Differentially Private Federated Learning with Normalized Updates

author names withheld

Under Review for OPT 2022

Abstract

The customary approach for client-level differentially private federated learning (FL) is to add Gaussian noise to the average of the clipped client updates. Clipping is associated with the following issue: as the client updates fall below the clipping threshold, they get drowned out by the added noise, inhibiting convergence. To mitigate this issue, we propose replacing clipping with normalization, where we use only a scaled version of the unit vector along the client updates. Normalization ensures that the noise does not drown out the client updates even when the original updates are small. We theoretically show that the resulting normalization-based private FL algorithm attains better convergence than its clipping-based counterpart on convex objectives in over-parameterized settings.

1. Introduction

The typical approach to *client/user*-level differentially private federated learning (FL), exemplified by `DP-FedAvg` [15, 29], involves each client first computing its update as it would have been under vanilla `FedAvg`[23], then clipping its norm to a pre-determined threshold, and adding noise to the clipped update before sending it to the server. The level of added noise is always proportional to the clipping threshold, with the proportionality constants depending on the desired level of privacy.

Our **key insight** is that this *clipping*-based approach has a fundamental issue: when the client updates have small magnitudes much lesser than the clipping threshold, they are completely drowned out by the added noise whose magnitude is always proportional to the clipping threshold. Based on this insight, we propose a variant algorithm – one that does not clip the client updates, but instead *normalizes updates, i.e., rescales them to always have a fixed norm*, before adding noise proportional to this fixed norm for privacy. We call our modified method `DP-NormFedAvg`. The rescaling of updates in `DP-NormFedAvg` ensures that client updates are never overwhelmed by the added noise, even if the original updates are small.

We succinctly state the updates of `FedAvg`, `DP-FedAvg` with clipping and `DP-NormFedAvg` in Table 1. `DP-FedAvg` with clipping and `DP-NormFedAvg` are stated in detail in Algorithm 1 Options (i) and (ii), respectively. We also show an example of the superior convergence of normalization compared to clipping in Figure 1.

Our **main contributions** are summarized next:
(a) In Section 3, we present `DP-NormFedAvg` (Alg. 1 Option (ii)) where we replace the usual practice of update clipping by update *normalization* (i.e., using a scaled version of the *unit vector* along the update) for bounding sensitivity in private FL. Our motivation for advocating normalization is that *normalization has a higher signal (viz., update norm) to noise ratio (SNR) than clipping* which should intuitively result in better convergence for normalization; this aspect is discussed

© .

Algorithm:	FedAvg	DP-FedAvg with Clipping	DP-NormFedAvg
Client Sends:	$\boldsymbol{u}$	$\boldsymbol{u}\min(1, \frac{C}{\lVert\boldsymbol{u}\rVert_2}) + C\boldsymbol{\zeta}$	$\frac{C\boldsymbol{u}}{\lVert\boldsymbol{u}\rVert_2} + C\boldsymbol{\zeta}$

Table 1: Summary of what each client in vanilla FedAvg, DP-FedAvg with clipping and DP-NormFedAvg sends to the server. Here, $\boldsymbol{u}$ is the client update of the vanilla FedAvg algorithm, $\boldsymbol{\zeta}$ is Gaussian noise whose variance depends on the desired privacy level and C is the clipping threshold/scaling factor.

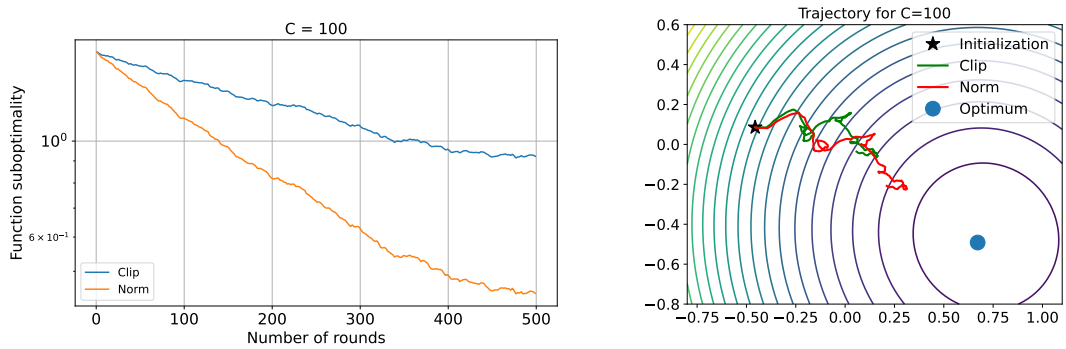


Figure 1: For a quadratic objective in the FL setting (described in Appendix F), function suboptimality (i.e., $f(\boldsymbol{w}) - \min_{\boldsymbol{w}'} f(\boldsymbol{w}')$) vs. round number on the left, and smoothed 2D projection of the trajectories of DP-FedAvg with clipping ("Clip") and DP-NormFedAvg ("Norm") on the right. DP-NormFedAvg reaches closer to the optimum and attains a smaller function suboptimality than DP-FedAvg with clipping.

in Section 3.1. We provide convergence results for DP-NormFedAvg as well as the standard algorithm of DP-FedAvg with clipping (Alg. 1 Option (i)) in the smooth convex case in Theorem 4. In Section 4.1, we compare the convergence results of both algorithms, showing that normalization has better asymptotic convergence than clipping in over-parameterized settings.

(b) In Appendix F and G, we demonstrate the superiority of normalization over clipping via experiments on a synthetic quadratic problem as well as on three benchmarking datasets, viz., Fashion MNIST, CIFAR-10 and CIFAR-100, respectively. For $\varepsilon = 5$, the improvement offered by normalization over clipping w.r.t. test accuracy is $> 2.8\%$, 2.1% and 1.5% for CIFAR-100, Fashion MNIST and CIFAR-10, respectively; see Table 2 in Appendix G.

2. Preliminaries

Federated Learning (FL) Setting: There are n clients, each with their own decentralized data, and a central server that has to train a model, parameterized by $\boldsymbol{w} \in \mathbb{R}^d$, using the clients' data. Suppose the ith client has m training examples drawn from some distribution $\mathcal{P}_i$. Then the ith client has an objective function $f_i(\boldsymbol{w})$ which is the average loss, w.r.t. some loss function, over its m samples, and the central server tries to minimize the average [1] loss $f(\boldsymbol{w})$, over the n clients, i.e., $f(\boldsymbol{w}) := \frac{1}{n}\sum_{i=1}^{n} f_i(\boldsymbol{w})$. The setting where the data distributions of all the clients are identical, i.e.

1. In general, each client may have different number of samples and this average is a weighted one with the weight of a client being proportional to the number of samples it has. We consider the case of equal number of samples per client for simplicity.

$\mathcal{P}_1 = \ldots = \mathcal{P}_n$, is known as the "homogeneous" setting. Other settings are known as "heterogeneous" settings. We are able to naturally quantify the effect of heterogeneity on convergence as follows.

Definition 1 (Heterogeneity) *Let $\boldsymbol{w}^* \in \arg\min_{\boldsymbol{w}' \in \mathbb{R}^d} f(\boldsymbol{w}')$ and $\Delta_i^* := f_i(\boldsymbol{w}^*) - \min_{\boldsymbol{w}' \in \mathbb{R}^d} f_i(\boldsymbol{w}')$. The system's heterogeneity is quantified by some increasing function of $\{\Delta_i^*\}_{i=1}^n$.*

The exact function of Δ_i^*'s quantifying heterogeneity depends on the algorithm as well as data, and this will become clear when we present the convergence results. Note that if the per-client distributions (i.e, $\mathcal{P}_i$'s) are similar, then we expect the Δ_i^*'s to be small indicating smaller heterogeneity.

Due to lack of space, we present the definitions of some standard concepts such as differential privacy and the Gaussian mechanism in Appendix A.

Definition 2 (A Key Quantity) *All the theoretical results in this paper are expressed in terms of the following key quantity*

$$\rho := \frac{\sqrt{qd\log(1/\delta)}}{n\varepsilon}, \tag{1}$$

where (ε, δ)-DP is the desired privacy level, n is the number of clients, d is the parameter dimension and q is the absolute constant in Theorem 3. We also assume that n is sufficiently large so that $\rho < 1$. Note that ρ increases as the level of privacy increases (i.e., ε and δ decrease), and vice versa.

Notation: Throughout the rest of this paper, we denote the ℓ_2 norm by $\|.\|$ (omitting the subscript 2). Vectors and matrices are written in boldface. For any $a \in \mathbb{N}$, we denote the set $\{1, \ldots, a\}$ by $[a]$, and the uniform distribution over $\{0, \ldots, a\}$ by unif$[0, a]$. For a function h and any point $\boldsymbol{\theta}$ in its domain Θ, the "*suboptimality gap*" at $\boldsymbol{\theta}$ means $h(\boldsymbol{\theta}) - \min_{\boldsymbol{\theta}' \in \Theta} h(\boldsymbol{\theta}')$.

The clipping function clip : $\mathbb{R}^d \times \mathbb{R}^+ \to \mathbb{R}^d$ is defined as $\text{clip}(\boldsymbol{z}, c) := \boldsymbol{z}\min(1, \frac{c}{\|\boldsymbol{z}\|})$; here, c is known as the clipping threshold. The normalization function norm : $\mathbb{R}^d - \{\mathbf{0}_d\} \times \mathbb{R}^+ \to \mathbb{R}^d$ is defined as $\text{norm}(\boldsymbol{z}, c) := \frac{c\boldsymbol{z}}{\|\boldsymbol{z}\|}$; here, c is the scaling factor and it is analogous to the clipping threshold in the clip(.) function. Also, note that $\|\text{clip}(\boldsymbol{z}, c)\| \leq \|\text{norm}(\boldsymbol{z}, c)\| \leq c$.

Related work has been deferred to Appendix B due to lack of space.

3. `DP-NormFedAvg`: Differentially Private FL with Client-Update Normalization

In Algorithm 1, we jointly state `DP-FedAvg` with client-update **clipping**, which is the standard algorithm for *client-level* private FL, and our proposed algorithm `DP-NormFedAvg`, which is `DP-FedAvg` with client-update **normalization** (instead of clipping)[2]. Note that both these algorithms only differ in line 9. Also, we call the parameter C in Algorithm 1 "clipping threshold" for `DP-FedAvg` with clipping, and "scaling factor" for `DP-NormFedAvg`.

In contrast to (non-private) `FedAvg` (presented for completeness in Appendix A), each client in the selected subset of clients in Alg. 1 sends its *clipped* or *normalized* update plus zero-mean Gaussian noise (for differential privacy) to the server. The server then computes the mean of the noisy *clipped* or *normalized* client updates that it received (i.e., $\boldsymbol{a}_k$) and uses it to update the global model similar to `FedAvg`, except with a potentially different global learning rate (β_k) than the local learning rate (η_k). Based on Thm. 1 in [1], we now specify the value of σ^2 required to make Alg. 1 (with either clipping or normalization) (ε, δ)-DP.

2. In Algorithm 1, we are using full gradients in the local updates for ease of analysis and to simplify presentation of results. Our results can be extended to stochastic gradients as well.

Theorem 3 ([1]) *For any* $0 < \varepsilon < \mathcal{O}\left(\frac{r^2 K}{n^2}\right)$*, Algorithm 1 is* (ε, δ)*-DP for* $\sigma^2 = qKC^2\left(\frac{\log(1/\delta)}{n^2\varepsilon^2}\right)$*, where* $q > 0$ *is an absolute constant.*

The DP-SGD algorithm of [1] returns the last iterate (i.e., $\boldsymbol{w}_K$) as the output, and Theorem 1 in [1] guarantees that the last iterate is (ε, δ)-DP by setting σ^2 as above. But if the last iterate is (ε, δ)-DP, then so is *any* other iterate (due to additivity of the privacy cost), from which Theorem 3 follows.

Algorithm 1 Option (i) is `DP-FedAvg` with Clipping, and Option (ii) is `DP-NormFedAvg`

Input: Initial point $\boldsymbol{w}_0$, number of rounds of communication K, number of local updates per round E, local learning rates $\{\eta_k\}_{k=0}^{K-1}$, global learning rates $\{\beta_k\}_{k=0}^{K-1}$, clipping threshold/scaling factor C, client sampling probability in each round r/n and noise variance σ^2.
for $k = 0, \ldots, K-1$ **do**
  Server sends $\boldsymbol{w}_k$ to a random set $\mathcal{S}_k$ of clients, formed by sampling each client $\in [n]$ with probability r/n.
  for client $i \in \mathcal{S}_k$ **do**
    Set $\boldsymbol{w}_{k,0}^{(i)} = \boldsymbol{w}_k$.
    for $\tau = 0, \ldots, E-1$ **do**
      Update $\boldsymbol{w}_{k,\tau+1}^{(i)} \leftarrow \boldsymbol{w}_{k,\tau}^{(i)} - \eta_k \nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})$.
    end for
    Let $\boldsymbol{u}_k^{(i)} = \frac{(\boldsymbol{w}_k - \boldsymbol{w}_{k,E}^{(i)})}{\eta_k}$. // $\boldsymbol{u}_k^{(i)}$ `is client` i`'s update.`
    Option (i): $\boldsymbol{g}_k^{(i)} = \mathrm{clip}\left(\boldsymbol{u}_k^{(i)}, C\right) = \boldsymbol{u}_k^{(i)} \min\left(1, \frac{C}{\|\boldsymbol{u}_k^{(i)}\|}\right)$. // `Clipping`
    Option (ii): $\boldsymbol{g}_k^{(i)} = \mathrm{norm}\left(\boldsymbol{u}_k^{(i)}, C\right) = \frac{C\boldsymbol{u}_k^{(i)}}{\|\boldsymbol{u}_k^{(i)}\|}$. // `Normalization`
    Send $(\boldsymbol{g}_k^{(i)} + \boldsymbol{\zeta}_k^{(i)})$ to the server, where $\boldsymbol{\zeta}_k^{(i)} \sim \mathcal{N}(\mathbf{0}_d, r\sigma^2 \mathrm{I}_d)$.
  end for
  Update $\boldsymbol{w}_{k+1} \leftarrow \boldsymbol{w}_k - \beta_k \boldsymbol{a}_k$, where $\boldsymbol{a}_k = \frac{1}{r}\sum_{i \in \mathcal{S}_k}(\boldsymbol{g}_k^{(i)} + \boldsymbol{\zeta}_k^{(i)})$.
end for
Return $\boldsymbol{w}_{\text{priv}} = \boldsymbol{w}_{\tilde{k}}$, where $\tilde{k} \sim \mathrm{unif}[0, K-1]$.

Now, let us talk about why normalization might be a better choice than clipping.

3.1. Intuition of why Normalization may outperform Clipping

Intuitively, clipping has the following issue with respect to optimization: as the client update norms decrease and fall below the clipping threshold, the norm of the added noise (which has constant expectation proportional to the clipping threshold, regardless of the client update norms) can become arbitrarily larger than the client update norms, which should hamper convergence. This issue is not as grave in `DP-NormFedAvg` because its update-*normalization* step ensures that the noise norm cannot become arbitrarily larger than the *normalized* update's norm, even if the original update's norm is small. In other words, *the signal (i.e., update norm) to noise ratio (SNR) of clipping eventually falls below that of normalization*, due to which we expect normalization to have better convergence.

Having said all this, it is worth noting that if the client update norms are lower bounded by C_{low}, then clipping with threshold $C \leq C_{\text{low}}$ is equivalent to normalization with the same scaling factor.

4. Convergence Results of `DP-FedAvg` with Clipping and `DP-NormFedAvg`

We now present convergence results for `DP-FedAvg` with clipping and `DP-NormFedAvg` for the same choices of hyper-parameters in Alg. 1. The detailed results and their proofs for `DP-FedAvg` with clipping and `DP-NormFedAvg` can be found in Appendices C and D. These results are for the non-vacuous privacy regime (i.e., when ε is finite and $\delta < 1$), where $\rho = \sqrt{qd\log(1/\delta)}/n\varepsilon > 0$.

Theorem 4 (Convex) *Suppose each f_i is convex, L-smooth*[3] *and G-Lipschitz*[4] *over $\mathbb{R}^d$. Pick some $\gamma > 0$ and $\alpha \geq 1$. In Algorithm 1, choose $E \leq \frac{\alpha}{2\rho}$ and set $\beta_k = \eta_k = \frac{\rho}{2\alpha L}$, $C = GE$ and $K = \left(\frac{2\alpha\gamma}{C}\right)\frac{1}{\rho^2}$. Sample $\tilde{k} \sim \text{unif}[0, K-1]$. Then for any $\boldsymbol{w}^* \in \arg\min_{\boldsymbol{w}' \in \mathbb{R}^d} f(\boldsymbol{w}')$ and $\Delta_i^* := f_i(\boldsymbol{w}^*) - \min_{\boldsymbol{w}' \in \mathbb{R}^d} f_i(\boldsymbol{w}')$,*
(a) *Algorithm 1 Option (i), i.e. `DP-FedAvg` with clipping, has the following guarantee:*

$$\left(2 - \frac{\rho E}{\alpha} - \frac{\rho^2 E^2}{\alpha^2}\right)\mathbb{E}\left[f(\boldsymbol{w}_{\tilde{k}}) - f(\boldsymbol{w}^*)\right] \leq \underbrace{G\left(\frac{L\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|^2}{\gamma} + \frac{\gamma}{L}\right)\rho}_{:=\text{A (effect of \textbf{initialization})}} + \underbrace{\left(\frac{3E}{2\alpha}\right)\left(\frac{1}{n}\sum_{i=1}^{n}\Delta_i^*\right)\rho}_{:=\text{B}_1\text{ (effect of \textbf{heterogeneity})}} . \tag{2}$$

(b) *Algorithm 1 Option (ii), i.e. `DP-NormFedAvg`, has the following guarantee:*

$$\left(2 - \frac{\rho^2 E^2}{\alpha^2}\right)\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\left(\frac{GE}{\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|}\right)(f_i(\boldsymbol{w}_{\tilde{k}}) - f_i(\boldsymbol{w}^*))\right]$$
$$\leq \underbrace{G\left(\frac{L\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|^2}{\gamma} + \frac{\gamma}{L}\right)\rho}_{:=\text{A (effect of \textbf{initialization})}} + \underbrace{\left(\frac{E}{\alpha}\right)\left(\frac{G^2}{2L} + \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\left(\frac{GE}{\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|}\right)\Delta_i^*\right]\frac{\rho E}{\alpha}\right)\rho}_{:=\text{B}_2\text{ (effect of \textbf{heterogeneity})}} . \tag{3}$$

In the results above, we remind the reader that $\boldsymbol{u}_{\tilde{k}}^{(i)}$ is the ith client's update at a random round number $\tilde{k}$ (see line 9 in Algorithm 1).

Effect of initialization and heterogeneity: The convergence results above depend on two things: (i) the distance of the **initialization** $\boldsymbol{w}_0$ from the optimum $\boldsymbol{w}^*$, appearing in term A in both eq. (2) and eq. (3), and (ii) the degree of **heterogeneity** which depends on the Δ_i^*'s (as per Definition 1), appearing in terms B_1 and B_2 in eq. (2) and eq. (3), respectively. A high (respectively, low) degree of heterogeneity implies high (respectively, low) values of Δ_i^*'s, which leads to worse (respectively, better) convergence. Also, as we increase α in Theorem 4, i.e., increase the number of rounds K, the effect of heterogeneity dies down for both clipping and normalization. However, the effect of the initialization term (A) cannot be diminished by increasing α.

4.1. Asymptotic Comparison of `DP-FedAvg` with Clipping and `DP-NormFedAvg`

Let us theoretically compare clipping and normalization when the number of rounds $K \to \infty$, i.e. *asymptotically*. The asymptotic comparison is reasonable in practice where training happens for a very large number of rounds. In Theorem 4, recall that we set $K = \left(\frac{2\alpha\gamma}{CE}\right)\frac{1}{\rho^2}$, where $\alpha \geq 1$ is a

3. i.e., $\|\nabla f_i(\boldsymbol{w}_1) - \nabla f_i(\boldsymbol{w}_2)\| \leq L\|\boldsymbol{w}_1 - \boldsymbol{w}_2\| \; \forall \, \boldsymbol{w}_1, \boldsymbol{w}_2 \in \mathbb{R}^d$.
4. i.e., $\sup_{\boldsymbol{w} \in \mathbb{R}^d} \|\nabla f_i(\boldsymbol{w})\| \leq G$. We assume Lipschitzness in Theorem 4 to simplify the presentation of results. The detailed results in Appendices C and D do not assume Lipschitzness, but are messier.

parameter of our choice; thus we can make $K \to \infty$ by choosing $\alpha \to \infty$. Here we keep $E = \mathcal{O}(1)$. Then, the *asymptotic* convergence bound of `DP-FedAvg` with clipping can be written as:

$$\underbrace{\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}(f_i(\boldsymbol{w}_{\tilde{k}}) - f_i(\boldsymbol{w}^*))\right]}_{=\mathbb{E}[f(\boldsymbol{w}_{\tilde{k}})-f(\boldsymbol{w}^*)]} \leq \frac{G}{2}\Big(\frac{L\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|^2}{\gamma} + \frac{\gamma}{L}\Big)\rho, \tag{4}$$

where $\tilde{k} \sim \text{unif}[0, K-1]$. In comparison, the corresponding bound of `DP-NormFedAvg` is:

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\underbrace{\left(\frac{GE}{\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|}\right)}_{\geq 1}(f_i(\boldsymbol{w}_{\tilde{k}}) - f_i(\boldsymbol{w}^*))\right] \leq \frac{G}{2}\Big(\frac{L\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|^2}{\gamma} + \frac{\gamma}{L}\Big)\rho. \tag{5}$$

Note that the RHS of both equations is the same; the RHS depends only on the initialization as the effect of heterogeneity is eliminated asymptotically. The only difference between the LHS of the two equations is due to the $\left({}^{GE}/_{\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|}\right)$ terms in eq. (5). Note that $\left({}^{GE}/_{\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|}\right) \geq 1$ because f_i is G-Lipschitz [5]. So in situations where $f_i(\boldsymbol{w}_{\tilde{k}}) \geq f_i(\boldsymbol{w}^*)$ for all $i \in [n]$, the LHS of eq. (5) is at least as large the LHS of eq. (4); in this case, we expect the convergence of normalization to be at least as good as that of clipping in private optimization (because both equations have the same RHS). An example of such a situation is over-parameterization where each minimizer of f is also a minimizer of *all* the f_i's but not the other way around [22]; this can be expected in the homogeneous case.

Due to lack of space, we defer the empirical results to Appendix F and G.

5. Conclusion

We proposed `DP-NormFedAvg` which normalizes client updates instead of clipping them. Theoretically, we argued that `DP-NormFedAvg` should have better asymptotic convergence than `DP-FedAvg` with clipping, at least in the over-parameterized case. Intuitively, this happens because normalization has a higher signal (i.e., update norm) to noise ratio than clipping.

In this work, we did not theoretically compare normalization and clipping in general convex settings where over-parameterization may not hold as well as in the nonconvex case. These limitations pave the way for interesting future works.

References

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318, 2016.

[2] Naman Agarwal, Ananda Theertha Suresh, Felix Yu, Sanjiv Kumar, and H Brendan Mcmahan. cpsgd: Communication-efficient and differentially-private distributed sgd. *arXiv preprint arXiv:1805.10559*, 2018.

5. $\|\boldsymbol{u}_{\tilde{k}}^{(i)}\| = \|\sum_{\tau=0}^{E-1}\nabla f_i(\boldsymbol{w}_{\tilde{k},\tau}^{(i)})\| \leq \sum_{\tau=0}^{E-1}\|\nabla f_i(\boldsymbol{w}_{\tilde{k},\tau}^{(i)})\| \leq GE$, where the last inequality follows by using the G-Lipschitzness of f_i.

[3] Hilal Asi, Vitaly Feldman, Tomer Koren, and Kunal Talwar. Private stochastic convex optimization: Optimal rates in l1 geometry. *arXiv preprint arXiv:2103.01516*, 2021.

[4] Raef Bassily, Adam Smith, and Abhradeep Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, pages 464–473. IEEE, 2014.

[5] Raef Bassily, Vitaly Feldman, Kunal Talwar, and Abhradeep Thakurta. Private stochastic convex optimization with optimal rates. *arXiv preprint arXiv:1908.09970*, 2019.

[6] Léon Bottou. Stochastic gradient descent tricks. In *Neural networks: Tricks of the trade*, pages 421–436. Springer, 2012.

[7] Zachary Charles, Zachary Garrett, Zhouyuan Huo, Sergei Shmulyian, and Virginia Smith. On large-cohort training for federated learning. *Advances in Neural Information Processing Systems*, 34, 2021.

[8] Kamalika Chaudhuri and Claire Monteleoni. Privacy-preserving logistic regression. In *NIPS*, volume 8, pages 289–296. Citeseer, 2008.

[9] Kamalika Chaudhuri, Claire Monteleoni, and Anand D Sarwate. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12(3), 2011.

[10] Ashok Cutkosky and Harsh Mehta. Momentum improves normalized sgd. In *International Conference on Machine Learning*, pages 2260–2268. PMLR, 2020.

[11] Rudrajit Das, Satyen Kale, Zheng Xu, Tong Zhang, and Sujay Sanghavi. Beyond uniform lipschitz condition in differentially private optimization. *arXiv preprint arXiv:2206.10713*, 2022.

[12] John C Duchi, Michael I Jordan, and Martin J Wainwright. Local privacy and statistical minimax rates. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 429–438. IEEE, 2013.

[13] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014.

[14] Vitaly Feldman, Tomer Koren, and Kunal Talwar. Private stochastic convex optimization: optimal rates in linear time. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pages 439–449, 2020.

[15] Robin C Geyer, Tassilo Klein, and Moin Nabi. Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557*, 2017.

[16] Antonious Girgis, Deepesh Data, Suhas Diggavi, Peter Kairouz, and Ananda Theertha Suresh. Shuffled model of differential privacy in federated learning. In *International Conference on Artificial Intelligence and Statistics*, pages 2521–2529. PMLR, 2021.

[17] Elad Hazan, Kfir Y Levy, and Shai Shalev-Shwartz. Beyond convexity: Stochastic quasi-convex optimization. *arXiv preprint arXiv:1507.02030*, 2015.

[18] 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.

[19] Roger Iyengar, Joseph P Near, Dawn Song, Om Thakkar, Abhradeep Thakurta, and Lun Wang. Towards practical differentially private convex optimization. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 299–316. IEEE, 2019.

[20] Daniel Kifer, Adam Smith, and Abhradeep Thakurta. Private convex empirical risk minimization and high-dimensional regression. In *Conference on Learning Theory*, pages 25–1. JMLR Workshop and Conference Proceedings, 2012.

[21] Tian Li, Zaoxing Liu, Vyas Sekar, and Virginia Smith. Privacy for free: Communication-efficient learning with differential privacy using sketches. *arXiv preprint arXiv:1911.00972*, 2019.

[22] Siyuan Ma, Raef Bassily, and Mikhail Belkin. The power of interpolation: Understanding the effectiveness of sgd in modern over-parametrized learning. In *International Conference on Machine Learning*, pages 3325–3334. PMLR, 2018.

[23] 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.

[24] Yurii Nesterov et al. *Lectures on convex optimization*, volume 137. Springer, 2018.

[25] Thien Duc Nguyen, Phillip Rieger, Hossein Yalame, Helen Möllering, Hossein Fereidooni, Samuel Marchal, Markus Miettinen, Azalia Mirhoseini, Ahmad-Reza Sadeghi, Thomas Schneider, et al. Flguard: Secure and private federated learning. *arXiv preprint arXiv:2101.02281*, 2021.

[26] Daniel Peterson, Pallika Kanani, and Virendra J Marathe. Private federated learning with domain adaptation. *arXiv preprint arXiv:1912.06733*, 2019.

[27] Shuang Song, Kamalika Chaudhuri, and Anand D Sarwate. Stochastic gradient descent with differentially private updates. In *2013 IEEE Global Conference on Signal and Information Processing*, pages 245–248. IEEE, 2013.

[28] Kunal Talwar, Abhradeep Guha Thakurta, and Li Zhang. Nearly optimal private lasso. *Advances in Neural Information Processing Systems*, 28:3025–3033, 2015.

[29] Om Thakkar, Galen Andrew, and H Brendan McMahan. Differentially private learning with adaptive clipping. *arXiv preprint arXiv:1905.03871*, 2019.

[30] Di Wang, Minwei Ye, and Jinhui Xu. Differentially private empirical risk minimization revisited: Faster and more general. *arXiv preprint arXiv:1802.05251*, 2018.

[31] Xi Wu, Fengan Li, Arun Kumar, Kamalika Chaudhuri, Somesh Jha, and Jeffrey Naughton. Bolt-on differential privacy for scalable stochastic gradient descent-based analytics. In *Proceedings of the 2017 ACM International Conference on Management of Data*, pages 1307–1322, 2017.

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

[33] Yang You, Igor Gitman, and Boris Ginsburg. Large batch training of convolutional networks. *arXiv preprint arXiv:1708.03888*, 2017.

[34] Yang You, Jing Li, Sashank Reddi, Jonathan Hseu, Sanjiv Kumar, Srinadh Bhojanapalli, Xiaodan Song, James Demmel, Kurt Keutzer, and Cho-Jui Hsieh. Large batch optimization for deep learning: Training bert in 76 minutes. *arXiv preprint arXiv:1904.00962*, 2019.

[35] Jiaqi Zhang, Kai Zheng, Wenlong Mou, and Liwei Wang. Efficient private erm for smooth objectives. In *IJCAI*, 2017.

Appendix

Contents

A More Preliminaries **11**

B Related Work and Significance **11**

C Detailed Result for `DP-FedAvg` with Clipping and its Proof **12**
C.1 Proof of Theorem 6 13

D Detailed Result for `DP-NormFedAvg` and its Proof **17**
D.1 Proof of Theorem 7 18

E Lemmas used in the Proofs **19**

F Synthetic Experiments **22**

G Experiments on Datasets **24**

H Experimental Details **25**

Appendix A. More Preliminaries

Differential Privacy (DP): Suppose we have a set of datasets D_c and a query function $h : D_c \to \mathcal{X}$. Two datasets $\mathcal{D}, \mathcal{D}' \in D_c$ are said to be neighboring if they differ in exactly one sample, and this is denoted by $|\mathcal{D} - \mathcal{D}'| = 1$. A randomized mechanism $\mathcal{M} : \mathcal{X} \to \mathcal{Y}$ is said to be (ε, δ)-DP, if for any two neighboring datasets $\mathcal{D}, \mathcal{D}' \in D_c$ and for any measurable subset of outputs $\mathcal{R} \in \mathcal{Y}$,

$$\mathbb{P}(\mathcal{M}(h(\mathcal{D})) \in \mathcal{R}) \leq e^{\varepsilon}\mathbb{P}(\mathcal{M}(h(\mathcal{D}')) \in \mathcal{R}) + \delta. \tag{6}$$

Adding zero-mean Gaussian noise to the output of $h(.)$ above is the customary approach to guarantee DP. This is known as the Gaussian mechanism [13] and it is also employed in private optimization [1]. We formally define the Gaussian mechanism next.

Definition 5 (Gaussian mechanism [13]) *Suppose $\mathcal{X}$ (i.e., the range of the query function h above) is $\mathbb{R}^p$. Let $\Delta_2 := \sup_{\mathcal{D},\mathcal{D}' \in D_c : |\mathcal{D}-\mathcal{D}'|=1} \|h(\mathcal{D}) - h(\mathcal{D}')\|_2$. If we set $\mathcal{M}(h(\mathcal{D})) = h(\mathcal{D}) + \boldsymbol{Z}$, where $\boldsymbol{Z} \sim \mathcal{N}\Big(\mathbf{0}_p, \frac{2\log(1.25/\delta)\Delta_2^2}{\varepsilon^2}\mathrm{I}_p\Big)$, then the mechanism $\mathcal{M}$ is (ε, δ)-DP.*

We also state the famous `FedAvg` algorithm of [23] (with local updates using full gradients).

Algorithm 2 `FedAvg` [23]

Input: Initial point $\boldsymbol{w}_0$, number of rounds of communication K, number of local updates per round E, local learning rates $\{\eta_k\}_{k=0}^{K-1}$ and number of participating clients in each round r.
for $k = 0, \ldots, K-1$ **do**
  Server sends $\boldsymbol{w}_k$ to a random set $\mathcal{S}_k$ of r clients chosen uniformly at random.
  for client $i \in \mathcal{S}_k$ **do**
    Set $\boldsymbol{w}_{k,0}^{(i)} = \boldsymbol{w}_k$.
    for $\tau = 0, \ldots, E-1$ **do**
      Update $\boldsymbol{w}_{k,\tau+1}^{(i)} \leftarrow \boldsymbol{w}_{k,\tau}^{(i)} - \eta_k \nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})$.
    end for
    Send $\boldsymbol{w}_k - \boldsymbol{w}_{k,E}^{(i)}$ to the server.
  end for
  Update $\boldsymbol{w}_{k+1} \leftarrow \boldsymbol{w}_k - \frac{1}{r}\sum_{i \in \mathcal{S}_k}(\boldsymbol{w}_k - \boldsymbol{w}_{k,E}^{(i)})$.
  // (`The above is equivalent to` $\boldsymbol{w}_{k+1} \leftarrow \frac{1}{r}\sum_{i \in \mathcal{S}_k} \boldsymbol{w}_{k,E}^{(i)}$`, so the clients might as well just send the` $\boldsymbol{w}_{k,E}^{(i)}$`'s.)`
end for

Appendix B. Related Work and Significance

Differentially private optimization: Most differentially private optimization algorithms for training ML models (both in the centralized and federated settings) are based off of DP-SGD [1], wherein

the optimizer receives a Gaussian noise-perturbed average of the *clipped* per-sample gradients to guarantee DP. Similar to and/or related to DP-SGD, there are several papers on private optimization in the centralized setting [3–5, 8, 9, 11, 12, 14, 19, 20, 27, 28, 30, 31, 35] as well as in the federated and distributed (without multiple local updates) setting [2, 15, 16, 21, 25, 26, 29]. `DP-FedAvg` with clipping [15, 29] (stated in Algorithm 1 Option (i)) is the most standard private algorithm in the federated setting for client-level DP. [4] show that in the convex Lipschitz centralized case, the suboptimality gap bound of $\mathcal{O}(\rho)$ (recall $\rho = \sqrt{qd\log(1/\delta)}/n\varepsilon$, as defined in Definition 2) bound is tight.

Normalized gradient descent (GD) and related methods: In the centralized setting, [17] propose (Stochastic) Normalized GD. This is based on a similar idea as `DP-NormFedAvg` – instead of using the (stochastic) gradient, use the *unit vector* along the (stochastic) gradient for the update. Extensions of this method incorporating momentum [10, 33, 34] have been shown to significantly improve the training time of very large models such as BERT in the centralized setting. In the FL setting, [7] propose Normalized `FedAvg`, where the server uses a normalized version of the average of client updates (and *not* the average of normalized client updates, which is what we do) to improve training. However, it must be noted here that these works perform (some kind of) normalization to accelerate *non-private* training, whereas we are proposing normalization as an alternative *sensitivity bounding* mechanism to improve *private* training compared to the usual mechanism of clipping.

Appendix C. Detailed Result for `DP-FedAvg` with Clipping and its Proof

Theorem 6 (`DP-FedAvg` with Clipping) *Suppose each f_i is convex and L-smooth over $\mathbb{R}^d$. Let $\hat{C} := \frac{C}{E}$, where C is the clipping threshold used in Algorithm 1 Option (i). For any $\boldsymbol{w}^* \in \arg\min_{\boldsymbol{w}'\in\mathbb{R}^d} f(\boldsymbol{w}')$ and $\Delta_i^* := f_i(\boldsymbol{w}^*) - \min_{\boldsymbol{w}'\in\mathbb{R}^d} f_i(\boldsymbol{w}') \geq 0$, Algorithm 1 Option (i) with $\hat{C} \geq 4\sqrt{L\max_{j\in[n]}\Delta_j^*}$, $\beta_k = \eta_k = \eta = \left(\frac{\gamma}{\hat{C}LEK}\right)\frac{1}{\rho}$ and $K > \left(\frac{2\gamma}{\hat{C}E}\right)\frac{1}{\rho}$, where $\gamma > 0$ is a constant of our choice, has the following convergence guarantee:*

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\left(\mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\| \leq \hat{C}E)\Big(2-\frac{2\gamma}{\hat{C}K\rho}-\frac{4\gamma^2}{\hat{C}^2K^2\rho^2}\Big)(f_i(\boldsymbol{w}_{\tilde{k}})-f_i(\boldsymbol{w}^*))+\mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\| > \hat{C}E)\Big(\frac{3\hat{C}\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|}{8LE}\Big)\right)\right]$$
$$\leq \hat{C}\Big(\frac{L\|\boldsymbol{w}_0-\boldsymbol{w}^*\|^2}{\gamma}+\frac{\gamma}{L}\Big)\rho+\left(\frac{2\gamma}{\hat{C}K\rho}\right)\left(1+\frac{2\gamma}{\hat{C}K\rho}\right)\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\| \leq \hat{C}E)\Delta_i^*\right],$$

with $\tilde{k} \sim \text{unif}[0, K-1]$.
Specifically, with $K = \left(\frac{2\alpha\gamma}{\hat{C}E}\right)\frac{1}{\rho^2}$ and $E \leq \frac{\alpha}{2\rho}$, where $\alpha \geq 1$ is another constant of our choice, Algorithm 1 Option (i) has the following convergence guarantee:

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\left(\mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\| \leq \hat{C}E)\Big(2-\frac{\rho E}{\alpha}-\frac{\rho^2E^2}{\alpha^2}\Big)(f_i(\boldsymbol{w}_{\tilde{k}})-f_i(\boldsymbol{w}^*))+\mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\| > \hat{C}E)\Big(\frac{3\hat{C}\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|}{8LE}\Big)\right)\right]$$
$$\leq \hat{C}\Big(\frac{L\|\boldsymbol{w}_0-\boldsymbol{w}^*\|^2}{\gamma}+\frac{\gamma}{L}\Big)\rho+\Big(\frac{3E}{2\alpha}\Big)\Big(\frac{1}{n}\sum_{i=1}^{n}\Delta_i^*\Big)\rho.$$

Theorem 4 **(a)** is obtained by further assuming that each f_i is G-Lipschitz and choosing $\hat{C} = G$ above, in which case $\mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\| \leq \hat{C}E) = 1$.

C.1. Proof of Theorem 6

Proof Let us set $\eta_k = \beta_k = \eta$ for all $k \geq 0$.

The update rule of the global iterate is:

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k - \eta\Big(\frac{1}{r}\sum_{i\in\mathcal{S}_k} \text{clip}\big(\boldsymbol{u}_k^{(i)}, C\big) + \boldsymbol{\zeta}_k\Big), \tag{7}$$

where $\boldsymbol{\zeta}_k = \frac{1}{r}\sum_{i\in\mathcal{S}_k} \boldsymbol{\zeta}_k^{(i)} \sim \mathcal{N}(\mathbf{0}_d, \frac{qK\log(1/\delta)C^2}{n^2\varepsilon^2}\mathrm{I}_d)$ and

$$\boldsymbol{u}_k^{(i)} = \frac{(\boldsymbol{w}_k - \boldsymbol{w}_{k,E}^{(i)})}{\eta} = \sum_{\tau=0}^{E-1} \nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)}). \tag{8}$$

Taking expectation with respect to the randomness in the current round, we get for any $\boldsymbol{w}^* \in \arg\min_{\boldsymbol{w}'\in\mathbb{R}^d} f(\boldsymbol{w}')$:

$$\mathbb{E}[\|\boldsymbol{w}_{k+1} - \boldsymbol{w}^*\|^2] = \mathbb{E}\Big[\Big\|\boldsymbol{w}_k - \eta\Big(\frac{1}{r}\sum_{i\in\mathcal{S}_k} \text{clip}(\boldsymbol{u}_k^{(i)}, C) + \boldsymbol{\zeta}_k\Big) - \boldsymbol{w}^*\Big\|^2\Big] \tag{9}$$

$$= \|\boldsymbol{w}_k - \boldsymbol{w}^*\|^2 - 2\eta\mathbb{E}_{\mathcal{S}_k}\Big[\frac{1}{r}\sum_{i\in\mathcal{S}_k}\langle \text{clip}(\boldsymbol{u}_k^{(i)}, C), \boldsymbol{w}_k - \boldsymbol{w}^*\rangle\Big] + \eta^2\mathbb{E}\Big[\Big\|\frac{1}{r}\sum_{i\in\mathcal{S}_k} \text{clip}(\boldsymbol{u}_k^{(i)}, C) + \boldsymbol{\zeta}_k\Big\|^2\Big] \tag{10}$$

$$= \|\boldsymbol{w}_k - \boldsymbol{w}^*\|^2 + \frac{1}{n}\sum_{i=1}^{n} -2\eta\langle \text{clip}(\boldsymbol{u}_k^{(i)}, C), \boldsymbol{w}_k - \boldsymbol{w}^*\rangle + \eta^2\mathbb{E}_{\mathcal{S}_k}\Big[\Big\|\frac{1}{r}\sum_{i\in\mathcal{S}_k} \text{clip}(\boldsymbol{u}_k^{(i)}, C)\Big\|^2\Big] \tag{11}$$

$$+ \eta^2\Big(\frac{qKd\log(1/\delta)C^2}{n^2\varepsilon^2}\Big)$$

$$\leq \|\boldsymbol{w}_k - \boldsymbol{w}^*\|^2 + \frac{1}{n}\sum_{i=1}^{n} -2\eta\langle \text{clip}(\boldsymbol{u}_k^{(i)}, C), \boldsymbol{w}_k - \boldsymbol{w}^*\rangle + \eta^2\mathbb{E}_{\mathcal{S}_k}\Big[\frac{1}{r}\sum_{i\in\mathcal{S}_k} \big\|\text{clip}(\boldsymbol{u}_k^{(i)}, C)\big\|^2\Big] \tag{12}$$

$$+ \eta^2\Big(\frac{qKd\log(1/\delta)C^2}{n^2\varepsilon^2}\Big)$$

$$= \|\boldsymbol{w}_k - \boldsymbol{w}^*\|^2 + \frac{1}{n}\sum_{i=1}^{n}\Big\{\underbrace{-2\eta\langle \text{clip}(\boldsymbol{u}_k^{(i)}, C), \boldsymbol{w}_k - \boldsymbol{w}^*\rangle + \eta^2\big\|\text{clip}(\boldsymbol{u}_k^{(i)}, C)\big\|^2}_{A_i}\Big\} \tag{13}$$

$$+ \eta^2\Big(\frac{qKd\log(1/\delta)C^2}{n^2\varepsilon^2}\Big).$$

Note that eq. (12) is obtained by using Fact 2. Let us examine A_i for each i.

Case 1: $\|\boldsymbol{u}_k^{(i)}\| > C$. So we have $\text{clip}(\boldsymbol{u}_k^{(i)}, C) = \frac{C}{\|\boldsymbol{u}_k^{(i)}\|}\boldsymbol{u}_k^{(i)}$. Thus,

$$A_i = -2\eta C\Big\langle \frac{\boldsymbol{u}_k^{(i)}}{\|\boldsymbol{u}_k^{(i)}\|}, \boldsymbol{w}_k - \boldsymbol{w}^*\Big\rangle + \eta^2C^2 \tag{14}$$

$$= \frac{-C}{\|\boldsymbol{u}_k^{(i)}\|}\Big(\|\boldsymbol{w}_k - \boldsymbol{w}^*\|^2 + \eta^2\big\|\boldsymbol{u}_k^{(i)}\big\|^2 - \|\underbrace{\boldsymbol{w}_k - \eta\boldsymbol{u}_k^{(i)}}_{=\boldsymbol{w}_{k,E}^{(i)}} - \boldsymbol{w}^*\|^2\Big) + \eta^2C^2, \tag{15}$$

where the last step follows by using the fact for any two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, $\langle \boldsymbol{a}, \boldsymbol{b}\rangle = \frac{1}{2}\Big(\|\boldsymbol{a}\|^2 + \|\boldsymbol{b}\|^2 - \|\boldsymbol{a}-\boldsymbol{b}\|^2\Big)$. Next, notice that $\boldsymbol{w}_k - \eta \boldsymbol{u}_k^{(i)} = \boldsymbol{w}_{k,E}^{(i)}$. Since f_i is convex, we use Lemma 8 to get:

$$\|\boldsymbol{w}_k - \boldsymbol{w}^*\|^2 - \|\boldsymbol{w}_{k,E}^{(i)} - \boldsymbol{w}^*\|^2 \geq \frac{\eta}{2L}\sum_{\tau=0}^{E-1}\|\nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})\|^2 - 2\eta E \Delta_i^*, \tag{16}$$

for $\eta \leq \frac{1}{2L}$ with $\Delta_i^* := f_i(\boldsymbol{w}^*) - \min_{\boldsymbol{w}' \in \mathbb{R}^d} f_i(\boldsymbol{w}') \geq 0$. But:

$$\big\|\boldsymbol{u}_k^{(i)}\big\|^2 = \Big\|\sum_{\tau=0}^{E-1}\nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})\Big\|^2 \leq E\sum_{\tau=0}^{E-1}\big\|\nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})\big\|^2. \tag{17}$$

The inequality above follows from Fact 2. Using this in eq. (16), we get:

$$\|\boldsymbol{w}_k - \boldsymbol{w}^*\|^2 - \|\boldsymbol{w}_{k,E}^{(i)} - \boldsymbol{w}^*\|^2 \geq \frac{\eta}{2LE}\big\|\boldsymbol{u}_k^{(i)}\big\|^2 - 2\eta E\Delta_i^*. \tag{18}$$

Plugging this back in eq. (15), we get:

$$A_i \leq -C\Big(\eta^2 + \frac{\eta}{2LE}\Big)\|\boldsymbol{u}_k^{(i)}\| + 2\eta \underbrace{\left(\frac{C}{\|\boldsymbol{u}_k^{(i)}\|}\right)}_{<1} E\Delta_i^* + \eta^2 C^2, \tag{19}$$

for $\eta \leq \frac{1}{2L}$.

Let us choose $C^2 \geq 16LE^2 \max_{j\in[n]} \Delta_j^*$. Then, we have $E\Delta_i^* \leq \frac{C^2}{16LE} \leq \frac{C\|\boldsymbol{u}_k^{(i)}\|}{16LE}$. Using this in eq. (19), we get:

$$\begin{aligned} A_i &\leq -C\Big(\eta^2 + \frac{\eta}{2LE}\Big)\|\boldsymbol{u}_k^{(i)}\| + \frac{\eta C}{8LE}\|\boldsymbol{u}_k^{(i)}\| + \eta^2 C^2 && (20)\\ &= -\frac{3\eta C}{8LE}\|\boldsymbol{u}_k^{(i)}\| + \eta^2 C \underbrace{(C - \|\boldsymbol{u}_k^{(i)}\|)}_{<0} && (21)\\ &\leq -\frac{3\eta C}{8LE}\|\boldsymbol{u}_k^{(i)}\|, && (22) \end{aligned}$$

for $C \geq 4E\sqrt{L\max_{j\in[n]}\Delta_j^*}$ and $\eta \leq \frac{1}{2L}$.

Case 2: $\|\boldsymbol{u}_k^{(i)}\| \leq C$. So we have $\text{clip}(\boldsymbol{u}_k^{(i)}, C) = \boldsymbol{u}_k^{(i)}$. Thus,

$$A_i = -2\eta\langle \boldsymbol{u}_k^{(i)}, \boldsymbol{w}_k - \boldsymbol{w}^*\rangle + \eta^2\|\boldsymbol{u}_k^{(i)}\|^2 \leq -2\eta \underbrace{\langle \boldsymbol{u}_k^{(i)}, \boldsymbol{w}_k - \boldsymbol{w}^*\rangle}_{B_i} + 2\eta^2 LE^2(f_i(\boldsymbol{w}_k) - f_i^*), \tag{23}$$

for $\eta L \leq 1$; the inequality $\|\boldsymbol{u}_k^{(i)}\|^2 \leq 2LE^2(f_i(\boldsymbol{w}_k) - f_i^*)$ (for $\eta L \leq 1$) is obtained from Lemma 9. Now:

$$
\begin{aligned}
B_i &= \langle \boldsymbol{u}_k^{(i)}, \boldsymbol{w}_k - \boldsymbol{w}^* \rangle && (24)\\
&= \sum_{\tau=0}^{E-1} \langle \nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)}), \boldsymbol{w}_k - \boldsymbol{w}^* \rangle && (25)\\
&= \sum_{\tau=0}^{E-1} \{ \langle \nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)}), \boldsymbol{w}_{k,\tau}^{(i)} - \boldsymbol{w}^* \rangle + \langle \nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)}), \boldsymbol{w}_k - \boldsymbol{w}_{k,\tau}^{(i)} \rangle \} && (26)\\
&\geq \sum_{\tau=0}^{E-1} \{ f_i(\boldsymbol{w}_{k,\tau}^{(i)}) - f_i(\boldsymbol{w}^*) + \langle \nabla f_i(\boldsymbol{w}_k), \boldsymbol{w}_k - \boldsymbol{w}_{k,\tau}^{(i)} \rangle + \langle \nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)}) - \nabla f_i(\boldsymbol{w}_k), \boldsymbol{w}_k - \boldsymbol{w}_{k,\tau}^{(i)} \rangle \} && (27)\\
&\geq \sum_{\tau=0}^{E-1} \{ f_i(\boldsymbol{w}_{k,\tau}^{(i)}) - f_i(\boldsymbol{w}^*) + f_i(\boldsymbol{w}_k) - f_i(\boldsymbol{w}_{k,\tau}^{(i)}) - L\|\boldsymbol{w}_k - \boldsymbol{w}_{k,\tau}^{(i)}\|^2 \} && (28)\\
&= E(f_i(\boldsymbol{w}_k) - f_i(\boldsymbol{w}^*)) - L \sum_{\tau=0}^{E-1} \|\boldsymbol{w}_k - \boldsymbol{w}_{k,\tau}^{(i)}\|^2. && (29)
\end{aligned}
$$

Note that eq. (27) follows from the convexity of f_i, while eq. (28) follows by once again using the convexity of f_i, the smoothness of f_i as well as the Cauchy-Schwarz inequality.
Again, from Lemma 9, we have

$$
\|\boldsymbol{w}_k - \boldsymbol{w}_{k,\tau}^{(i)}\|^2 \leq 2\eta^2 L \tau^2 (f_i(\boldsymbol{w}_k) - f_i^*), \tag{30}
$$

for $\eta L \leq 1$. Using eq. (30) in eq. (29), we get

$$
B_i \geq E(f_i(\boldsymbol{w}_k) - f_i(\boldsymbol{w}^*)) - 2\eta^2 L^2 \sum_{\tau=0}^{E-1} \tau^2 (f_i(\boldsymbol{w}_k) - f_i^*) \geq E(f_i(\boldsymbol{w}_k) - f_i(\boldsymbol{w}^*)) - 2\eta^2 L^2 E^3 (f_i(\boldsymbol{w}_k) - f_i^*). \tag{31}
$$

Now using eq. (31) in eq. (23), we get

$$
\begin{aligned}
A_i &\leq -2\eta E(f_i(\boldsymbol{w}_k) - f_i(\boldsymbol{w}^*)) + 4\eta^3 L^2 E^3 (f_i(\boldsymbol{w}_k) - f_i^*) + 2\eta^2 L E^2 (f_i(\boldsymbol{w}_k) - f_i^*)\\
&= -\eta E\Big(2 - 2\eta L E - 4\eta^2 L^2 E^2\Big)(f_i(\boldsymbol{w}_k) - f_i(\boldsymbol{w}^*)) + (2\eta^2 L E^2 + 4\eta^3 L^2 E^3)\Delta_i^*, && (32)
\end{aligned}
$$

for $\eta \leq \frac{1}{L}$.

Combining the results of Case 1 and 2, i.e. eq. (22) and eq. (32), we get

$$
\begin{aligned}
A_i \leq\ & \mathbb{1}(\|\boldsymbol{u}_k^{(i)}\| > C)\Big(-\frac{3\eta C}{8LE}\|\boldsymbol{u}_k^{(i)}\|\Big)\\
&+ \mathbb{1}(\|\boldsymbol{u}_k^{(i)}\| \leq C)\Big(-\eta E(2 - 2\eta L E - 4\eta^2 L^2 E^2)(f_i(\boldsymbol{w}_k) - f_i(\boldsymbol{w}^*)) + (2\eta^2 L E^2 + 4\eta^3 L^2 E^3)\Delta_i^*\Big), && (33)
\end{aligned}
$$

for $C \geq 4E\sqrt{L\max_{j\in[n]}\Delta_j^*}$ and $\eta \leq \frac{1}{2L}$. Let us define $\hat{C} := \frac{C}{E}$. Then eq. (33) can be re-written as:

$$A_i \leq -\eta E\Bigg\{\mathbb{1}(\|\boldsymbol{u}_k^{(i)}\| \leq \hat{C}E)\Big((2-2\eta LE-4\eta^2L^2E^2)(f_i(\boldsymbol{w}_k)-f_i(\boldsymbol{w}^*))\Big) + \mathbb{1}(\|\boldsymbol{u}_k^{(i)}\| > \hat{C}E)\Big(\frac{3\hat{C}}{8LE}\|\boldsymbol{u}_k^{(i)}\|\Big) - \mathbb{1}(\|\boldsymbol{u}_k^{(i)}\| \leq \hat{C}E)(2\eta LE+4\eta^2L^2E^2)\Delta_i^*\Bigg\}, \quad (34)$$

where $\hat{C} \geq 4\sqrt{L\max_{j\in[n]}\Delta_j^*}$ and $\eta \leq \frac{1}{2L}$. Now using eq. (34) in eq. (13), we get:

$$\mathbb{E}[\|\boldsymbol{w}_{k+1}-\boldsymbol{w}^*\|^2] \leq \|\boldsymbol{w}_k-\boldsymbol{w}^*\|^2 - \frac{\eta E}{n}\sum_{i=1}^{n}\Big\{\mathbb{1}(\|\boldsymbol{u}_k^{(i)}\| > \hat{C}E)\Big(\frac{3\hat{C}}{8LE}\|\boldsymbol{u}_k^{(i)}\|\Big) + \mathbb{1}(\|\boldsymbol{u}_k^{(i)}\| \leq \hat{C}E)\Big((2-2\eta LE-4\eta^2L^2E^2)(f_i(\boldsymbol{w}_k)-f_i(\boldsymbol{w}^*))\Big)\Big\} + \eta E(2\eta LE+4\eta^2L^2E^2)\Big(\frac{1}{n}\sum_{i=1}^{n}\mathbb{1}(\|\boldsymbol{u}_k^{(i)}\| \leq \hat{C}E)\Delta_i^*\Big) + \eta^2E^2\hat{C}^2\Big(\frac{qKd\log(1/\delta)}{n^2\varepsilon^2}\Big). \quad (35)$$

Solving the above recursion after taking expectation throughout and some rearranging, we get:

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\Big[\frac{1}{n}\sum_{i=1}^{n}\Big\{\mathbb{1}(\|\boldsymbol{u}_k^{(i)}\| \leq \hat{C}E)\Big((2-2\eta LE-4\eta^2L^2E^2)(f_i(\boldsymbol{w}_k)-f_i(\boldsymbol{w}^*))\Big)+\mathbb{1}(\|\boldsymbol{u}_k^{(i)}\| > \hat{C}E)\Big(\frac{3\hat{C}\|\boldsymbol{u}_k^{(i)}\|}{8LE}\Big)\Big\}\Big] \leq \frac{\|\boldsymbol{w}_0-\boldsymbol{w}^*\|^2}{\eta EK}+\eta EK\hat{C}^2\Big(\frac{qd\log(1/\delta)}{n^2\varepsilon^2}\Big)+\frac{2\eta LE(1+2\eta LE)}{K}\mathbb{E}\Bigg[\sum_{k=0}^{K-1}\Big(\frac{1}{n}\sum_{i=1}^{n}\mathbb{1}(\|\boldsymbol{u}_k^{(i)}\| \leq \hat{C}E)\Delta_i^*\Big)\Bigg]. \quad (36)$$

Let us choose $\eta = \frac{\gamma}{\hat{C}LEK}\frac{n\varepsilon}{\sqrt{qd\log(1/\delta)}}$ for some constant $\gamma > 0$. Note that we must have $K > \frac{2\gamma}{\hat{C}E}\frac{n\varepsilon}{\sqrt{qd\log(1/\delta)}}$ for our condition of $\eta L \leq \frac{1}{2}$ to be satisfied. With that, we get:

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\Bigg[\frac{1}{n}\sum_{i=1}^{n}\Bigg\{\mathbb{1}(\|\boldsymbol{u}_k^{(i)}\| \leq \hat{C}E)\Big(2-\frac{2\gamma}{\hat{C}K}\frac{n\varepsilon}{\sqrt{qd\log(1/\delta)}}-\frac{4\gamma^2}{\hat{C}^2K^2}\frac{n^2\varepsilon^2}{qd\log(1/\delta)}\Big)(f_i(\boldsymbol{w}_k)-f_i(\boldsymbol{w}^*)) + \mathbb{1}(\|\boldsymbol{u}_k^{(i)}\| > \hat{C}E)\Big(\frac{3\hat{C}}{8LE}\|\boldsymbol{u}_k^{(i)}\|\Big)\Bigg\}\Bigg] \leq \Big(\frac{L\|\boldsymbol{w}_0-\boldsymbol{w}^*\|^2}{\gamma}+\frac{\gamma}{L}\Big)\frac{\hat{C}\sqrt{qd\log(1/\delta)}}{n\varepsilon} + \left(\frac{2\gamma}{\hat{C}K}\frac{n\varepsilon}{\sqrt{qd\log(1/\delta)}}\right)\left(1+\frac{2\gamma}{\hat{C}K}\frac{n\varepsilon}{\sqrt{qd\log(1/\delta)}}\right)\mathbb{E}\Bigg[\Bigg(\frac{1}{K}\sum_{k=0}^{K-1}\Big(\frac{1}{n}\sum_{i=1}^{n}\mathbb{1}(\|\boldsymbol{u}_k^{(i)}\| \leq \hat{C}E)\Delta_i^*\Big)\Bigg)\Bigg], \quad (37)$$

with $\hat{C} \geq 4\sqrt{L\max_{j\in[n]}\Delta_j^*}$.

The above equation is equivalent to:

$$\mathbb{E}\Bigg[\frac{1}{n}\sum_{i=1}^{n}\Bigg\{\mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|\leq\hat{C}E)\Big(2-\frac{2\gamma}{\hat{C}K}\frac{n\varepsilon}{\sqrt{qd\log(1/\delta)}}-\frac{4\gamma^2}{\hat{C}^2K^2}\frac{n^2\varepsilon^2}{qd\log(1/\delta)}\Big)(f_i(\boldsymbol{w}_{\tilde{k}})-f_i(\boldsymbol{w}^*))$$
$$+\mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|>\hat{C}E)\Big(\frac{3\hat{C}}{8LE}\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|\Big)\Bigg\}\Bigg]\leq\Big(\frac{L\|\boldsymbol{w}_0-\boldsymbol{w}^*\|^2}{\gamma}+\frac{\gamma}{L}\Big)\frac{\hat{C}\sqrt{qd\log(1/\delta)}}{n\varepsilon}$$
$$+\left(\frac{2\gamma}{\hat{C}K}\frac{n\varepsilon}{\sqrt{qd\log(1/\delta)}}\right)\left(1+\frac{2\gamma}{\hat{C}K}\frac{n\varepsilon}{\sqrt{qd\log(1/\delta)}}\right)\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|\leq\hat{C}E)\Delta_i^*\right],\quad(38)$$

where $\tilde{k}\sim\text{unif}[0,K-1]$. Let us set $K=\frac{2\alpha\gamma}{\hat{C}E}\Big(\frac{n\varepsilon}{\sqrt{qd\log(1/\delta)}}\Big)^2$ in eq. (38), where $\alpha\geq1$ is a constant of our choice and $E\leq\frac{\alpha}{2}\Big(\frac{n\varepsilon}{\sqrt{qd\log(1/\delta)}}\Big)$. That gives us:

$$\mathbb{E}\Bigg[\frac{1}{n}\sum_{i=1}^{n}\Bigg\{\mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|\leq\hat{C}E)\left(2-\left(\frac{E\sqrt{qd\log(1/\delta)}}{\alpha n\varepsilon}\right)-\left(\frac{E\sqrt{qd\log(1/\delta)}}{\alpha n\varepsilon}\right)^2\right)(f_i(\boldsymbol{w}_{\tilde{k}})-f_i(\boldsymbol{w}^*))$$
$$+\mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|>\hat{C}E)\Big(\frac{3\hat{C}}{8LE}\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|\Big)\Bigg\}\Bigg]\leq\hat{C}\Big(\frac{L\|\boldsymbol{w}_0-\boldsymbol{w}^*\|^2}{\gamma}+\frac{\gamma}{L}\Big)\frac{\sqrt{qd\log(1/\delta)}}{n\varepsilon}$$
$$+\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\underbrace{\mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|\leq\hat{C}E)}_{\leq1}\Delta_i^*\right]\frac{E\sqrt{qd\log(1/\delta)}}{\alpha n\varepsilon}\underbrace{\left(1+\frac{E\sqrt{qd\log(1/\delta)}}{\alpha n\varepsilon}\right)}_{\leq\frac{3}{2}\text{ from our constraint on }E}.\quad(39)$$

The final result follows by substituting $\rho=\frac{\sqrt{qd\log(1/\delta)}}{n\varepsilon}$. ■

Appendix D. Detailed Result for `DP-NormFedAvg` and its Proof

Theorem 7 (`DP-NormFedAvg`) *Suppose each f_i is convex and L-smooth over $\mathbb{R}^d$. Let $\hat{C}:=C/E$, where C is the scaling factor used in Algorithm 1 Option (ii). For any $\boldsymbol{w}^*\in\arg\min_{\boldsymbol{w}'\in\mathbb{R}^d}f(\boldsymbol{w}')$ and $\Delta_i^*:=f_i(\boldsymbol{w}^*)-\min_{\boldsymbol{w}'\in\mathbb{R}^d}f_i(\boldsymbol{w}')\geq0$, Algorithm 1 Option (ii) with $\hat{C}\geq4\sqrt{L\max_{j\in[n]}\Delta_j^*}$, $\beta_k=\eta_k=\eta=\big(\frac{\gamma}{\hat{C}LEK}\big)\frac{1}{\rho}$ and $K>\big(\frac{2\gamma}{\hat{C}E}\big)\frac{1}{\rho}$, where $\gamma>0$ is a constant of our choice, has the following convergence guarantee:*

$$\mathbb{E}\Bigg[\frac{1}{n}\sum_{i=1}^{n}\Bigg\{\mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|\leq\hat{C}E)\Big(2-\frac{4\gamma^2}{\hat{C}^2K^2\rho^2}\Big)\left(\frac{\hat{C}E}{\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|}\right)(f_i(\boldsymbol{w}_{\tilde{k}})-f_i(\boldsymbol{w}^*))+\mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|>\hat{C}E)\Big(\frac{3\hat{C}\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|}{8LE}\Big)\Bigg\}\Bigg]$$
$$\leq\hat{C}\Big(\frac{L\|\boldsymbol{w}_0-\boldsymbol{w}^*\|^2}{\gamma}+\frac{\gamma}{L}\Big)\rho+\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|\leq\hat{C}E)\left\{\frac{\gamma\hat{C}}{LK\rho}+\left(\frac{\hat{C}E}{\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|}\right)\frac{4\gamma^2\Delta_i^*}{\hat{C}^2K^2\rho^2}\right\}\right],$$

with $\tilde{k}\sim\text{unif}[0,K-1]$. Further, this result holds for any $\boldsymbol{w}^*\in\arg\min_{\boldsymbol{w}'\in\mathbb{R}^d}f(\boldsymbol{w}')$. Specifically, with $K=\big(\frac{2\alpha\gamma}{\hat{C}E}\big)\frac{1}{\rho^2}$ and $E\leq\frac{\alpha}{2\rho}$, where $\alpha\geq1$ is another constant of our choice,

Algorithm 1 Option (ii) has the following convergence guarantee:

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\left\{\mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|\leq\hat{C}E)\Big(2-\frac{\rho^2E^2}{\alpha^2}\Big)\left(\frac{\hat{C}E}{\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|}\right)(f_i(\boldsymbol{w}_{\tilde{k}})-f_i(\boldsymbol{w}^*))+\mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|>\hat{C}E)\Big(\frac{3\hat{C}\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|}{8LE}\Big)\right\}\right]$$
$$\leq\hat{C}\Big(\frac{L\|\boldsymbol{w}_0-\boldsymbol{w}^*\|^2}{\gamma}+\frac{\gamma}{L}\Big)\rho+\Big(\frac{E}{\alpha}\Big)\left(\frac{\hat{C}^2}{2L}+\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|\leq\hat{C}E)\left(\frac{\hat{C}E}{\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|}\right)\Delta_i^*\right]\frac{\rho E}{\alpha}\right)\rho.$$

Theorem 4 **(b)** is obtained by further assuming that each f_i is G-Lipschitz and choosing $\hat{C}=G$ above, in which case $\mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|\leq\hat{C}E)=1$.

D.1. Proof of Theorem 7

Proof Let us again set $\eta_k=\beta_k=\eta$, for all $k\geq 0$.

Everything remains the same till eq. (13) in the proof of Theorem 6, with clip(.) replaced by norm(.).

$$\mathbb{E}[\|\boldsymbol{w}_{k+1}-\boldsymbol{w}^*\|^2]\leq\|\boldsymbol{w}_k-\boldsymbol{w}^*\|^2+\frac{1}{n}\sum_{i=1}^{n}\Big\{\underbrace{-2\eta\langle\text{norm}(\boldsymbol{u}_k^{(i)},C),\boldsymbol{w}_k-\boldsymbol{w}^*\rangle+\eta^2\big\|\text{norm}(\boldsymbol{u}_k^{(i)},C)\big\|^2}_{A_i}\Big\}$$
$$+\eta^2\Big(\frac{qKd\log(1/\delta)C^2}{n^2\varepsilon^2}\Big). \quad (40)$$

Again, let us examine A_i for each i. Also, as used in the proof of Theorem 6, let $\hat{C}=\frac{C}{E}$.

Case 1: $\|\boldsymbol{u}_k^{(i)}\|>\hat{C}E$. Everything remains the same as Case 1 in the proof of Theorem 6. Thus,

$$A_i\leq-\frac{3\eta\hat{C}}{8L}\|\boldsymbol{u}_k^{(i)}\|, \quad (41)$$

for $\eta L\leq\frac{1}{2}$ and $\hat{C}\geq 4\sqrt{L\max_{j\in[n]}\Delta_j^*}$.

Case 2: $\|\boldsymbol{u}_k^{(i)}\|\leq\hat{C}E$. Here:

$$A_i\leq\left(\frac{\hat{C}E}{\|\boldsymbol{u}_k^{(i)}\|}\right)\Big(-2\eta\underbrace{\langle\boldsymbol{u}_k^{(i)},\boldsymbol{w}_k-\boldsymbol{w}^*\rangle}_{B_i}\Big)+\eta^2\hat{C}^2E^2. \quad (42)$$

For ease of notation henceforth, let us define:

$$z_k^{(i)}:=\left(\frac{\hat{C}E}{\|\boldsymbol{u}_k^{(i)}\|}\right). \quad (43)$$

The bound for B_i remains the same as the one in the proof of Theorem 6 (in eq. (31)), i.e.,

$$B_i\geq E(f_i(\boldsymbol{w}_k)-f_i(\boldsymbol{w}^*))-2\eta^2L^2E^3(f_i(\boldsymbol{w}_k)-f_i^*), \quad (44)$$

for $\eta L \leq 1$. Using this in eq. (42), we get:

$$\begin{aligned} A_i &\leq -2\eta E z_k^{(i)} \Big\{ (f_i(\boldsymbol{w}_k) - f_i(\boldsymbol{w}^*)) - 2\eta^2 L^2 E^2 (f_i(\boldsymbol{w}_k) - f_i^*) \Big\} + \eta^2 \hat{C}^2 E^2 \\ &= -2\eta E z_k^{(i)} \Big\{ (f_i(\boldsymbol{w}_k) - f_i(\boldsymbol{w}^*))(1 - 2\eta^2 L^2 E^2) - 2\eta^2 L^2 E^2 \Delta_i^* \Big\} + \eta^2 \hat{C}^2 E^2. \end{aligned} \tag{45}$$

Combining the results of Case 1 and 2, i.e. eq. (41) and eq. (45), we get:

$$\begin{aligned} A_i \leq \eta E \Bigg\{ & \mathbb{1}(\|\boldsymbol{u}_k^{(i)}\| \leq \hat{C}E) \Big(4\eta^2 L^2 E^2 \Delta_i^* z_k^{(i)} + \eta \hat{C}^2 E \Big) \\ & - \mathbb{1}(\|\boldsymbol{u}_k^{(i)}\| \leq \hat{C}E)(2 - 4\eta^2 L^2 E^2) z_k^{(i)} (f_i(\boldsymbol{w}_k) - f_i(\boldsymbol{w}^*)) - \mathbb{1}(\|\boldsymbol{u}_k^{(i)}\| > \hat{C}E) \Big(\frac{3\hat{C}\|\boldsymbol{u}_k^{(i)}\|}{8LE} \Big) \Bigg\}, \end{aligned} \tag{46}$$

for $\eta L \leq \frac{1}{2}$ and $\hat{C} \geq 4\sqrt{L \max_{j \in [n]} \Delta_j^*}$.

Now using the above bound in eq. (40), plugging in $z_k^{(i)} = \frac{\hat{C}E}{\|\boldsymbol{u}_k^{(i)}\|}$, and following the same process and choice of $\eta = \frac{\gamma}{\hat{C}LEK} \frac{n\varepsilon}{\sqrt{qd\log(1/\delta)}}$ that we used in Theorem 6, we get:

$$\begin{aligned} \mathbb{E}\Bigg[\frac{1}{n} \sum_{i=1}^{n} \Bigg\{ & \mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\| \leq \hat{C}E) \Big(2 - \frac{4\gamma^2}{\hat{C}^2 K^2} \frac{n^2 \varepsilon^2}{qd\log(1/\delta)} \Big) \left(\frac{\hat{C}E}{\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|} \right) (f_i(\boldsymbol{w}_{\tilde{k}}) - f_i(\boldsymbol{w}^*)) \\ & + \mathbb{1}(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\| > \hat{C}E) \Big(\frac{3\hat{C}\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|}{8LE} \Big) \Bigg\} \Bigg] \leq \Big(\frac{L\|\boldsymbol{w}_0 - \boldsymbol{w}^*\|^2}{\gamma} + \frac{\gamma}{L} \Big) \frac{\hat{C}\sqrt{qd\log(1/\delta)}}{n\varepsilon} \\ & + \mathbb{E}\Bigg[\frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\big(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\| \leq \hat{C}E\big) \Bigg\{ \frac{\gamma \hat{C}}{LK} \frac{n\varepsilon}{\sqrt{qd\log(1/\delta)}} + \frac{4\gamma^2 \Delta_i^*}{\hat{C}^2 K^2} \frac{n^2 \varepsilon^2}{qd\log(1/\delta)} \left(\frac{\hat{C}E}{\|\boldsymbol{u}_{\tilde{k}}^{(i)}\|} \right) \Bigg\} \Bigg], \end{aligned} \tag{47}$$

with $\tilde{k} \sim$ unif $[0, K-1]$ and $K > \frac{2\gamma}{\hat{C}E} \frac{n\varepsilon}{\sqrt{qd\log(1/\delta)}}$ (so that $\eta LE \leq \frac{1}{2}$). Now setting $K = \frac{2\alpha\gamma}{\hat{C}E} \Big(\frac{n\varepsilon}{\sqrt{qd\log(1/\delta)}} \Big)^2$ and $\rho = \frac{\sqrt{qd\log(1/\delta)}}{n\varepsilon}$ above, and using the fact that $\mathbb{1}\big(\|\boldsymbol{u}_{\tilde{k}}^{(i)}\| \leq \hat{C}E\big) \leq 1$ gives us the final result. ■

Appendix E. Lemmas used in the Proofs

Lemma 8 *Suppose f_i is convex and L-smooth over $\mathbb{R}^d$. Let us set $\eta_k \leq \frac{1}{2L}$ in round k of Algorithm 1. Then:*

$$\|\boldsymbol{w}_{k,E}^{(i)} - \boldsymbol{w}^*\|^2 \leq \|\boldsymbol{w}_k - \boldsymbol{w}^*\|^2 - \frac{\eta_k}{2L} \sum_{\tau=0}^{E-1} \|\nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})\|^2 + 2\eta_k E \Delta_i^*,$$

where $\Delta_i^* := f_i(\boldsymbol{w}^*) - \min_{\boldsymbol{w}' \in \mathbb{R}^d} f_i(\boldsymbol{w}')$.

Proof Let us define $f_i^* := \min_{\boldsymbol{w}' \in \mathbb{R}^d} f_i(\boldsymbol{w}')$. Then, $\Delta_i^* = f_i(\boldsymbol{w}^*) - f_i^*$.

For any $\tau \geq 0$, we have:

$$\begin{aligned}
\|\boldsymbol{w}_{k,\tau+1}^{(i)} - \boldsymbol{w}^*\|^2 &= \|\boldsymbol{w}_{k,\tau}^{(i)} - \boldsymbol{w}^*\|^2 - 2\eta_k \langle \nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)}), \boldsymbol{w}_{k,\tau}^{(i)} - \boldsymbol{w}^* \rangle + \eta_k^2 \|\nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})\|^2 \\
&\leq \|\boldsymbol{w}_{k,\tau}^{(i)} - \boldsymbol{w}^*\|^2 - 2\eta_k (f_i(\boldsymbol{w}_{k,\tau}^{(i)}) - f_i(\boldsymbol{w}^*)) + \eta_k^2 \|\nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})\|^2 && (48) \\
&\leq \|\boldsymbol{w}_{k,\tau}^{(i)} - \boldsymbol{w}^*\|^2 - 2\eta_k (f_i(\boldsymbol{w}_{k,\tau}^{(i)}) - f_i^*) + 2\eta_k \underbrace{(f_i(\boldsymbol{w}^*) - f_i^*)}_{=\Delta_i^*} + \eta_k^2 \|\nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})\|^2 && (49) \\
&\leq \|\boldsymbol{w}_{k,\tau}^{(i)} - \boldsymbol{w}^*\|^2 - \frac{\eta_k}{L} \|\nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})\|^2 + 2\eta_k \Delta_i^* + \eta_k^2 \|\nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})\|^2. && (50)
\end{aligned}$$

Equation (48) follows by using the fact that each f_i is convex. Equation (50) follows using Fact 1.

Now if we set $\eta_k \leq \frac{1}{2L}$, then we get:

$$\|\boldsymbol{w}_{k,\tau+1}^{(i)} - \boldsymbol{w}^*\|^2 \leq \|\boldsymbol{w}_{k,\tau}^{(i)} - \boldsymbol{w}^*\|^2 - \frac{\eta_k}{2L} \|\nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})\|^2 + 2\eta_k \Delta_i^*. \tag{51}$$

Doing this recursively for $\tau = 0$ through to $\tau = E - 1$ and adding everything up gives us the desired result. ■

Lemma 9 *Suppose each f_i is L-smooth over $\mathbb{R}^d$ and $f_i^* := \min_{\boldsymbol{w}' \in \mathbb{R}^d} f_i(\boldsymbol{w}')$. Let us set $\eta_k \leq \frac{1}{L}$ in round k of Algorithm 1. Then:*

$$\|\boldsymbol{w}_k - \boldsymbol{w}_{k,\tau}^{(i)}\|^2 \leq 2\eta_k^2 L \tau^2 (f_i(\boldsymbol{w}_k) - f_i^*) \; \forall \, \tau \geq 1.$$

Thus,

$$\left\|\boldsymbol{u}_k^{(i)}\right\|^2 \leq 2LE^2 (f_i(\boldsymbol{w}_k) - f_i^*).$$

Proof

$$\|\boldsymbol{w}_k - \boldsymbol{w}_{k,\tau}^{(i)}\|^2 = \left\|\eta_k \sum_{t=0}^{\tau-1} \nabla f_i(\boldsymbol{w}_{k,t}^{(i)})\right\|^2 \leq \eta_k^2 \tau \sum_{t=0}^{\tau-1} \|\nabla f_i(\boldsymbol{w}_{k,t}^{(i)})\|^2, \tag{52}$$

where the last step follows from Fact 2. Next, since f_i is L-smooth, we have using Fact 1:

$$\|\nabla f_i(\boldsymbol{w}_{k,t}^{(i)})\|^2 \leq 2L(f_i(\boldsymbol{w}_{k,t}^{(i)}) - f_i^*).$$

Applying this in eq. (52), we get:

$$\|\boldsymbol{w}_k - \boldsymbol{w}_{k,\tau}^{(i)}\|^2 \leq 2\eta_k^2 L \tau \sum_{t=0}^{\tau-1} (f_i(\boldsymbol{w}_{k,t}^{(i)}) - f_i^*). \tag{53}$$

But using the L-smoothness of f_i, we have for any $t \geq 1$:

$$\begin{aligned}
f_i(\boldsymbol{w}_{k,t}^{(i)}) - f_i^* &= f_i(\boldsymbol{w}_{k,t-1}^{(i)} - \eta_k \nabla f_i(\boldsymbol{w}_{k,t-1}^{(i)})) - f_i^* && (54) \\
&\leq (f_i(\boldsymbol{w}_{k,t-1}^{(i)}) - f_i^*) - \eta_k \|\nabla f_i(\boldsymbol{w}_{k,t-1}^{(i)})\|^2 + \frac{\eta_k^2 L}{2} \|\nabla f_i(\boldsymbol{w}_{k,t-1}^{(i)})\|^2 && (55) \\
&\leq (f_i(\boldsymbol{w}_{k,t-1}^{(i)}) - f_i^*) - \frac{\eta_k}{2} \|\nabla f_i(\boldsymbol{w}_{k,t-1}^{(i)})\|^2, && (56)
\end{aligned}$$

for $\eta_k L \leq 1$. Doing this recursively (and recalling that $\boldsymbol{w}_{k,0}^{(i)} = \boldsymbol{w}_k$), we get:

$$f_i(\boldsymbol{w}_{k,t}^{(i)}) - f_i^* \leq (f_i(\boldsymbol{w}_k) - f_i^*) - \frac{\eta_k}{2} \sum_{t'=0}^{t-1} \|\nabla f_i(\boldsymbol{w}_{k,t'}^{(i)})\|^2 \leq f_i(\boldsymbol{w}_k) - f_i^*. \tag{57}$$

Plugging this in eq. (53), we get:

$$\|\boldsymbol{w}_k - \boldsymbol{w}_{k,\tau}^{(i)}\|^2 \leq 2\eta_k^2 L \tau^2 (f_i(\boldsymbol{w}_k) - f_i^*). \tag{58}$$

The upper bound on $\left\|\boldsymbol{u}_k^{(i)}\right\|^2$ follows by recalling that $\boldsymbol{u}_k^{(i)} = (\boldsymbol{w}_k - \boldsymbol{w}_{k,E}^{(i)})/\eta_k$. ■

Lemma 10 *Suppose each f_i is L-smooth over $\mathbb{R}^d$. Then in Algorithm 1, we have:*

$$\|\nabla f_i(\boldsymbol{w}_{k,\tau+1}^{(i)})\|^2 \leq \|\nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})\|^2 - \left(\frac{2}{\eta_k L} - 1\right) \|\nabla f_i(\boldsymbol{w}_{k,\tau+1}^{(i)}) - \nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})\|^2,$$

for any $i \in [n]$, $k \in \{0, \ldots, K-1\}$ and $\tau \in \{0, \ldots, E-1\}$.

Proof Since each f_i is L-smooth, we have by using the co-coercivity of the gradient:

$$\langle \nabla f_i(\boldsymbol{w}_{k,\tau+1}^{(i)}) - \nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)}), \boldsymbol{w}_{k,\tau+1}^{(i)} - \boldsymbol{w}_{k,\tau}^{(i)} \rangle \geq \frac{1}{L} \|\nabla f_i(\boldsymbol{w}_{k,\tau+1}^{(i)}) - \nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})\|^2. \tag{59}$$

Now using the fact that $\boldsymbol{w}_{k,\tau+1}^{(i)} - \boldsymbol{w}_{k,\tau}^{(i)} = -\eta_k \nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})$ above, we get:

$$\begin{aligned} L\langle \nabla f_i(\boldsymbol{w}_{k,\tau+1}^{(i)}) - \nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)}), -\eta_k \nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)}) \rangle \geq \|\nabla f_i(\boldsymbol{w}_{k,\tau+1}^{(i)})\|^2 + \|\nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})\|^2 \\ - 2\langle \nabla f_i(\boldsymbol{w}_{k,\tau+1}^{(i)}), \nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)}) \rangle. \end{aligned} \tag{60}$$

Rearranging the above a bit, we get:

$$(2 - \eta_k L) \langle \nabla f_i(\boldsymbol{w}_{k,\tau+1}^{(i)}), \nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)}) \rangle \geq \|\nabla f_i(\boldsymbol{w}_{k,\tau+1}^{(i)})\|^2 + (1 - \eta_k L) \|\nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})\|^2. \tag{61}$$

But, we also have:

$$\langle \nabla f_i(\boldsymbol{w}_{k,\tau+1}^{(i)}), \nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)}) \rangle = \frac{1}{2} \left(\|\nabla f_i(\boldsymbol{w}_{k,\tau+1}^{(i)})\|^2 + \|\nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})\|^2 - \|\nabla f_i(\boldsymbol{w}_{k,\tau+1}^{(i)}) - \nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})\|^2 \right). \tag{62}$$

Using this in eq. (61) and simplifying a bit, we get:

$$\|\nabla f_i(\boldsymbol{w}_{k,\tau+1}^{(i)})\|^2 \leq \|\nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})\|^2 - \left(\frac{2}{\eta_k L} - 1\right) \|\nabla f_i(\boldsymbol{w}_{k,\tau+1}^{(i)}) - \nabla f_i(\boldsymbol{w}_{k,\tau}^{(i)})\|^2. \tag{63}$$

This completes the proof. ■

Fact 1 ([24]) *For an L-smooth function $h : \mathbb{R}^d \to \mathbb{R}$ with $h^* = \min_{\boldsymbol{x} \in \mathbb{R}^d} h(\boldsymbol{x})$ and $L > 0$, $\|\nabla h(\boldsymbol{x})\|^2 \leq 2L(h(\boldsymbol{x}) - h^*)$.*

Fact 2 *For any $p > 1$ vectors $\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_p\}$, $\| \sum_{i=1}^p \boldsymbol{y}_i \|^2 \leq p \sum_{i=1}^p \|\boldsymbol{y}_i\|^2$.*

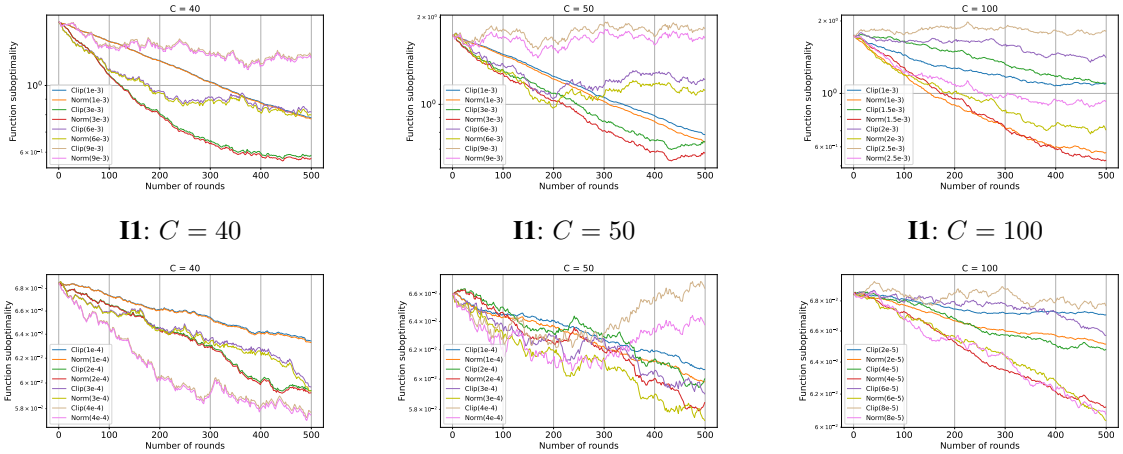
Fact 2 follows from Jensen's inequality.

Appendix F. Synthetic Experiments

Here we compare `DP-FedAvg` with clipping and `DP-NormFedAvg` on a simple but illustrative quadratic problem. We consider $f_i(\boldsymbol{w}) = \frac{1}{2}(\boldsymbol{w} - \boldsymbol{w}_i^*)^T \boldsymbol{Q}_i(\boldsymbol{w} - \boldsymbol{w}_i^*)$, where $i \in [100]$ (so, $n = 100$) and $\boldsymbol{w} \in \mathbb{R}^{200}$ (so, $d = 200$). Further, $\boldsymbol{w}_i^*$ is drawn i.i.d. from $\mathcal{N}(\mathbf{0}_{200}, \mathrm{I}_{200})$ and $\boldsymbol{Q}_i = \boldsymbol{A}_i \boldsymbol{A}_i^T$, where $\boldsymbol{A}_i$ is a 200×20 matrix whose entries are drawn i.i.d from $\mathcal{N}(0, \frac{1}{20^2})$; hence, $\boldsymbol{Q}_i$ is a PSD matrix with bounded maximum eigenvalue, due to which f_i is convex and smooth.

We set $(\varepsilon, \delta) = (5, 10^{-6})$, $K = 500$ and $E = 20$ for this set of experiments. We consider two different initializations with different distances from the global optimum $\boldsymbol{w}^*$ – **(i) I1**: $\boldsymbol{w}_0 = \boldsymbol{w}^* + \boldsymbol{z}$, and **(ii) I2**: $\boldsymbol{w}_0 = \boldsymbol{w}^* + \frac{\boldsymbol{z}}{5}$, where each coordinate of $\boldsymbol{z}$ is drawn i.i.d. from the continuous uniform distribution with support (0,1). We set $\eta_k = \beta_k = \eta$ for all rounds k, and also have full-device participation. In Figure 2, we plot the function suboptimality (i.e., $f(\boldsymbol{w}_k) - f(\boldsymbol{w}^*)$ at round number k) of `DP-FedAvg` with Clipping and `DP-NormFedAvg` for different values of η and clipping threshold/scaling factor C, for I1 and I2. In Figure 3, for each round k, we plot the corresponding SNR $:= \frac{\left\| \frac{1}{r} \sum_{i \in \mathcal{S}_k} \boldsymbol{g}_k^{(i)} \right\|}{\left\| \frac{1}{r} \sum_{i \in \mathcal{S}_k} \boldsymbol{\zeta}_k^{(i)} \right\|}$, where $\boldsymbol{g}_k^{(i)}$ and $\boldsymbol{\zeta}_k^{(i)}$ are the clipped/normalized per-client update and per-client noise, respectively, as defined in Algorithm 1. All plots are averaged over 3 independent runs. For a fair comparison, in each run, the exact same noise vectors (sampled randomly at each round) are used in both algorithms. The captions of Figures 2 and 3 discuss the results in detail. For further illustration, in Figure 4, we show the 2D projection of the trajectories of `DP-FedAvg` with clipping and `DP-NormFedAvg` for 1) $C = 100$ and $\eta = 0.001$, and 2) $C = 50$ and $\eta = 0.003$, both with initialization **I1**. In both cases, note that `DP-NormFedAvg` reaches closer to the optimum than `DP-FedAvg` with clipping.

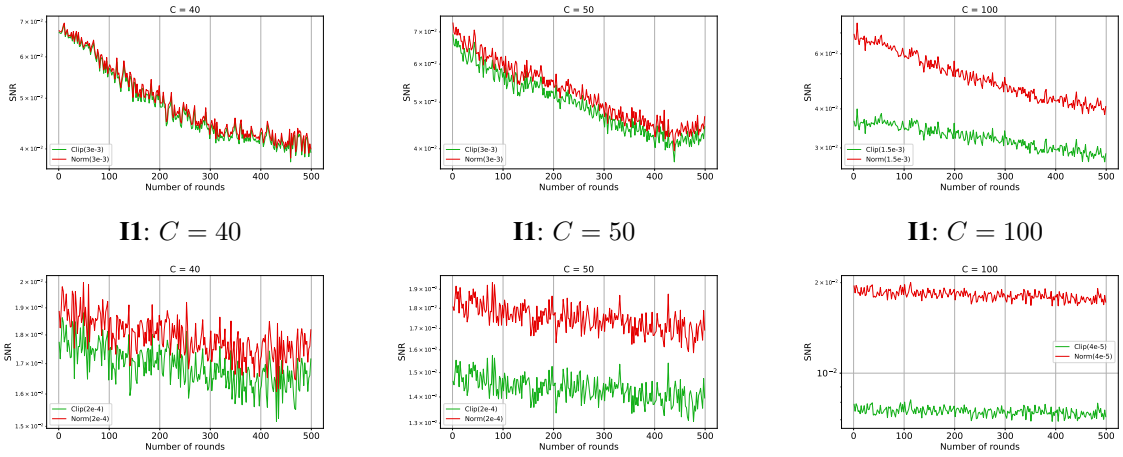
These plots corroborate our theoretical predictions and intuition.



I1: $C = 40$ **I1**: $C = 50$ **I1**: $C = 100$

I2: $C = 40$ **I2**: $C = 50$ **I2**: $C = 100$

Figure 2: Function suboptimality (i.e., $f(\boldsymbol{w}_k) - f(\boldsymbol{w}^*)$ at round number k) of `DP-FedAvg` with Clipping and `DP-NormFedAvg` for different values of η (recall, $\eta_k = \beta_k = \eta$ for all rounds k) and clipping threshold/scaling factor C, for I1 and I2 described in Appendix F. Specifically, "Clip(η)" and "Norm(η)" in the legend denote `DP-FedAvg` with Clipping and `DP-NormFedAvg` with $\eta_k = \beta_k = \eta$, respectively. All plots are averaged over three independent runs. For $C = \{50, 100\}$ and all values of η, normalization attains an appreciably lower function suboptimality than clipping. For $C = 40$ and lower, normalization and clipping are nearly equivalent, but clipping never does better than normalization. As mentioned before, if the client update norms are lower bounded by C_{low}, then clipping with threshold $C \leq C_{\text{low}}$ is equivalent to normalization with the same scaling factor.



I1: $C = 40$ **I1**: $C = 50$ **I1**: $C = 100$

I2: $C = 40$ **I2**: $C = 50$ **I2**: $C = 100$

Figure 3: In the same setting and with the same notation as Figure 2, comparison of SNR $:= \dfrac{\left\| \frac{1}{r} \sum_{i \in \mathcal{S}_k} \boldsymbol{g}_k^{(i)} \right\|}{\left\| \frac{1}{r} \sum_{i \in \mathcal{S}_k} \boldsymbol{\zeta}_k^{(i)} \right\|}$, where $\boldsymbol{g}_k^{(i)}$ and $\boldsymbol{\zeta}_k^{(i)}$ are the clipped/normalized per-client update and per-client noise, as defined in Algorithm 1. The SNR for only one value of η is shown here as the trend for other values of η is similar (and to avoid congestion). As per our discussion in Section 3.1, the SNR of normalization is never lower than that of clipping, explaining the superiority of the former. Also, similar to the trend in Figure 2, the improvement in SNR of normalization is much higher for $C = \{50, 100\}$ than $C = 40$.

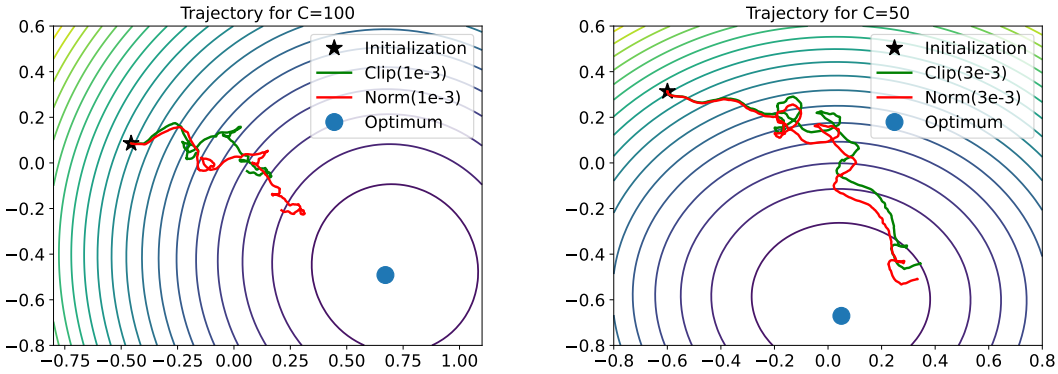


Figure 4: Smoothed 2D projection of the trajectories of `DP-FedAvg` with clipping and `DP-NormFedAvg` for the cases of $C = 100, \eta = 0.001$ (left) and $C = 50, \eta = 0.003$ (right) with initialization **I1** considered in Figure 2.

Appendix G. Experiments on Datasets

We consider the task of *private* multi-class classification to compare `DP-FedAvg` with clipping against `DP-NormFedAvg`; for brevity, we will often call them just clipping and normalization, respectively. Our experiments are performed on three benchmarking datasets, Fashion-MNIST [32] (abbreviated as FMNIST henceforth), CIFAR-10 and CIFAR-100, where the first two datasets have 10 classes each and the last one has 100 classes. Specifically, we consider logistic regression on FMNIST, CIFAR-10 and CIFAR-100 with ℓ_2-regularization; the weight decay value in PyTorch for ℓ_2-regularization is set to 1e-4. For FMNIST, we flatten each image into a 784-dimensional vector and use that as the feature vector. For CIFAR-10 and CIFAR-100, we use 512-dimensional features extracted from the last layer of a ResNet-18 [18] model pretrained on ImageNet. Similar to [23], we simulate a heterogeneous setting by distributing the data among the clients such that each client can have data from at most five classes. The exact procedure is described in Appendix H. For the CIFAR-10 and CIFAR-100 (respectively, FMNIST) experiment, the number of clients n is set to 5000 (respectively, 3000), with each client having the same number of samples. The number of participating clients in each round is set to $r = 0.2n$ for all datasets, with 20 local client updates per-round. We consider two privacy levels: $\varepsilon = \{5, 2\}$ with $\delta = 10^{-5}$; note that $\varepsilon = 5$ (resp., 2) corresponds to the low (resp., high) privacy regime. For clipping and normalization, the values of C that we tune over are $\{500, 250, 125, 62.5, 31.25, 15.625\}$. Details about the learning rate schedule are in Appendix H.

In Table 2, we show the comparison between clipping and normalization (in terms of test accuracy) for the two aforementioned privacy levels as well as vanilla `FedAvg` (without any privacy) as the baseline. The results reported here are the best ones for each algorithm by tuning over C and the learning rates, and have been averaged over three different runs. In all cases, normalization is clearly superior to clipping. It is worth noting that the improvement obtained with normalization is more for the low privacy regime (i.e., $\varepsilon = 5$).

(a)

Dataset	Algorithm	$\varepsilon = 5$	$\varepsilon = 2$
FMNIST	Clipping	75.59 ($\pm$ 0.04) %	57.42 ($\pm$ 0.08) %
	Normalization	**77.72** ($\pm$ 0.10) %	**58.32** ($\pm$ 0.12) %
	FedAvg (w/o privacy)	`83.43` ($\pm$ 0.02) %	

(b)

Dataset	Algorithm	$\varepsilon = 5$	$\varepsilon = 2$
CIFAR-10	Clipping	82.63 ($\pm$ 0.13) %	81.64 ($\pm$ 0.15) %
	Normalization	**84.21** ($\pm$ 0.19) %	**82.53** ($\pm$ 0.24) %
	FedAvg (w/o privacy)	`85.64` ($\pm$ 0.06) %	

(c)

Dataset	Algorithm	$\varepsilon = 5$	$\varepsilon = 2$
CIFAR-100	Clipping	56.53 ($\pm$ 0.10) %	41.69 ($\pm$ 0.21) %
	Normalization	**59.36** ($\pm$ 0.17) %	**43.12** ($\pm$ 0.25) %
	FedAvg (w/o privacy)	`64.61` ($\pm$ 0.07) %	

Table 2: Avg. test accuracy ($\pm$ 1 std.) over the last 5 rounds for (a) FMNIST, (b) CIFAR-10 and (c) CIFAR-100. "Clipping" and "Normalization" denote `DP-FedAvg` with Clipping and `DP-NormFedAvg`, respectively. In all cases, $\delta = 10^{-5}$. `FedAvg`, *without DP*, which is our baseline is at the bottom.

Appendix H. Experimental Details

First, we explain the procedure we have used to generate heterogeneous data for our FL experiments in Appendix G. For each dataset (individually), the training data was first sorted based on labels and then divided into $5n$ equal data-shards, where n is the number of clients. Splitting the data in this way ensures that each shard contains data from only one class for all datasets (and because n was chosen appropriately). Now, each client is assigned 5 shards chosen uniformly at random without replacement which ensures that each client can have data belonging to at most 5 distinct classes.

Next, we specify the learning rate schedule for our experiments in Appendix G. We use $\beta_k = 0.5\eta_k$ for all k. We employ the learning rate scheme suggested in [6] where we decrease the local learning rate by a factor of 0.99 after every round, i.e. $\eta_k = (0.99)^k \eta_0$. We search the best initial local learning rates η_0 over $\{10^{-3}, 2 \times 10^{-3}, 4 \times 10^{-3}, 8 \times 10^{-3}, 10^{-2}, 2 \times 10^{-2}, 4 \times 10^{-2}, 8 \times 10^{-2}, 10^{-1}, 2 \times 10^{-1}, 4 \times 10^{-1}, 8 \times 10^{-1}\}$ in each case. Server momentum = 0.8 is also applied.

All experiments are run on a single NVIDIA TITAN Xp GPU.