	485	400
55	200	0	0	0	53	10	400	200	736	401
56	496	5	268	0	200	0	327	200	304	200
57	266	0	199	0	202	132	400	200	401	200
58	200	0	0	0	200	200	400	200	400	400
59	200	0	0	0	200	115	400	200	485	400
60	200	0	0	0	53	10	400	200	736	401
61	496	5	268	0	200	0	327	200	304	200
62	266	0	199	0	202	132	400	200	401	200
63	200	0	0	0	200	200	400	200	400	400
64	200	0	0	0	200	115	400	200	485	400
65	200	0	0	0	53	10	400	200	736	401
66	496	5	268	0	200	0	327	200	304	200
67	266	0	199	0	202	132	400	200	401	200
68	200	0	0	0	200	200	400	200	400	400
69	200	0	0	0	200	115	400	200	485	400
70	200	0	0	0	53	10	400	200	736	401
71	496	5	268	0	200	0	327	200	304	200
72	266	0	199	0	202	132	400	200	401	200
73	200	0	0	0	200	200	400	200	400	400
74	200	0	0	0	200	115	400	200	485	400
75	200	0	0	0	53	10	400	200	736	401

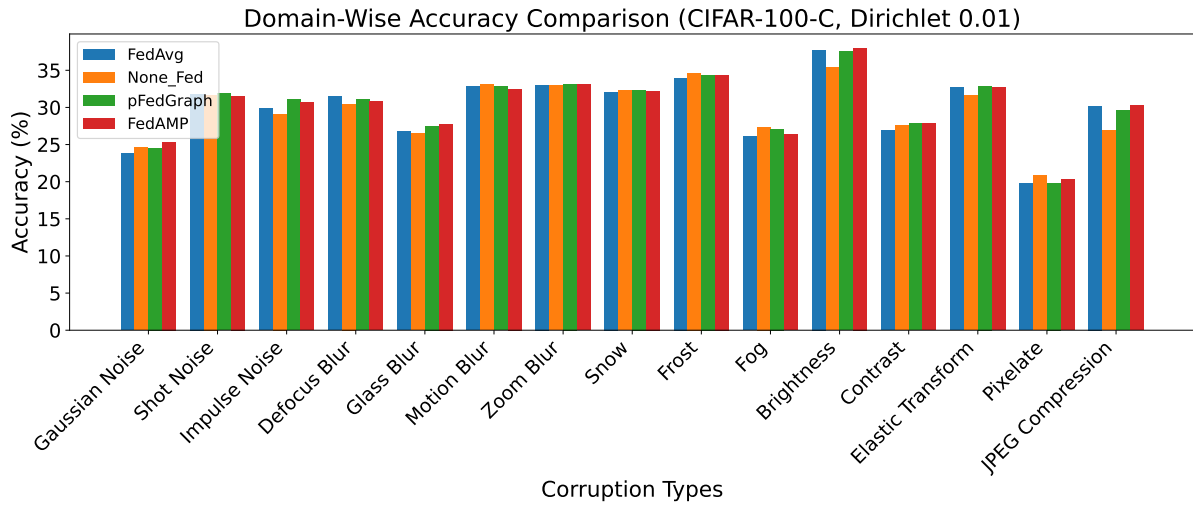


Figure 20: Domain-wise accuracy comparison of our method (Dirichlet  $\alpha = 0.01$ ) across CIFAR-100-C corruptions.

Table 6: Domain-wise average accuracy (%) on CIFAR-10-C (severity-5) under Dirichlet heterogeneity  $\alpha = 0.01$ , comparing five TTA methods (TENT, CoTTA, ROID, RoTTA, BBN) across four federation schemes: FedAvg, None-Fed, pFedGraph, and FedAMP. Each block of five columns corresponds to one federation strategy, with “None-Fed” values indicating no federated strategy. The accuracy values under 15 corruption types are shown in the table.

Corruption	FedAvg					None-Fed				
	Tent	CoTTA	ROID	RoTTA	BBN	Tent	CoTTA	ROID	RoTTA	BBN
Gaussian Noise	27.61	38.48	29.54	49.35	50.81	27.60	32.90	29.53	38.21	48.39
Shot Noise	27.93	43.30	29.57	58.66	64.45	27.86	36.98	29.60	32.25	56.37
Impulse Noise	25.49	36.81	26.60	48.14	51.47	25.46	33.53	26.61	41.64	37.20
Defocus Blur	30.88	40.44	33.11	49.99	50.70	30.82	40.44	33.08	14.10	38.01
Glass Blur	24.02	39.19	25.10	54.35	56.74	24.01	30.82	25.16	37.63	36.58
Motion Blur	30.92	46.89	32.67	62.86	66.80	30.92	40.37	32.69	33.85	51.72
Zoom Blur	30.76	46.89	33.04	63.01	70.50	30.68	46.89	33.02	65.85	59.04
Snow	29.75	50.29	31.35	72.82	68.50	29.76	51.16	31.36	48.01	58.49
Frost	30.55	53.32	32.10	76.09	76.74	30.54	47.54	32.08	50.95	68.36
Fog	31.30	49.44	33.01	67.57	73.26	31.29	49.44	32.98	52.19	60.41
Brightness	32.54	56.46	34.25	81.75	81.17	32.43	56.03	34.28	46.79	74.21
Contrast	31.18	39.17	32.82	47.16	49.33	31.10	36.03	32.83	9.07	41.22
Elastic Transform	28.05	42.33	28.76	56.61	61.05	28.05	36.57	28.76	17.90	52.74
Pixelate	28.97	36.65	30.22	44.33	50.08	28.97	40.20	30.19	33.02	47.09
JPEG Compression	26.89	39.17	27.45	57.82	62.05	26.90	33.90	27.46	24.82	56.64

Corruption	pFedGraph					FedAMP				
	Tent	CoTTA	ROID	RoTTA	BBN	Tent	CoTTA	ROID	RoTTA	BBN
Gaussian Noise	50.81	39.21	29.52	50.81	52.58	50.75	39.21	29.48	50.75	52.35
Shot Noise	59.36	43.64	29.59	59.36	64.08	60.88	44.41	29.54	60.88	61.31
Impulse Noise	49.04	37.27	26.62	49.04	52.84	50.59	38.05	26.49	50.59	51.98
Defocus Blur	49.97	40.42	33.11	49.97	47.74	44.32	37.63	32.92	44.32	44.75
Glass Blur	54.27	39.15	25.10	54.27	54.30	54.23	39.12	24.96	54.23	48.67
Motion Blur	62.99	46.95	32.74	62.99	64.46	60.95	45.95	32.60	60.95	58.09
Zoom Blur	62.88	46.92	33.04	62.88	68.93	63.27	46.86	32.95	63.27	66.86
Snow	72.95	50.36	31.37	72.95	66.66	70.67	48.05	31.19	70.67	64.76
Frost	76.10	53.35	32.05	76.10	76.49	74.63	52.04	31.92	74.63	73.62
Fog	67.55	49.43	33.00	67.55	70.61	64.76	44.66	32.88	64.76	68.73
Brightness	81.77	56.49	34.21	81.77	80.76	81.02	53.95	34.10	81.02	77.39
Contrast	47.40	39.22	32.86	47.40	47.40	39.90	39.21	32.72	39.90	45.21
Elastic Transform	56.39	35.52	28.75	56.39	59.69	55.67	36.88	28.58	55.67	53.76
Pixelate	42.07	42.61	30.17	42.07	49.52	44.79	42.74	30.14	44.79	48.36
JPEG Compression	58.32	39.29	27.48	58.32	61.09	58.56	42.74	27.38	58.56	57.48

Table 7: Client-wise accuracy (%) on CIFAR-10-C under Dirichlet  $\alpha = 0.01$  Accuracy is reported for each client (0–9) using five TTA methods (Tent, CoTTA, ROID, RoTTA, BBN) across four federated learning setups: FedAvg, pFedGraph, FedAMP, and None-Fed. The bottom row shows the average accuracy across clients, highlighting the impact of personalization under non-IID settings.

Client	FedAvg					None-Fed				
	Tent	CoTTA	ROID	RoTTA	BBN	Tent	CoTTA	ROID	RoTTA	BBN
0	18.40	22.74	18.79	27.08	28.08	18.06	17.57	18.95	16.66	17.66
1	2.02	4.32	2.02	6.62	7.62	2.02	4.32	2.03	4.07	5.07
2	2.03	4.55	2.10	7.07	8.07	2.02	4.55	2.13	4.91	5.91
3	13.19	16.58	13.82	19.97	20.97	13.13	16.05	13.83	10.77	11.77
4	5.41	8.88	5.62	12.36	13.36	5.38	8.88	5.72	8.42	9.42
5	17.28	19.11	17.69	20.94	21.94	17.33	19.11	17.72	13.86	14.86
6	13.57	15.75	14.06	16.43	17.43	13.53	15.75	14.02	11.66	12.66
7	2.02	4.13	2.31	6.23	7.23	2.00	4.13	2.32	4.16	5.16
8	10.41	12.31	10.46	14.16	15.16	10.41	12.31	11.18	7.77	8.77
9	2.29	4.46	2.59	6.63	7.63	2.27	4.46	2.56	3.32	4.32
Mean	8.66	11.21	8.95	13.75	14.75	8.61	11.21	9.05	8.56	9.56
Client	pFedGraph					FedAMP				
	Tent	CoTTA	ROID	RoTTA	BBN	Tent	CoTTA	ROID	RoTTA	BBN
0	18.40	22.74	18.79	27.45	28.45	18.40	22.74	18.76	27.09	28.09
1	2.02	4.32	2.02	6.38	7.38	2.02	4.32	2.00	6.15	7.15
2	2.03	4.55	2.10	7.14	8.14	2.03	4.55	2.08	6.78	7.78
3	13.19	16.58	13.83	20.10	21.10	13.19	16.58	13.81	20.14	21.14
4	5.41	8.88	5.61	12.44	13.44	5.41	8.88	5.59	11.65	12.65
5	17.28	19.11	17.69	20.56	21.56	17.29	19.11	17.69	20.30	21.30
6	13.57	15.75	14.06	16.97	17.97	13.57	15.75	14.04	16.31	17.31
7	2.02	4.13	2.31	6.28	7.28	2.03	4.13	2.29	6.38	7.38
8	10.41	12.31	10.45	14.27	15.27	10.41	12.31	10.44	14.16	15.16
9	2.29	4.46	2.60	6.73	7.73	2.29	4.46	2.59	6.72	7.72
Mean	8.66	11.21	8.95	13.83	14.83	8.66	11.21	8.93	13.57	14.57