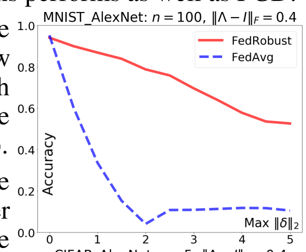
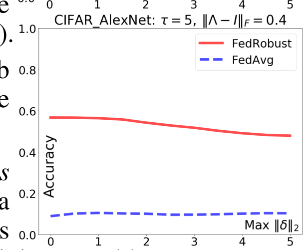
We thank the reviewers for their careful consideration and their feedback. We provide our responses below.

Response regarding numerical results. -*"Adversarial robustness gained by `FedRobust` compared to distributed PGD and FGM methods:"* **(R1, R3)** As noted by the reviewers, `FedRobust` achieved a similar (or superior) adversarial robustness to the standard PGD training. This observation can be explained by analyzing the generalization properties of these algorithms. We note that `FedRobust`'s improved robustness was obtained over the **test** samples. On the other hand, PGD consistently outperformed `FedRobust` on the **training** samples, achieving a near perfect training accuracy. However, `FedRobust` generalized better to the test samples and could overall outperform PGD on the test set. We will include the training performance scores in the final version for clarification. Also, the similar performance of FGM and PGD can be explained via the random Gaussian perturbations used for simulating the heterogeneity across clients and the results of Wong et al. (ICLR 2020) indicating FGM initialized at random perturbations performs as well as PGD.





-*"Effect of network size n and # of local updates τ; other datasets:"* **(R1, R2)** We have performed several new experiments to study the effect of larger n and τ and will add the new results. Here, we report some preliminary results for $n = 100$ (top) and $\tau = 5$ (bottom). Both results indicate that `FedRobust` still offers a significant robustness gain over `FedAvg`. We will also conduct experiments using the suggested LEAF framework (FEMNIST dataset).

-*"Computation and communication times in speed comparisons:"* **(R1, R2)** The time comparison made in the main body is in terms of the computation time for the same number of training iterations. We note that in our experiments the methods will share the same communication time as they have been trained for the same $10,000$ iterations (and rounds).

-*"Step-size of PGD and FGM:"* **(R3)** For PGD training, we used the standard rule of thumb to choose step-size $\frac{2}{k}\epsilon_{\text{pgd}}$ for each of $k = 10$ PGD steps. For FGM training, the effective step-size is the same as ϵ_{fgm}, since FGM normalizes the single-step perturbation.

Response regarding theoretical results. -*"Summary of contributions, technical advances and Theorems 1-4 in tandem:"* **(R3)** We first consider a heterogeneity model where the data distribution at each node is an affine transformation of a mother distribution. Using this model, we formulate a robust federated optimization problem in eq. (3). To solve this minimax problem, we propose a communication-computation efficient optimization algorithm (`FedRobust`) and show its convergence (Theorems 1, 2). This paper is the first work that integrates proof techniques from local SGD, minimax optimization, federated optimization, and provides provable robust federated methods. Other existing results in distributed minimax optimization and non-robust federated learning can be retrieved as special cases of our results. Then in Theorem 3, we ensure that when a new client with unseen data joins the federated network, the model learned by solving (3) is properly generalized. Finally, in Theorem 4 we connect our proposed minimax formulation (3) to distributionally robust optimization by showing that it indeed optimizes a lower-bound on the distributionally robust problem with Wasserstein cost in (7).

-*"Theorem 2 vs. prior results in distributed minimax optimization, effect of τ:"* **(R1, R2)** Theorems 1, 2 characterize the convergence rates of `FedRobust` in which, each of the clients runs τ local updates in each round. General distributed minimax optimization algorithms can be viewed as special cases for $\tau = 1$. The effect of running more than one local update ($\tau > 1$) in the convergence rates are demonstrated in both theorems by the terms containing $(\tau - 1)$. Analysing the effect of $\tau > 1$ is indeed a technical challenge in our convergence analysis **(R2)**. At a high-level, τ controls the computation-communication trade-off as larger τ implies less communication at the expense of more computation **(R1)**.

-*"Technical novelty of Theorem 3:"* **(R2, R3)** We note that the distribution shift considered in this paper is device-dependent, i.e. all the samples stored at node i undergo the same transformation $\Lambda^i \mathbf{x} + \delta^i$. This is unlike the prior works in adversarial training such as Farnia et al. (2018), where *each* data sample is affected by a different transformation. Moreover, the affine shift considered in our paper is specified with two variable Λ, δ, generalizing over the prior works considering only δ. These two challenges distinguish Theorem 3 **(R2)**. We also note that our proof of Theorem 3 uses the PAC-Bayes framework (McAllester, 1999), while Bartlett et al. (2017) analyzes the Rademacher complexity **(R3)**.

-*"Discussion on learning rates:"* **(R3)** The conditions on η_1, η_2 in Theorem 1 can be rewritten as linear constraints and are always feasible. Rewriting the last condition as linear in η_1, η_2: $\eta_1 \hat{L} + 40(3\eta_1 L_1^2 + \eta_2 L_{21}^2)(\tau - 1)\mu_1^{-1}(1 - \frac{\mu_1}{8\sqrt{2}L_1})^{-1} \leq 1$. E.g.: $\eta_1 = c_1 \ln(T)/T, \eta_2 = c_2 \ln(T)/T$ where $c_2 = [160\kappa n L]^{-1}$ and $c_1 = \min\{[(482\kappa + 6)\tau L]^{-1}, [1440\kappa^3 n L]^{-1}\}$.

-*"Practicality of the affine model in FL; nonparametric regime; effect of matrices Λ"* **(R4)** The affine model considered in this paper is particularly practical for image classification tasks in FL as also elaborated in the introduction, where each camera's imperfections affect its pictures (Robey et al., 2020). While this model provides significant robustness compared to additive-only perturbation models (i.e. $\Lambda = I$), it lays out potential new directions to study more complicated (non-affine) models such as neural network transformations. The nonparametric regime is another interesting generalization of this work, however in this case, nodes might need to solve a maximization problem at each iteration which can be problematic due to limited computation in federated settings. Diagonal Λ is also another interesting special case, however it may also fall short in capturing different filtering functions, e.g. rotation of images.