HAL open science

Differentially Private Federated Learning on Heterogeneous Data

Maxence Noble, Aurélien Bellet, Aymeric Dieuleveut

To cite this version:

Maxence Noble, Aurélien Bellet, Aymeric Dieuleveut. Differentially Private Federated Learning on Heterogeneous Data. 2021. hal-03498158

HAL Id: hal-03498158

https://inria.hal.science/hal-03498158v1

Preprint submitted on 20 Dec 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

arXiv:2111.09278v1 [cs.LG] 17 Nov 2021

Differentially Private Federated Learning on Heterogeneous Data

Maxence Noble
Centre de Mathématiques Appliquées
Ecole Polytechnique, France
Institut Polytechnique de Paris

Aurélien Bellet
Univ. Lille, Inria, CNRS,
Centrale Lille,
UMR 9189 - CRIStAL,
F-59000 Lille, France

Aymeric Dieuleveut
Centre de Mathématiques Appliquées
Ecole Polytechnique, France
Institut Polytechnique de Paris

Abstract

Federated Learning (FL) is a paradigm for large-scale distributed learning which faces two key challenges: (i) training efficiently from highly heterogeneous user data, and (ii) protecting the privacy of participating users. In this work, we propose a novel FL approach (DP-SCAFFOLD) to tackle these two challenges together by incorporating Differential Privacy (DP) constraints into the popular SCAFFOLD algorithm. We focus on the challenging setting where users communicate with a "honest-but-curious" server without any trusted intermediary, which requires to ensure privacy not only towards a third party observing the final model but also towards the server itself. Using advanced results from DP theory and optimization, we establish the convergence of our algorithm for convex and non-convex objectives. Our paper clearly highlights the trade-off between utility and privacy and demonstrates the superiority of DP-SCAFFOLD over the state-of-the-art algorithm DP-FedAvg when the number of local updates and the level of heterogeneity grows. Our numerical results confirm our analysis and show that DP-SCAFFOLD provides significant gains in practice.

1 INTRODUCTION

Federated Learning (FL) enables a set of users with local datasets to collaboratively train a machine learning model without centralizing data (Kairouz et al., 2021). Compared to machine learning in the cloud, the promise of FL is to avoid the costs of moving data and to mitigate privacy concerns. Yet, this promise can only be fulfilled if two key challenges are addressed. First, FL algorithms must be able to *efficiently deal with the high heterogeneity of data across users*, which stems from the fact that each local dataset reflects the usage and production patterns specific to a given user. Heterogeneous data may prevent FL algorithms from converging unless they use a large number of communication rounds between the users and the server, which is often considered as a bottleneck in FL (Khaled et al., 2020; Karimireddy et al., 2020b). Second, when training data contains sensitive or confidential information, FL algorithms must *provide rigorous privacy guarantees* to ensure that the server (or a third party) cannot accurately reconstruct this information from model updates shared by users (Geiping et al., 2020). The widely recognized way to quantify such guarantees is Differential Privacy (DP) (Dwork and Roth, 2013).

Since the seminal `FedAvg` algorithm proposed by McMahan et al. (2017a), a lot of effort has gone into addressing these two challenges *separately*. FL algorithms like `SCAFFOLD` (Karimireddy et al., 2020b) and `FedProx` (Li et al., 2020a) can better deal with heterogeneous data, while versions of `FedAvg` with Differential Privacy (DP) guarantees have been proposed based on the addition of random noise to the model updates (McMahan et al., 2017b; Geyer et al., 2018; Triastcyn and Faltings, 2019). Yet, we are not aware of any approach designed to tackle data heterogeneity while ensuring differential privacy, or of any work studying the associated trade-offs. This appears to be a challenging problem: on the one hand, data heterogeneity can hurt the privacy-utility trade-off of DP-FL algorithms (by requiring more communication rounds and thus more noise). On the other hand, it is not clear how to extend existing heterogeneous FL algorithms to satisfy DP and what the resulting privacy-utility trade-off would be in theory and in practice.

Our work precisely aims to tackle the issue of data heterogeneity in the context of FL under DP constraints. We aim to protect the privacy of any user's data against a honest-but-curious server observing all user updates, and against a third-party observing only

the final model. We present DP-SCAFFOLD, a novel differential private FL algorithm for training a global model from heterogeneous data based on SCAFFOLD (Karimireddy et al., 2020b) augmented with the addition of noise in the local model updates. Our convergence analysis leverages a particular initialization of the algorithm, and controls a different set of quantities than in the original proof.

Relying on recent tools for tightly keeping track of the privacy loss of the subsampled Gaussian mechanism (Wang et al., 2020) under Rényi Differential Privacy (RDP) (Mironov, 2017), we formally characterize the privacy-utility trade-off of DP-FedAvg, considered as the state-of-the-art DP-FL algorithm (Geyer et al., 2018), and DP-SCAFFOLD in convex and non-convex regimes. Our results show the superiority of DP-SCAFFOLD over DP-FedAvg when the number of local updates is large and/or the level of heterogeneity is high. Finally, we provide experiments on simulated and real-world data which confirm our theoretical findings and show that the gains achieved by DP-SCAFFOLD are significant in practice.

The rest of the paper is organized as follows. Section 2 reviews some background and related work on FL, data heterogeneity and privacy. Section 3 describes the problem setting and introduces DP-SCAFFOLD. In Section 4, we provide theoretical guarantees on both privacy and utility for DP-SCAFFOLD and DP-FedAvg. Finally, Section 5 presents the results of our experiments and we conclude with some perspectives for future work in Section 6.

2 RELATED WORK

Federated learning & heterogeneity. The baseline FL algorithm FedAvg (McMahan et al., 2017a) is known to suffer from instability and convergence issues in heterogeneous settings, related to device variability or non-identically distributed data (Khaled et al., 2020). In the last case, these issues stem from a *user-drift* in the local updates, which occurs even if all users are available or full-batch gradients are used (Karimireddy et al., 2020b). Several FL algorithms have been proposed to better tackle heterogeneity. FedProx (Li et al., 2020a) features a proximal term in the objective function of local updates. However, it is often numerically outperformed by SCAFFOLD (Karimireddy et al., 2020b), which relies on variance reduction through control variates. In a nutshell, the update direction of the global model at the server (c) and the update direction of each user i's local model (c_i) are estimated and combined in local Stochastic Gradient Descent (SGD) steps ($c - c_i$) to correct the user-drift (see Section 3.3 for more details).

MIME (Karimireddy et al., 2020a) also focuses on client heterogeneity and improves on SCAFFOLD by using the *stochastic gradient* evaluated on the global model as the local variate c_i and the synchronized *full-batch gradient* as the global control variate c. However, computing full-batch gradients is very costly in practice. Similarly, incorporating DP noise into FedDyn (Acar et al., 2020), which is based on the exact minimization of a proxy function, is not straightforward. On the other hand, the adaptation of SCAFFOLD to DP-SCAFFOLD is more natural as control variates only depend on stochastic gradients and thus do not degrade the privacy level throughout the iterations (see details in Section 4.1).

Extension to other optimization schemes. While Fed-Opt (Reddi et al., 2020) generalizes FedAvg by using different optimization methods locally (e.g., Adam (Kingma and Ba, 2014), AdaGrad (Duchi et al., 2011), etc., instead of vanilla local SGD steps) or a different aggregation on the central server, these methods may also suffer from user-drift. Their main objective is to improve the convergence rate (Wang et al., 2021) without focusing on heterogeneity. We thus choose to focus on the simplest algorithm to highlight the impact of DP and heterogeneity.

Federated learning & differential privacy. Even if datasets remain decentralized in FL, the privacy of users may still be compromised by the fact that the server (which may be "honest-but-curious") or a third-party has access to model parameters that are exchanged during or after training (Fredrikson et al., 2015; Shokri et al., 2017; Geiping et al., 2020). Differential Privacy (DP) (Dwork and Roth, 2013) provides a robust mathematical way to quantify the information that an algorithm A leaks about its input data. DP relies on a notion of *neighboring datasets*, which in the context of FL may refer to pairs of datasets differing by one user (*user-level* DP) or by one data point of one user (*record-level* DP).

Definition 2.1 (Differential Privacy, Dwork and Roth, 2013)**.** *Let $\epsilon, \delta > 0$. A randomized algorithm $A : \mathcal{X}^n \to \mathcal{Y}$ is (ϵ, δ)-DP if for all pairs of neighboring datasets D, D' and every subset $S \subset \mathcal{Y}$, we have:*

$$\mathbb{P}[A(D) \in S] \leq e^{\epsilon}\mathbb{P}[A(D') \in S] + \delta.$$

The privacy level is controlled by the parameters ϵ and δ (the lower, the more private). A standard building block to design DP algorithms is the Gaussian mechanism (Dwork and Roth, 2013), which adds Gaussian noise to the output of a non-private computation. The variance of the noise is calibrated to the sensitivity of the computation, i.e., the worst-case change (measured in ℓ_2 norm) in its output on two neighboring datasets.

The design of private ML algorithms heavily relies on the Gaussian mechanism to randomize intermediate data-dependent computations (e.g. gradients). The privacy guarantees of the overall procedure are then obtained via *composition* (Dwork et al., 2010; Kairouz et al., 2017). Recent theoretical tools like *Rényi Differential Privacy* (Mironov, 2017) (see Appendix B) allow to obtain tighter privacy bounds for the Gaussian mechanism under composition and data subsampling (Wang et al., 2020).

In the context of FL, the output of an algorithm A in the sense of Definition 2.1 contains all information observed by the party we aim to protect against. Some work considered a trusted server and thus only protect against a third-party who observes the final model. In this setting, McMahan et al. (2017b) introduced `DP-FedAvg` and `DP-FedSGD` (i.e., `DP-FedAvg` with a single local update), which was also proposed independently by Geyer et al. (2018). These algorithms extend `FedAvg` and `FedSGD` by having the server add Gaussian noise to the aggregated user updates. Triastcyn and Faltings (2019) used a relaxation of DP known as Bayesian DP to provide sharper privacy loss bounds. However, these papers do not discuss the theoretical trade-off between utility and privacy. Some recent work by Wei et al. (2020) has formally examined this trade-off for `DP-FedSGD`, providing a utility guarantee for strongly convex loss functions. However, they do not consider multiple local updates.

Some papers also considered the setting with a "honest-but-curious" server, where users must randomize their updates locally before sharing them. This corresponds to a stronger version of DP, referred to as *Local Differential Privacy* (LDP) (Duchi et al., 2014; Zhao et al., 2021; Duchi et al., 2018). `DP-FedAvg` and `DP-FedSGD` can be easily adapted to this setting by pushing the Gaussian noise addition to the users, which induces a cost in utility. Zhao et al. (2021) consider `DP-FedSGD` in this setting but do not provide any utility analysis. Girgis et al. (2020) provide utility and compression guarantees for variants of `DP-FedSGD` in an intermediate model where a trusted *shuffler* between the server and the users randomly permutes the user contributions, which is known to amplify privacy (Balle et al., 2019; Cheu et al., 2019; Ghazi et al., 2019; Erlingsson et al., 2019). However, both of these studies do not consider multiple local updates, which is key to reduce the number of communication rounds. Li et al. (2020b) consider the server as "honest-but-curious" but does not ensure end-to-end privacy to the users. Finally, Hu et al. (2020) present a personalized DP-FL approach as a way to tackle data heterogeneity, but it is limited to linear models.

Summary. To the best of our knowledge, there exists no FL approach designed to tackle *data heterogeneity under DP constraints*, or any study of existing DP-FL algorithms capturing the impact of data heterogeneity on the privacy-utility trade-off.

3 DP-SCAFFOLD

In this section, we first describe the framework that we consider for FL and DP, before giving a detailed description of `DP-SCAFFOLD`. A table summarizing all notations is provided in Appendix A.

3.1 Federated Learning Framework

We consider a setting with a central server and M users. Each user $i \in [M]$, holds a private local dataset $D_i = \{d_1^i, ..., d_R^i\} \subset \mathcal{X}^R$, composed of R observations living in a space $\mathcal{X}$. We denote by $D := D_1 \sqcup ... \sqcup D_M$ the disjoint union of all user datasets. Each dataset D_i is supposed to be independently sampled from distinct distributions. The objective is to solve the following empirical risk minimization problem over parameter x:

$$\min_{x \in \mathbb{R}^d} F(x) := \frac{1}{M} \sum_{i=1}^{M} F_i(x),$$

where $F_i(x) := \frac{1}{R} \sum_{j=1}^{R} f_i(x, d_j^i)$ is the empirical risk on user i, and for all $x \in \mathbb{R}^d$ and $d \in \mathcal{X}$, $f_i(x, d)$ is the loss of the model x on observation d. We denote by $\nabla f_i(x, d_j^i)$ the gradient of the loss f_i computed on a sample $d_j^i \in D_i$, and by extension, for any $S_i \subset D_i$, $\nabla f_i(x, S_i) := \frac{1}{|S_i|} \sum_{j \in S_i} \nabla f_i(x, d_j^i)$ is the averaged mini-batch gradient. We note that our results can easily be adapted to optimize any weighted average of the loss functions and to imbalanced local datasets.

3.2 Privacy Model

We aim at controlling the information leakage from individual datasets D_i in the updates shared by the users. For simplicity, our analysis focuses on *record-level* DP with respect to (w.r.t) the joint dataset D. We thus consider the following notion of neighborhood: $D, D' \in \mathcal{X}^{MR}$ are *neighboring datasets* (denoted $||D - D'|| \leq 1$) if they differ by at most one record, that is if there exists at most one $i \in [M]$ such that D_i and D_i' differ by one record. We want to ensure privacy (or quantify privacy level) (i) towards a third party observing the final model and (ii) towards an honest-but-curious server. Our DP budget is set in advance and denoted by (ϵ, δ), and corresponds to the desired level of privacy towards a third party observing the final model (or any model during the training pro-

Algorithm 1: DP-SCAFFOLD$(T, K, l, s, \sigma_g, \mathcal{C})$

Server Input: initial x^0, initial c^0
i-th User Input: initial c_i^0
Output: x^T

1 **for** $t = 1, ..., T$ **do**
2 **User subsampling by the server:**
3 Sample $C^t \subset [M]$
4 **Server sends** (x^{t-1}, c^{t-1}) to users $i \in C^t$
5 **for** *user* $i \in C^t$ **do**
6 Initialize model: $y_i^0 \leftarrow x^{t-1}$
7 **for** $k = 1, ..., K$ **do**
8 **Data subsampling by user** i**:**
 $S_i^k \subset D_i$
9 **for** *sample* $j \in S_i^k$ **do**
10 **Compute gradient:**
 $g_{ij} \leftarrow \nabla f_i(y_i^{k-1}, d_j^i)$
11 **Clip gradient:**
 $\tilde{g}_{ij} \leftarrow g_{ij} / \max\left(1, ||g_{ij}||_2 / \mathcal{C}\right)$
12 **Add DP noise to local gradients:**
13 $\tilde{H}_i^k \leftarrow \frac{1}{sR} \sum_{j \in S_i^k} \tilde{g}_{ij} + \frac{2\mathcal{C}}{sR} \mathcal{N}(0, \sigma_g^2)$
14 $y_i^k \leftarrow y_i^{k-1} - \eta_l(\tilde{H}_i^k - c_i^{t-1} + c^{t-1})$
15 $\tilde{c}_i^t \leftarrow c_i^{t-1} - c^{t-1} + \frac{1}{K\eta_l}(x^{t-1} - y_i^K)$
16 $(\Delta y_i^t, \Delta c_i^t) \leftarrow (y_i^K - x^{t-1}, \tilde{c}_i^t - c_i^{t-1})$
17 **User** i **sends to server** $(\Delta y_i^t, \Delta c_i^t)$
18 $c_i^t \leftarrow \tilde{c}_i^t$
19 **Server aggregates**:
20 $(\Delta x^t, \Delta c^t) \leftarrow \frac{1}{lM} \sum_{i \in C^t} (\Delta y_i^t, \Delta c_i^t)$
21 $x^t \leftarrow x^{t-1} + \eta_g \Delta x^t$, $c^t \leftarrow c^{t-1} + l \Delta c^t$

cess).[1] We will also report the corresponding (weaker) DP guarantees towards the server.

3.3 Description of DP-SCAFFOLD

We now explain how our algorithm DP-SCAFFOLD is constructed. DP-SCAFFOLD proceeds similarly as standard FL algorithms like FedAvg: all users perform a number of local updates K, before communicating with the central server. We denote T the number of communication rounds. As SCAFFOLD, DP-SCAFFOLD relies on the use of control variates that are updated throughout the iterations of the algorithm: (i) on the server side (c, downloaded by the users) and (ii) on the user side ($\{c_i\}_{i \in [M]}$, uploaded to the server).

At any round $t \in [T]$, a subset C^t of users with cardinality $\lfloor lM \rfloor$ is uniformly selected by the server, where l is the user sampling ratio. Each user $i \in C^t$ downloads the global model x^{t-1} held by the central server and performs K local updates on their local copy y_i of the model (with step-size $\eta_l \geq 0$), starting from $y_i^0 = x^{t-1}$.

At iteration $k \in [K]$, user $i \in C^t$ samples an independent $\lfloor sR \rfloor$-mini-batch of data $S_i^k \subset D_i$, where s is the data sampling ratio. Given a clipping parameter $\mathcal{C} > 0$, for all $j \in S_i^k$, the gradient $\nabla f_i(y_i^{k-1}, d_j^i)$ is computed and clipped at threshold $\mathcal{C}$ (Abadi et al., 2016), giving $\tilde{g}_{ij}$. The resulting average stochastic gradient $H_i^k(y_i^{k-1})$ is made private w.r.t. D_i using Gaussian noise calibrated to the ℓ_2-sensitivity $S = 2\mathcal{C}/sR$ and to the scale σ_g (a parameter which will depend on the privacy budget), giving $\tilde{H}_i^k(y_i^{k-1})$ such that

$$\tilde{H}_i^k(y_i^{k-1}) := H_i^k(y_i^{k-1}) + S\mathcal{N}(0, \sigma_g^2).$$

Finally, we update the model y_i^{k-1} (omitting index t):

$$y_i^k \leftarrow y_i^{k-1} - \eta_l\Big(\underbrace{\tilde{H}_i^k(y_i^{k-1})}_{\text{“noisy” gradient}} + \underbrace{c^{t-1} - c_i^{t-1}}_{\text{drift correction}} \Big), \quad (1)$$

using the control variates which are updated at the end of each inner loop:

$$c_i^t \leftarrow c_i^{t-1} - c^{t-1} + \frac{1}{K\eta_l}(x^{t-1} - y_i^K) = \frac{1}{K} \sum_{k=1}^{K} \tilde{H}_i^k(y_i^{k-1}).$$

After K local iterations, each user communicates $(y_i^K - x^{t-1})$ and $(c_i^t - c_i^{t-1})$ to the central server, and updates the global model with step-size η_g, as described in Step 21 of Alg. 1.

From the privacy point of view, the updates $(\Delta y_i, \Delta c_i)$ that are transmitted to the server are private w.r.t. D (proved in Section 4.1), thus making private (x, c) w.r.t D by postprocessing.

The complete pseudo-code is given in Algorithm 1. Subsampling steps, which amplify privacy (Kasiviswanathan et al., 2011), are highlighted in red, and steps specifically related to DP are highlighted in yellow. Setting $\sigma_g = 0$ and $\mathcal{C} = \infty$ recovers the classical SCAFFOLD algorithm, and removing control variates (i.e., setting c_i^t to 0 for all $t \in [T], i \in [M]$) recovers DP-FedAvg, which we describe in Appendix A (Algorithm 2) for completeness.

Intuition for control variates. In SCAFFOLD, the local control variate c_i converges to the local gradient $\nabla f_i(x^*)$ at the optimal, while c approximates $\frac{1}{M}\sum_{i=1}^{M} c_i$ (Karimireddy et al., 2020b, Appendix E). Therefore, adding $(c - c_i)$ in the update balances the local stochastic gradient and limits user-drift.

Warm-start version of DP-SCAFFOLD. We adapt the *warm-start strategy* from Karimireddy et al. (2020b, Appendix E) to accommodate DP constraints,

[1] The use of composition for analyzing the privacy guarantee for the final model implies that the same guarantee holds even if every intermediate *global* model is observed.

leading to `DP-SCAFFOLD-warm`. The first few rounds of communication are saved to set[2] the initial values of the control variates to $c^0 = \frac{1}{M}\sum_{i=1}^M c_i^0$, with $c_i^0 = \frac{1}{K}\sum_{k=1}^K \tilde{H}_i^k(x^0)$ (perturbed by DP-noise), without updating the global model. Note that as we leverage user sampling in the privacy analysis, the server cannot communicate with all users at a single round and the users have to be *randomly* picked to ensure privacy. We prove the convergence of `DP-SCAFFOLD-warm` in Section 4.2 (assuming that every user participated to the warm-start phase). Our experiments in Section 5 are conducted with this version of `DP-SCAFFOLD`.

User-level privacy. Our framework can easily be adapted to *user-level* privacy, by setting $S = 2\mathcal{C}/s$.

4 THEORETICAL ANALYSIS

We first provide the analysis of the privacy level in Section 4.1, then analyze utility in Section 4.2.

4.1 Privacy

We first establish that the setting of our algorithms `DP-SCAFFOLD` and `DP-FedAvg` enables a fair comparison in terms of privacy.

Claim 4.1. *For some given noise scale $\sigma_g > 0$, x^t has the same level of privacy at any round $t \in [T]$ in `DP-SCAFFOLD(-warm)` and `DP-FedAvg` after the server aggregation.*

This claim can be proved by induction, see Appendix B. Consequently, the analysis of privacy is similar for `DP-FedAvg` or `DP-SCAFFOLD`. Theorem 4.1 gives the order of magnitude of σ_g (same for `DP-FedAvg` and `DP-SCAFFOLD`) to ensure DP towards the server or any third party. In order to give simple closed forms for the privacy guarantees, we make the following assumption.

Assumption 1. *We consider a noise level σ_g, a privacy budget $\epsilon > 0$ and a data-subsampling ratio s s.t.: (i) $s = o(1)$, (ii) $\epsilon < 1$ and (iii) $\sigma_g = \Omega(s\sqrt{K/\log(2Tl/\delta)})$ (high privacy regime).*

Theorem 4.1 (Privacy guarantee)**.** *Let $\epsilon, \delta > 0$. Under Assumption 1, suppose we set $\sigma_g = \Omega\big(s\sqrt{lTK\log(2Tl/\delta)\log(2/\delta)}/\epsilon\sqrt{M}\big)$. Then, for `DP-SCAFFOLD(-warm)` and `DP-FedAvg`, x^T is:*

1. *$\big(\mathcal{O}(\epsilon), \delta\big)$-DP towards a third-party,*
2. *$\big(\mathcal{O}(\epsilon_s), \delta_s\big)$-DP towards the server, where $\epsilon_s = \epsilon\sqrt{\frac{M}{l}}$ and $\delta_s = \frac{\delta}{2}(\frac{1}{l}+1)$.*

[2]This happens with high probability: typically, after $4/l$ where $l = o(1)$, all users have been selected at least once with probability $1 - e^{-4} \approx 0.98$.

Sketch of proof. We here summarize the main steps of the proof. Let σ_g be a given DP noise level. Our proof stands for the privacy analysis over a query function of sensitivity 1 (since calibration is made with constant S in Section 3.2). We denote $\mathrm{GM}(\sigma_g)$ the corresponding Gaussian mechanism. We first provide the result for any third party.

We combine the following steps:

- *Data-subsampling with Rényi DP.* Let $t \in [T]$ be an arbitrary round. We first estimate an upper DP bound ϵ_a (w.r.t. D) of the privacy loss after the *aggregation* by the server of lM individual contributions (Step 21 in Alg. 1). Those are *private* w.r.t. to the corresponding local datasets, say (α, ϵ_i)-RDP w.r.t. D_i where $i \in C^t$ stands for the i-th user, each one being the result of the composition of K adaptative s-subsampled $\mathrm{GM}(\sigma_g)$. For any $\alpha > 1$, we know that $\mathrm{GM}(\sigma_g)$ is $(\alpha, \alpha/2\sigma_g^2)$-RDP (Mironov, 2017). Wang et al. (2020) proves that the s-subsampled $\mathrm{GM}(\sigma_g)$ is $(\alpha, \mathcal{O}(s^2\alpha/\sigma_g^2))$-RDP under Assumption 1-(i). By the RDP composition rule over the K local iterations, we have $\epsilon_i(\alpha) \leq \mathcal{O}(Ks^2\alpha/\sigma_g^2)$. Therefore, the aggregation over all users considered in C^t is private w.r.t. D with a corresponding Gaussian noise of variance $S^2\sigma_a^2$ where $\sigma_a^2 = \frac{1}{lM}\frac{\sigma_g^2}{Ks^2}$ (mean of independent Gaussian noises). Yet, making the whole aggregation private w.r.t. D only requires a l_2 calibration equal to $S' = S/lM$ (by triangle inequality) which means we can quantify the gain of privacy as $(\alpha, \mathcal{O}(Ks^2\alpha/lM\sigma_g^2))$-RDP. After converting this result into a DP bound (Mironov, 2017), we get that for any $\delta' > 0$, the whole mechanism is $(\epsilon_a(\alpha, \delta'), \delta')$-DP where $\epsilon_a(\alpha, \delta') = \mathcal{O}\big(\frac{Ks^2\alpha}{lM\sigma_g^2} + \frac{\log(1/\delta')}{\alpha-1}\big)$.
- *User-subsampling with DP.* In order to get explicit bounds (that may not be optimal), we then use classical DP tools to estimate an upper DP bound ϵ_T after T rounds. By combining amplification by subsampling results (Kasiviswanathan et al., 2011) over users and strong composition (Kairouz et al., 2017) (with Assumption 1-(ii)) over communication rounds, we finally get that, for any $\delta'' > 0$, x^T is $(\epsilon_T(\alpha, \delta', \delta''), Tl\delta' + \delta'')$-DP where $\epsilon_T(\alpha, \delta', \delta'') = \mathcal{O}(l\epsilon_a(\alpha, \delta')\sqrt{T\log(1/\delta'')})$.
- *Fixing parameters.* Considering our final privacy budget δ for any third party, we fix $\delta' := \delta/2Tl$ and $\delta'' := \delta/2$. Following the method of the *Moments Accountant* (Abadi et al., 2016), we then minimize the bound on ϵ_T w.r.t. $\alpha > 1$, which gives that $\epsilon_T = \mathcal{O}(\tilde{\epsilon})$ where

$$\tilde{\epsilon} = l\sqrt{T\log(2/\delta)}\left(\frac{s\sqrt{K\log(2Tl/\delta)}}{\sigma_g\sqrt{lM}} + \frac{Ks^2}{lM\sigma_g^2}\right).$$

Finally, under Assumption 1-(iii), we can bound the second term by the first one. We then invert the formula of this upper bound of $\tilde{\epsilon}$ to express σ_g as a func-

tion of a given privacy budget ϵ, which proves the first statement.

To prove the second statement, we recall that the server has access to individual contributions before aggregation (which prevents a reduction by a factor lM of the variance) and that it knows the selected users at each round, which cancels the user-sampling effect (factor l). We refer to Appendix B for the full proof as well as the non-asymptotic (tighter) formulas.

Remarks. Our RDP analysis allows to limit the impact of K in the expression of σ_g. A standard analysis would require a noise level increased by an extra factor $\mathcal{O}(\sqrt{\log(TKls/\delta)})$ (see Appendix B for more details).

We also stress the fact that the approximations in the privacy upper bounds are meant to give readable closed-form results. In practice, we do not use these approximations: instead, we numerically compute a tighter σ_g directly from the non-asymptotic formulas.

4.2 Utility

We denote by $\|\cdot\|$ the Euclidean ℓ_2-norm. We assume that F is bounded from below by $F^* = F(x^*)$, for an $x^* \in \mathbb{R}^d$. Furthermore, we make standard assumptions on the functions $(F_i)_{i\in[M]}$.

Assumption 2. *For all $i \in [M]$, F_i is differentiable and ν-smooth (i.e., ∇F_i is ν-Lipschitz).*

We also make the following assumption on the stochastic gradients and data sampling.

Assumption 3. *For any iteration $t \in [T], k \in [K]$,*

1. the stochastic gradient $\nabla f_i(y_i^{k-1}, d_j^i)$ is conditionally unbiased, i.e., $\mathbb{E}_{d_j^i}[\nabla f_i(y_i^{k-1}, d_j^i)|y_i^{k-1}] = \nabla F_i(y_i^{k-1})$.
2. the stochastic gradient has bounded variance, i.e., for any $y \in \mathbb{R}^d$, $\mathbb{E}_{d_j^i}[\|\nabla f_i(y, d_j^i) - \nabla F_i(y)\|^2] \leq \varsigma^2$.
3. there exists a clipping constant $\mathcal{C}$ independent of i, j such that $\|\nabla f_i(y_i^{k-1}, d_j^i)\| \leq \mathcal{C}$.

The first condition is naturally satisfied when d_j^i is uniformly sampled in $[R]$. The second condition is classical in the literature, and can be relaxed to only assume that the noise is bounded at the optimal point x^* (Gower et al., 2019). Remark that consequently, the variance of a mini-batch of size sR uniformly sampled over D_i is upper bounded by ς^2/sR. Finally, the third point ensures that we can safely ignore the impact of gradient clipping.

Lastly, to obtain a convergence guarantee for `DP-FedAvg` (but not for `DP-SCAFFOLD`), we use Assumption 4 on the data-heterogeneity, which bounds gradients ∇f_i towards ∇f.

Assumption 4 (Bounded Gradient dissimilarity)**.** *There exist constants $G \geq 0$ and $B \geq 1$ such that:*

$$\forall x \in \mathbb{R}^d, \frac{1}{M}\sum_{i=1}^{M} ||\nabla F_i(x)||^2 \leq G^2 + B^2||\nabla F(x)||^2.$$

Quantifying the heterogeneity between users by controlling the difference between the local gradients and the global one is classical in federated optimization (e.g. Kairouz et al., 2021). We can now state a utility result in the convex case, by considering $\sigma_g^* := s\sqrt{lTK\log(2Tl/\delta)\log(2/\delta)}/\epsilon\sqrt{M}$ (order of magnitude of noise scale to approximately ensure end-to-end (ϵ, δ)-DP w.r.t. D according to Theorem 4.1). This result is extended to the strongly convex and non-convex cases in Appendix C.

Theorem 4.2 (Utility result - convex case)**.** *Assume that for all $i \in [M]$, F_i is convex. Let $x^0 \in \mathbb{R}^d$ and denote $D_0 := ||x^0 - x^*||$. Under Assumptions 2 and 3, we consider the sequence of iterates $(x^t)_{t\geq 0}$ of Algorithm 1 (`DP-SCAFFOLD`) and Algorithm 2 (`DP-FedAvg`), starting from x^0, and with DP noise $\sigma_g := \sigma_g^*$. Then there exist step-sizes (η_g, η_l) and weights $(w_t)_{t\in[T]}$ such that the expected excess of loss $\mathbb{E}[F(\overline{x}^T)] - F^*$, where $\overline{x}^T = \sum_{t=1}^{T} w_t x^t$, is bounded by:*

- *For `DP-FedAvg`, under Assumption 4:*

$$\mathcal{O}\Bigg(\underbrace{\frac{D_0\mathcal{C}\sqrt{d\log(Tl/\delta)\log(1/\delta)}}{\epsilon MR}}_{\textit{privacy bound}} + \underbrace{\frac{\varsigma D_0}{\sqrt{sRlMKT}} + \frac{B^2\nu D_0^2}{T} + \frac{GD_0\sqrt{1-l}}{\sqrt{lMT}} + \frac{D_0^{4/3}\nu^{1/3}G^{2/3}}{T^{2/3}}}_{\textit{optimization bound}}\Bigg).$$

- *For `DP-SCAFFOLD-warm`:*

$$\mathcal{O}\Bigg(\underbrace{\frac{D_0\mathcal{C}\sqrt{d\log(Tl/\delta)\log(1/\delta)}}{\epsilon MR}}_{\textit{privacy bound}} + \underbrace{\frac{\varsigma D_0}{\sqrt{sRlMKT}} + \frac{\nu D_0^2}{l^{2/3}T}}_{\textit{optimization bound}}\Bigg).$$

The two bounds given in Theorem 4.2 consist of two and three terms respectively:

1. A classical convergence rate resulting from (non-private) first order optimization. This term is highlighted in green. The dominant part, as $T \to \infty$, is $\frac{\varsigma D_0}{\sqrt{sRlMKT}}$. This term is inversely proportional to the square root of the *total number of iterations* TK times the *average number of gradients computed per iteration* $lM \times sR$, and increases proportionally to the stochastic gradients' standard deviation ς and the initial distance to the optimal point D_0.

2. An extra term showing that heterogeneity hinders the convergence of `DP-FedAvg`, for which Assumption 4 is required. This term is highlighted in blue. Here, as $T \to \infty$, the dominant term in it is $\frac{GD_0\sqrt{1-l}}{\sqrt{lMT}}$, except if the user sampling ratio $l = 1$, then the dominating term becomes $\frac{D_0^{4/3}\nu^{1/3}G^{2/3}}{T^{2/3}}$. Both these terms do not decrease with the number of local iterations K, and increase with heterogeneity constant G. **This extra term for `DP-FedAvg` highlights the superiority of `DP-SCAFFOLD` over `DP-FedAvg` under data heterogeneity.**
3. Lastly, an additional term showing the impact of DP appears. This term is diverging with the number of iterations T, which results in the privacy-utility trade-off on T. Moreover, this term decreases proportionally to the whole number of data records MR. It also outlines the cost of DP since it sublinearly grows with the size of the model d and dramatically increases inversely to the DP budget ϵ.

Take-away messages. Our analysis highlights that: (i) `DP-SCAFFOLD` improves on `DP-FedAvg` in the presence of heterogeneity; and (ii) increasing the number of local updates K is very profitable to `DP-SCAFFOLD`, as it improves the dominating optimization bound without degrading the privacy bound. These aspects are numerically confirmed in Section 5.

Sketch of proof and originality. To establish Theorem 4.2, we adapt the proof of Theorems V and VII in Karimireddy et al. (2020b). However, we consider a weakened assumption on stochastic gradients due the addition of Gaussian noise in the local updates. Consequently, in order to limit the impact of this additional noise, we change the quantity (Lyapunov function) that is controlled during the proof: we combine the squared distance to the optimal point $\|x^t - x_*\|^2$ to a control of the lag at iteration t, $\frac{1}{KM}\sum_{k=1}^{K}\sum_{i=1}^{M}\mathbb{E}\|\alpha_{i,k-1}^t - x^t\|^2$; where we ensure that our control variates $(c_i^t)_{i\in[M]}$ at iteration t correspond to noisy stochastic gradients measured at points $(\alpha_{i,k-1}^t)_{i\in[M]}$, that is, $c_i^t = \frac{1}{K}\sum_{k=1}^{K}\tilde{H}_i^k\left(\alpha_{i,k-1}^t\right)$.[3] This proof is detailed in Appendix C.

Remark. To obtain the utility result, we have to ensure that initial users' controls c_i^0 are set as follows: $c_i^0 = \frac{1}{K}\sum_{k=1}^{K}\tilde{H}_i^k(x^0)$ (notations of Alg. 1). Our theoretical result thus only holds for the `DP-SCAFFOLD-warm` version.

Extension to other local randomizers. Instead of the Gaussian mechanism, other randomizers could be applied, possibly to the *per-example* gradients. Our utility analysis would easily carry over as long as the chosen mechanism is unbiased and has explicit variance (see Appendix C). On the other hand, a tight RDP bound on the subsampling of this mechanism would be needed to provide the same proof of privacy as in Section 4.1 (see the work of Wang et al., 2020, for more details). Otherwise, classic DP results for composition and subsampling must be used instead.

[3] In contrast, the proof in the convex case in Karimireddy et al. (2020b) relies on controlling $\frac{1}{M}\sum_{i=1}^{M}\mathbb{E}\|c_i^t - \nabla f_i(x^*)\|^2$.

5 EXPERIMENTS

Global setting. In our experiments,[4] we perform federated logistic regression with a regularization parameter (5.10^{-3}) for simulated and real-world data over 3 random runs. We split each dataset in train/test sets with proportion 80%/20%. Both train and test data are preprocessed (standardization and normalization) before the training phase. We fix sampling constants $l = 0.2, s = 0.2$, privacy parameter $\delta = 1/MR$, global step-size $\eta_g = 1$ and $\eta_l = \eta_0/sK$, where η_0 is carefully tuned (see Appendix D.1). Details on the clipping heuristic are also given in Appendix D.1. For each dataset, we plot the *test accuracy* or the *train loss* (averaged over the 3 runs) w.r.t. the communication rounds of 6 algorithms: `FedAvg`, `FedSGD` (`FedAvg` with $Ks = 1$), `SCAFFOLD-warm`, with and without DP. For the sake of fairness, we choose $T \times K$ (total number of iterations) to be constant throughout the settings.

Simulated data. To generate synthetic data, we follow a setup detailed by Li et al. (2020a), which enables to control heterogeneity between users' local models and between users' data distributions, respectively with parameters α and β (the higher, the more heterogeneous). We tackle a 10-classes problem with input dimension $d' = 40$ over $M = 100$ users, each holding $R = 5000$ samples. Details on data generation are given in Appendix D.2. We consider $\epsilon = 1.5$ in this setting (thus $\epsilon_s = 7$), and compare three levels of heterogeneity $\big((\alpha, \beta) \in \{(0,0), (1,1), (5,5)\}\big)$ for two situations: (i) with 10 local epochs ($K = 50$) and $T = 1,600$ (see Fig 1, first row), (ii) with 20 local epochs ($K = 100$) and $T = 800$ (see Fig 1, second row).

Real-world data. This experiment is conducted on the EMNIST-'balanced' dataset (Cohen et al., 2017), which consists of 47 classes (letters and numbers) containing all together $131,600$ samples. Our dataset is divided between $M = 40$ users, who all have $r = 2500$ data records. Heterogeneity is controlled by parameter γ. For γ% similar data, we allocate to each user

[4] Code available at: Github

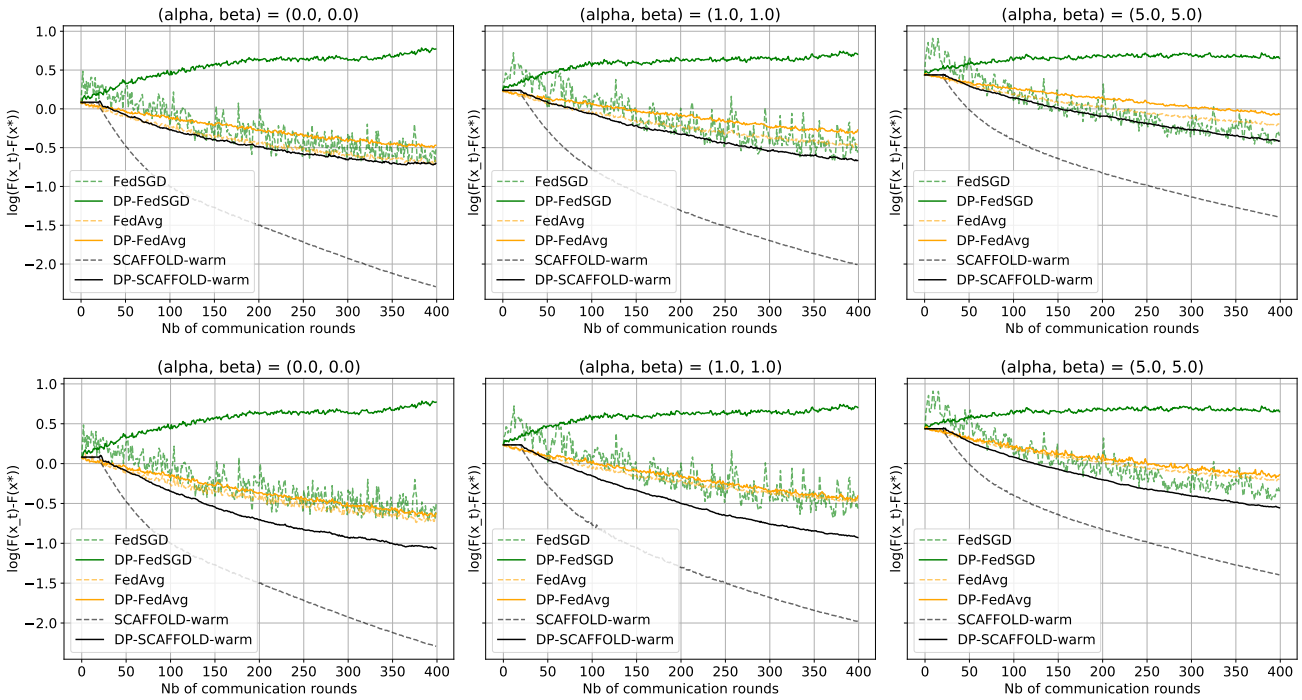

Figure 1: Train loss on simulated data with $(1.5, 2.10^{-6})$-DP. First row: $K = 50$; Second row: $K = 100$.

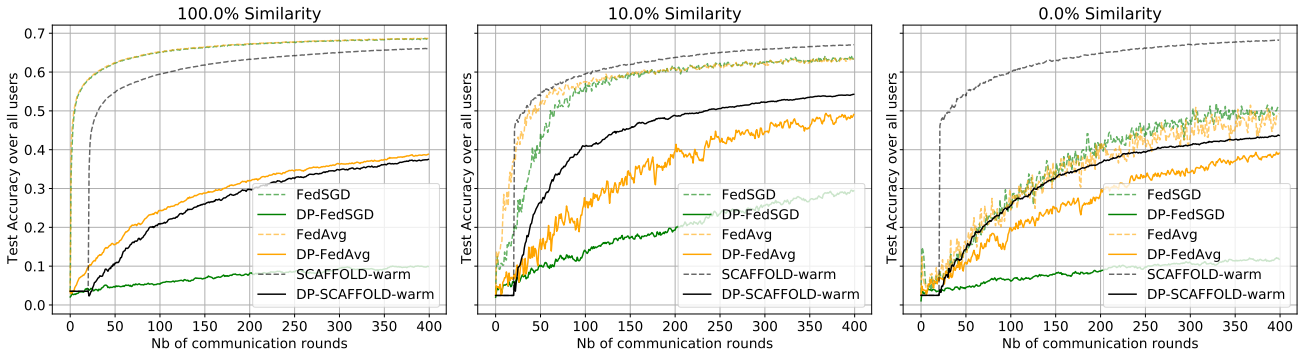

Figure 2: Test accuracy on FEMNIST data with $(4.5, 10^{-5})$-DP: $K = 50$.

γ% i.i.d. data and the remaining $(100 - \gamma)$% by sorting according to the label (Hsu et al., 2019), which corresponds to 'FEMNIST'. We consider $\epsilon = 4.5$ in this setting (thus $\epsilon_s = 20$) and we compare $\gamma \in \{0\%, 10\%, 100\%\}$, for the following setting for T, K: with 10 local epochs ($K = 50$) and $T = 800$ (see Fig 2).

Results. We make the following observations. (I) For both simulated and real data, we observe that `DP-SCAFFOLD` outperforms `DP-FedAvg` under the same level of privacy, and that this difference increases with heterogeneity and the number of local updates K, which confirms the theory. (II) We provide other experiments with various settings in Appendix D.3. We also study the effect of K on convergence for `DP-FedAvg` and `DP-SCAFFOLD-warm`.

6 CONCLUSION

Our paper introduced a novel FL algorithm, `DP-SCAFFOLD`, to tackle data heterogeneity under DP constraints, and showed that it improves over the baseline `DP-FedAvg` from both the theoretical and empirical point of view. In particular, our theoretical analysis highlights an interesting trade-off between the parameters of the problem, involving a term of heterogeneity in `DP-FedAvg` which does not appear in the rate of `DP-SCAFFOLD`. As future work, we aim at providing additional numerical experiments with deep learning models and various sizes of local datasets across users, for more realistic use-cases. Besides, our paper opens other perspectives. `DP-SCAFFOLD` may be improved by incorporating other ML techniques such as momen-

tum. On the experimental side, a larger number of samples and a more precise tuning of the trade-off between T and K may dramatically improve the utility for real-world data cases under a given privacy budget. From a theoretical perspective, investigating an adaptation of our approach to a personalized FL setting (Fallah et al., 2020; Sattler et al., 2020; Marfoq et al., 2021), where formal privacy guarantees have seldom been studied (at the exception of Bellet et al., 2018; Hu et al., 2020), is a direction of interest.

7 Acknowledgments

We thank Baptiste Goujaud and Constantin Philippenko for interesting discussions. The work of A. Dieuleveut is partially supported by ANR-19-CHIA-0002-01 /chaire SCAI, and Hi! Paris. The work of Aurélien Bellet is supported by grants ANR-16-CE23-0016 (Project PAMELA) and ANR-20-CE23-0015 (Project PRIDE).

References

Martín 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. doi: 10.1145/2976749.2978318. arXiv: 1607.00133.

Durmus Alp Emre Acar, Yue Zhao, Ramon Matas, Matthew Mattina, Paul Whatmough, and Venkatesh Saligrama. Federated learning based on dynamic regularization. In *International Conference on Learning Representations*, 2020.

Galen Andrew, Om Thakkar, H. Brendan McMahan, and Swaroop Ramaswamy. Differentially Private Learning with Adaptive Clipping. 2021. arXiv: 1905.03871.

Borja Balle, James Bell, Adria Gascon, and Kobbi Nissim. The Privacy Blanket of the Shuffle Model. In *Annual International Cryptology Conference*, pages 638–667. Springer, 2019. arXiv: 1903.02837.

A Bellet, R Guerraoui, M Taziki, and M Tommasi. Personalized and Private Peer-to-Peer Machine Learning. *PMLR*, 84, 2018.

Albert Cheu, Adam Smith, Jonathan Ullman, David Zeber, and Maxim Zhilyaev. Distributed Differential Privacy via Shuffling. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 375–403. Springer, 2019. arXiv: 1808.01394.

Gregory Cohen, Saeed Afshar, Jonathan Tapson, and André van Schaik. EMNIST: an extension of MNIST to handwritten letters. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 2921–2926, 2017. doi: 10.1109/IJCNN.2017.7966217. arXiv: 1702.05373.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

John C. Duchi, Michael I. Jordan, and Martin J. Wainwright. Local Privacy, Data Processing Inequalities, and Statistical Minimax Rates. 2014. arXiv: 1302.3203.

John C. Duchi, Michael I. Jordan, and Martin J. Wainwright. Minimax optimal procedures for locally private estimation. *Journal of the American Statistical Association*, 113(521):182–201, 2018. doi: 10.1080/01621459.2017.1389735.

Cynthia Dwork and Aaron Roth. The Algorithmic Foundations of Differential Privacy. *Foundations and Trends® in Theoretical Computer Science*, 9 (3-4):211–407, 2013. ISSN 1551-305X, 1551-3068. doi: 10.1561/0400000042.

Cynthia Dwork, Guy N. Rothblum, and Salil Vadhan. Boosting and Differential Privacy. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 51–60, Las Vegas, NV, USA, 2010. IEEE. ISBN 978-1-4244-8525-3. doi: 10.1109/FOCS.2010.12.

Ulfar Erlingsson, Vitaly Feldman, Ilya Mironov, Ananth Raghunathan, Kunal Talwar, and Abhradeep Thakurta. Amplification by Shuffling: From Local to Central Differential Privacy via Anonymity. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2019. ISBN 978-1-61197-548-2. doi: 10.1137/1.9781611975482.

Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning: A meta-learning approach. In *34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, 2020.

Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, pages 1322–1333, 2015.

Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller. Inverting gradients - how easy is it to break privacy in federated learning? In *NeurIPS*, 2020.

Robin C. Geyer, Tassilo Klein, and Moin Nabi. Differentially Private Federated Learning: A Client Level Perspective. 2018. arXiv: 1712.07557.

Badih Ghazi, Rasmus Pagh, and Ameya Velingker. Scalable and Differentially Private Distributed Aggregation in the Shuffled Model. 2019. arXiv: 1906.08320.

Antonious M. Girgis, Deepesh Data, Suhas Diggavi, Peter Kairouz, and Ananda Theertha Suresh. Shuffled Model of Federated Learning: Privacy, Communication and Accuracy Trade-offs. 2020. arXiv: 2008.07180.

Robert Mansel Gower, Nicolas Loizou, Xun Qian, Alibek Sailanbayev, Egor Shulgin, and Peter Richtárik. SGD: General Analysis and Improved Rates. In *International Conference on Machine Learning*, pages 5200–5209. PMLR, May 2019. ISSN: 2640-3498.

Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. Measuring the Effects of Non-Identical Data Distribution for Federated Visual Classification. 2019. arXiv: 1909.06335.

Rui Hu, Yuanxiong Guo, Hongning Li, Qingqi Pei, and Yanmin Gong. Personalized federated learning with differential privacy. *IEEE Internet of Things Journal*, 7(10):9530–9539, 2020. doi: 10.1109/JIOT.2020.2991416.

Peter Kairouz, Sewoong Oh, and Pramod Viswanath. The Composition Theorem for Differential Privacy. *IEEE Transactions on Information Theory*, 63(6): 4037–4049, 2017. ISSN 0018-9448, 1557-9654. doi: 10.1109/TIT.2017.2685505.

Peter Kairouz, H. Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, Rafael G. L. D'Oliveira, Hubert Eichner, Salim El Rouayheb, David Evans, Josh Gardner, Zachary Garrett, Adrià Gascón, Badih Ghazi, Phillip B. Gibbons, Marco Gruteser, Zaid Harchaoui, Chaoyang He, Lie He, Zhouyuan Huo, Ben Hutchinson, Justin Hsu, Martin Jaggi, Tara Javidi, Gauri Joshi, Mikhail Khodak, Jakub Konečný, Aleksandra Korolova, Farinaz Koushanfar, Sanmi Koyejo, Tancrède Lepoint, Yang Liu, Prateek Mittal, Mehryar Mohri, Richard Nock, Ayfer Özgür, Rasmus Pagh, Mariana Raykova, Hang Qi, Daniel Ramage, Ramesh Raskar, Dawn Song, Weikang Song, Sebastian U. Stich, Ziteng Sun, Ananda Theertha Suresh, Florian Tramèr, Praneeth Vepakomma, Jianyu Wang, Li Xiong, Zheng Xu, Qiang Yang, Felix X. Yu, Han Yu, and Sen Zhao. Advances and Open Problems in Federated Learning. 2021. arXiv: 1912.04977.

Sai Praneeth Karimireddy, Martin Jaggi, Satyen Kale, Mehryar Mohri, Sashank J Reddi, Sebastian U Stich, and Ananda Theertha Suresh. Mime: Mimicking centralized stochastic algorithms in federated learning. *arXiv preprint arXiv:2008.03606*, 2020a.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank J. Reddi, Sebastian U. Stich, and Ananda Theertha Suresh. SCAFFOLD: Stochastic Controlled Averaging for Federated Learning. *PMLR*, 119, 2020b. arXiv: 1910.06378.

Shiva Prasad Kasiviswanathan, Homin K. Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. What Can We Learn Privately? *SIAM Journal on Computing*, 40(3):793–826, 2011. ISSN 0097-5397, 1095-7111. doi: 10.1137/090756090.

Ahmed Khaled, Konstantin Mishchenko, and Peter Richtárik. Tighter Theory for Local SGD on Identical and Heterogeneous Data. *PMLR*, 108, 2020. arXiv: 1909.04746.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated Optimization in Heterogeneous Networks. In *Proceedings of the 3rd MLSys Conference*, Austin, TX, USA, 2020a. arXiv: 1812.06127.

Yiwei Li, Tsung-Hui Chang, and Chong-Yung Chi. Secure federated averaging algorithm with differential privacy. In *2020 IEEE 30th International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6, 2020b. doi: 10.1109/MLSP49062.2020.9231531.

Othmane Marfoq, Giovanni Neglia, Aurélien Bellet, Laetitia Kameni, and Richard Vidal. Federated Multi-Task Learning under a Mixture of Distributions. In *NeurIPS*, 2021.

H Brendan McMahan, Eider Moore, Daniel Ramage, and Seth Hampson. Communication-Efficient Learning of Deep Networks from Decentralized Data. *JMLR*, 54, 2017a.

H. Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. Learning Differentially Private Recurrent Language Models. 2017b. arXiv: 1710.06963.

Ilya Mironov. Renyi Differential Privacy. *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pages 263–275, 2017. doi: 10.1109/CSF.2017.11. arXiv: 1702.07476.

Yurii Nesterov et al. *Lectures on convex optimization*, volume 137. Springer, 2004.

Sashank J Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečnỳ, Sanjiv Kumar, and Hugh Brendan McMahan. Adaptive

federated optimization. In *International Conference on Learning Representations*, 2020.

Felix Sattler, Klaus-Robert Müller, and Wojciech Samek. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 3–18. IEEE, 2017.

Aleksei Triastcyn and Boi Faltings. Federated learning with bayesian differential privacy. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 2587–2596. IEEE, 2019.

Tim van Erven and Peter Harremoës. Renyi Divergence and Kullback-Leibler Divergence. *IEEE Transactions on Information Theory*, 60(7):3797–3820, 2014. ISSN 0018-9448, 1557-9654. doi: 10.1109/TIT.2014.2320500. arXiv: 1206.2459.

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

Yu-Xiang Wang, Borja Balle, and Shiva Kasiviswanathan. Subsampled Rényi Differential Privacy and Analytical Moments Accountant. *Journal of Privacy and Confidentiality*, 10(2), 2020. ISSN 2575-8527. doi: 10.29012/jpc.723.

Kang Wei, Jun Li, Ming Ding, Chuan Ma, Howard H. Yang, Farhad Farokhi, Shi Jin, Tony Q. S. Quek, and H. Vincent Poor. Federated learning with differential privacy: Algorithms and performance analysis. *IEEE Transactions on Information Forensics and Security*, 15:3454–3469, 2020. doi: 10.1109/TIFS.2020.2988575.

Yang Zhao, Jun Zhao, Mengmeng Yang, Teng Wang, Ning Wang, Lingjuan Lyu, Dusit Niyato, and Kwok-Yan Lam. Local Differential Privacy based Federated Learning for Internet of Things. *IEEE Internet of Things Journal*, 8(11):8836–8853, 2021. doi: 10.1109/JIOT.2020.3037194. arXiv: 2004.08856.

Differentially Private Federated Learning on Heterogeneous Data
Supplementary Material

Organization of the Appendix

This appendix is organized as follows. Appendix A summarizes the main notations and provides the detailed `DP-FedAvg` algorithm for completeness. Appendix B provides details on our privacy analysis. Appendix C gives the full proofs of our utility results for the convex, strongly-convex and nonconvex cases. Finally, Appendix D provides more details on the experiments of Section 5, as well as additional results.

A Additional Information

A.1 Table of Notations

Table 1 summarizes the main notations used throughout the paper.

Table 1: Summary of the main notations.

Symbol	Description
$[n]$	set $\{1, 2, ..., n\}$ for any $n \in \mathbb{N}$
M, $i \in [M]$	number and index of users
T, $t \in [T]$	number and index of communication rounds
K, $k \in [K]$	number and index of local updates (for each user)
D_i	local dataset held by the i-th user, composed of points $d_1^i, \ldots, d_R^i$
R	size of any local dataset D_i
D	joint dataset ($\bigsqcup_{i=1}^M D_i$)
$f_i(x, d)$	loss of the i-th user for model x on data record d
F_i	local empirical risk function of the i-th user ($\frac{1}{R}\sum_{j=1}^R f_i(\cdot, d_j^i)$)
F	global objective function ($\frac{1}{M}\sum_{i=1}^M F_i$)
$x^t \in \mathbb{R}^d$	server model after round t
$y_i^k \in \mathbb{R}^d$	model of i-th user after local update k
$c^t \in \mathbb{R}^d$	server control variate after round t
$c_i^t \in \mathbb{R}^d$	control variate of the i-th user after round t
$l \in (0, 1)$	user sampling ratio
$s \in (0, 1)$	data sampling ratio
ϵ, δ	differential privacy parameters
σ_g	standard deviation of Gaussian noise added for privacy
$\mathcal{C}$	gradient clipping threshold
ν	Lipschitz-smoothness constant
μ	strong convexity parameter
ς^2	variance of stochastic gradients

A.2 `DP-FedAvg` Algorithm

The code of `DP-FedAvg` is given in Algorithm 2.

Algorithm 2: DP-FedAvg$(T, K, l, s, \sigma_g, \mathcal{C})$

Server Input: initial x^0
Output: x^T
for $t = 1, ..., T$ **do**
  User subsampling by the server: $C^t \subset [M]$
  Server communicates x^{t-1} to users $i \in C^t$
  for *user* $i \in C^t$ **do**
    Initialize model: $y_i^0 \leftarrow x^t$
    for $k = 1, ..., K$ **do**
      Data subsampling by user: $S_i^k \subset D_i$
      for *sample* $j \in S_i^k$ **do**
        Compute gradient: $g_{ij} \leftarrow \nabla f_i(y_i^{k-1}, d_j^i)$
        Clip gradient: $\tilde{g}_{ij} \leftarrow g_{ij} / \max\left(1, ||g_{ij}||_2 / \mathcal{C}\right)$
      Add DP noise to local gradients: $\tilde{H}_i^k \leftarrow \frac{1}{sR} \sum_{j \in S_i^k} \tilde{g}_{ij} + \frac{2\mathcal{C}}{sR} \mathcal{N}(0, \sigma_g^2)$
      $y_i^k \leftarrow y_i^{k-1} - \eta_l \tilde{H}_i^k$
    $\Delta y_i^t \leftarrow y_i^K - x^{t-1}$
    User i **communicates to server:** Δy_i^t
  Server aggregates: $\Delta x^t \leftarrow \frac{1}{lM} \sum_{i \in C^t} \Delta y_i^t$
  $x^t \leftarrow x^{t-1} + \eta_g \Delta x^t$

B Details on Privacy Analysis

In this section, we provide the proof of our privacy results. We start by recalling standard differential privacy results on composition and amplification by subsampling in Section B.1. Section B.2 reviews recent results in Rényi Differential Privacy (RDP) which allow to obtain tighter privacy bounds. We then formally state and prove Claim 4.1 in Section B.3. Finally, we provide the proof of our main result (Theorem 4.1) in Section B.4.

B.1 Reminders on Differential Privacy

In the following, we denote by $D \in \mathcal{X}^n$ to a dataset of size n. Two datasets $D, D' \in \mathcal{X}^n$ are said to be neighboring (denoted by $||D - D'|| \leq 1$) if they differ in at most one element.

Composition. Let $M_1(\cdot; A_1), ..., M_T(\cdot; A_T)$ be a sequence of T *adaptive* DP mechanisms where A_t stands for the auxiliary input to the t-th mechanism, which may depend on the outputs of previous mechanisms $(M_{t'})_{t'<t}$. The ability to choose the sequences of mechanisms adaptively is crucial for the design of iterative machine learning algorithms. DP allows to keep track of the privacy guarantees when such a sequence of private mechanisms is run on the same dataset D. Simple composition (Dwork et al., 2010, Theorem III.1.) states that the privacy parameters grow linearly with T. Dwork et al. (2010) provide a *strong composition* result where the ϵ parameter grows sublinearly with T. This result is restated in Lemma B.1.

Lemma B.1 (Strong adaptive composition, Dwork et al., 2010)**.** *Let $M_1, ..., M_T$ be T adaptive (ϵ, δ)-DP mechanisms. Then, for any $\delta' > 0$, the mechanism $M = (M_1, ..., M_T)$ is $(\overline{\epsilon}, \overline{\delta})$-DP where:*

$$\overline{\epsilon} = \epsilon\sqrt{2T\log(1/\delta')} + T\epsilon(e^{\epsilon} - 1) \text{ and } \overline{\delta} = T\delta + \delta'.$$

Remark. When stating theoretical results, $\overline{\epsilon}$ is typically approximated by $\mathcal{O}(\epsilon\sqrt{T\log(1/\delta')})$ when $\epsilon < 1$.

Privacy amplification by subsampling. A key result in DP is that applying a private algorithm on a random subsample of the dataset amplifies privacy guarantees (Kasiviswanathan et al., 2011). In this work, we are interested in subsampling without replacement.

Definition B.1 (Subsampling without replacement)**.** *The subsampling procedure $Samp_{n,m} : \mathcal{X}^n \to \mathcal{X}^m$ (where $m \in \mathbb{N}$, with $m \leq n$) takes D as input and chooses uniformly among its elements a subset $\underline{D}$ of m elements. We may also denote $Samp_{n,m}$ as $Samp_q$ where $q = m/n$ in the rest of the paper.*

Lemma B.2 quantifies the associated privacy amplification effect.

Lemma B.2 (Amplification by subsampling, Kasiviswanathan et al., 2011)**.** *Let $M' : \mathcal{X}^m \to \mathcal{Y}$ be a (ϵ, δ)-DP mechanism w.r.t a given dataset $\underline{D} \in \mathcal{X}^m$. Then, mechanism $M : \mathcal{X}^n \to \mathcal{Y}$ defined as $M := M' \circ Samp_{n,m}$ is (ϵ', δ')-DP w.r.t. to any dataset $D \in \mathcal{X}^n$ such that $\underline{D} = Samp_{n,m}(D)$, where:*

$$\epsilon' = \log(1 + q(e^{\epsilon} - 1)),\ \delta' = q\delta,\ q = m/n.$$

Remark. In theoretical results, ϵ' is often approximated by $\mathcal{O}(q\epsilon)$ when $\epsilon \leq 1$.

B.2 Rényi Differential Privacy

Abadi et al. (2016) demonstrated in practice that the privacy bounds provided by standard (ϵ, δ)-DP theory (see Section B.1) often overestimate the actual privacy loss. In order to better express inequalities on the tails of the output distributions of private algorithms, we introduce the *privacy loss random variable* (Dwork and Roth, 2013; Abadi et al., 2016; Wang et al., 2020). Given a random mechanism M, let $M(D)$ and $M(D')$ be the distributions of the output when M is run on D and D' respectively. The privacy loss $L^M_{D,D'}$ is defined as:

$$L^M_{D,D'}(\theta) := \log\left(\frac{M(D)(\theta)}{M(D')(\theta)}\right) \quad \text{where } \theta \sim M(D). \tag{2}$$

The interpretation of this quantity is easy to understand: (ϵ, δ)-DP ensures that the absolute value of the privacy loss is bounded by ϵ with probability at least $(1 - \delta)$ for all pairs of neighboring datasets D and D' (Dwork and Roth, 2013, Lemma 3.17).

We will reason on the *Cumulant Generating Function* (CGF) of the privacy loss, denoted K_M, rather than on the privacy loss L^M itself. This CGF is expressed as follows for any $\lambda > 0$:

$$K_M(D, D', \lambda) = \mathbb{E}_{\theta\sim M(D)}\left[e^{\lambda L^M_{D,D'}(\theta)}\right] = \mathbb{E}_{\theta\sim M(D)}\left[\left(\frac{M(D)(\theta)}{M(D')(\theta)}\right)^{\lambda}\right],$$

which is also equivalent to:

$$K_M(D, D', \lambda) = \mathbb{E}_{\theta\sim M(D')}\left[\left(\frac{M(D)(\theta)}{M(D')(\theta)}\right)^{\lambda+1}\right]. \tag{3}$$

By the property of the moment generating function, $K_M(D, D', \cdot)$ fully determines the distribution of the privacy loss random variable $L^M_{D,D'}$. We also define $K_M(\lambda) := \sup_{||D-D'||\leq 1} K_M(D, D', \lambda)$, which is the upper bound on the CGF for any pair of neighboring datasets.

We can now introduce *Rényi Differential Privacy* (RDP), which generalizes DP using the Rényi divergence D_α.

Definition B.2 (Rényi Differential Privacy, Mironov, 2017)**.** *For any $\alpha \in (1, \infty)$ and any $\epsilon > 0$, a mechanism $M : \mathcal{X}^n \to \mathcal{Y}$ is said to be (α, ϵ)-RDP, if for all neighboring datasets D and D',*

$$D_\alpha(M(D)||M(D')) := \frac{1}{\alpha - 1}\log \mathbb{E}_{\theta\sim M(D')}\left[\left(\frac{M(D)(\theta)}{M(D')(\theta)}\right)^{\alpha}\right] \leq \epsilon. \tag{4}$$

Given a mechanism M and a RDP parameter α, we can thus determine from Definition B.2 the lowest value of the ϵ-RDP bound, denoted $\epsilon_M(\alpha)$, such that M is $(\alpha, \epsilon_M(\alpha))$-RDP. Indeed, $\epsilon_M(\alpha)$ is such that:

$$\epsilon_M(\alpha) = \inf_{\epsilon\in\varepsilon(M)} \epsilon \quad \text{where} \quad \varepsilon(M) := \{\epsilon > 0 : \sup_{||D-D'||\leq 1} D_\alpha(M(D)||M(D')) \leq \epsilon\}.$$

The obvious similarity between Eq. (3) and Eq. (4) shows the link between the CGF and the notion of RDP. Indeed, for any $\alpha \in (1, \infty)$, it is easy to see that $(\alpha - 1)\epsilon_M(\alpha)$ is equal to $K_M(\lambda)$ where $\lambda + 1 = \alpha$ (restated in Lemma B.3).

Lemma B.3 (**Equivalence RDP-CGF**)**.** *Any mechanism M is $(\lambda + 1, K_M(\lambda)/\lambda)$-RDP for all $\lambda > 0$.*

We now recall how we can convert RDP guarantees into standard DP guarantees.

Lemma B.4 (RDP to DP conversion, Mironov, 2017)**.** *If M is (ϵ, α)-RDP, then M is $(\epsilon + \log(1/\delta)/(\alpha-1), \delta)$-DP for any $0 < \delta < 1$.*

Given Lemma B.4 and Lemma B.3, it is possible to find the smallest ϵ from some fixed parameter δ or the smallest δ from some fixed parameter ϵ so as to achieve (ϵ, δ)-DP:

$$\epsilon(\delta) = \min_{\lambda > 0} \frac{\log(1/\delta) + K_M(\lambda)}{\lambda}, \tag{5}$$

$$\delta(\epsilon) = \min_{\lambda > 0} e^{K_M(\lambda) - \lambda\epsilon}. \tag{6}$$

Moreover, $\lambda \to K_M(\lambda)/\lambda$ is monotonous (van Erven and Harremoës, 2014, Theorem 3) and $\lambda \to K_M(\lambda)$ is convex (van Erven and Harremoës, 2014, Theorem 11). This last property enables to bound K_M by a linear interpolation between the values of K_M evaluated at integers, as stated below:

$$\forall \lambda > 0, K_M(\lambda) \leq (1 - \lambda + \lfloor \lambda \rfloor) K_M(\lfloor \lambda \rfloor) + (\lambda - \lfloor \lambda \rfloor) K_M(\lceil \lambda \rceil). \tag{7}$$

Therefore, Problem (5) is quasi-convex and Problem (6) is log-convex, and both can be solved if we know the expression of $K_M(\lambda)$ for any $\lambda > 0$.

We provide below other useful results from RDP theory, which we will use in our privacy analysis.

Lemma B.5 (RDP Composition, Mironov, 2017)**.** *Let $\alpha \in (1, \infty)$. Let M_1 and M_2 be two mechanisms such that M_1 is (α, ϵ_1)-RDP and M_2, which takes the output of M_1 as auxiliary input, is (α, ϵ_2)-RDP. Then the composed mechanism $M_2 \circ M_1$ is $(\alpha, \epsilon_1 + \epsilon_2)$-RDP.*

Lemma B.6 (RDP Gaussian mechanism, Mironov, 2017)**.** *If $f : \mathcal{X}^n \to \mathbb{R}^d$ has ℓ_2-sensitivity 1, then the Gaussian mechanism $G_f(\cdot) := f(\cdot) + \mathcal{N}(0, \sigma_g^2 I_d)$ is $(\alpha, \alpha/2\sigma_g^2)$-RDP for any $\alpha > 1$.*

Lemma B.7 (RDP for subsampled Gaussian mechanism, Wang et al., 2020)**.** *Let $\alpha \in \mathbb{N}$ with $\alpha \geq 2$ and $0 < q < 1$ be a subsampling ratio. Suppose $f : \mathcal{X}^n \to \mathbb{R}^d$ has ℓ_2-sensitivity equal to 1. Let $G'_f(\cdot) := G_f \circ Samp_q(\cdot)$ be a subsampled Gaussian mechanism. Then G'_f is $(\alpha, \epsilon'(\alpha, \sigma_g^2))$-RDP where*

$$\epsilon'(\alpha, \sigma_g^2) \leq \frac{1}{\alpha - 1} \log \left(1 + 2q^2 \binom{\alpha}{2} \min\{2(e^{1/\sigma_g^2} - 1), e^{1/\sigma_g^2}\} + \sum_{j=3}^{\alpha} 2q^j \binom{\alpha}{j} e^{j(j-1)/2\sigma_g^2} \right).$$

Remark. By considering $q = o(1)$, the dominant term in the upper bound of $\epsilon'(\alpha, \sigma_g^2)$ comes from the term of the sum of the order of q^2. In particular, when σ_g^2 is large (i.e. high privacy regime), the term $\min\{2(e^{1/\sigma_g^2} - 1), e^{1/\sigma_g^2}\}$ simplifies to $2(e^{1/\sigma_g^2} - 1) \leq 4/\sigma_g^2$. This thus simplifies the whole upper bound to $\mathcal{O}(\alpha q^2/\sigma_g^2)$.

B.3 Proof of Claim 4.1

We restate below a more formal version of Claim 4.1 along with its proof. For any $t \in [T]$, we define *subversions* of algorithms `DP-SCAFFOLD` (Alg. 1) and `DP-FedAvg` (Alg. 2), which stop at round t and reveal an output, either to the server or to a third party:

- **To the server.** We assume that the sampling of users C^t is known by the server. Formally, we define $\mathcal{A}^t_{\texttt{DP-SCAFFOLD}}$, which outputs (reveals) $\{y_i^t, c_i^t\}_{i \in C^t}$, and $\mathcal{A}^t_{\texttt{DP-FedAvg}}$, which outputs $\{y_i^t\}_{i \in C^t}$ (those quantities being private w.r.t. $\{D_i\}_{i \in C^t}$).
- **To a third party.** We define $\tilde{\mathcal{A}}^t_{\texttt{DP-SCAFFOLD}}$, which outputs (x^t, c^t) and $\tilde{\mathcal{A}}^t_{\texttt{DP-FedAvg}}$, which outputs x^t (those quantities being private w.r.t. D).

In both privacy models, `DP-SCAFFOLD` and `DP-FedAvg` can be seen as T adaptive compositions of these *sub*-algorithms.

Claim B.1 (Formal version of Claim 4.1)**.** *For any $t \in [T]$, the following holds:*

- $\mathcal{A}^t_{\texttt{DP-SCAFFOLD}}$ *and* $\mathcal{A}^t_{\texttt{DP-FedAvg}}$ *have the same level of privacy (towards the server),*
- $\tilde{\mathcal{A}}^t_{\texttt{DP-SCAFFOLD}}$ *and* $\tilde{\mathcal{A}}^t_{\texttt{DP-FedAvg}}$ *have the same level of privacy (towards a third party).*

Proof. We prove the claim by reasoning by induction on the number of communication rounds t. We only give the proof for the first statement (including the `DP-SCAFFOLD-warm` version). The second one can be proved in a similar manner.

First, consider $t = 1$. For any $i \in C^t$, control variates c_i^0 are either all set to 0 (`DP-SCAFFOLD`), or c_i^0 are at least as private as y_i^1 (`DP-SCAFFOLD-warm`). The level of privacy for $\mathcal{A}^1_{\texttt{DP-SCAFFOLD}}$ is thus *fully* determined by the level of privacy of $\{y_i^1\}_{i\in C^t}$, which is the same as $\mathcal{A}^1_{\texttt{DP-FedAvg}}$. Therefore the claim is true for $t = 1$.

Then, let $t \in [T]$ and suppose that the claim is verified for all $t' < t$. Let $i \in C^t$ and first consider $\mathcal{A}^t_{\texttt{DP-SCAFFOLD}}$. The update of the i-th user model (see Eq. 1) at round t shows that an additional information leakage may come from the correction $(c^{t-1} - c_i^{t-1})$, or more precisely from c_i^{t-1} since c^{t-1} is known by the server. By assumption of induction, c_i^{t-1} is also known by the server. Therefore, using the post-processing property of DP, the y_i^t as updated in `DP-SCAFFOLD` is as private w.r.t. D_i as the y_i^t as updated in `DP-FedAvg`. Besides this, the update of the i-th control variate *fully* depends on the local updates of y_i^t through the average of the DP-noised stochastic gradients calculated over the local iterations. Therefore, considering all the contributions from C^t, $\mathcal{A}^t_{\texttt{DP-FedAvg}}$ and $\mathcal{A}^t_{\texttt{DP-SCAFFOLD}}$ have the same level of privacy. □

B.4 Proof of Theorem 4.1

Preliminaries. Lemma B.7 only gives an upper bound of the RDP privacy for a subsampled Gaussian mechanism when $\alpha \in \mathbb{N}$ with $\alpha \geq 2$. However we will need to optimize our privacy bound w.r.t. $\alpha \in \mathbb{R}$ with $\alpha > 1$. We thus use Lemma B.3 and the convexity of the CGF (see Eq. 7) to generalize this upper bound to the following result.

Let $\alpha \in \mathbb{R}$ with $\alpha > 1$. Under the same assumptions as in Lemma B.7, G'_f is $(\alpha, \epsilon''(\alpha, \sigma_g^2))$-RDP with

$$\epsilon''(\alpha, \sigma_g^2) \leq (1 - \alpha + \lfloor\alpha\rfloor)\frac{\lfloor\alpha\rfloor - 1}{\alpha - 1}\epsilon'(\lfloor\alpha\rfloor - 1, \sigma_g^2) + (\alpha - \lfloor\alpha\rfloor)\frac{\lceil\alpha\rceil - 1}{\alpha - 1}\epsilon'(\lceil\alpha\rceil - 1, \sigma_g^2), \tag{8}$$

where $\epsilon'(\cdot, \sigma_g^2)$ admits the upper bound given in Lemma B.7.

Details of the proof. Our privacy analysis assumes that the query function has sensitivity 1, since the calibration of the Gaussian noise is locally adjusted in our algorithms with the constant $S = 2\mathcal{C}/sR$ (see Section 3.2). We simply denote by G the Gaussian mechanism with variance σ_g^2, which is $(\alpha, \alpha/2\sigma_g^2)$-RDP (Lemma B.6). Below, we first prove privacy guarantees towards a third-party observing only the final result, and then deduce the guarantees towards the honest-but-curious server.

Step 1: data subsampling. Let $t \in [T]$ be an arbitrary round. We first provide an upper bound ϵ_a for the privacy loss *after the aggregation* by the server of the lM individual contributions (line 20 in Alg. 1) thanks to the local addition of noise.

Let $i \in C^t$, $\alpha > 1$. We denote by $\epsilon_i(\alpha)$ the α-RDP budget (w.r.t. D_i) used to "hide" the individual contribution of the i-th user from the server. This contribution is the result of the composition of K adaptative s-subsampled mechanisms G:

- We first obtain an upper RDP bound for the s-subsampled mechanism with Lemma B.7. Suppose first $\alpha \in \mathbb{N}$ and $\alpha \geq 2$, which is the case covered by Lemma B.7. Under Assumption 1-(i) and Assumption 1-(iii), the resulting mechanism is $(\alpha, \mathcal{O}(s^2\alpha/\sigma_g^2))$-RDP. To extend this result to $\alpha > 1$, we use the result provided in (8): by factoring by s^2/σ_g^2 in the upper bound of $\epsilon''(\alpha, \sigma_g^2)$, and bounding the rest of the inequality (a convex combination between $(\lfloor\alpha\rfloor - 1)^2/(\alpha - 1)$ and $(\lceil\alpha\rceil - 1)^2/(\alpha - 1)$) by α, we also obtain that this mechanism is $(\alpha, \mathcal{O}(s^2\alpha/\sigma_g^2))$-RDP.
- We then use the result of Lemma B.5 for the RDP composition rule over the K local iterations, which gives that $\epsilon_i(\alpha) \leq \mathcal{O}(Ks^2\alpha/\sigma_g^2)$.

We now consider the aggregation step. Taking into account all the contributions of the users from C^t, we get a Gaussian noise of variance $S^2\sigma_a^2$ where $\sigma_a^2 = \frac{1}{lM}\sigma_g^2$. Note that the sensitivity of the aggregation (w.r.t. the joint dataset D) is lM times smaller than when considering an individual contribution. Therefore, with the previous approximation, the aggregated contributions satisfy $(\alpha, \mathcal{O}(Ks^2\alpha/lM\sigma_g^2))$-RDP w.r.t. D.

After converting this result into a DP bound (Lemma B.4), we get that for any $0 < \delta' < 1$, the aggregation at line 20 in Alg. 1 is $(\epsilon_a(\alpha, \delta'), \delta')$-DP w.r.t. D where $\epsilon_a(\alpha, \delta') = \mathcal{O}\big(\frac{Ks^2\alpha}{lM\sigma_g^2} + \frac{\log(1/\delta')}{\alpha-1}\big)$.

Without approximation: we would obtain at this step an exact upper bound $\epsilon_a(\alpha, \delta') = K\epsilon''(\alpha, lM\sigma_g^2) + \frac{\log(1/\delta')}{\alpha-1}$.

Step 2: user subsampling. In order to get explicit bounds (which are slightly suboptimal), we then use classical DP tools to estimate an upper DP bound after T rounds taking into account the amplification by subsampling from the set of users.

- Using Lemma B.2, the subsampling of users enables a gain of privacy of the order of l, which gives $\big(\mathcal{O}(l\epsilon_a(\alpha, \delta')), l\delta'\big)$-DP.
- Using Lemma B.1, we compose this mechanism over T iterations, which under Assumption 1-(ii) gives for any $\delta'' > 0$, $\big(\mathcal{O}(\sqrt{T\log(1/\delta'')}l\epsilon_a(\alpha, \delta')), Tl\delta' + \delta''\big)$-DP.

Without approximation: the mechanism is $\big(\epsilon^*(\alpha, \delta')\sqrt{2T\log(1/\delta'')} + T\epsilon^*(\alpha, \delta')(e^{\epsilon^*(\alpha,\delta')} - 1), Tl\delta' + \delta''\big)$ where $\epsilon^*(\alpha, \delta') = \log(1 + l(e^{\epsilon_a(\alpha,\delta')} - 1))$.

Step 3: setting parameters. We denote $\epsilon_T(\alpha, \delta', \delta'') = l\sqrt{T\log(1/\delta'')}\big(\frac{Ks^2\alpha}{lM\sigma_g^2} + \frac{\log(1/\delta')}{\alpha-1}\big)$. Given what is stated above, the final output of the algorithm is $(\mathcal{O}(\epsilon_T), Tl\delta' + \delta'')$-DP.

Considering our final privacy budget δ, we *arbitrarily* fix $\delta' := \delta/2Tl$ and $\delta'' := \delta/2$. We now aim to find an expression of σ_g such that the privacy bound is minimized. *By considering the approximated bound*, this gives the following minimization problem:

$$\min_{\alpha>1} \epsilon_T(\alpha) := l\sqrt{T\log(2/\delta)}\left(\frac{Ks^2\alpha}{lM\sigma_g^2} + \frac{\log(2Tl/\delta)}{\alpha-1}\right).$$

Using DP rather than RDP has the advantage to solve this minimization problem pretty easily since only the second factor in $\epsilon_T(\alpha)$ depends on α, that is:

$$\min_{\alpha>1} \tilde{\epsilon_T}(\alpha) := \frac{Ks^2\alpha}{lM\sigma_g^2} + \frac{\log(2Tl/\delta)}{\alpha-1}.$$

By omitting constants, we obtain the expression for the minimum value of $\epsilon_T(\alpha)$:

$$\tilde{\epsilon} = l\sqrt{T\log(2/\delta)}\left(\frac{s\sqrt{K\log(2Tl/\delta)}}{\sigma_g\sqrt{lM}} + \frac{Ks^2}{lM\sigma_g^2}\right).$$

Under Assumption 1-(iii), we can bound the second term by the first one, which gives:

$$\tilde{\epsilon} = \mathcal{O}\left(\frac{s\sqrt{lTK\log(2/\delta)\log(2Tl/\delta)}}{\sigma_g\sqrt{M}}\right).$$

We then invert the formula of this upper bound of $\tilde{\epsilon}$ to express σ_g as a function of a given privacy budget ϵ:

$$\sigma_g = \Omega\big(s\sqrt{lTK\log(2Tl/\delta)\log(2/\delta)}/\epsilon\sqrt{M}\big),$$

which proves that the algorithm is $(\mathcal{O}(\epsilon), \delta)$-DP towards a third-party observing its final output.

Without approximation: the minimization problem is much more complex and has to be solved numerically

$$\min_{\alpha>1,\delta'>0,\delta''>0} \epsilon^*(\alpha, \delta')\sqrt{2T\log(1/\delta'')} + T\epsilon^*(\alpha, \delta')(e^{\epsilon^*(\alpha,\delta')} - 1) \quad \text{s.t. } \delta = Tl\delta' + \delta',$$

or:

$$\min_{\alpha>1,x\in(0,1)} \epsilon^*(\alpha, x\delta/Tl)\sqrt{2T\log(1/(1-x)\delta)} + T\epsilon^*(\alpha, x\delta/Tl)(e^{\epsilon^*(\alpha,x\delta/Tl)} - 1).$$

Extension to privacy towards the server. The crucial difference with the third-party case is that the server observes individual contributions and knows which users are subsampled at each step. Removing the privacy amplification effect of the l-subsampling of users and the aggregation step, the minimization problem becomes

$$\min_{\alpha>1} \epsilon_T(\alpha) := \sqrt{T\log(2/\delta)}\left(\frac{Ks^2\alpha}{\sigma_g^2} + \frac{\log(2T/\delta)}{\alpha-1}\right),$$

where the minimizing value can be approximated by:

$$\tilde{\epsilon} = \sqrt{T\log(2/\delta)}\left(\frac{s\sqrt{K\log(2T/\delta)}}{\sigma_g} + \frac{Ks^2}{\sigma_g^2}\right).$$

Under Assumption 1-(iii), we can bound the second term by the first one:

$$\tilde{\epsilon} = \mathcal{O}\left(\frac{s\sqrt{TK\log(2/\delta)\log(2T/\delta)}}{\sigma_g}\right),$$

which proves that we obtain $(\mathcal{O}(\epsilon_s), \delta_s)$-DP towards the server where $\epsilon_s = \epsilon\sqrt{\frac{M}{l}}$ and $\delta_s = \frac{\delta}{2}(\frac{1}{l}+1)$.

Remark. For privacy towards a third-party, it is actually possible to combine the subsampling ratios (user and data) to determine a bound upon the subsampling of data *directly* from D and thus to quantify a more precise gain in privacy (Girgis et al., 2020). The difficulty in this setup is that this combined subsampling is *not uniform overall*, which requires extending the proof of Lemma B.2 as done by Girgis et al. (2020).

C Proof of Utility

In this section, we provide the proof of our utility results. We first establish in Section C.1 some preliminary results about the impact of DP noise over stochastic gradients. In Section C.2, we provide the complete version of our utility result for `DP-SCAFFOLD-warm` (Theorem C.1), from which Theorem 4.2 is an immediate corollary. We prove this theorem for convex local loss functions in Section C.3 and non-convex loss functions in Section C.4. We finally state in Section C.5 our complete result for `DP-FedAvg` (Theorem C.2).

For any $\mathcal{C}, \sigma_g > 0$, we define $\Sigma_g(\mathcal{C}) := 2\mathcal{C}\sqrt{2d}\sigma_g/sR$. We recall that we assume that F is bounded from below by $F^* = F(x^*)$, for an $x^* \in \mathbb{R}^d$.

C.1 Preliminaries

Properties of DP-noised stochastic gradients. Let $i \in [M]$, $x \in \mathbb{R}^d$, $S_i \subset D_i$ and $\mathcal{C}, \sigma_g > 0$. Suppose Assumptions 2 and 3.3 are verified (the last assumption ensures that the clipping on per-example local gradients with threshold $\mathcal{C}$ is not effective).

We recall below the expression of $\tilde{H}_i(x)$ from Section 3.3, which is the noised version of the local gradient $H_i(x)$ of the i-th user over S_i evaluated at x (omitting index k):

$$\tilde{H}_i(x) := H_i(x) + \frac{2\mathcal{C}}{sR}\mathcal{N}(0, \sigma_g^2), \quad \text{where} \quad H_i(x) := \frac{1}{sR}\sum_{d_j^i \in S_i} \nabla f_i(x, d_j^i).$$

We recall that the ℓ_2-sensitivity of $H_i(x)$ w.r.t. S_i is upper bounded by $2C/sR$, which explains the scaling of the Gaussian noise in the expression of $\tilde{H}_i(x)$. Since the variance of $\mathcal{N}(0, \mathrm{I}_d)$ is $2d$, the following statement holds directly:

$$\mathbb{E}\big[\tilde{H}_i(x)\big] = H_i(x) \text{ and } \mathbb{E}\big[||\tilde{H}_i(x) - H_i(x)||^2\big] \leq \frac{8\mathcal{C}^2 d\sigma_g^2}{s^2R^2} = \Sigma_g(\mathcal{C})^2.$$

By combining our utility assumptions with the result stated above, we can deduce the following lemma.

Lemma C.1 (Regularity of DP-noised stochastic gradients)**.** *Under Assumptions 2 and 3, for any iteration* $t \in [T], k \in [K]$,

1. $\mathbb{E}\big[\tilde{H}_i^k(y_i^{k-1})|y_i^{k-1}\big] = \nabla F_i(y_i^{k-1})$,
2. $\mathbb{E}\big[||\tilde{H}_i^k(y_i^{k-1}) - \nabla F_i(y_i^{k-1})||^2|y_i^{k-1}\big] \leq \dfrac{\varsigma^2}{sR} + \Sigma_g^2(\mathcal{C})$.

The proof of Lemma C.1 is easily obtained by conditioning on the two sources of randomness (i.e., mini-batch sampling and Gaussian noise) which are *independent*, thus the variance is additive. This result can be seen as a *degraded* version of Assumption 3 due to the local injection DP noise, a fact that we will strongly leverage to derive convergence rates.

We now enumerate several statements that will be used in the utility proof. First, Lemma C.2 enables to control $||\nabla F||^2$ using the assumption of smoothness over the local loss functions. Second, Lemma C.3 provides separation inequalities of mean and variance (Karimireddy et al., 2020b, Lemma 4), which enables to state a result on quantities of interest in Corollary C.1.

Lemma C.2 (Nesterov inequality)**.** *Suppose Assumption 2 is verified and assume that for all* $i \in [M]$, F_i *is convex. Then,*

$$\forall x \in \mathbb{R}^d, ||\nabla F(x)||^2 \leq 2\nu(F(x) - F^*).$$

Proof. Let $x \in \mathbb{R}^d$.

$$\begin{aligned}
||\nabla F(x)||^2 = ||\nabla F(x) - \nabla F(x^*)||^2 &= ||\frac{1}{M}\sum_{i=1}^{M} \nabla F_i(x) - \nabla F_i(x^*)||^2 \\
&\leq \frac{1}{M}\sum_{i=1}^{M} ||\nabla F_i(x) - \nabla F_i(x^*)||^2 && \text{(Jensen inequality)} \\
&\leq 2\nu(F(x) - F^*) && \text{(Nesterov et al., 2004, Theorem 2.1.5)}
\end{aligned}$$

□

Lemma C.3 (Separating mean and variance)**.** *Let* $(A_1, ..., A_n)$ *be* n *random variables in* $\mathbb{R}^d$ not necessarily independent.

1. Suppose that their mean is $\mathbb{E}[A_i] = a_i$ *and their variance is uniformly bounded, i.e. for all* $i \in [n]$, $\mathbb{E}[||A_i - a_i||^2] \leq \sigma_A^2$. *Then,*

$$\mathbb{E}\left\|\sum_{i=1}^{n} A_i\right\|^2 \leq \left\|\sum_{i=1}^{n} a_i\right\|^2 + n^2\sigma_A^2.$$

2. Suppose that their conditional mean is $\mathbb{E}[A_i|A_{i-1}, ...A_1] = a_i$ *and their variance is uniformly bounded, i.e. for all* $i \in [n]$, $\mathbb{E}[||A_i - a_i||^2] \leq \sigma_A^2$. *Then,*

$$\mathbb{E}\left\|\sum_{i=1}^{n} A_i\right\|^2 \leq 2\left\|\sum_{i=1}^{n} a_i\right\|^2 + 2n\sigma_A^2.$$

Corollary C.1. *Let* $t \in [T]$. *In the following statements, the expectation is taken w.r.t. the randomness from their local data sampling and from the Gaussian DP noise, conditionally to the users' sampling* C^t *and initial value of variables* y_i, *that is* $y^0 = x^{t-1}$ *(same for all users). We have:*

- $\mathbb{E}\left[\left\|\dfrac{1}{KlM}\sum_{i\in C^t}\sum_{k\in[K]}(\tilde{H}_i^k(y_i^{k-1}) - \nabla F_i(y_i^{k-1}))\right\|^2 \middle| C^t, y^0\right] \leq \dfrac{\Sigma_g^2(\mathcal{C}) + \varsigma^2/sR}{KlM}$,

- $\mathbb{E}\Big[\Big\|\frac{1}{lM}\sum_{i\in C^t} c_i^t - \mathbb{E}[c_i^t]\Big\|^2 | C^t, y^0\Big] \leq \frac{\Sigma_g^2(\mathcal{C}) + \varsigma^2/sR}{KlM}$,
- $\mathbb{E}\Big[\Big\|c^t - \mathbb{E}[c^t]\Big\|^2 | y^0\Big] \leq \frac{\Sigma_g^2(\mathcal{C}) + \varsigma^2/sR}{KlM}$.

Proof. **First inequality.** We define a random variable A such as $A := \frac{1}{KlM}\sum_{i\in C^t}\sum_{k\in[K]} A_{i,k}$, with $A_{i,k} := \tilde{H}_i^k(y_i^{k-1})$.

From Lemma C.1, we have that for all $i \in C^t, k \in [K]$: $\mathbb{E}[A_{i,k}|y_i^{k-1}] = \mathbb{E}[\tilde{H}_i^k(y_i^{k-1})|y_i^{k-1}] = \nabla F_i(y_i^{k-1})$. Furthermore, by Lemma C.1, $\mathbb{E}\left[||\tilde{H}_i^k(y_i^{k-1}) - \nabla F_i(y_i^{k-1})||^2 | y_i^{k-1}\right] \leq \Sigma_g^2(\mathcal{C}) + \varsigma^2/sR$.

Furthermore, (a) for $i, j \in C^t$, $\sum_{k\in[K]} A_{i,k} - \nabla F_i(y_i^{k-1})$ and $\sum_{k\in[K]} A_{j,k} - \nabla F_j(y_j^{k-1})$ are independent conditionally to y^0; (b) for any $i \in C^t$, $(A_{i,k} - \nabla F_i(y_i^{k-1}))_{k\in[K]}$ is a martingale increment, i.e., $\mathbb{E}[A_{i,k} - \nabla F_i(y_i^{k-1})|\sigma(\{y_i^{k'}\}_{k'\in[k-1]})] = 0$. Then

Consequently:

$$\begin{aligned}
\mathbb{E}\left[\Big\|\frac{1}{KlM}\sum_{i\in C^t}\sum_{k\in[K]} A_{i,k} - \nabla F_i(y_i^{k-1})\Big\|^2 \Big| C^t, y^0\right] &\overset{(a)}{=} \frac{1}{(KlM)^2}\sum_{i\in C^t}\mathbb{E}\left[\Big\|\sum_{k\in[K]} A_{i,k} - \nabla F_i(y_i^{k-1})\Big\|^2 \Big| y^0\right] \\
&\overset{(b)}{=} \frac{1}{(KlM)^2}\sum_{i\in C^t}\sum_{k\in[K]}\mathbb{E}\left[\underbrace{\mathbb{E}\left[\Big\|A_{i,k} - \nabla F_i(y_i^{k-1})\Big\|^2 \Big| \sigma\big(\{y_i^{k'}\}_{k'\in[k]}\big)\right]}_{\leq \Sigma_g^2(\mathcal{C}) + \varsigma^2/sR} \Big| y^0\right] \\
&\leq \frac{\Sigma_g^2(\mathcal{C}) + \varsigma^2/sR}{KlM}.
\end{aligned}$$

To prove the second equality, we need to "iteratively" expand the squared norm and take the conditional expectation w.r.t. $\sigma\big(\{y_i^{k'}\}_{k'\in[k]}\big)$ for $k = K, K-1, \ldots, 1$ and use the martingale property to obtain that the scalar products are equal to 0.

Second inequality. We recall that for any $i \in C^t$, $c_i^t = \frac{1}{K}\sum_{k=1}^K \tilde{H}_i^k(y_i^{k-1})$. Thus $\frac{1}{lM}\sum_{i\in C^t} c_i^t = A$ and we can directly use the results from the first inequality.

Third inequality. We recall that $c^t = \frac{1}{M}\sum_{i=1}^M c_i^t$ (even if local control variates are not updated). Therefore, we can use the previous results and take the expectation over C^t, which gives:

$$\mathbb{E}\Big[\Big\|c^t - \mathbb{E}[c^t]\Big\|^2 | y^0\Big] \leq \frac{\varsigma^2/sR + \Sigma_g^2(\mathcal{C})}{KM} \leq \frac{\varsigma^2/sR + \Sigma_g^2(\mathcal{C})}{KlM}.$$

□

C.2 Theorem of Convergence for `DP-SCAFFOLD-warm`

Theorem C.1 (Utility rates for `DP-SCAFFOLD-warm`, σ_g chosen arbitrarily)**.** *Let $\sigma_g, \mathcal{C} > 0$, $x^0 \in \mathbb{R}^d$. Suppose we run `DP-SCAFFOLD-warm`$(T, K, l, s, \sigma_g, \mathcal{C})$ with initial local controls such that $c_i^0 = \frac{1}{K}\sum_{k=1}^K \tilde{H}_i^k(x^0)$ for any $i \in [M]$. Under Assumptions 2 and 3, we consider the sequence of iterates $(x^t)_{t\geq 0}$ of the algorithm, starting from x^0.*

1. *If F_i are μ-**strongly convex** ($\mu > 0$), $\eta_g = \sqrt{lM}$, $\eta_l = \min\big(\frac{l^{\frac{2}{3}}}{24\nu K\eta_g}, \frac{l}{54\mu K\eta_g}\big)$, and $T \geq \max(\frac{108}{l}, \frac{48\nu}{\mu l^{\frac{2}{3}}})$, then, there exist weights $\{w_t\}_{t\in[T]}$ such that the averaged output of `DP-SCAFFOLD-warm`$(T, K, l, s, \sigma_g, \mathcal{C})$, defined by $\overline{x}^T = \sum_{t\in[T]} w_t x^t$, has expected excess of loss such that:*

$$\mathbb{E}[F(\overline{x}^T)] - F(x^*) \leq \mathcal{O}\Big(\frac{\varsigma^2/sR + \Sigma_g^2(\mathcal{C})}{\mu TKlM} + \mu D_0^2 \exp(-\min\big(\frac{l}{108}, \frac{\mu l^{\frac{2}{3}}}{48\nu}\big)T)\Big),$$

2. *If F_i are* ***convex****, $\eta_g = \sqrt{lM}$, $\eta_l = \frac{l^{\frac{2}{3}}}{24\nu K\eta_g}$ and $T \geq 1$,then, there exist weights $\{w_t\}_{t\in[T]}$ such that the averaged output of* `DP-SCAFFOLD-warm`*$(T, K, l, s, \sigma_g, \mathcal{C})$, defined by $\overline{x}^T = \sum_{t\in[T]} w_t x^t$, has expected excess of loss such that:*

$$\mathbb{E}[F(\overline{x}^T)] - F(x^*) \leq \mathcal{O}\Big(\frac{\varsigma/\sqrt{sR} + \Sigma_g(\mathcal{C})}{\sqrt{TKlM}} D_0 + \frac{\nu}{l^{\frac{2}{3}}T} D_0^2\Big),$$

3. *If F_i are* ***non-convex****, $\eta_g = \sqrt{lM}$, $\eta_l = \frac{l^{\frac{2}{3}}}{24\nu K\eta_g}$ and $T \geq 1$, then there exist weights $\{w_t\}_{t\in[T]}$ such that the randomized output of* `DP-SCAFFOLD-warm`*$(T, K, l, s, \sigma_g, \mathcal{C})$, defined by $\{\overline{x}^T = x^t$ with probability w_t for all $t\}$, has expected squared gradient of the loss such that:*

$$\mathbb{E}||\nabla F(\overline{x}^T)||^2 \leq \mathcal{O}\Big(\frac{\varsigma/\sqrt{sR} + \Sigma_g(\mathcal{C})}{\sqrt{TKlM}} \sqrt{F_0} + \frac{\nu}{l^{\frac{2}{3}}T} F_0\Big),$$

where $D_0 = ||x^0 - x^*||$ and $F_0 := F(x^0) - F^*$.

We recover the result of Theorem 4.2 for `DP-SCAFFOLD-warm` where F_i are convex by setting $\sigma_g = \sigma_g^*$ where $\sigma_g^* := s\sqrt{lTK\log(2Tl/\delta)\log(2/\delta)}/\epsilon\sqrt{M}$, which gives $\Sigma_g(\mathcal{C}) = 2\mathcal{C}d\sqrt{2lTK\log(2Tl/\delta)\log(2/\delta)}/\epsilon R\sqrt{M}$ (with numerical constants omitted for the asymptotic bound).

C.3 Proof of Theorem C.1 (Convex case)

In this section, we give a detailed proof of convergence of `DP-SCAFFOLD-warm` with convex local loss functions. Our analysis is adapted from the proof given by Karimireddy et al. (2020b) without DP noise, but requires original modifications (see below). Throughout this part, we re-use the notations from Section 3.3.

Summary of the main steps. Let $t \in [T]$ be an arbitrary communication round of the algorithm. We detail below the updates that occur at this round.

- Let $i \in C^t$. Starting from $y_i^0 = x^{t-1}$, the random variable y_i is updated at local step $k \in [K]$ such that $y_i^k := y_i^{k-1} - \eta_l v_{i,k}^t$ where $v_{i,k}^t = \tilde{H}_i^k(y_i^{k-1}) - c_i^{t-1} + c^{t-1}$.
- Then we define the local control variate $\tilde{c}_i^t$ for this user by:

$$\tilde{c}_i^t := c^{t-1} - c_i^{t-1} + \frac{1}{K\eta_l}(x^{t-1} - y_i^K) = \frac{1}{K}\sum_{k=1}^{K} \tilde{H}_i^k(y_i^{k-1}).$$

- For any $i \in [M]$, we update the control variate c_i^t such that:
  - $c_i^t := \tilde{c}_i^t$ if $i \in C^t$,
  - $c_i^t := c_i^{t-1}$ otherwise.
- Finally, the global update is computed as:

$$x^t = x^{t-1} + \frac{\eta_g}{lM}\sum_{i\in C^t}(y_i^K - x^{t-1}) \text{ and } c^t = \frac{1}{M}\Big(\sum_{i\in C^t} c_i^t + \sum_{i\notin C^t} c_i^{t-1}\Big).$$

To keep track of the lag in the update of c_i^t, we introduce $\alpha_{i,k-1}^t$ defined for any $i \in [M]$, any $t \in [T]$ and any $k \in [K]$ by:

$$\alpha_{i,k-1}^t = \begin{cases} y_i^{k-1} & \text{if } i \in C^t \\ \alpha_{i,k-1}^{t-1} & \text{otherwise} \end{cases}$$

with $\alpha_{i,k-1}^0 = x^0$.

We hence have the following property for any $i \in [M]$ and any $t \in [T]$: $c_i^t = \frac{1}{K}\sum_{k=1}^{K} \tilde{H}_i^k(\alpha_{i,k-1}^t)$.

Additional definitions.

- Model gap: $\Delta x^t := x^t - x^{t-1}$,
- Global step-size: $\tilde{\eta} := K\eta_l\eta_g$ which gives $\Delta x^t = -\frac{\tilde{\eta}}{KlM}\sum_{k\in[K],i\in C^t}\tilde{H}_i^k(y_i^{k-1}) + c^{t-1} - c_i^{t-1}$,
- User-drift: $\mathcal{E}_t := \frac{1}{KM}\sum_{k=1}^{K}\sum_{i=1}^{M}\mathbb{E}||y_i^k - x^{t-1}||^2$,
- Control lag: $\mathcal{F}_t := \frac{1}{KM}\sum_{k=1}^{K}\sum_{i=1}^{M}\mathbb{E}||\alpha_{i,k-1}^t - x^t||^2$ with $\mathcal{F}_0 = 0$.

Originality of the proof. The proof substantially differs form the proof by Karimireddy et al. (2020b) in the convex case. Indeed, Karimireddy et al. (2020b) control a combination of the quadratic distance to the optimum and a control of the deviation between the controls and the gradients at the optimal point $\|c_i^t - \nabla F_i(x^*)\|$. Leveraging such a quantity in our proof would result in a worse upper bound on the utility than the one we get, as either the noise added to ensure DP (if c_i^0 is defined w.r.t. a noised gradient) or the heterogeneity (if $c_i^0 = 0$) would also appear in the initial condition $\|c_i^t - \nabla F_i(x^*)\|$. On the other hand, in our approach, we combine the quadratic distance to the optimum to a control of the lag and user-drift. In some sense this resembles some aspects of the proof in the non-convex regime in (Karimireddy et al., 2020b), in which the excess risk $(F(x^t) - F^*)$ is combined with the lag. Nevertheless, our result (in the convex case), strongly leverages the convexity of the function in the proof.

Details of the proof. The idea of the proof is to find a contraction inequality involving $||x^t - x^*||^2$, $\mathbb{E}[F(x^{t-1})] - F(x^*)$, $\mathcal{F}_t$ and $\varsigma/\sqrt{sR} + \Sigma_g(\mathcal{C})$. To do so, we will first bound the variance of the server's update. Then we will see how the control lag evolves through the communication rounds. We will also bound the user drift. To make the proof more readable, the index t may be omitted on random variables when the only communication round that is considered is the t-th one.

Lemma C.4 (Variance of the server's update)**.** $\forall\tilde{\eta} \in [0, 1/\nu]$

$$\mathbb{E}||x^t - x^{t-1}||^2 \leq 4\tilde{\eta}^2\nu^2\mathcal{E}_t + 8\nu^2\tilde{\eta}^2\mathcal{F}_{t-1} + 8\nu\tilde{\eta}^2\mathbb{E}\big(F(x^{t-1}) - F(x^*)\big) + \frac{9\tilde{\eta}^2}{KlM}(\varsigma^2/sR + \Sigma_g^2(\mathcal{C})).$$

Proof. We consider the model gap $\Delta x^t = x^t - x^{t-1}$.

$$\mathbb{E}||\Delta x^t||^2 = \tilde{\eta}^2\mathbb{E}\Big\|\underbrace{\left(\frac{1}{KlM}\sum_{k\in[K],i\in C^t}\tilde{H}_i(y_i^{k-1})\right)}_{A_1} + \underbrace{c^{t-1}}_{A_2} - \underbrace{\frac{1}{lM}\sum_{i\in C^t}c_i^{t-1}}_{A_3}\Big\|^2$$

We combine Lemma C.3-1 on A_1, A_2, A_3 with Corollary C.1 which controls their individual variance (conditionally to the users' sampling and the local parameters) by $\frac{\varsigma^2/sR + \Sigma_g^2(\mathcal{C})}{KlM}$. We first get rid of the terms related to the variance of the data sampling and the DP noise, before bounding the quantities of interest. It leads to:

$$\begin{aligned}\mathbb{E}||\Delta x^t||^2 &= \tilde{\eta}^2\mathbb{E}\left[\mathbb{E}\left[\left\|\left(\frac{1}{KlM}\sum_{k\in[K],i\in C^t}\tilde{H}_i(y_i^{k-1})\right) + c^{t-1} - \frac{1}{lM}\sum_{i\in C^t}c_i^{t-1}\right\|^2\Big|C^t, y^0\right]\right] \\ &\leq \tilde{\eta}^2\mathbb{E}\left[\left\|\frac{1}{KlM}\sum_{k\in[K],i\in C^t}\mathbb{E}[\tilde{H}_i(y_i^{k-1})|y^0] + \mathbb{E}[c^{t-1}|y^0] - \mathbb{E}[c_i^{t-1}|y^0]\right\|^2\right] + \frac{9\tilde{\eta}^2}{KlM}(\varsigma^2/sR + \Sigma_g^2(\mathcal{C})) \quad \text{(Lemma C.3-1)}\end{aligned}$$

For any $i \in C^t, k \in [K]$, we have $\mathbb{E}[\tilde{H}_i(y_i^{k-1})|y^0] = \mathbb{E}\left[\mathbb{E}[\tilde{H}_i(y_i^{k-1})|y_i^{k-1}]\big|y^0\right] = \mathbb{E}\left[\nabla F_i(y_i^{k-1})|y^0\right] = \nabla F_i(y_i^{k-1})$. Then,

$$
\begin{aligned}
\mathbb{E}||\Delta x^t||^2 &\leq \tilde{\eta}^2 \mathbb{E}\left[\frac{1}{KlM}\sum_{k\in[K],i\in C^t}\left\|\nabla F_i(y_i^{k-1}) + \mathbb{E}[c^{t-1}|y^0] - \mathbb{E}[c_i^{t-1}|y^0]\right\|^2\right] + \frac{9\tilde{\eta}^2}{KlM}(\varsigma^2/sR + \Sigma_g^2(\mathcal{C})) \qquad \text{(convexity of } ||.||^2) \\
&= \tilde{\eta}^2 \frac{1}{KM}\sum_{k\in[K],i\in[M]} \mathbb{E}\Big[\Big\|\underbrace{\nabla F_i(y_i^{k-1}) + \mathbb{E}[c^{t-1}|y^0] - \mathbb{E}[c_i^{t-1}|y^0]}_{\substack{\nabla F_i(y_i^{k-1}) - \nabla F_i(x^{t-1}) \\ +\mathbb{E}[c^{t-1}|y^0] - \nabla F(x^{t-1}) \\ -\mathbb{E}[c_i^{t-1}|y^0] + \nabla F_i(x^{t-1}) \\ +\nabla F(x^{t-1})}}\Big\|^2\Big] + \frac{9\tilde{\eta}^2}{KlM}(\varsigma^2/sR + \Sigma_g^2(\mathcal{C})) \qquad \text{(definition of } C^t) \\
&= \tilde{\eta}^2 \frac{1}{KM}\sum_{k\in[K],i\in[M]} \mathbb{E}\left[\left\|\mathbb{E}\left[\nabla F_i(y_i^{k-1}) - \nabla F_i(x^{t-1}) + c^{t-1} - \nabla F(x^{t-1}) - c_i^{t-1} + \nabla F_i(x^{t-1}) + \nabla F(x^{t-1})\Big|y^0\right]\right\|^2\right] \\
&+ \frac{9\tilde{\eta}^2}{KlM}(\varsigma^2/sR + \Sigma_g^2(\mathcal{C})) \qquad \text{(all variables are measurable wrt } y^0) \\
&\leq \tilde{\eta}^2 \frac{1}{KM}\sum_{k\in[K],i\in[M]} \mathbb{E}\left[\mathbb{E}\left[\left\|\nabla F_i(y_i^{k-1}) - \nabla F_i(x^{t-1}) + c^{t-1} - \nabla F(x^{t-1}) - c_i^{t-1} + \nabla F_i(x^{t-1}) + \nabla F(x^{t-1})\right\|^2\Big|y^0\right]\right] \\
&+ \frac{9\tilde{\eta}^2}{KlM}(\varsigma^2/sR + \Sigma_g^2(\mathcal{C})) \qquad \text{(Jensen inequality)} \\
&\leq \frac{4\tilde{\eta}^2}{KM}\sum_{k\in[K],i\in[M]} \mathbb{E}\left[\left\|\nabla F_i(y_i^{k-1}) - \nabla F_i(x^{t-1})\right\|^2\right] + \frac{8\tilde{\eta}^2}{KM}\sum_{k\in[K],i\in[M]} \mathbb{E}\left[\left\|\nabla F_i(\alpha_{i,k-1}^{t-1}) - \nabla F_i(x^{t-1})\right\|^2\right] \\
&+ 4\tilde{\eta}^2\mathbb{E}\left\|\nabla F(x^{t-1})\right\|^2 + \frac{9\tilde{\eta}^2}{KlM}(\varsigma^2/sR + \Sigma_g^2(\mathcal{C})).
\end{aligned}
$$

The last inequality is obtained by definition of c and c_i and by applying Jensen inequality. With Lemma C.2, this leads to the result. □

Lemma C.5 (Lag in the control variate)**.** $\forall \alpha \in [1/2, 1], \forall \tilde{\eta} \leq \frac{1}{24\nu} l^\alpha$,

$$
\mathcal{F}_t \leq \left(1 - \frac{17}{36}l\right)\mathcal{F}_{t-1} + \frac{1}{24\nu}l^{2\alpha-1}\mathbb{E}\big(F(x^{t-1}) - F(x^*)\big) + \frac{97}{48}l^{2\alpha-1}\mathcal{E}_t + \frac{l}{\nu^2}\frac{\varsigma^2/sR + \Sigma_g^2(\mathcal{C})}{32KlM}.
$$

Proof. We adapt the original proof made in the non-convex case (Karimireddy et al., 2020b, Lemma 16) and use Lemma C.1 and Lemma C.2. □

Lemma C.6 (Bounding the user drift)**.** $\forall \eta_g \geq 1, \forall \eta_l \leq 1/24\nu K\eta_g$,

$$
\frac{9}{2}\nu^2\tilde{\eta}\mathcal{E}_t \leq \frac{9}{2}\nu^3\tilde{\eta}^2\mathcal{F}_{t-1} + \frac{9}{40}\frac{\tilde{\eta}\nu}{\eta_g^2}\mathbb{E}\big(F(x^{t-1}) - F(x^*)\big) + \frac{27}{40}\frac{\tilde{\eta}^2\nu}{K\eta_g^2}\big(\varsigma^2/sR + \Sigma_g^2(\mathcal{C})\big).
$$

Proof. We once again adapt the original proof made in the non-convex case (Karimireddy et al., 2020b, Lemma 17), use Lemma C.2 and multiply on each side of the inequality by $\frac{9}{2}\nu^2\tilde{\eta}$. □

Lemma C.7 (Progress made at each round)**.** $\forall \eta_g \geq 1, \forall \eta_l \leq \min\left(\frac{1}{24K\eta_g\nu}l^{2/3}, \frac{l}{54\mu K\eta_g}\right)$,

$$
\begin{aligned}
\mathbb{E}||x^t - x^*||^2 + 27\nu^2\tilde{\eta}^2\frac{1}{l}\mathcal{F}_t \leq &\left(1 - \frac{\mu\tilde{\eta}}{2}\right)\left[\mathbb{E}||x^{t-1} - x^*||^2 + 27\nu^2\tilde{\eta}^2\frac{1}{l}\mathcal{F}_{t-1}\right] \\
&- \frac{\tilde{\eta}}{2}\mathbb{E}\big(F(x^{t-1}) - F(x^*)\big) + \frac{10\tilde{\eta}^2}{KlM}\left(1 + \frac{lM}{\eta_g^2}\right)(\varsigma^2/sR + \Sigma_g^2(\mathcal{C})).
\end{aligned}
$$

Proof. We recall that $\Delta x^t = -\frac{\tilde{\eta}}{KlM}\sum_{k\in[K],i\in C^t}\tilde{H}_i^k(y_i^{k-1}) + c^{t-1} - c_i^{t-1}$. Then,

$$
\mathbb{E}[\Delta x^t|y^0] = \mathbb{E}[\Delta x^t|x^{t-1}] = -\tilde{\eta}\mathbb{E}[c^{t-1}|y_0] = -\frac{\tilde{\eta}}{KM}\sum_{k\in[K],i\in[M]}\mathbb{E}[\nabla F_i(y_i^{k-1})|y_0]. \tag{9}
$$

We denote $\mathbb{E}_{t-1}[.]$ as the expectation conditioned on randomness generated (strictly) prior to round t, i.e. conditionally to $\sigma(x^\tau, \tau \leq t-1)$. We first bound the quantity $\mathbb{E}_{t-1}||x^t - x^*||^2 = \mathbb{E}_{t-1}||x^{t-1} + \Delta x^t - x^*||^2$,

$$
\begin{aligned}
\mathbb{E}_{t-1}||x^t - x^*||^2 &= \mathbb{E}_{t-1}||x^{t-1} - x^*||^2 + \mathbb{E}_{t-1}||\Delta x^t||^2 + 2\left[\left\langle \mathbb{E}_{t-1}[\Delta x^t|y_0], x^{t-1} - x^* \right\rangle\right] \\
&= ||x^{t-1} - x^*||^2 + \mathbb{E}_{t-1}||\Delta x^t||^2 + 2\Bigg[\Bigg\langle \underbrace{-\frac{\tilde{\eta}}{KM}\sum_{k\in[K],i\in[M]} \mathbb{E}[\nabla F_i(y_i^{k-1})|y_0]}_{\text{by (9)}}, x^{t-1} - x^* \Bigg\rangle\Bigg] \\
&\leq \mathbb{E}_{t-1}||x^{t-1} - x^*||^2 + 4\tilde{\eta}^2\nu^2\mathcal{E}_t + 8\nu^2\tilde{\eta}^2\mathcal{F}_{t-1} + 8\nu\tilde{\eta}^2\left(F(x^{t-1}) - F(x^*)\right) \\
&+ \frac{9\tilde{\eta}^2}{KlM}(\varsigma^2/sR + \Sigma_g^2(\mathcal{C})) + \underbrace{\frac{2\tilde{\eta}}{KM}\mathbb{E}_{t-1}\left[\sum_{k\in[K],i\in[M]} \langle \nabla F_i(y_i^{k-1}), x^* - x^{t-1}\rangle\right]}_{\mathcal{A}}, \qquad \text{(Lemma C.4)}
\end{aligned}
$$

where

$$
\begin{aligned}
\mathbb{E}[\mathcal{A}] &\leq \frac{2\tilde{\eta}}{KM}\mathbb{E}\left(\sum_{k\in[K],i\in[M]} F_i(x^*) - F_i(x^{t-1}) + \nu||y_i^{k-1} - x^{t-1}||^2 - \frac{\mu}{4}||x^{t-1} - x^*||^2\right) \quad \text{(convexity and } \nu\text{-smoothness)} \\
&= -2\tilde{\eta}\left(\mathbb{E}(F(x^{t-1})) - F(x^*) + \frac{\mu}{4}\mathbb{E}||x^{t-1} - x^*||\right) + 2\nu\tilde{\eta}\mathcal{E}_t.
\end{aligned}
$$

Hence, by taking the expectation:

$$
\begin{aligned}
\mathbb{E}||x^t - x^*||^2 \leq \mathbb{E}||x^{t-1} - x^*||^2 &- 2\tilde{\eta}\left(\mathbb{E}(F(x^{t-1})) - F(x^*) + \frac{\mu}{4}\mathbb{E}||x^{t-1} - x^*||^2\right) + 2\nu\tilde{\eta}\mathcal{E}_t \\
&+ 4\tilde{\eta}^2\nu^2\mathcal{E}_t + 8\nu^2\tilde{\eta}^2\mathcal{F}_{t-1} + 8\nu\tilde{\eta}^2\mathbb{E}\left(F(x^{t-1}) - F(x^*)\right) + \frac{9\tilde{\eta}^2}{KlM}(\varsigma^2/sR + \Sigma_g^2(\mathcal{C})).
\end{aligned}
$$

By combining all terms and multiplying by ν on each side of the inequality, it comes:

$$
\begin{aligned}
\nu\mathbb{E}||x^t - x^*||^2 \leq \left(1 - \frac{\mu\tilde{\eta}}{2}\right)\nu\mathbb{E}||x^{t-1} - x^*||^2 &+ (8\nu^2\tilde{\eta}^2 - 2\tilde{\eta}\nu)\left(\mathbb{E}(F(x^{t-1})) - F(x^*)\right) \\
&+ \frac{9\tilde{\eta}^2\nu}{KlM}(\varsigma^2/sR + \Sigma_g^2(\mathcal{C})) + (2\nu^2\tilde{\eta} + 4\nu^3\tilde{\eta}^2)\mathcal{E}_t + 8\nu^3\tilde{\eta}^2\mathcal{F}_{t-1}.
\end{aligned} \tag{10}
$$

We now consider $\alpha \in [1/2, 1]$, $\eta_l \leq \frac{1}{24K\nu\eta_g}l^\alpha$ and $\eta_g \geq 1$. We use the result of Lemma C.5 where each side is multiplied by $27\nu^3\tilde{\eta}^2\frac{1}{l}$ to obtain:

$$
\begin{aligned}
27\nu^3\tilde{\eta}^2\frac{1}{l}\mathcal{F}_t \leq \big(1 - \frac{\mu\tilde{\eta}}{2}\big)27\nu^3\tilde{\eta}^2\frac{1}{l}\mathcal{F}_{t-1} &+ 27\big(\frac{\mu\tilde{\eta}}{2l} - \frac{17}{36}\big)\nu^3\tilde{\eta}^2\mathcal{F}_{t-1} \\
&+ \frac{9}{8}l^{2\alpha-2}\nu^2\tilde{\eta}^2\big(\mathbb{E}(F(x^{t-1})) - F(x^*)\big) + \frac{873}{16}l^{2\alpha-2}\nu^3\tilde{\eta}^2\mathcal{E}_t \\
&+ \frac{27}{32}\nu\tilde{\eta}^2\frac{\varsigma^2/sR + \Sigma_g^2(\mathcal{C})}{KlM}.
\end{aligned} \tag{11}
$$

Since we have $\eta_l \leq \frac{1}{24K\nu\eta_g}$, we recall the result from Lemma C.6:

$$
\frac{9}{2}\nu^2\tilde{\eta}\mathcal{E}_t \leq \frac{9}{2}\nu^3\tilde{\eta}^2\mathcal{F}_{t-1} + \frac{9}{40}\frac{\tilde{\eta}\nu}{\eta_g^2}\mathbb{E}\big(F(x^{t-1}) - F(x^*)\big) + \frac{27}{40}\frac{\tilde{\eta}^2\nu}{K\eta_g^2}\big(\varsigma^2/sR + \Sigma_g^2(\mathcal{C})\big). \tag{12}
$$

By summing inequalities (10), (11), (12), we obtain:

$$\begin{aligned}
\nu\mathbb{E}||x^t - x^*||^2 + 27\nu^3\tilde{\eta}^2\frac{1}{l}\mathcal{F}_t \leq \big(1 - \frac{\mu\tilde{\eta}}{2}\big)\big(\nu\mathbb{E}||x - x^*||^2 + 27\nu^3\tilde{\eta}^2\frac{1}{l}\mathcal{F}_{t-1}\big) \\
+ \big(\frac{9}{8}l^{2\alpha-2}\nu^2\tilde{\eta}^2 + \frac{9}{40}\frac{\tilde{\eta}\nu}{\eta_g^2} + 8\nu^2\tilde{\eta}^2 - 2\tilde{\eta}\nu\big)\big(\mathbb{E}(F(x^{t-1})) - F(x^*)\big) \quad (13) \\
+ \big(\frac{315}{32} + \frac{27}{40}\frac{lM}{\eta_g^2}\big)\frac{\tilde{\eta}^2\nu}{KlM}\big(\varsigma^2/sR + \Sigma_g^2(\mathcal{C})\big) \quad (14) \\
+ \big(-\frac{5}{2}\nu\tilde{\eta} + 4\nu^2\tilde{\eta}^2 + \frac{873}{16}l^{2\alpha-2}\nu^2\tilde{\eta}^2\big)\nu\mathcal{E}_t \quad (15) \\
+ \Big(27\big(\frac{\mu\tilde{\eta}}{2l} - \frac{17}{36}\big)\nu^2\tilde{\eta}^2 + \frac{25}{2}\nu^2\tilde{\eta}^2\Big)\nu\mathcal{F}_{t-1}. \quad (16)
\end{aligned}$$

We now consider $\eta_l \leq l/54\mu K\eta_g$. Then $\tilde{\eta} \leq l/54\mu$ and we recall that $\nu\tilde{\eta} \leq 1/24$. We fix $\alpha = 2/3$ (then $2-2\alpha = \alpha$).

In this part, we aim at simplifying the terms on the right side of the last inequality.

Simplifying (13):

$$\begin{aligned}
\frac{9}{8}l^{2\alpha-2}\nu^2\tilde{\eta}^2 + \frac{9}{40}\frac{\tilde{\eta}\nu}{\eta_g^2} + 8\nu^2\tilde{\eta}^2 - 2\tilde{\eta}\nu &\leq \big(\frac{9}{8\times 24} + \frac{9}{40} + \frac{8}{24} - 2\big)\nu\tilde{\eta} \\
&= -\underbrace{\frac{1339}{960}}_{\sim 1.39}\nu\tilde{\eta} \leq -\frac{\nu\tilde{\eta}}{2}.
\end{aligned}$$

Simplifying (14):

$$\frac{315}{32} + \frac{27}{40}\frac{lM}{\eta_g^2} \leq 10\big(1 + \frac{lM}{\eta_g^2}\big).$$

Simplifying (15):

Since $l^{2\alpha-2}\nu\tilde{\eta} = \nu\tilde{\eta}\big(\frac{1}{l}\big)^{2/3} \leq 1/24$,

$$\begin{aligned}
-\frac{5}{2}\nu\tilde{\eta} + 4\nu^2\tilde{\eta}^2 + \frac{873}{16}l^{2\alpha-2}\nu^2\tilde{\eta}^2 &\leq \big(-\frac{5}{2} + \frac{4}{24} + \frac{873}{16}\frac{1}{24}\big)\nu\tilde{\eta} \\
&= -\frac{23}{384}\nu\tilde{\eta} \leq 0.
\end{aligned}$$

Simplifying (16):

Since $\frac{\mu\tilde{\eta}}{2l} \leq 1/108$,

$$27\big(\frac{\mu\tilde{\eta}}{2l} - \frac{17}{36}\big)\nu^2\tilde{\eta}^2 + \frac{25}{2}\nu^2\tilde{\eta}^2 \leq \big(27\big(\frac{1}{108} - \frac{17}{36}\big) + \frac{25}{2}\big)\nu^2\tilde{\eta}^2 = 0.$$

We then obtain the final result by dividing by ν on each side of the inequality. □

Lemma C.8 (Convergence of `DP-SCAFFOLD-warm` with convex loss functions)**.** *There exist weights $\{w_t\}$ such that $\overline{x}^T = \sum_{t\in[T]} w_t x^t$ and:*

*If f_i are μ-**strongly convex** ($\mu > 0$), $\eta_g \geq 1$, $\eta_l \leq \min\big(\frac{l^{\frac{2}{3}}}{24\nu K\eta_g}, \frac{l}{54\mu K\eta_g}\big)$, and $T \geq \max(\frac{108}{l}, \frac{48\nu}{\mu l^{\frac{2}{3}}})$,*

$$\mathbb{E}[F(\overline{x}^T)] - F(x^*) \leq \mathcal{O}\bigg(\frac{\varsigma^2/sR + \Sigma_g^2(\mathcal{C})}{\mu TKlM}\bigg(1 + \frac{lM}{\eta_g^2}\bigg) + \mu D_0^2 \exp\bigg(-\min\bigg(\frac{l}{108}, \frac{\mu l^{\frac{2}{3}}}{48\nu}\bigg)T\bigg)\bigg),$$

*If f_i are **convex**, $\eta_g \geq 1$, $\eta_l \leq \frac{l^{\frac{2}{3}}}{24\nu K\eta_g}$ and $T \geq 1$,*

$$\mathbb{E}[F(\overline{x}^T)] - F(x^*) \leq \mathcal{O}\bigg(D_0\frac{\varsigma/\sqrt{sR} + \Sigma_g(\mathcal{C})}{\sqrt{TKlM}}\sqrt{1 + \frac{lM}{\eta_g^2}} + \frac{\nu}{T}D_0^2\bigg(\frac{1}{l}\bigg)^{\frac{2}{3}}\bigg),$$

where $D_0 := ||x^0 - x^*||$.

Proof. The result of Lemma C.8 is obtained by combining the contraction inequality from Lemma C.7 and the results from technical contraction results (Karimireddy et al., 2020b, Lemmas 1 and 2). □

We then obtain the result of Theorem C.1 by setting $\eta_g := \sqrt{lM} \geq 1$ and η_l as low as possible.

C.4 Proof of Theorem C.1 (Non-Convex case)

To state this result, we adapt the original proof in the case with a larger variance for DP-noised stochastic gradients (see Lemma C.1), which gives the following result.

Lemma C.9 (Convergence of `DP-SCAFFOLD-warm` with non-convex loss functions)**.** *There exist weights $\{w_t\}$ such that $\overline{x}^T = \sum_{t\in[T]} w_t x^t$ and:*

*If f_i are **non-convex**, $\eta_g \geq 1$, $\eta_l \leq \frac{l^{\frac{2}{3}}}{24\nu K\eta_g}$ and $T \geq 1$,*

$$\mathbb{E}||\nabla F(\overline{x}^T)||^2 \leq \mathcal{O}\bigg(\sqrt{F_0}\frac{\varsigma/\sqrt{sR} + \Sigma_g(\mathcal{C})}{\sqrt{TKlM}}\sqrt{1 + \frac{lM}{\eta_g^2}} + \frac{\nu}{T}F_0\bigg(\frac{1}{l}\bigg)^{\frac{2}{3}}\bigg),$$

where $F_0 := F(x^0) - F(x^*)$.

We obtain the result of Theorem C.1 by setting $\eta_g := \sqrt{lM} \geq 1$ and η_l as low as possible.

C.5 Theorem of Convergence for `DP-FedAvg`

Theorem C.2 (Utility rates of `DP-FedAvg`$(T, K, l, s, \sigma_g, \mathcal{C})$, σ_g chosen arbitrarily)**.** *Let $\sigma_g, \mathcal{C} > 0$, $x^0 \in \mathbb{R}^d$. Suppose we run `DP-FedAvg`$(T, K, l, s, \sigma_g, \mathcal{C})$ (see Algorithm 2). Under Assumptions 2 and 3, we consider the sequence of iterates $(x^t)_{t\geq 0}$ of the algorithm, starting from x^0.*

1. *If F_i are **μ-strongly convex** ($\mu > 0$), $\eta_g = \sqrt{lM}$, $\eta_l = \frac{1}{8(1+B^2)\nu K\eta_g}$ and $T \geq \frac{8(1+b^2)\nu}{\mu}$, then there exist weights $\{w_t\}_{t\in[T]}$ such that the averaged output of `DP-FedAvg`$(T, K, l, s, \sigma_g, \mathcal{C})$, defined by $\overline{x}^T = \sum_{t\in[T]} w_t x^t$, has expected excess of loss such that:*

$$\mathbb{E}[F(\overline{x}^T)] - F(x^*) \leq \mathcal{O}\bigg(\frac{\varsigma^2/sR + \Sigma_g^2(\mathcal{C})}{\mu TKlM} + (1-l)\frac{G^2}{\mu TlM} + \frac{\nu G^2}{\mu^2 T^2} + \mu D_0^2 \exp\bigg(-\frac{\mu}{16(1+B^2)\nu}T\bigg)\bigg),$$

2. *If F_i are **convex**, $\eta_g = \sqrt{lM}$, $\eta_l = \frac{1}{8(1+B^2)\nu K\eta_g}$ and $T \geq 1$, then there exist weights $\{w_t\}_{t\in[T]}$ such that the averaged output of `DP-FedAvg`$(T, K, l, s, \sigma_g, \mathcal{C})$, defined by $\overline{x}^T = \sum_{t\in[T]} w_t x^t$, has expected excess of loss such that:*

$$\mathbb{E}[F(\overline{x}^T)] - F(x^*) \leq \mathcal{O}\bigg(D_0\frac{\varsigma/\sqrt{sR} + \Sigma_g(\mathcal{C})}{\sqrt{TKlM}} + \frac{GD_0\sqrt{1-l}}{\sqrt{TlM}} + \frac{D_0^{4/3}\nu^{1/3}G^{2/3}}{T^{2/3}} + \frac{B^2\nu D_0^2}{T}\bigg),$$

3. *If F_i are **non-convex**, $\eta_g = \sqrt{lM}$, $\eta_l = \frac{1}{8(1+B^2)\nu K\eta_g}$ and $T \geq 1$, then there exist weights $\{w_t\}_{t\in[T]}$ such that the randomized output of `DP-SCAFFOLD-warm`$(T, K, l, s, \sigma_g, \mathcal{C})$, defined by $\{\overline{x}^T = x^t$ with probability w_t for all $t\}$, has expected squared gradient of the loss such that:*

$$\mathbb{E}||\nabla F(\overline{x}^T)||^2 \leq \mathcal{O}\left(\nu\sqrt{F}\frac{\varsigma/\sqrt{sR}+\Sigma_g(\mathcal{C})}{\sqrt{TKlM}} + \frac{\nu G\sqrt{F(1-l)}}{\sqrt{TlM}} + \frac{F^{2/3}\nu^{1/3}G^{2/3}}{T^{2/3}} + \frac{B^2\nu F}{T}\right),$$

where $D_0 := ||x^0 - x^*||$ *and* $F := F(x^0) - F(x^*)$.

Proof. To state the result of Theorem C.2, we combine the original result (Karimireddy et al., 2020b, Theorem V) provided for any type of loss functions with the result of Lemma C.1. □

D Additional Experimental Details and Results

In this section, we give additional details on our experimental setup (Section D.1) and simulated data generation process (Section D.2), and provide additional results (Section D.3).

D.1 Algorithms Setup

Hyperparameter tuning. We tuned the step size hyperparameter η_0 for each dataset, each algorithm and each noise version (with or without DP) over a grid of 10 values with the lowest level of heterogeneity (5-fold cross validation conducted on the training set). We kept the same η_0 for the experiments with higher heterogeneity.

Clipping heuristic. Setting a good clipping threshold $\mathcal{C}$ while preserving accuracy can be difficult (McMahan et al., 2017b). Indeed, if $\mathcal{C}$ is too small, the clipped gradients may become biased, thereby affecting the convergence rate. On the other hand, if $\mathcal{C}$ is too large, we have to add more noise to stochastic gradients to ensure differential privacy (since the variance of the Gaussian noise is proportional to $\mathcal{C}^2$). In practice, we follow the strategy proposed by Abadi et al. (2016), which consists in setting $\mathcal{C}$ as the median of the norms of the unclipped gradients over each stage of local training. Throughout the iterations, $\mathcal{C}$ will then decrease. However, we are aware that locally setting $\mathcal{C}$ may leak information to the server about the magnitude of stochastic gradients. We here consider this leak as minor and neglect its impact on privacy guarantees. Adaptive clipping (Andrew et al., 2021) could be used to mitigate these concerns.

D.2 Simulated Data Generation

Each ground-truth model for a user i consists of weights $W_i \in \mathbb{R}^{d'\times 10}$ and bias $b_i \in \mathbb{R}^{10}$, which are sampled from the following distributions: $W_i|u_i \sim \mathcal{N}_{d'\times 10}(u_i, \mathrm{Id})$ and $b_i|u_i' \sim \mathcal{N}_{10}(u_i', \mathrm{Id})$ where $u_i \sim \mathcal{N}_{d'\times 10}(0, \alpha\mathrm{Id})$ and $u_i' \sim \mathcal{N}_{10}(0, \alpha\mathrm{Id})$. The data matrix X_i of user i is sampled according to $X_i|v_i \sim \mathcal{N}_{d'}(v_i, \Sigma)$ where Σ is the covariance matrix defined by its diagonal $\Sigma_{j,j} = j^{-1.2}$ and $v_i|B_i \sim \mathcal{N}_{d'}(B_i, \mathrm{Id})$ where $B_i \sim \mathcal{N}_{d'}(0, \nu\mathrm{Id})$. The label of each data point is obtained by independently changing the label given by the ground-truth model with probability 0.05.

D.3 Experiments

In this section, we provide more results on the experiments described in Section 5, and also present additional experiments.

Metrics. To measure the convergence and performance of the algorithms at any communication round $t \in [T]$, we consider the following metrics:

- *Accuracy(t)*: the average test accuracy of the model over all users,
- *Train Loss(t)*$= \log_{10}(F(x^t) - F^*)$: the log-gap between the objective function evaluated at parameter x^t and its minimum,
- *Train Gradient Dissimilarity(t)*$= \frac{1}{M}\sum_{i=1}^{M} ||\nabla F_i(x^t)||^2 - ||\nabla F(x^t)||^2$, which measures how the local gradients differ from the global gradient (i.e., the average across users) when evaluated at x^t, and hence quantifies the *user-drift* over the rounds of communication. Remark that, without any kind of heterogeneity, this variable would converge to 0.

Experiments with TK fixed. In Section 5 of the main text, we conducted experiments on both FEMNIST and simulated data with a total number of iterations that is *constant* ($TK = 4.10^4$ for FEMNIST data and $TK = 8.10^4$ for simulated data). As observed in previous work (Karimireddy et al., 2020b), these experiments recover the superiority of `SCAFFOLD` over `FedAvg` and `FedSGD` under heterogeneous data, but most importantly they show that this hierarchy is preserved in our DP-FL framework with privacy constraints: this is especially clear with growing heterogeneity and with growing number K of local updates. Besides this, the results provided for logistic regression in the high-privacy regime ($\epsilon = 1.5$) numerically demonstrate that `DP-SCAFFOLD-warm` **actually outperforms (non-private)** `FedAvg` (see Fig. 1), despite the local injection of Gaussian noise! Therefore, our results are quite promising with respect to obtaining efficient DP-FL algorithms under heterogeneous data.

We provide below some additional results which complement those provided in Section 5.

- **Simulated data.** We plot in Fig. 5 the evolution of the accuracy over the rounds, which is consistent with the evolution of the train loss in Fig. 1. While the variance of the accuracy for `DP-FedAvg` grows with the heterogeneity, the results of `DP-SCAFFOLD-warm` are not affected. We can observe an average difference of 10% in the accuracy for these two algorithms over the various heterogeneity settings. We provide in Fig. 6 the evolution of the gradient dissimilarity for the same settings as in Fig. 1 and Fig. 5, which once again shows a better convergence of `DP-SCAFFOLD-warm` compared to `DP-FedAvg` for the same privacy level. We also provide the evolution of the train loss when varying a single heterogeneity parameter: either α (which controls *model* heterogeneity across users) in Fig. 7 or β (which controls *data* heterogeneity across users) in Fig. 8. In both of these settings, `DP-SCAFFOLD-warm` performs consistently better.
- **FEMNIST data.** In Fig. 9, we put in perspective the accuracy observed with $K = 50$ (see Fig. 2) with the one observed with $K = 100$. We also show the evolution of the gradient dissimilarity in Fig. 10. These results on real data again show the superior performance of `DP-SCAFFOLD-warm`, consistently with our observations on simulated data.

In the following experiments, we keep the same privacy budget ($\epsilon = 4.5$ for FEMNIST data and $\epsilon = 1.5$ for simulated data, $\delta = 1/MR$) as described in Section 5 as well as the same values for l and s.

Experiments with T fixed and K growing. In this part, we aim to highlight the impact of K in the performance of the DP versions of `SCAFFOLD` and `FedAvg`. We fix $T = 400$ and compute the metrics for simulated data (Fig. 4, $\gamma \in \{0\%, 10\%, 100\%\}$) and for FEMNIST data (Fig. 3, $(\alpha, \beta) \in \{(0,0), (1,1), (5,5)\}$) with $K \in \{5, 20, 40\}$, while keeping the same privacy budget (average over 3 random runs). Remark that the value of σ_g is lower for `DP-SCAFFOLD-warm` than `DP-FedAvg`, due to the saving of the first rounds for the warm-start version (no model is trained yet). Local learning rate factor η_0 was carefully tuned for each value of K (5-fold cross-validation). Our results show that `DP-SCAFFOLD-warm` does not suffer from an increasing value of K (on the contrary, `DP-FedAvg` is more fluctuating as K grows) and consistently outperforms `DP-FedAvg`. However, we do not observe a clear improvement of the precision of the model trained with `DP-SCAFFOLD-warm` as K grows.

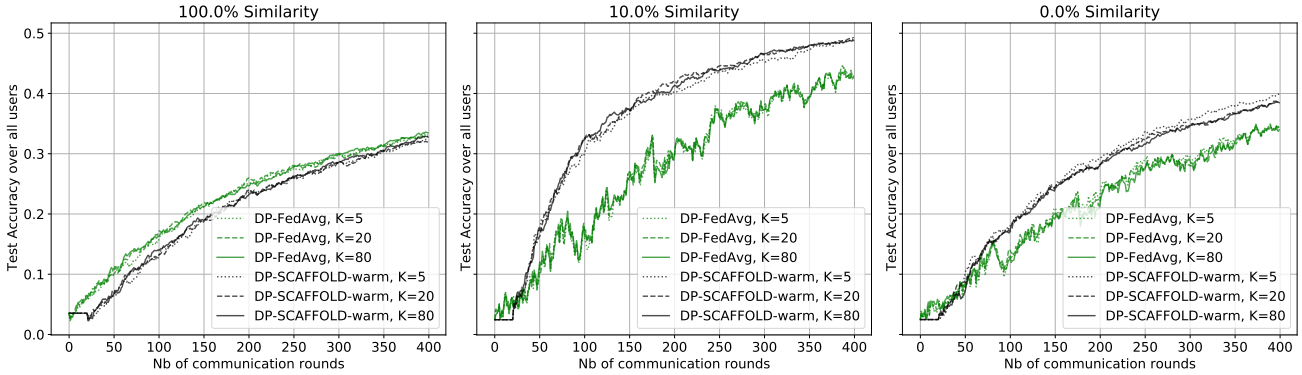

Figure 3: Test Accuracy on FEMNIST data with $(4.5, 10^{-5})$-DP for $T = 400$.

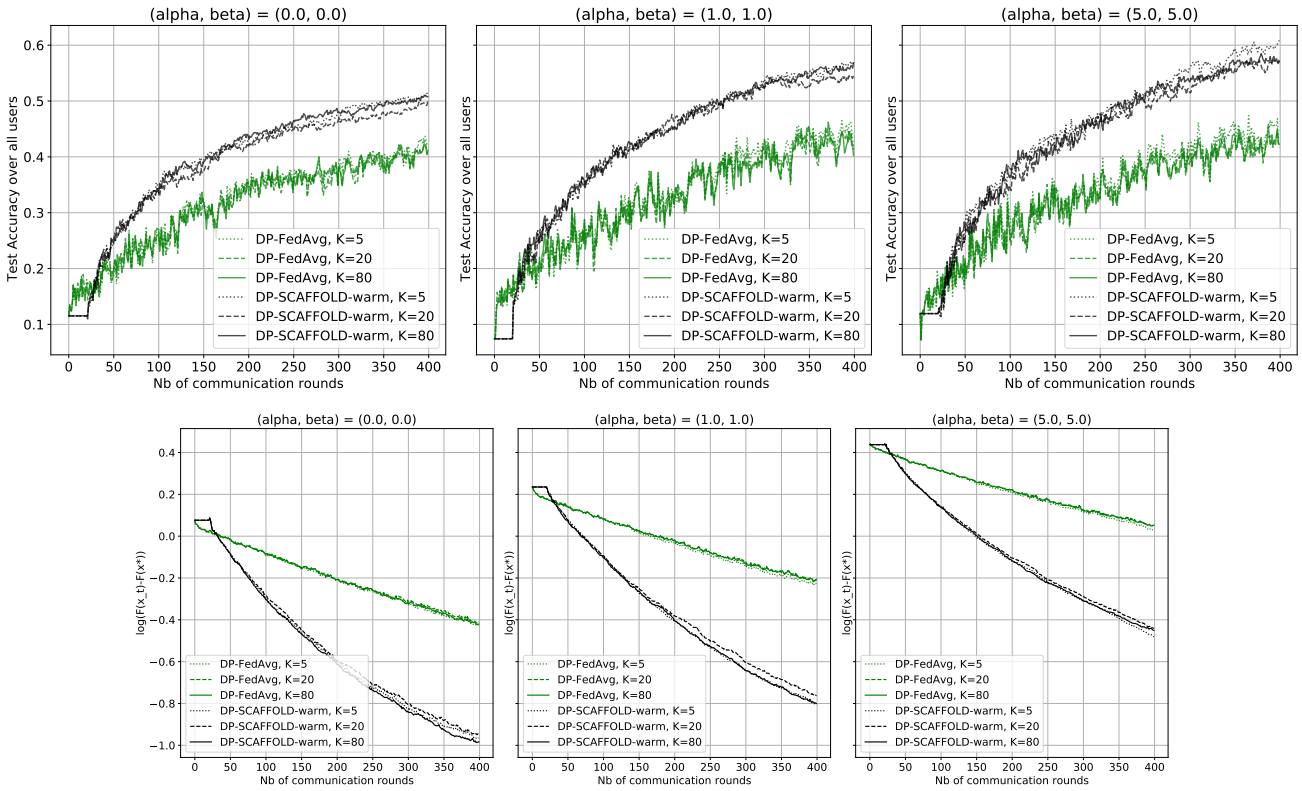

Figure 4: Metrics on simulated data with $(1.5, 2.10^{-6})$-DP for $T = 400$. First row: Accuracy; Second row: Loss.

Experiments with K fixed and T maximized (simulated data). Given some values for σ_g and K, we calculate the maximal value of T such that the privacy guarantee is still maintained in the output after T communication rounds. We compute in Table 2 the test accuracy obtained after these iterations (average over 3 random runs) for a low heterogeneity setting $(\alpha, \beta) = (0, 0)$ and a strong heterogeneity setting $(\alpha, \beta) = (5, 5)$. Our results show that, given a constraint on T, K (and thus σ_g), `DP-SCAFFOLD-warm` is even more efficient when the heterogeneity is high. Our analysis also highlights the trade-off between T and K (which relates to hardware and communication constraints in real deployments) to achieve some given performance. For example, we show that under high heterogeneity, we obtain roughly the same accuracy if we run `DP-SCAFFOLD-warm` with $(T, K) = (266, 5)$ or $(T, K) = (182, 20)$ (in blue), as well as $(T, K) = (62, 20)$ or $(T, K) = (42, 80)$ (in red). In practical deployments, T is often the bottleneck, so smaller T and larger K are often preferred. Note also that we observe a threshold due to a DP noise when $(\sigma_g, K) = (40, 20)$, which leads to similar performance as the setting $(\sigma_g, K) = (20, 5)$ (in brown). For our experiments, we did not take into account the communication rounds from the warm-start ($4/l = 20$ rounds) to compute T.

Table 2: Test accuracy for `DP-SCAFFOLD-warm` with T maximized (simulated data, $\epsilon = 1.5$)

σ_g	(α, β)	$K = 5$		$K = 20$		$K = 80$	
15	(0,0)	48.93% ∓ 0.73	$T = 266$	42.46% ∓ 0.25	$T = 62$	24.11% ∓ 1.75	$T = 14$
	(5,5)	**58.02**% ∓ 3.16		**47.76**% ∓ 0.09		**28.14**% ∓ 2.80	
20	(0,0)	51.83% ∓ 0.95	$T = 480$	46.48% ∓ 1.67	$T = 114$	34.70% ∓ 1.25	$T = 26$
	(5,5)	**64.97**% ∓ 1.34		**55.34**% ∓ 1.17		**37.41**% ∓ 3.41	
25	(0,0)	*Not done*	$T = 758$	49.72% ∓ 0.14	$T = 182$	39.39% ∓ 0.15	$T = 42$
	(5,5)			**59.74**% ∓ 1.63		**44.02**% ∓ 1.36	
40	(0,0)	*Not done*	$T = 1972$	51.83% ∓ 0.81	$T = 480$	45.23% ∓ 1.33	$T = 114$
	(5,5)			**62.11**% ∓ 1.83		**54.49**% ∓ 1.61	

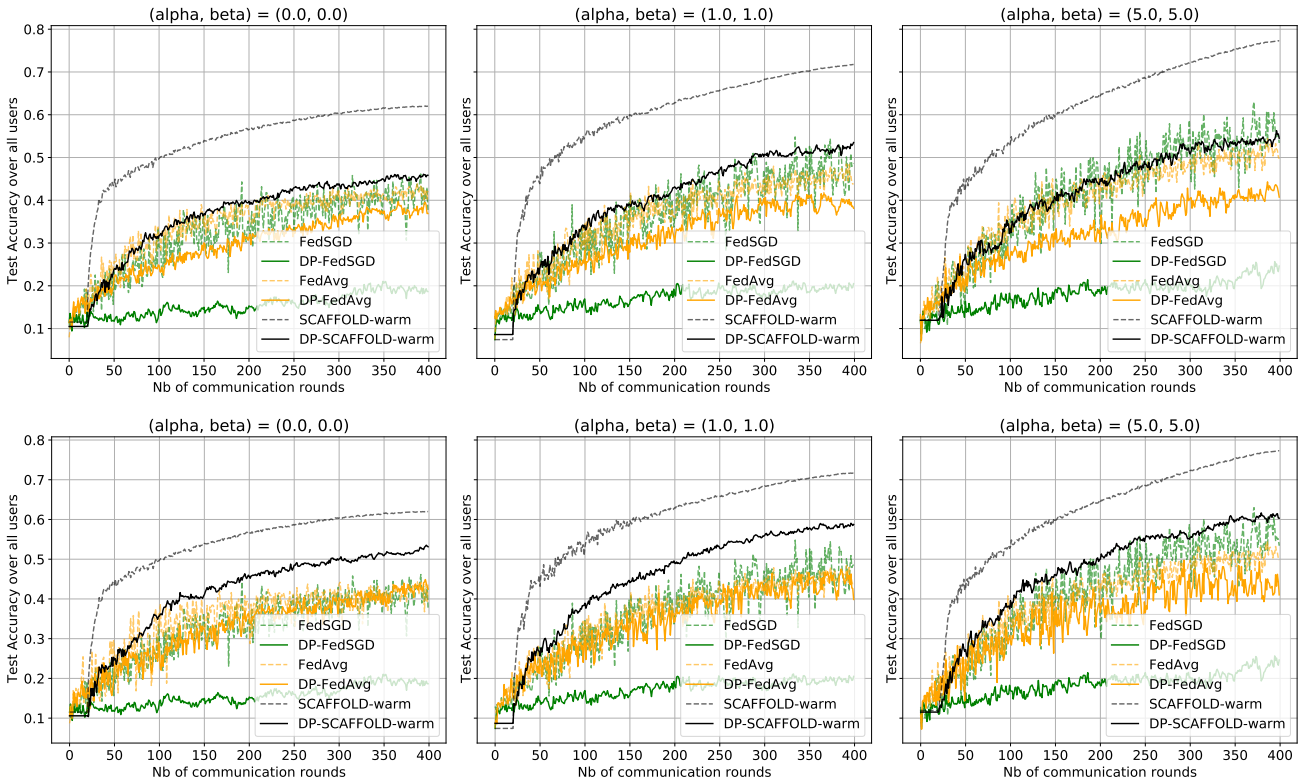

Figure 5: Test Accuracy on simulated data with $(1.5, 2.10^{-6})$-DP. First row: $K = 50$; Second row: $K = 100$.

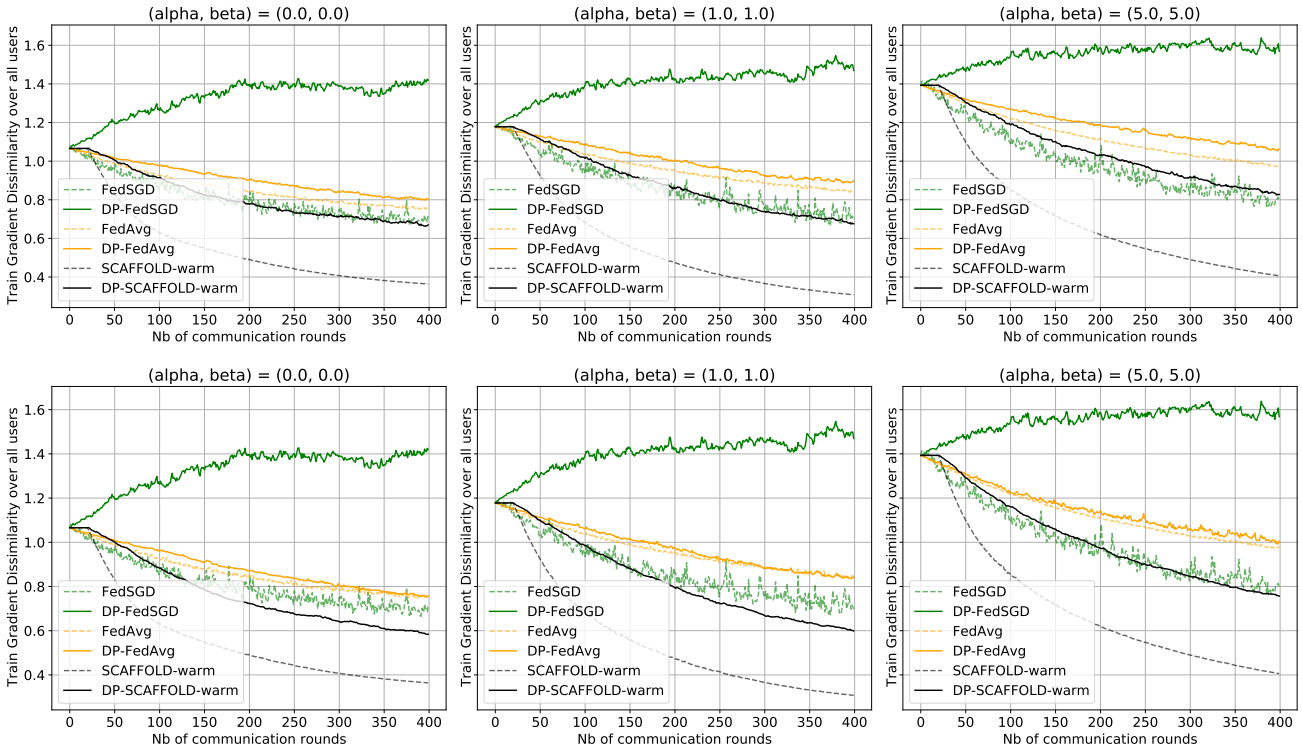

Figure 6: Train Grad. Diss. on simulated data with $(1.5, 2.10^{-6})$-DP. First row: $K = 50$; Second row: $K = 100$.

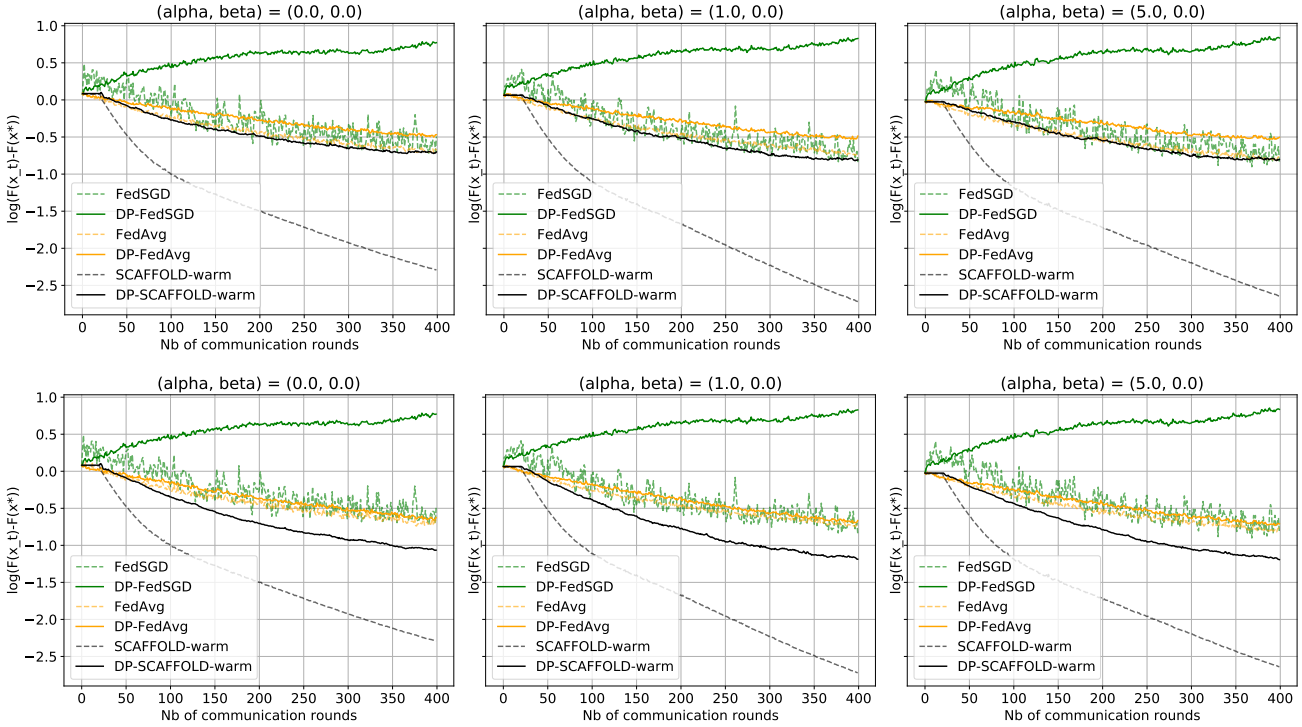

Figure 7: Model Heterogeneity (varying α): Train Loss on simulated data with $(1.5, 2.10^{-6})$-DP ($K = 50$ and $K = 100$)

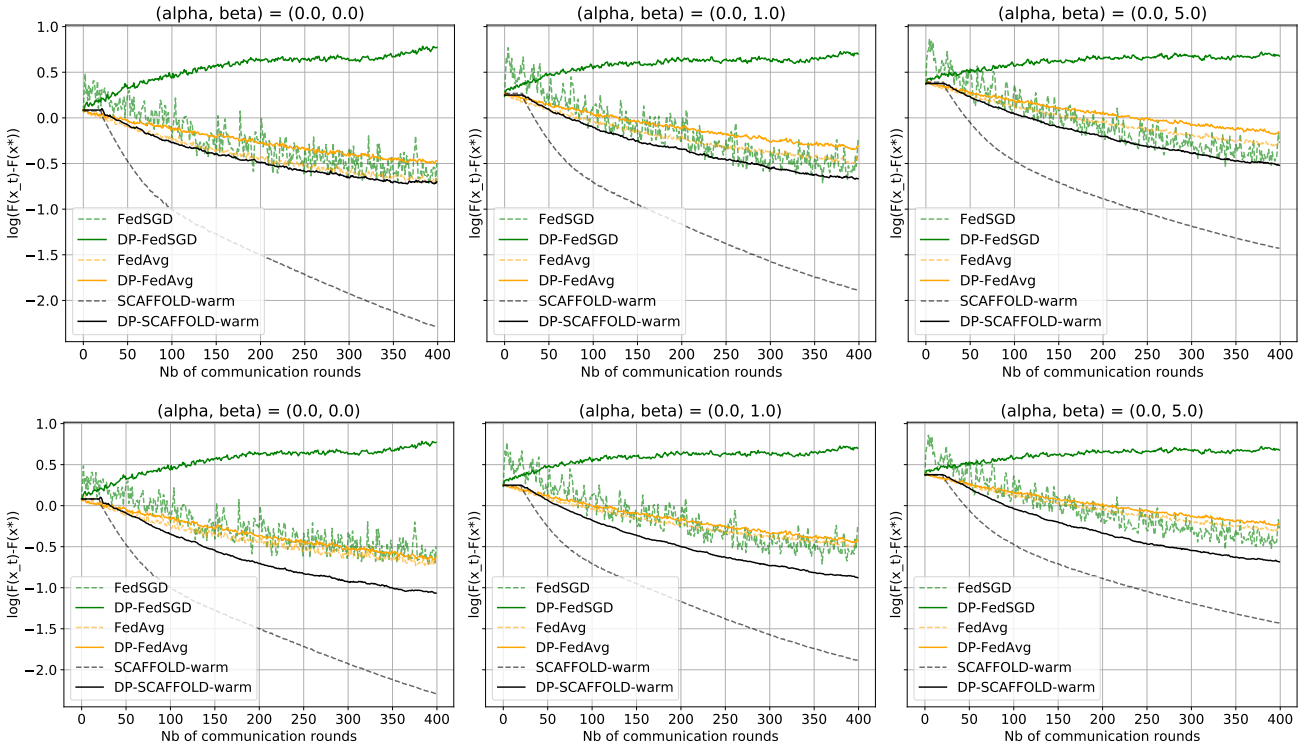

Figure 8: Data Heterogeneity (varying β): Train Loss on simulated data with $(1.5, 2.10^{-6})$-DP ($K = 50$ and $K = 100$)

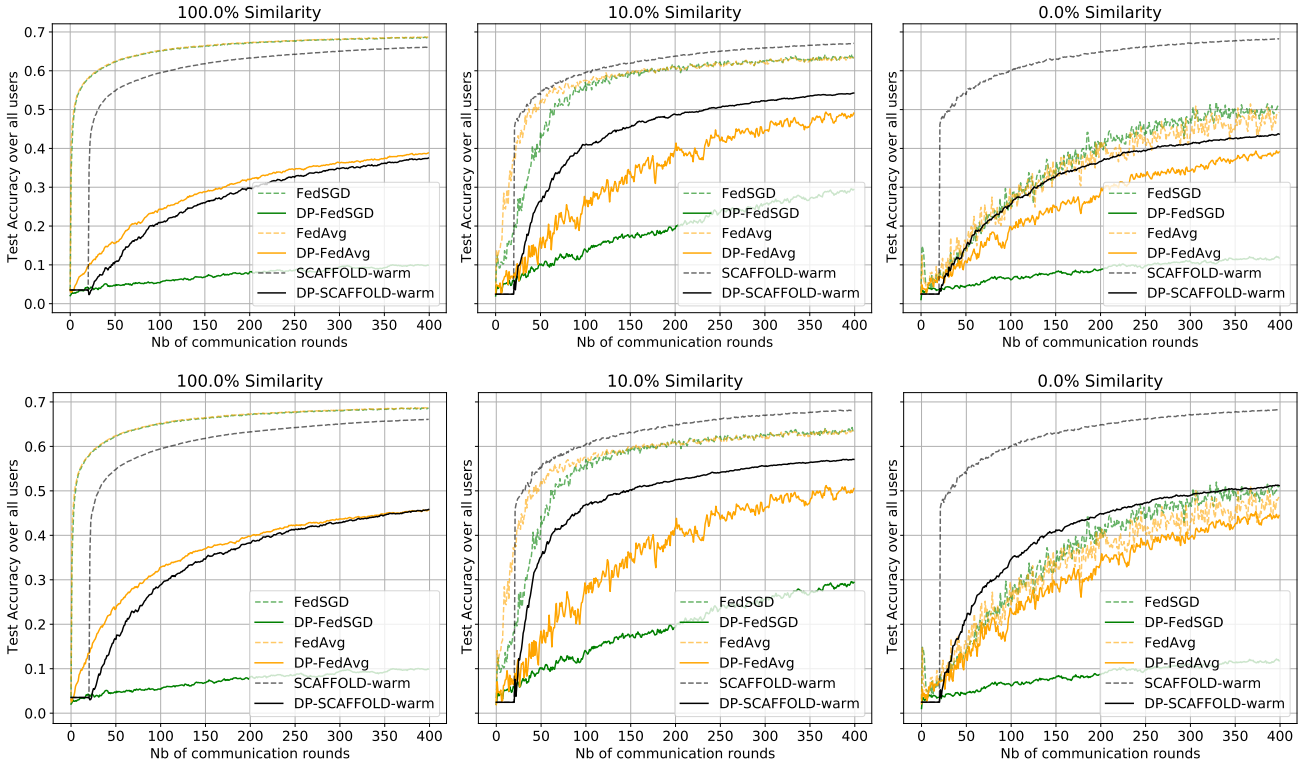

Figure 9: Test Accuracy on FEMNIST data with $(4.5, 10^{-5})$-DP. First row: $K = 50$; Second row: $K = 100$.

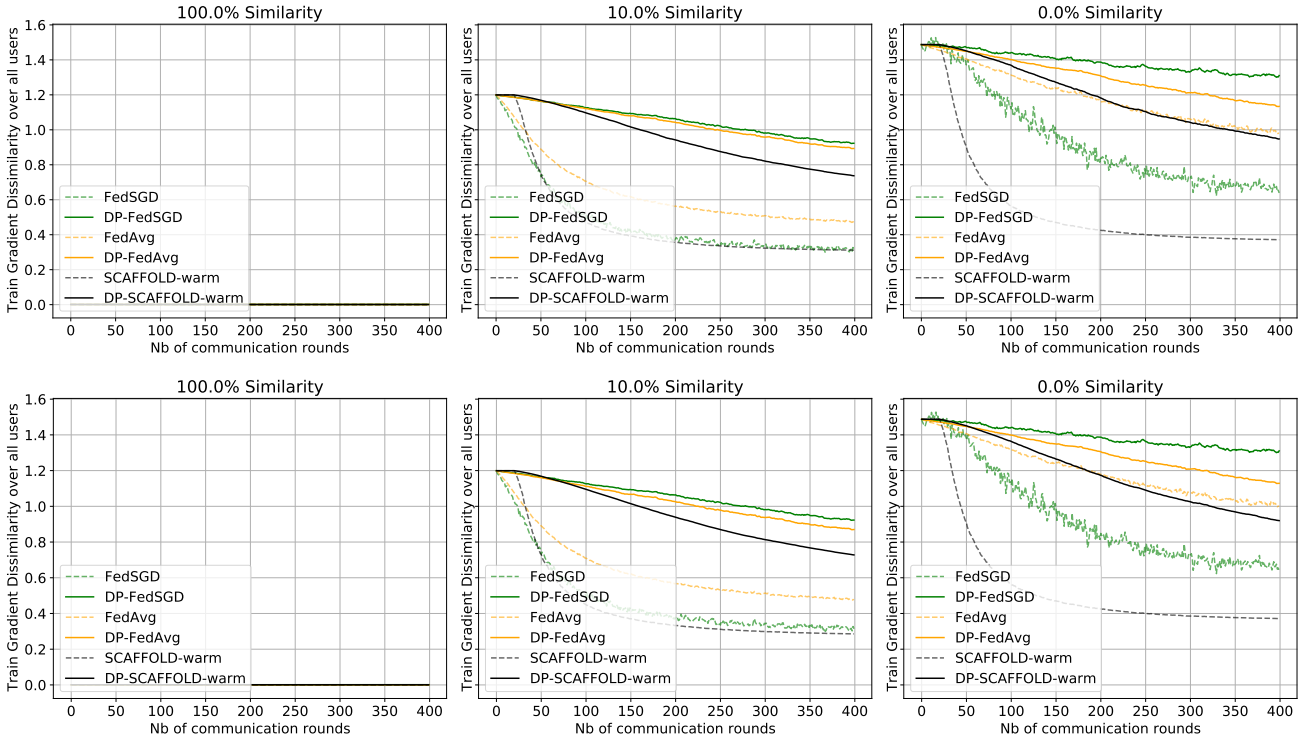

Figure 10: Train Grad. Diss. on FEMNIST data with $(4.5, 10^{-5})$-DP. First row: $K = 50$; Second row: $K = 100$.