Cross-silo Federated Learning with Record-level Personalized Differential Privacy

Junxu Liu
Renmin University of China
Beijing, China
geminiljx@gmail.com

Jian Lou
Zhejiang University
Hangzhou, China
jian.lou@zju.edu.cn

Li Xiong
Emory University
Atlanta, USA
lxiong@emory.edu

Jinfei Liu
Zhejiang University
Hangzhou, China
jinfeiliu@zju.edu.cn

Xiaofeng Meng*
Renmin University of China
Beijing, China
xfmeng@ruc.edu.cn

ABSTRACT

Federated learning (FL) enhanced by differential privacy has emerged as a popular approach to better safeguard the privacy of client-side data by protecting clients' contributions during the training process. Existing solutions typically assume a uniform privacy budget for all records and provide one-size-fits-all solutions that may not be adequate to meet each record's privacy requirement. In this paper, we explore the uncharted territory of cross-silo FL with record-level personalized differential privacy. We devise a novel framework named *rPDP-FL*, employing a two-stage hybrid sampling scheme with both uniform client-level sampling and non-uniform record-level sampling to accommodate varying privacy requirements.

A critical and non-trivial problem is how to determine the ideal per-record sampling probability q given the personalized privacy budget ε. We introduce a versatile solution named *Simulation-CurveFitting*, allowing us to uncover a significant insight into the nonlinear correlation between q and ε and derive an elegant mathematical model to tackle the problem. Our evaluation demonstrates that our solution can provide significant performance gains over the baselines that do not consider personalized privacy preservation.

CCS CONCEPTS

• **Security and privacy → Privacy protections**.

KEYWORDS

Federated Learning, Differential Privacy, Personalized Privacy Protection

ACM Reference Format:
Junxu Liu, Jian Lou, Li Xiong, Jinfei Liu, and Xiaofeng Meng. 2024. Cross-silo Federated Learning with Record-level Personalized Differential Privacy. In *Proceedings of the 2024 ACM SIGSAC Conference on Computer and Communications Security (CCS '24), October 14–18, 2024, Salt Lake City, UT, USA.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3658644.3670351

*Corresponding author: Xiaofeng Meng.

CCS '24, October 14–18, 2024, Salt Lake City, UT, USA

ACM ISBN 979-8-4007-0636-3/24/10
https://doi.org/10.1145/3658644.3670351

1 INTRODUCTION

Federated Learning (FL) [23, 41] is a recent machine learning (ML) framework that was motivated by data privacy. In comparison to centralized ML, it eliminates the need for centralized data sharing and has the potential to harness decentralized data for powerful predictive models while alleviating individual privacy concerns. The distinctive feature is its decentralized architecture, where multiple institutions (e.g., hospitals or banks) or devices (e.g., smartphones, IoT devices [6, 15, 38]) collaborate in training a joint model under the coordination of a central *server* while keeping the data local. This paper primarily focuses on the former case, also known as cross-silo FL [36], where each *client* (institution) holds a local dataset comprising personal data *records*. For simplicity, we assume each *record* is associated with a single *user* (e.g. patient or customer) who does not contribute the same record or multiple records to multiple clients simultaneously.

Although data are not directly shared in FL, potential adversaries (e.g., the honest-but-curious server or untrusted clients) might engage in indirect privacy violations via reconstruction or inference attacks [21, 37, 46, 51, 53, 57–59]. Differential privacy (DP), known as the de facto standard for private data analysis, has been introduced to FL algorithm design [13, 14, 32, 34, 35, 39, 40, 56]. This integration ensures rigorous privacy protection for participants (clients or records) by introducing controlled perturbation into the computation of the intermediate model parameters transferred between clients and the server [42]. While standard DP provides the means to quantify the extent of privacy protection through a positive real-valued parameter ε (aka *privacy budget*), it imposes identical privacy safeguards on every participant involved. This uniformity cannot reflect the reality of diverse privacy expectations among people and can lead to significant utility costs [3, 11, 22]. It is desirable to allow each participant to set their expected privacy budgets reflecting their personal privacy preferences.

With this objective in mind, personalized differential privacy (PDP) [11, 22] was introduced and has been investigated in various scenarios including statistical analysis [5, 11, 22], centralized machine learning (ML) [3, 12], and federated learning [33]. For FL, Liu et al. [33] proposed the concept of heterogeneous DP in FL, where

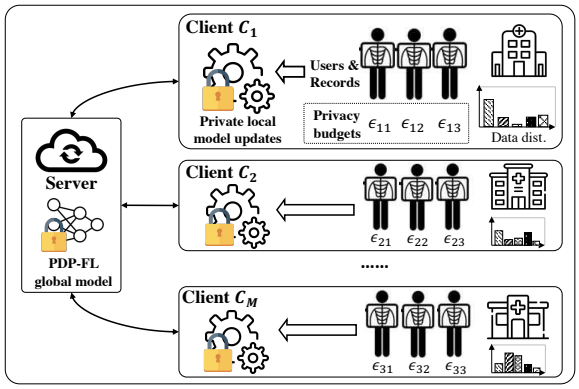

Figure 1: An illustration of the cross-silo federated learning with record-level personalized differential privacy. In this framework, each user is given the autonomy to independently opt for a personalized privacy preference (specified by a personalized differential privacy (PDP) budget ε) for their respective records. The goal is to train a private global model that satisfies record-level PDP.

records within a single client (institution) share the same privacy budget, but different clients may have varying privacy budgets. We refer to this specific setting as *client-level* PDP-FL for clarity. In contrast, this paper introduces a broader setting where even records within the same client may have distinctive privacy preferences, referred to as *record-level* PDP-FL (rPDP-FL in short). Figure 1 provides an illustrative example of the latter case in a healthcare context. To the best of our knowledge, *record-level* PDP-FL has not yet been investigated.

From a technical standpoint, the essence of implementing *record-level* PDP lies in ensuring each record's accumulative privacy cost aligns with its predetermined privacy budget during the entire training process. This emphasizes the need for an effective privacy budget allocation strategy. When it comes to achieving PDP in a centralized ML setting (centralized PDP), some studies [12, 47] use even privacy budgets across all records during every iteration of training, and records with smaller privacy budgets will be filtered out of the training process once their privacy budgets run out. This approach may trigger *catastrophic forgetting* [18, 26], a situation where the learned model could potentially "forget" the knowledge from records that terminate early, eventually leading to degraded model performance. A more promising strategy involves achieving simultaneous depletion of privacy budgets for all records by developing DP mechanisms coupled with non-uniform sampling [3, 22]. The fundamental theory underlying it is the "privacy amplification by random sampling" theorem [1, 2, 29, 52, 60] which implies individuals with lower inclusion (sampling) probabilities q will incur less privacy cost (leading to smaller privacy budget). Boenisch et al. [3] introduced a binary search-based approach to determine an approximate optimal $q \in [0, 1]$ for each record with a specific privacy budget ε. The computation of the accumulative privacy cost for each record is based on Mironov et al. [44] in which *Rényi Differential Privacy* (RDP) [43, 44] is employed for tight privacy accounting. However, their approach is impractical when all records' privacy budgets are distributed continuously and can cover a spectrum of values, due to the scalability and computational costs associated with the per-record search process.

Overall, there exist significant research gaps in achieving record-level PDP-FL. First, the existing theoretical findings on RDP-based privacy accounting [44, 60] are no longer adequate for the needs of analyzing each record's accumulative privacy cost in FL applications. Intuitively, the two-stage sampling process, i.e., client-level and record-level sampling, will further increase uncertainty for potential adversaries to infer whether a "target" record is a member of a client and hence further amplify the privacy protection. Second, the existing binary search-based approach [3] for finding the optimal sampling probability q in centralized PDP is not efficient. A desirable way is to directly compute a sampling probability given a privacy budget for each record. However, it's non-trivial to derive an explicit closed-form solution due to the highly nonlinear and less interpretable RDP-based privacy accounting.

Contributions. Our key contributions are outlined as follows.

(1) We formalize a real-world problem in federated learning concerning record-level personalized differential privacy. To solve this problem, we propose a novel framework called *rPDP-FL*. The essence of this framework is a *two-stage hybrid sampling* scheme which comprises a uniform client-level sampling process and a non-uniform record-level sampling process. Specifically, the per-client sampling probability is assumed as a hyperparameter and publicly known by both the server and all clients, while the per-record sampling probability is proportional to each record's privacy budget and determined by the client to which it belongs.
(2) We formally analyze the enhanced privacy amplification effect of the two-stage hybrid sampling scheme. This RDP-based theoretical investigation fills a gap in existing research and facilitates a more favorable trade-off between privacy and utility.
(3) We devise an efficient and general strategy named *Simulation-CurveFitting* (SCF) to identify the sampling probabilities for all records given their personalized privacy budgets. Our simulations with varying sampling probabilities enable the identification of an elegant mathematical function discerning the relationship between per-record sampling probabilities and their accumulative privacy costs. An important insight arises: the tight upper bound on the accumulative privacy cost of rPDP-FL can be modeled by a simple exponential function w.r.t. its record-level sampling probability.
(4) We simulate three potential personalized privacy scenarios and conduct a comprehensive evaluation on two real-world datasets. We first show that our SCF strategy outperforms the existing PDP methods for centralized ML [3, 12] in model utility and computational efficiency. Additionally, we demonstrate that rPDP-FL significantly enhances the utility of the global model compared to baseline methods that do not incorporate personalized privacy preservation.

2 RELATED WORK

Personalized Differential Privacy. The concept of personalized DP (PDP) was initially introduced by Ebadi et al. [11] and Jorgensen et al. [22], focusing on basic private statistical analysis tasks with the standard ε-DP framework. Notably, the *Sample* mechanism proposed in [22] demonstrated the feasibility of implementing PDP by

combining DP mechanisms (e.g., Laplace or Gaussian mechanism) with non-uniform record-level sampling.

Recent work studied PDP in centralized ML [3] built on top of the non-uniform sampling strategy and proposed a binary search-based approach to find a suitable sampling probability q as a decimal value within the range of $[0, 1]$ for each record given a target privacy budget ε. It is, however, computationally demanding for the more realistic settings where records' privacy budgets are distributed continuously (e.g., follow a Gaussian or Pareto distribution) and can cover a range of values. Another line of work [12, 47] considered all records' privacy budgets to be uniform during each iteration of the training process. Two individual privacy accounting techniques named *privacy odometer* and *privacy filter* are designed to monitor and restrict accumulative privacy costs for individual records throughout the training process so that a record will be excluded from the subsequent training iterations once its privacy budget is exhausted. This poses a potential risk of *catastrophic forgetting* [18, 26] and may lead to downgraded model performance.

Federated Learning with DP and Personalized DP. We discuss existing work on FL with DP in two aspects: (1) the granularity of the DP guarantee, i.e., what information is protected (each client or each record), and (2) the level of personalization for DP, i.e., who has the right to specify the privacy budget (the central server, each client, or each record).

- *Client- vs. record-level privacy protection.* There is rich literature exploring the DP-FL framework concerning potential adversaries. Specifically, these adversaries may be either solely recipients of the global model parameters (i.e., the other *untrusted* clients or third parties) or recipients of local model updates (i.e. the *honest-but-curious* central server). Within this framework, two categories of DP guarantees are recognized: client- and record-level DP. The former is achieved by adding random Gaussian noise to the aggregated local model updates to hide a single client's contribution [16], while the latter requires clients to perturb their computed gradients locally to obscure a single record's contribution [36, 42, 54]. Our primary focus lies on achieving record-level protection against both attack scenarios.
- *Client- vs. record-level privacy personalization.* As mentioned earlier, the majority of approaches offer uniform privacy guarantees for all records involved, based on the one-sided considerations of the central server. Only a few studies recognize the necessity of privacy personalization within FL applications. Liu et al. [33] introduced the concept of heterogeneous DP and developed a projection-based framework to accommodate diverse privacy budgets among different clients. Although the work [36] also proposed a similar notion known as *silo-specific sample-level* DP, it primarily focused on addressing data heterogeneity challenges and did not address varying privacy needs. Liu et al. [35], on the other hand, centered on cross-device FL and achieved personalized local differential privacy (PLDP) for clients' local model gradients. However, it requires a large number of clients for reasonable utility. Our research represents the first attempt to explore record-level privacy personalization in cross-silo FL.

Tight Privacy Analysis for DP-FL. Conducting a tight analysis of the accumulative privacy cost is crucial for designing DP algorithms effectively. The predominant focus of research on this issue centers around centralized ML [1, 44, 52, 60], with limited attention directed towards the FL scenarios [17, 45] where the employment of both data and client sampling may lead to an enhanced privacy amplification effect. Girgis et al. [17] investigated a related issue but focused on offering local differential privacy (LDP) guarantees for clients' gradients. In their framework, only one step of the local Stochastic Gradient Descent (SGD) update is executed per client per round, whereas our algorithm allows for multiple local updates. Noble et al. [45] adopted RDP to track the privacy cost over the local SGD iterations, while using (ε, δ)-DP to evaluate privacy costs over global communication rounds. This conventional privacy notion is often considered suboptimal in practical applications. Our work extends existing findings by leveraging RDP tools to estimate the gain of privacy caused by client sampling, see Section 5.

3 PRELIMINARIES

Differential Privacy (DP) is a robust and mathematically rigorous definition of privacy. It allows for the quantification of the information leaked by an algorithm about its input data. Note that when two datasets D and D' differ by only one record[1], denoted as $D \sim D'$, we refer to them as adjacent datasets.

Definition 1 ((ε, δ)-Differential Privacy [8, 9])**.** A randomized algorithm $\mathcal{A} : \mathbb{D} \to \mathbb{O}$ satisfies (ε, δ)-DP if for any pair of adjacent datasets $D, D' \in \mathbb{D}$ and any subsets of outputs $o \subseteq \mathbb{O}$, it holds that

$$\Pr[\mathcal{A}(D) \in o] \leq e^{\varepsilon} \Pr[\mathcal{A}(D') \in o] + \delta.$$

The privacy guarantee is controlled by the "privacy budget" $\varepsilon > 0$ and the parameter $\delta \geq 0$ which captures the probability that the pure ε-DP (i.e., $(\varepsilon, 0)$-DP) is broken. While the standard (ε, δ)-DP is widely used in a broad range of literature, it may not be suitable for some settings. The following are two notable limitations associated with (ε, δ)-DP recognized in literature:

(1) (ε, δ)-DP provides *uniform* privacy guarantees for the entire dataset regardless of the individuals' preferences;
(2) (ε, δ)-DP offers a relatively *loose* composition bound and thus it is not suitable to track and analyze the overall privacy cost of complex iterative algorithms which will lead to poor privacy and utility trade-off.

In this study, our aim is to design a finely tailored algorithm with personalized privacy that effectively tackles the aforementioned challenges in the context of FL applications. We first review the notions of personalized differential privacy (PDP) and Rényi differential privacy (RDP), both of which form building blocks for our privacy analysis and algorithm design. More specifically, PDP tailors the level of privacy protection based on the specific privacy preferences of each record. RDP offers a versatile framework for tight privacy accounting and better privacy-utility trade-offs.

3.1 Personalized Differential Privacy

Personalized DP is a variation of DP that bounds the *individual* privacy cost for each record in the dataset. For example, the privacy guarantee for a specific record d_j is defined over all pairs of adjacent datasets that differ by d_j, denoted as $D \overset{d_j}{\sim} D_{-j}$. For clarity, we refer

[1] In this work, we consider the presence/absence model of privacy, where protection is w.r.t. the presence/absence of a record in the analyzed dataset, e.g., $D' \triangleq D \setminus \{d\}$, instead of the replacement of a record with another.

to this variant as *record-specific* adjacent datasets and we have the relationship $\{(D, D_{-j})\} \subset \{(D, D')\}$.

Definition 2 ($(\mathcal{E}, \delta)$-Personalized Differential Privacy [22])**.** Given a dataset D with each record $d_j \in D$ corresponding to a specific privacy budget $\varepsilon_j > 0$. Let $\mathcal{E} = \{\varepsilon_j\}_{j\in[N]}$. A randomized algorithm $\mathcal{A} : \mathbb{D} \to \mathbb{O}$ satisfies $(\mathcal{E}, \delta)$-personalized differential privacy (PDP) if it guarantees (ε_j, δ)-DP w.r.t. the specific record d_j. That is, for any pair of record-specific adjacent datasets $D, D_{-j} \in \mathbb{D}$ and any subsets of output $o \subseteq \mathbb{O}$, it holds that

$$\Pr[\mathcal{A}(D) \in o] \leq e^{\varepsilon_j} \Pr[\mathcal{A}(D_{-j}) \in o] + \delta.$$

Remark 1. Although δ is also an important DP parameter and technically its value could be randomly specified like ε, we assume all records share a common δ with a small, positive default value in this paper based on the following two considerations.

- On the one hand, δ is commonly taken to be "sub-polynomially small", that is, a rule-of-thumb is that it should be much smaller than the inverse of any polynomial in the size of the dataset [9, 10]. Since individuals may not have access to the complete dataset or information about its size, it becomes difficult for them to properly set a value for δ that meets the desired privacy budgets.
- On the other hand, the choices of ε and the choices of δ are statistically independent, that is, for two different records $d_1, d_2 \in D$, if $\varepsilon_1 \geq \varepsilon_2$, it is not necessarily always $\delta_1 \geq \delta_2$ (and vice versa). We argue this issue is complicated and leave it as an open problem.

The Sample Mechanism. Building upon the findings of privacy amplification by random sampling [1, 2, 29, 52, 60], Jorgensen et al. [22] proposed the Sample mechanism. It achieves $(\mathcal{E}, 0)$-PDP by applying an arbitrary mechanism that satisfies ε-DP on a subset of data records which is obtained by a *non-uniform Poisson sampling* procedure. Our work is inspired by this idea but encounters greater challenges due to the utilization of RDP, detailed in Section 4.2.

Definition 3 (Poisson Sampling [60])**.** Given a dataset D with size N and a set of per-record sampling probabilities $\mathbf{q} = \{q_i | q_i \in [0, 1], i \in [N]\}$, the Poisson sampling procedure outputs a subset $\{d_i | \beta_i = 1, i \in [N]\}$ by sampling a Bernoulli random variable $\beta_i \sim Ber(q_i)$ independently. Here $\beta_i \in \{0, 1\}$ denotes an indicator that depicts each individual's participation in the dataset.

Definition 4 (Poisson-Sampled Gaussian (PoiSG) mechanism)**.** Let $D \in \mathbb{D}$ be an input dataset and $\mathbf{q} = \{q_1, \ldots, q_N\}$ denote the set of sampling probabilities of each record $d_i \in D$. Consider a function $f : \mathbb{D} \to \mathbb{O}$ with ℓ_2-sensitivity L, then the Poisson-Sampled Gaussian (PoiSG) mechanism is defined as:

$$PoiSG_{\mathbf{q},\sigma}(D) \triangleq f(S) + \zeta, \quad \zeta \sim \mathcal{N}(0, \sigma^2 L^2),$$

where each element $d_i \in S \subseteq D$ is selected via Poisson sampling, and $\mathcal{N}(0, \sigma^2 L^2)$ is a Gaussian distribution with standard deviation σL. Note that we assume $L = 1$ throughout the rest of this paper.

3.2 Rényi Differential Privacy

Rényi differential privacy (RDP) utilizes the asymmetric measure of Rényi divergence to quantify the privacy guarantee. Note that with a controlling parameter $\alpha \neq 1$, the Rényi divergence of order α from distribution Q to P is:

$$D_\alpha(P\|Q) \triangleq \frac{1}{\alpha - 1} \log \mathbb{E}_{o\sim Q} \left[\left(\frac{P(o)}{Q(o)} \right)^\alpha \right].$$

Let $P = \mathcal{A}(D)$ and $Q = \mathcal{A}(D')$, then (α, ρ)-RDP is achieved by simultaneously bounding the Rényi divergence of two directions, denoted by $D^{\leftrightarrow}_\alpha(P\|Q) \triangleq \max\{D_\alpha(P\|Q), D_\alpha(Q\|P)\}$.

Definition 5 ((α, ρ)-Rényi Differential Privacy [43])**.** A randomized mechanism $\mathcal{A}$ satisfies (α, ρ)-RDP with order $\alpha \in (1, \infty)$ if for any pair of adjacent datasets $D, D' \in \mathbb{D}$, it holds that

$$D^{\leftrightarrow}_\alpha(\mathcal{A}(D)\|\mathcal{A}(D')) \leq \rho. \tag{1}$$

Different from the traditional (ε, δ)-DP, which measures privacy leakage by utilizing the max divergence of two output distributions, RDP allows for a continuous spectrum of privacy measures. More specifically, as $\alpha \to \infty$, $D_\infty(\mathcal{A}(D)\|\mathcal{A}(D'))$ is equal to the max divergence [9]; and $\lim_{\alpha\to 1} D_\alpha(\mathcal{A}(D)\|\mathcal{A}(D'))$ can be verified to be equal to the expected value of the privacy cost random variable $c(o; \mathcal{A}, D, D') \triangleq \ln \frac{\Pr[\mathcal{A}(D)\in o]}{\Pr[\mathcal{A}(D')\in o]}$ [43]. This characteristic enables RDP to provide a sharper privacy quantification and become one of the most popular privacy analysis tools, especially adept at handling composite mechanisms like differentially private stochastic gradient descent (DP-SGD) [1].

We provide the following useful lemmas which are important primitives for the design of our FL algorithm and privacy analysis.

Lemma 1 (Transition from RDP to DP [43]). *If $\mathcal{A}$ is an (α, ρ)-RDP mechanism, it also satisfies $(\rho + \frac{\log 1/\delta}{\alpha-1}, \delta)$-DP for any $0 < \delta < 1$.*

Lemma 2 (Adaptive sequential composition [43]). *If $\mathcal{A}_1 : \mathbb{D} \to \mathbb{O}_1$ is (α, ρ_1)-RDP and $\mathcal{A}_2 : \mathbb{D} \times \mathbb{O}_1 \to \mathbb{O}_2$ is (α, ρ_2)-RDP, then the composed mechanism $\mathcal{A} \triangleq \mathcal{A}_1 \circ \mathcal{A}_2 : \mathbb{D} \to \mathbb{O}_1 \times \mathbb{O}_2$ satisfies $(\alpha, \rho_1 + \rho_2)$-RDP.*

Lemma 3 (Post-processing [43]). *If $\mathcal{A}$ is (α, ρ)-RDP and $\mathcal{F} : \mathbb{O} \to \mathbb{O}'$ is an arbitrary data-independent randomized mapping, then $\mathcal{F} \circ \mathcal{A}$ is (α, ρ)-RDP.*

Lemma 4 (Privacy amplification via (uniform) Poisson sampling for Gaussian mechanism [44, 60]). *Consider a PoiSG mechanism and a uniform sampling probability q among all records. For all pairs of adjacent datasets D, D' and integer $\alpha > 1$, we have*[2]

$$\rho_{PoiSG}(\alpha, q) \leq \frac{1}{\alpha - 1} \log \Big\{ (1-q)^{\alpha-1}(\alpha q - q + 1) + \sum_{\ell=2}^{\alpha} \binom{\alpha}{\ell} (1-q)^{\alpha-\ell} q^\ell e^{(\ell-1)\rho(\ell)} \Big\}. \tag{2}$$

Note that $\rho(\alpha) = \frac{\alpha}{2\sigma^2}$ for any $\alpha > 1$.

Privacy Bounds Visualization. The privacy guarantee under RDP can be depicted as a curve of Rényi divergence, aka., the *RDP budget curve*, over the continuous range of α values [43]. For a clear understanding, we visualize the RDP budget curve of the PoiSG mechanism with uniform sampling probability q in Figure 2

[2]Note that [44] and [60] demonstrated similar RDP upper bounds for the PoiSG mechanism. The presented Lemma 3 is mainly rooted in the findings of [60]. Specifically, when we work with the Gaussian mechanism, Proposition 10 [60] implies Theorem 8 holds and the lower bound in Theorem 6 is a tighter RDP upper bound for the PoiSG mechanism.

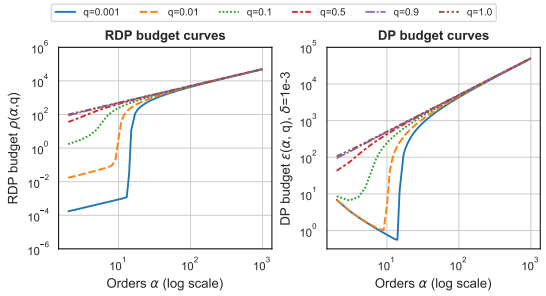

Figure 2: The RDP and DP budget curves w.r.t. order α and sampling probability q of a sequential combination of T=100 PoiSG mechanisms with noise multiplier σ=1.0 and δ=1e-3.

(left). When $q = 1$, the PoiSG mechanism is reduced to a standard Gaussian mechanism whose RDP budget curve is a straight line [43]. For the RDP budget curves with $q < 1$, there exists a phase transition that happens around $\alpha q e^{\rho(\alpha)} \approx q^{-1}$ [60]. As q gets larger, this transition tends to appear earlier and get more indistinct.

Based on Lemma 1, we can obtain the corresponding *DP budget curve* given a desired δ and then find the smallest ε by solving the optimization problem below [1]:

$$\varepsilon^* \triangleq \min_{\alpha} \left\{ \rho + \frac{\log(1/\delta)}{\alpha - 1} \right\}. \tag{3}$$

Corollary 38 in [52] proves the unimodality/quasi-convexity of this optimization problem. Figure 2 (right) demonstrates the existence of an optimal order α^*, corresponding to the minimum ε^*.

Remark 2. Instead of exploring an infinite range of real numbers $\alpha \in (1, \infty)$, practitioners often opt to predefine a finite collection of RDP orders to effectively capture the minimum ε^*. This trick has been implemented in leading DP libraries such as Opacus[3], Tensorflow Privacy[4], etc.

4 FEDERATED LEARNING WITH RECORD-LEVEL PERSONALIZED DP

We target the typical supervised FL task with a central server and a set of M clients $\mathcal{C} = \{C_1, \ldots, C_M\}$. Consider each client $C_i \in \mathcal{C}$ holds a private training dataset $D_i = \{d_{i,1}, \ldots, d_{i,N_i}\}$. Each record $d_{i,j} \in D_i$ is associated with a privacy budget $\varepsilon_{i,j} > 0$, which reflects the privacy preference of the record's owner. Our goal is to learn a globally shared model with parameters $\mathbf{x} \in \mathbb{R}^d$ by solving the following empirical risk problem

$$\min_{\mathbf{x} \in \mathbb{R}^d} \left\{ \mathcal{L}(\mathbf{x}) \triangleq \frac{1}{M} \sum_{i=1}^{M} \mathcal{L}_i(\mathbf{x}; D_i) \right\}, \tag{4}$$

$$\text{where } \mathcal{L}_i(\mathbf{x}; D_i) \triangleq \frac{1}{N_i} \sum_{j=1}^{N_i} l(\mathbf{x}, d_{i,j}), \tag{5}$$

with the privacy guarantee of record-level personalized differential privacy (as stated in Definition 6 below). Here $l(\cdot)$ denotes the loss function used for local optimization.

[3]https://github.com/pytorch/opacus
[4]https://github.com/tensorflow/privacy

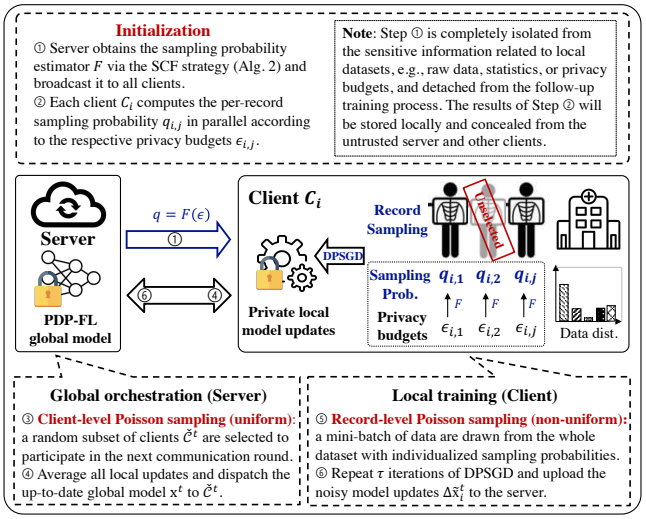

Figure 3: A step-by-step illustration of the rPDP-FL algorithm.

Definition 6 (Federated Learning with Record-level Personalized Differential Privacy). Given $\delta \geq 0$. Let $\mathcal{D} = \bigcup_{i=1}^{M} D_i$ and $\mathcal{E} = \bigcup_{i=1}^{M} \{\varepsilon_{i,j}\}_{j \in [N_i]}$. A randomized FL algorithm $\mathcal{A}_{FL} : \mathbb{D} \to \mathbb{O}$ satisfies $(\mathcal{E}, \delta)$-record-level personalized differential privacy (rPDP) if it guarantees $(\varepsilon_{i,j}, \delta)$-DP w.r.t. the specific record $d_{i,j}$, i.e., for any pair of record-specific adjacent datasets $\mathcal{D} \overset{d_{i,j}}{\sim} \mathcal{D}_{-i,j}$ and any subsets of output $o \subseteq \mathbb{O}$, it holds that

$$\Pr[\mathcal{A}_{FL}(\mathcal{D}) \in o] \leq e^{\varepsilon_{i,j}} \Pr[\mathcal{A}_{FL}(\mathcal{D}_{-i,j}) \in o] + \delta, \tag{6}$$

where $\mathcal{D}_{-i,j} \triangleq D_{i,-j} \cup \{\cup_{m \neq j} D_m\}$ and $D_{i,-j} \triangleq D_i \setminus \{d_{i,j}\}$.

FedAvg and Two-stage Sampling Scheme. The most fundamental approach for solving the non-private optimization problem in Eq. (4) is federated averaging (FedAvg) [41]. Despite numerous improved methods being proposed to tackle FedAvg's limitations like data heterogeneity or communication efficiency [19, 24, 30, 31], most of them still adhere to a *two-stage sampling* scheme (i.e., outer client sampling followed by the inner record sampling) together with SGD-based learning paradigm. Given the primary aim of this work is the establishment of a record-level personalized privacy protection mechanism, we choose FedAvg as the backbone of our FL framework and adopt DP-SGD [1] during each client's local training process to achieve record-level protection. We expect our proposed rPDP-FL to be extendable to work with other two-stage sampling FL methods listed above.

4.1 Solution Overview

We employ a two-stage *hybrid* sampling scheme within the FedAvg algorithm to obtain a global model using clients' local datasets while ensuring diverse individual privacy preferences. This innovative framework for private FL, termed *rPDP-FL*, differs from the classic approach in three key aspects:

- **Initialization**: Each client allocates a customized sampling probability $q_{i,j}$ to every record in its local dataset, tailored to the record's specific privacy budget.

Algorithm 1: Record-level Personalized Differentially Private Federated Learning (rPDP-FL, Pseudocode)

input : M clients with their local datasets $(D_1, \ldots, D_M)$; the total communication round T and the local SGD step τ; the client-level sampling probability λ.

```
// Initialization
foreach client C_i ∈ C do in parallel
    {q_{i,j}}_{j∈[|D_i|]} ← (pre-computation of sampling probabilities for all records)
for t ∈ [T] do
    // Client-level Poisson sampling with the uniform sampling probability λ
    Č^t ← (a random subset drawn from [M])
    foreach client C_i ∈ Č^t do in parallel
        for r ∈ [τ] do
            // Record-level Poisson sampling with non-uniform sampling probabilities {q_{i,j}}_{j∈[|D_i|]}
            S^r ← (a random mini-batch drawn from D_i)
            // Differentially private SGD
    // The central server averages the collected noisy model updates and obtains the updated global model parameters
```

- **Stage 1: Client-level Poisson sampling (uniform)**: at the beginning of round $t \in [T]$, the central server selects a random subset of clients $\check{\mathcal{C}}^t$ via Poisson sampling with *uniform* per-client sampling probability $\lambda \in [0, 1]$ and dispatches the up-to-date global model $\mathbf{x}^t$ to these selected clients.
- **Stage 2: Record-level Poisson sampling (non-uniform)**: each client selected in the above stage performs a certain number of DP-SGD iterations locally and independently and uploads the model updates to the central server. During each iteration, the mini-batches are drawn from the whole local dataset via Poisson sampling with non-uniform per-record sampling probabilities.

It's worth highlighting that rPDP-FL solely alters the sampling processes (except for the initialization step) and remains detached from the intricacies associated with the learning process. This feature enables its broader applicability to any non-private FL frameworks that incorporate a two-stage sampling process, as illustrated in Figure 3. The pseudocode is presented in Algorithm 1 and the complete version will be shown in Algorithm 3.

4.2 Challenges

To offer reasonable personalized privacy guarantees while maintaining the utility of the global model, the development of Algorithm 1 faces a *dual* challenge in both theory and practice.

- **The privacy analysis challenge.** From a theoretical perspective, it is essential to establish as "tight" upper bounds as possible for the overall privacy cost of each individual to enhance the trade-off between privacy and utility.
- **The parameter estimation challenge.** For practical purposes, an efficient and effective parameter estimation strategy must be adopted to select appropriate hyperparameters for the privacy algorithm, i.e., determining sampling probabilities for all records.

Our research aims to explore the mathematical relationship between privacy cost and their sampling probabilities. In particular, we provide the privacy analysis given the sampling probability of each record, as detailed in Section 5. Furthermore, we outline our approach for deriving the sampling probabilities in accordance with the predetermined individualized privacy budgets in Section 6.

5 PRIVACY ANALYSIS

5.1 Privacy Objectives and Key Results

We will analyze the upper bound of the accumulative privacy cost for any single record $d_{i,j}$ in Algorithm 1, assuming its sampling probability $q_{i,j}$ is given. This can be broken down into the following three basic nested routines:

(1) *Local multi-step update*, which can be abstracted as an adaptive combination of τ PoiSG mechanisms.
(2) *Global parameter aggregation*, which can be seen as a multi-phase procedure involving uniform client sampling and data-independent post-processing of the results derived from the local update performed by the chosen clients.
(3) *Global multi-round update*, which can be perceived as an adaptive combination of T parameter aggregation mechanisms above.

Symbolic Representations. Without loss of generality, our focus will be primarily on the first record $d_{1,1} \in D_1$ of client C_1. For the sake of conciseness, we use the symbolic representations as follows.

- $\mathcal{D} \triangleq \bigcup_{i=1}^{M} D_i$: the federated dataset.
- $\mathcal{D}, \mathcal{D}_{-1,1}$: the adjacent *federated* datasets concerning a specific data record $d_{1,1}$, i.e., $\mathcal{D} \overset{d_{1,1}}{\sim} \mathcal{D}_{-1,1}$.
- $D_1, D_{1,-1}$: the adjacent *local* datasets at client C_1 concerning a specific data record $d_{1,1}$, i.e., $D_1 \overset{d_{1,1}}{\sim} D_{1,-1}$.
- $\mathrm{CSamp}_\lambda(D_1, \ldots, D_M)$: client sampling using uniform Poisson sampling, where $\lambda \in (0, 1]$ denotes the sampling probabilities for all clients.
- $\mathrm{RSamp}_{\mathbf{q}_1}(D_1)$: record sampling at client C_1 using non-uniform Poisson sampling, where $\mathbf{q}_1 = \{q_{1,1}, \ldots, q_{1,|D_1|}\}$ denotes the set of sampling probabilities of each record $d_{1,j} \in D_1$.
- $\mathcal{A}_G(\cdot) \triangleq f(\cdot) + \mathcal{N}(0, \sigma^2\mathbb{I})$: the Gaussian mechanism satisfying (α, ρ_G)-RDP, where $\rho_G(\alpha) \triangleq \frac{\alpha L^2}{2\sigma^2}$ and L is the the sensitivity of function f [43]. For simplicity, we assume that $L = 1$ through the rest of this section.
- $\mathcal{A}(\cdot) \triangleq \mathcal{A}_G(\mathrm{RSamp}_{\mathbf{q}_1}(\cdot))$: the PoiSG mechanism.
- $\mathcal{A}_{in}(\cdot) \triangleq \mathcal{A}^{\otimes\tau} = (\mathcal{A}_1(\cdot), \mathcal{A}_2(\mathcal{A}_1(\cdot), \cdot), \ldots, \mathcal{A}_\tau(\mathcal{A}_1(\cdot), \ldots, \cdot))$: local multi-step update.
- $\mathcal{A}_{out}(\cdot) \triangleq \mathrm{Avg}(\mathrm{CSamp}_\lambda \circ \mathcal{A}_{in})$: global parameter aggregation.
- $\mathcal{A}_{FL}(\cdot) \triangleq \mathcal{A}_{out}(\cdot)^{\otimes t}$: global multi-round update.

To enjoy the strength of tight privacy accounting offered by the RDP privacy analysis framework, we need to overcome the incompatibility challenge that all existing PDP techniques fail to provide tight privacy analysis under the RDP framework. In detail, we consider analyzing the RDP bound of $\mathcal{A}_{FL}$ first, and then convert it into the form of a standard DP guarantee by applying Lemma 1. For example, in order to show that $\mathcal{A}_{FL}$ satisfies $(\varepsilon_{1,1}, \delta)$-DP w.r.t. $d_{1,1}$, we need to show that for any pair of adjacent federated datasets $\mathcal{D}, \mathcal{D}_{-1,1}$ and arbitrary output o, we have

$$D(\mathcal{A}_{FL}(\mathcal{D}) \| \mathcal{A}_{FL}(\mathcal{D}_{-1,1})) \leq \rho_{FL},$$

$$\text{s.t.} \quad \min_{\alpha>1}\left\{\rho_{FL}+\frac{\log(1/\delta)}{\alpha-1}\right\} \le \varepsilon_{1,1}. \tag{7}$$

Privacy Objectives. As previously discussed in Section 2, FL scenarios typically account for two distinct types of potential adversaries. Consequently, the objectives of privacy analysis can be categorized into the following.

(1) *Type I privacy analysis* against the honest-but-curious server (which has access to the intermediate model updates): given the local model parameter $\mathbf{x}_1^t \sim \mathcal{A}_{in}(D_1)$ uploaded by Client C_1, the individual RDP privacy bound of $\mathcal{A}_{FL}$ for record $d_{1,1}$ is

$$D_\alpha(\mathcal{A}_{FL}(\mathcal{D})\|\mathcal{A}_{FL}(\mathcal{D}_{-1,1})) = \lambda T \cdot D_\alpha(\mathcal{A}_{in}(D_1)\|\mathcal{A}_{in}(D_{1,-1})) \le \rho_I.$$

(2) *Type II privacy analysis* against untrusted clients or third parties (which have access to the intermediate or final global model): for any global model parameter $\mathbf{x}^t \sim \mathcal{A}_{FL}(\mathcal{D})$, the individual RDP privacy bound of $\mathcal{A}_{FL}$ for record $d_{1,1}$ is

$$D_\alpha(\mathcal{A}_{FL}(\mathcal{D})\|\mathcal{A}_{FL}(\mathcal{D}_{-1,1})) = T \cdot D_\alpha(\mathcal{A}_{out}(\mathcal{D})\|\mathcal{A}_{out}(\mathcal{D}_{-1,1})) \le \rho_{II}.$$

Key Results. The key results are presented below. The detailed proofs will be presented in the next subsection.

Lemma 5. *Suppose that $\check{C}^t$ is a subset of clients selected at round $t \in [T]$. The simple average operation $Avg(\cdot)$ over all outputs $o_i \sim \mathcal{A}_{in}(D_i)$, where $C_i \in \check{C}^t$, will not incur any extra privacy cost to all records $d_{i,j} \in \mathcal{D}$.*

Proof. The proof follows from the fact that the RDP guarantee is preserved under post-processing, as shown in Lemma 3. □

Lemma 6 (Enhanced Privacy Amplification by Two-Stage Hybrid Sampling). *Assume the sampling probability for any clients is $\lambda \in (0,1]$, and the sampling probability for data record $d_{i,j} \in D_i$ is $q_{i,j} \in (0,1]$. If a random algorithm $\mathcal{A}_{in}(D_i, \mathbf{x}^{t-1})$ satisfies $(\alpha, \rho_{i,j}^\tau)$-RDP w.r.t $d_{i,j}$, then the algorithm $\mathcal{A}_{out}(\mathcal{D})$ satisfies $(\alpha, \rho_{i,j}^{\tau,\lambda})$-RDP w.r.t. $d_{i,j}$, where*

$$\rho_{i,j}^{\tau,\lambda}(\alpha, q_{i,j}) \le \frac{1}{\alpha-1}\ln\left\{1-\lambda+\lambda e^{(\alpha-1)\rho_{i,j}^\tau(\alpha,q_{i,j})}\right\}, \text{ and}$$

$$\rho_{i,j}^{\tau}(\alpha, q_{i,j}) \le \frac{\tau}{\alpha-1}\ln\Big\{(1-q_{i,j})^{\alpha-1}(\alpha q_{i,j}-q_{i,j}+1) + \sum_{\ell=2}^{\alpha}\binom{\alpha}{\ell}(1-q_{i,j})^{\alpha-\ell}q_{i,j}^\ell e^{(\ell-1)\rho_G(\ell)}\Big\}.$$

Theorem 1 (Individual Privacy Analysis in Federated Learning). *For any $\delta \in (0,1)$, the random algorithm $\mathcal{A}_{FL}(\mathcal{D})$ satisfies $(\hat{\varepsilon}_{i,j}^*, \delta)$-DP w.r.t. a specific record $d_{i,j} \in \mathcal{D}$, where*

$$\hat{\varepsilon}_{i,j}^* \triangleq \min_\alpha\left(\rho_{FL}(\alpha, q_{i,j})+\frac{\ln(1/\delta)}{\alpha-1}\right). \tag{8}$$

Note that: (1) for untrusted clients or third parties, $\rho_{FL}(\alpha, q_{i,j}) \triangleq T\rho_{i,j}^{\tau,\lambda}(\alpha, q_{i,j})$; (2) for the honest-but-curious server, $\rho_{FL}(\alpha, q_{i,j}) \triangleq \lambda T\rho_{i,j}^{\tau}(\alpha, q_{i,j})$.

5.2 Detailed Proofs

We first use a special case to explain the enhanced privacy effects at different stages and extend the conclusion to more general scenarios. Consider that a server collaborates with two clients, C_1 and C_2, to collectively train an FL model. Here C_2 is assumed to be an adversary and aims to infer whether $d_{1,1}$ is contained in D_1.

Considering the sequential composition of RDP as in Lemma 2, our major objective will be analyzing the increment of the individual RDP parameter between two successive rounds, that is,

$$D(\mathcal{A}_{out}(\mathcal{D})\|\mathcal{A}_{out}(\mathcal{D}_{-1,1})) \triangleq \frac{1}{\alpha-1}\log\mathbb{E}_{o\sim\mathcal{A}_{out}(\mathcal{D}_{-1,1})}\left[\left(\frac{\Pr[\mathcal{A}_{out}(\mathcal{D})\in o]}{\Pr[\mathcal{A}_{out}(\mathcal{D}_{-1,1})\in o]}\right)^\alpha\right].$$

5.2.1 Local Multi-step Update. According to Fact 1 below which aligns with the decentralized nature of FL, the privacy analysis of the local multi-step update process essentially follows the existing theoretical results based on the typical DP-SGD algorithm. The only difference is that now we need to characterize the privacy cost for each record since the sampling probabilities of the records are different from each other.

Fact 1. *Once the initial model parameters $\mathbf{x}^{t-1}$ are fixed at the beginning of round $t \in [T]$, each client performs local update independently, i.e., the distribution of the output $\mathbf{x}_i^t \sim \mathcal{A}_{in}(\mathbf{x}^{t-1}, D_i)$ $(i = 2, \ldots, M)$ is independent of any data records in D_1.*

For the local dataset D_1 with a size of N, let $\mathbf{s} = (s_1, \ldots, s_N) \subseteq \{0,1\}^N$ be the indicator vector of the record sampling outcome, i.e., $s_j = 1$ if $d_{1,j}$ is selected[5]. It is evident the probability that $\mathbf{s}$ appears is $p_\mathbf{s} = \prod_{j=1}^N (q_j)^{s_j}(1-q_j)^{1-s_j}$ and the total number of possible values of $\mathbf{s}$ is 2^N. For example, if $N = 3$ and $\mathbf{s} = [1,0,1]$, then $p_\mathbf{s} = q_1(1-q_2)q_3$ and total number of possible values of $\mathbf{s}$ is 8. Then for any pair of adjacent local datasets $D_1, D_{1,-1}$ and any subsets of output $o \subseteq \mathbb{O}$, the output distributions of a single DP-SGD step can be represented as:

$$\begin{aligned}\Pr[\mathcal{A}(D_1)\in o] &= \sum_{\mathbf{s}} p_\mathbf{s}\Pr[\mathcal{A}_G(\mathbf{s})\in o] \\ &= (1-q_1)\sum_{\mathbf{s}:s_1=0} p_\mathbf{s}\Pr[\mathcal{A}_G(\mathbf{s})\in o|s_1=0] \\ &\quad + q_1\sum_{\mathbf{s}:s_1=1} p_\mathbf{s}\Pr[\mathcal{A}_G(\mathbf{s})\in o|s_1=1] \\ \Pr[\mathcal{A}(D_{1,-1})\in o] &= \sum_{\mathbf{s}:s_1=0} p_\mathbf{s}\Pr[\mathcal{A}_G(\mathbf{s})\in o|s_1=0]\end{aligned}$$

As the local multi-step update process $\mathcal{A}_{in}$ can be viewed as a τ-fold adaptive composition of a PoiSG mechanism, we have the following Lemma 7 by directly applying the existing RDP composition and amplification results as shown in Lemma 2 and Lemma 4. The distinction lies in the privacy guarantee provided by $\mathcal{A}_{in}$ is specific to individual records, instead of being established on the wider scope of the adjacent datasets D_1 and D_1'.

Lemma 7. *For any client C_i, if the sampling probability of a specific record $d_{i,j} \in D_i$ is $q_{i,j} \in (0,1]$, then the local multi-step update process $\mathcal{A}_{in}(D_i)$ satisfies $(\alpha, \rho_{i,j}^\tau)$-RDP w.r.t. $d_{i,j}$, where*

$$\rho_{i,j}^\tau \triangleq \tau \cdot \rho_{PoiSG}(\alpha; D_i, D_{i,-j}, q_{i,j}) \tag{9}$$

[5]For notational convenience, we suppress the dependence on the client identifier i.

5.2.2 Global Parameter Aggregation. We consider the output distribution of $\mathcal{A}_{out}$ on the adjacent federated datasets $\mathcal{D}, \mathcal{D}_{-1,1}$ in the context of the above special case. Let $P_i \triangleq \Pr[\mathcal{A}_{in}(D_i) \in o_i]$ $(i = 1, 2)$ and $P'_1 \triangleq \Pr[\mathcal{A}_{in}(D_{1,-1}) \in o_1]$. It can be observed that for the federated dataset $\mathcal{D}$, the underlying distribution $\Phi \triangleq \Pr[\mathcal{A}_{out}(\mathcal{D}) \in o]$ can be represented as

- a mixture of P_1 and P_2, denoted as H_{11}, if both C_1 and C_2 are selected;
- the same as P_1, denoted as H_{10}, if only C_1 is selected;
- the same as P_2, denoted as H_{01}, if only C_2 is selected;
- independent of both, denoted as H_{00}, if neither C_1 nor C_2 is selected.

Similarly, the distribution $\Psi \triangleq \Pr[\mathcal{A}_{out}(\mathcal{D}_{-1,1}) \in o]$ will be

- a mixture of P'_1 and P_2, denoted as H'_{11}, if both C_1 and C_2 are selected;
- the same as P'_1, denoted as H'_{10}, if only C_1 is selected;
- the same as H_{01}, if only C_2 is selected;
- the same as H_{00}, if neither C_1 nor C_2 is selected.

Remark 3. Note that here we focus on the individual privacy cost for PDP which is measured on all pairs of record-level adjacent datasets w.r.t. the "target" record $d_{1,1}$. In the context of traditional (uniform) DP, we cannot simply assume the output from other clients $o_2, \ldots, o_M$ are constants when analyzing the impact of an individual record on the worst-case privacy cost, as the output distribution of $\text{Avg}(\cdot)$ is highly dependent on each record in $\mathcal{D}$.

Let $\omega \subseteq (\omega_1, \ldots, \omega_M) \in \{0, 1\}^M$ be the indicator vector of the outcome of client sampling and $\omega_i = 1$ denotes that Client C_i is selected. Then we have

$$\begin{aligned}\Phi &= \lambda(1-\lambda)H_{10} + \lambda^2 H_{11} + (1-\lambda)^2 H_{00} + (1-\lambda)\lambda H_{01},\\ &= \sum_{\omega:\omega_1=1} p_\omega H_\omega + \sum_{\omega:\omega_1=0} p_\omega H_\omega.\\ \Psi &= \lambda(1-\lambda)H'_{10} + \lambda^2 H'_{11} + (1-\lambda)^2 H_{00} + (1-\lambda)\lambda H_{01}.\\ &= \sum_{\omega:\omega_1=1} p_\omega H'_\omega + \sum_{\omega:\omega_1=0} p_\omega H_\omega.\end{aligned}$$

Now we try to bound $\mathbb{E}_\Psi\left[(\Phi/\Psi)^\alpha\right]$ by means of decomposition and simplification. More specifically, we have

$$\begin{aligned}&\mathbb{E}_\Psi\left[(\Phi/\Psi)^\alpha\right] \overset{(1)}{\leq} \mathbb{E}_\omega \mathbb{E}_\Psi\left[(\Phi/\Psi)^\alpha|\omega\right]\\ &= \mathbb{E}_{\omega_2}\left\{\lambda\mathbb{E}_{H'_\omega}\left[(H_\omega/H'_\omega)^\alpha|\omega_1=1\right] + (1-\lambda)\mathbb{E}_{H'_\omega}\left[(H_\omega/H'_\omega)^\alpha|\omega_1=0\right]\right\}\\ &= \lambda\mathbb{E}_{H'_\omega}\left[(H_\omega/H'_\omega)^\alpha|\omega_1=1\right] + (1-\lambda)\\ &\overset{(2)}{\leq} \lambda\mathbb{E}_{P'_1}\left[(P_1/P'_1)^\alpha\right] + (1-\lambda)\\ &= \lambda e^{(\alpha-1)D_\alpha(\mathcal{A}_{in}(D_1)\|\mathcal{A}_{in}(D_{1,-1}))} + (1-\lambda)\\ &\leq \lambda e^{(\alpha-1)\rho^\tau_{1,1}} + (1-\lambda),\end{aligned}$$

where the inequality (1) follows from the Jensen's inequality and Lemma 22 in [52] which proves bivariate function $f(x, y) = x^\alpha/y^{\alpha-1}$ is jointly convex on $\mathbb{R}^2_+$ for all $\alpha > 1$; the inequality (2) follows from Lemma 5 which implies that the individual privacy guarantee for $d_{1,1}$ is immune to post-processing, i.e.,

$$D_\alpha(H_\omega\|H'_\omega) \leq D_\alpha(\mathcal{A}_{in}(D_1)\|\mathcal{A}_{in}(D_{1,-1})) \leq \rho^\tau_{i,j}.$$

The equality holds if each client's sampling probability $\lambda = 1$. Similarly, we can also have

$$\mathbb{E}_\Phi\left[(\Psi/\Phi)^\alpha\right] \leq \lambda e^{(\alpha-1)D_\alpha(\mathcal{A}_{in}(D_{1,-1})\|\mathcal{A}_{in}(D_1))} + (1-\lambda)$$

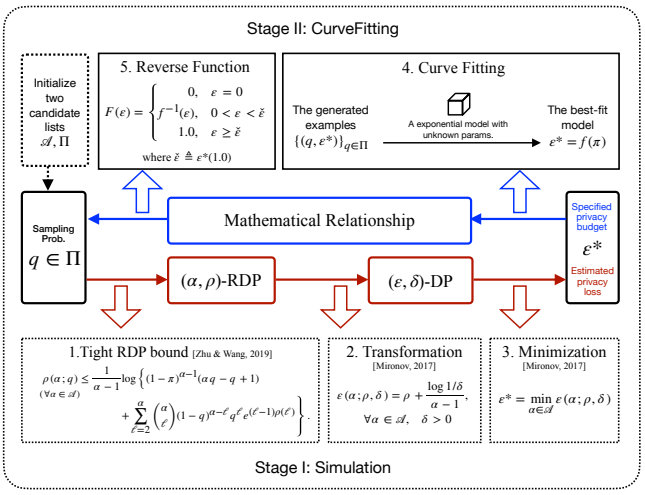

Figure 4: An illustration of Simulation-CurveFitting.

$$\leq \lambda e^{(\alpha-1)\rho^\tau_{1,1}} + (1-\lambda).$$

Now we have completed the proof of Lemma 6.

5.2.3 Global Multi-round Update. Putting all the pieces together, the proof of Theorem 1 can be further derived by leveraging the adaptive composition theorem of RDP as shown in Lemma 2.

6 SELECTING SAMPLING PROBABILITY

In this section, we explore how to select a sampling probability for every single record to achieve an estimated privacy cost that closely aligns with the desired privacy budget, on the condition that the other factors (e.g., T, τ, σ, and δ) remain constant.

6.1 Simulation-CurveFitting

Given the theoretical result in Theorem 1, it would be ideal if we could directly derive a sampling probability for each record given its predetermined personalized privacy budget. Yet, it is non-trivial to derive an explicit closed-form expression due to the complexity arising from the optimization process and the highly nonlinear nature of the tight bound ρ_{FL}. Existing approaches utilize numerical methods to handle this absence of closed-form issue and approximately obtain the sampling probability, e.g., by binary search algorithm [3]. However, these strategies become computationally demanding when applied to our case. We introduce a new and effective strategy termed *Simulation-CurveFitting* (SCF). As the name suggests, this approach consists of the following two stages, and the specific steps are illustrated in Figure 4 and outlined in Algorithm 2.

Stage I: Simulation. We aim to elucidate the relationship between q and ε[6] through a series of simulation experiments. In the beginning, we establish two finite sets of candidate values: the first consists of various sampling probabilities, which we refer to as Π; the second consists of a sequence of discrete RDP orders denoted as $\mathbb{A}$. For each $q \in \Pi$, we compute the DP budget curves and then find the corresponding minimum value ε^*. For illustration, we show a series of DP budget curves for the rPDP-FL algorithm in Figure 5

[6]For notational convenience, we suppress the dependence on the record identifier i, j.

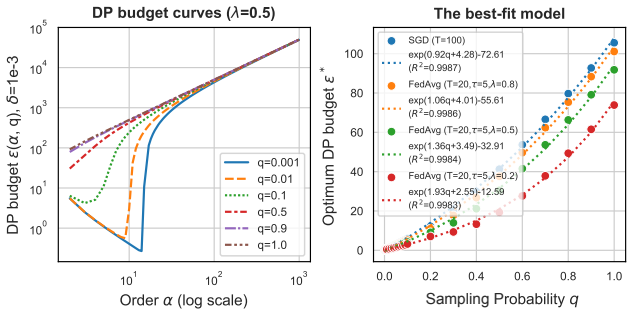

Figure 5: The DP budget curves w.r.t. order α (left) and the optimum DP budget w.r.t. sampling probability q (right) of a rPDP-FL algorithm with parameters T=20, τ=5, λ=0.5, σ=1.0, and δ=1e-3.

(left[7]), corresponding to varying values of q. The minimum value on each curve is then used to plot the optimum DP budget w.r.t. the corresponding sampling probability on the right figure. The pseudocode is illustrated in lines 3-6 of Algorithm 2.

Stage II: CurveFitting. In Figure 5 (right), we depict the one-to-one correspondence between the optimum DP budgets and their respective sampling probabilities (represented as a series of dots) across various parameter settings (represented by different colors). This visualization reveals a compelling observation – *a mathematical function may potentially model the tight upper bound on the accumulative privacy cost w.r.t. its sampling probability.* Armed with this insight, we employ curve-fitting tools[8] to approximate the correlation between q and ε. The best-fit solution obtained is a simple exponential function in the form

$$\hat{\varepsilon}^* \approx f(q) \triangleq e^{a \cdot q + b} + c,$$

as stated in Line 7 of Algorithm 2. Our best-fit function is more concise and elegant than the one given in Eq.(8) in Theorem 1, allowing inverse computation of the sampling probability q given the privacy cost. We refer to the inverse function as *sampling probability estimator*, denoted as $F(\varepsilon)$, which takes a privacy budget $\varepsilon > 0$ as input and outputs a valid sampling probability $q \in [0, 1]$. In particular, if an input ε exceeds the optimum DP budget corresponding to q=1.0, denoted as $\varepsilon^*(1.0)$, the output probability q is projected to be 1.0. See Line 8 of Algorithm 2 for more details.

Measures for goodness-of-fit. We utilize the R^2 value (also known as the coefficient of determination) to quantify how well the estimated privacy cost by the curve fitting function matches the privacy cost derived from the privacy accounting, ranging from 0 (no correlation) to 1 (perfect positive correlation). As per the empirical results illustrated in Figure 5 (right), the best-fit model exhibits an R^2 value exceeding 99%. This demonstrates strong evidence of the model's ability to derive the sampling probability for each record based on their desired privacy budgets.

[7]This figure closely resembles Figure 2 (right) which depicts results obtained in centralized settings. However, a key distinction is that the minimum values ε^* across all the curves in Figure 2 (right) are consistently greater than those in Figure 5 (left). This highlights the enhanced effect of privacy amplification resulting from the client-level sampling procedure in FL framework.
[8]SciPy: https://scipy.org/.

Algorithm 2: The Simulation-CurveFitting (SCF) strategy

input : The noise multiplier σ, the gradient clipping bound L, and the target DP parameter δ.
output : The sampling probability estimator
// Initialize two candidate lists of α, q
1 $\mathbb{A} \leftarrow$ a candidate list of RDP order $\alpha \in (1, \infty)$
2 $\Pi \leftarrow$ a candidate list of sampling probability $q \in [0, 1]$
3 **foreach** $q \in \Pi$ **do**
  // Numerical simulation analysis of PoiSGM with sampling probability q
4 $\rho_{FL}(\alpha, q) \leftarrow$ (the RDP budget curve w.r.t. order $\alpha \in \mathbb{A}$ calculated based on Theorem 1)
5 $\varepsilon(\alpha, \delta, q) = \rho_{FL}(\alpha, q) + \frac{\log 1/\delta}{\alpha - 1} \leftarrow$ (the DP budget curve w.r.t. order $\alpha \in \mathbb{A}$ calculated based on Lemma 1)
6 $\varepsilon^*(\delta, q) = \min_{\alpha \in \mathbb{A}} \varepsilon(\alpha, \delta, q) \leftarrow$ (the optimum DP budget w.r.t. sampling probability q)
// Curve fitting
7 $f(q) \leftarrow$ (the best-fit mathematical model to the generated observations $\{(q, \varepsilon^*)\}_{q \in \Pi}$)
// The sampling probability estimator
8

$$F = \begin{cases} f^{-1}(\varepsilon), & 0 < \varepsilon < \varepsilon^*(1.0) \\ 1.0, & \varepsilon \geq \varepsilon^*(1.0) \end{cases}$$

return F

6.2 Complete Algorithm of rPDP-FL

The SCF strategy has been further integrated into Algorithm 1, with some tweaks in the initialization and sampling procedures. A comprehensive outline of rPDP-FL is provided in Algorithm 3.

The per-record sampling probability initialization. At the onset of the learning process, the server will compute the sampling probability estimator F and distribute it to all clients. Note that this computation does not rely on personal data from sensitive records stored locally, so there is no risk of compromising the privacy of these records. On the client side, the per-record sampling probabilities will be calculated by directly plugging in their privacy budgets $\varepsilon_{i,j}$ into the received sampling probability estimator. All clients then employ non-uniform Poisson sampling to randomly select a subset of records based on these probabilities, and apply the DP-SGD algorithm [1] for local model updates.

The per-record privacy budget accountant. Another important task for completing rPDP-FL is to keep track of the usage of the privacy budget for each of the records in the course of training. Once the privacy budget runs out, individuals can opt out of the remaining training. In our work, we introduce a monitoring module, called the *budget accountant*, which is in charge of privacy budget accounting: (1) *Pre-check* at the beginning of the communication round if an individual has sufficient privacy budget to participate in the current round; (2) *Compute and update* the accumulated privacy cost of an individual after the current communication round is over.

6.3 Discussions

Generalization of the SCF strategy. A distinct characteristic of the SCF strategy lies in its independence from the inherent complexities of specific processes. While this paper primarily focuses on FL

Algorithm 3: Record-level Personalized Differentially Private Federated Learning (rPDP-FL, complete version)

input : M clients with their local datasets $D_{i\in[M]}$ and pre-specified privacy budgets $\{\varepsilon_{i,j}\}_{i\in|D_i|,j\in[M]}$; the total communication rounds T, the client-level sampling probability λ. Parameters shared by all clients: the local training steps τ; the learning rate η; the gradient clipping bound L, the noise multiplier σ and the target DP parameter δ.

// Initialization

$C \leftarrow$ (all participating clients with size M)

// Pre-computation

$F \leftarrow$ (the sampling probability estimator obtained through Alg. 2)

foreach $C_i \in C$ **do** *in parallel*

  $\{q_{i,j} = F(\varepsilon_{i,j})\}_{j\in[|D_i|]} \leftarrow$ (the per-record sampling probabilities)

$\mathbf{x}^0 \leftarrow$ (Initialize randomly)

for $t \in [T]$ **do**

  // Client-level Poisson sampling with the uniform sampling probability λ

  $\check{C}^t \leftarrow$ (a random subset drawn from M clients)

  foreach $C_i \in \check{C}^t$ **do** *in parallel*

    $\mathbf{x}_i^{t,0} = \mathbf{x}^t$

    for $r \in [\tau]$ **do**

      // Record-level Poisson sampling with the derived sampling probability $\{q_{i,j}\}_{j\in[|D_i|]}$

      $S^r \leftarrow$ (a random mini-batch drawn from D_i)

      foreach *microbatch* $\xi \in S^r$ **do**

        $\tilde{\mathbf{g}}_\xi^r \leftarrow \nabla\ell(\mathbf{x}_i^{t,r};\xi) \cdot \min(1, \frac{L}{\|\nabla\ell(\mathbf{x}_i^{t,r};\xi)\|_2})$

      $\tilde{\mathbf{g}}^r \leftarrow \frac{\eta}{|S^r|}\left(\sum \tilde{\mathbf{g}}_\xi^r + \mathcal{N}(0, \sigma^2 L^2)\right)$

      $\mathbf{x}_i^{t,r+1} \leftarrow \mathbf{x}_i^{t,r} - \eta\tilde{\mathbf{g}}^r$

    $\Delta\mathbf{x}_i = \mathbf{x}_i^{t,\tau} - \mathbf{x}_i^{t,0}$

  $\mathbf{x}^{t+1} \leftarrow$ (taking the average of all the local updates)

applications, the SCF approach serves as a versatile plug-in module applicable to a broad spectrum of tasks that incorporate data sampling and Gaussian mechanism, including private statistical analysis and other SGD-based optimization tasks.

Sampling/Noise trade-offs. In highly privacy-sensitive scenarios, the majority of individuals (e.g., patients) prefer stronger privacy protections, and thus their sensitive personal records are less likely to be included in analysis. We argue that this issue is not about our methodology itself, but an inevitable consequence of the personal privacy decision. One remedy is to adjust the parameters of the Gaussian mechanism, for example, setting a larger noise multiplier σ such that a higher level of Gaussian noise is used in the computation, resulting in universally increased sampling probabilities for everyone. In Figure 6, we illustrate the relationship between optimum DP budgets and their corresponding q (represented as a series of dots) across various σ (represented by different colors). Essentially, there is a trade-off between sampling probability and perturbation noise. However, this approach should be treated with extreme caution, as an improper σ could lead to a significant degradation in model performance.

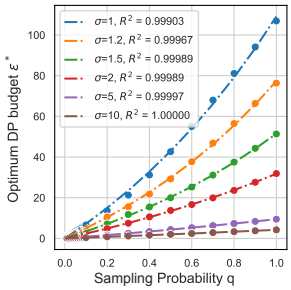

Figure 6: The optimum DP budget w.r.t. sampling probability q and noise multiplier σ of a rPDP-FL algorithm with parameters T=20, τ=5, and δ=1e-5.

The utility risks of the potentially suboptimal F. As a result of the intrinsic traits of the numerical simulation method, the obtained sampling probability estimator F could be suboptimal and lead to utility risks for certain records, e.g., slightly unused budget (due to low sampling probability) or early stop (due to high sampling probability). However, both cases have statistically insignificant effects since the best-fit curve achieves more than 99.9% R^2 value as shown in Figure 6.

7 EXPERIMENTAL EVALUATION

In this section, we conduct a thorough empirical analysis to evaluate the performance of both the SCF strategy and the rPDP-FL algorithm. In Subsection 7.1, we focus on evaluating the effectiveness of SCF by comparing it with the other two existing strategies employed for achieving PDP in centralized ML settings. Given the absence of alternative implementations achieving record-level PDP in the context of FL, we assess the utility improvement of rPDP-FL by contrasting it with two conventional methods that do not incorporate personalized privacy preservation in Subsection 7.2. The source code, data, and other artifacts have been made available[9].

Privacy Preference Distributions. We simulate different scenarios where users have diverse privacy preferences for their data.

- **ThreeLevels**: each record has the option to select a preferred privacy budget from three distinct choices (categories) $\{\varepsilon_1, \varepsilon_2, \varepsilon_3\}$ where $\varepsilon_1 < \varepsilon_2 < \varepsilon_3$, each denoting strong, moderate, and weak privacy protection, respectively. This simulates practical PDP implementation scenarios where the users can choose from a few predefined privacy categories.
- **BoundedPareto**: each record has an arbitrary $\varepsilon_{i,j} \in [0.1, 10]$ that approximately follows a Pareto distribution. This simulates the scenario where a majority of individuals lean towards stringent privacy safeguards, while a smaller subset opts for less restrictive protections in exchange for improved services or other incentives.
- **BoundedMixGauss**: each record has an arbitrary $\varepsilon_{i,j} \in [0.1, 10]$ that approximately follows a mixture of three Gaussian distributions with means $\{\varepsilon_1, \varepsilon_2, \varepsilon_3\}$ where $\varepsilon_1 < \varepsilon_2 < \varepsilon_3$. This simulates the scenarios where the privacy choices are multi-modal as in many other complex social systems [48, 49, 55].

Datasets and Models. We consider four classification tasks with the consistent objective of training a global model privately on the

[9] https://github.com/Emory-AIMS/rPDP-FL.git

Table 1: Overview of the datasets and baseline models used in our experiments.

Dataset	Non-/IID	# clients	# features	# labels	# examples (per client)	Train/Test split	Model	# params (trainable)	Training steps (local τ / global T)	Client-level samp. prob. λ
Heart-Disease [50]	Non-IID	4	13	2	303 / 261 / 46 / 130	66% / 34%	Logistic Regression (training from scratch)	20	10 / 15	1.0
MNIST [28]	Non-/IID	10	28×28×1	10	≈ 6,000	66% / 34%	Two-Layer CNN (training from scratch)	26,010	50 / 15	0.5
CIFAR10 [27]	IID	10	32×32×3	10	5,000	66% / 34%	ResNet-18 [20] (training from scratch)	11,181,642	50 / 30	0.5
SNLI [4]	IID	10	Premise-hypothesis pairs	3	54,936	95% / 5%	Pretrained BERT [7] (fine-tuning)	7,680,771	50 / 15	0.5

federated *Heart Disease* [50], *MNIST* [28], *CIFAR10* [27], and *SNLI* [4], separately. Note that *Heart-Disease* is a real healthcare dataset comprising records from 920 patients across four hospitals in Cleveland, Hungary, Switzerland, and Long Beach V. On the other hand, *MNIST* and *CIFAR10* are two commonly used benchmarks for image classification tasks, while the *SNLI* dataset is a benchmark for natural language inference (NLP) tasks. In these cases, we apply the IID and non-IID partitioning strategies introduced in [41] to split total training examples into $M = 10$ subsets. For a more comprehensive overview of the datasets, along with details on the corresponding baseline models, please refer to Table 1.

Implementations. Our implementation utilizes the Opacus library. All experiments are conducted on a machine with one NVIDIA A40 GPU running on Ubuntu with 256 GB memory. Given that the model training is a randomized process, we repeat all the experiments five times and report the mean test accuracy across all clients.

7.1 Comparison of SCF with Existing Strategies

In this section, we show the effectiveness and efficiency of our SCF strategy in terms of model utility and computational cost by comparing it with the following representative approaches:

- **Filter**: also known as Rényi privacy filter (Algorithm 3 in [12]), is an individual privacy accounting method that monitors the accumulation of *squared gradient norms* B_{norm} for each record during the training process. The record will be filtered out if this accumulation exceeds a pre-specified threshold.
- **BinarySearch**: also known as Individual DP-SGD (IDP-SGD) with the Sample mechanism (Algorithm 2 in [3]), is a *binary search*-based approach aiming for finding the optimal sampling probability within the range of $[0, 1]$ for a target privacy budget.

Given that both approaches were initially tailored for *centralized* ML scenarios, our experiments adhere to this context to maintain fairness in comparisons. Specifically, we implement a variant of the DP-SGD [1] algorithm, incorporating refinements in the predetermination of record-level sampling probabilities through the SCF strategy. Note that here the RDP budget curve (line 4 in Alg. 2) should be calculated based on Lemma 4 instead of Theorem 1.

7.1.1 Comparison of SCF with Filter. While both *SCF* and *Filter* share a common objective of achieving personalized privacy protection, they significantly differ in the definition of "budgets": *Filter* considers a budget for the accumulative squared gradient norms for the records in the training process, while we focus on the DP privacy budget ε for the accumulative privacy cost. Achieving a smooth transition between the two quantities is challenging[10]. Therefore, our evaluation only focuses on a simplified personalized privacy scenario *ThreeLevels*, where the percentage of records with ε_1, ε_2, and ε_3 is 70%, 20%, and 10%, respectively. We incorporate *Filter* into the private gradient descent algorithm[11] as implemented in [12]. Then we assess the model utility and computational efficiency of both approaches on the pooled Heart-Disease and MNIST datasets.

Table 2: Comparison of SCF with Filter in centralized ML.

Priv. Pref. Dist.	Dataset	Method	Batch Size	Test Acc	Runtime (s/step)
ThreeLevels:	Heart-Disease (pooled)	Filter	486	0.7717	0.0128
$\varepsilon_1 \approx 2.0$ (70%)		SCF	32 (Expected)	**0.8189**	**0.0050**
$\varepsilon_2 \approx 4.7$ (20%)	MNIST (pooled)	Filter	60000	0.8411	7.1016
$\varepsilon_3 \approx 11.8$ (10%)		SCF	64 (Expected)	**0.9477**	**2.1824**

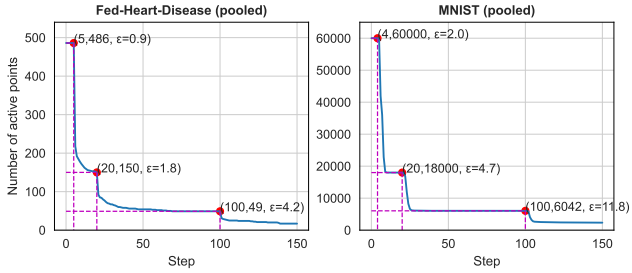

Figure 7: Number of active records during one run of private GD with Filter evaluated on the pooled Fed-Heart-Disease dataset (left) and MNIST (right). The red points indicate the specific steps that records with different privacy budgets start to get filtered.

Number of active records. In Figure 7, we plot the number of active records during one run of private GD with *Filter*, i.e. those that have not yet exhausted their privacy budgets. The results reveal that in scenarios where personalized privacy protection is implemented, particularly when a significant portion of the records adheres to conservative privacy preferences, a considerable number of records undergo early filtration in the training process, which

[10]We highlight that the final privacy budget ε depends on the sampling probability q, noise multiplier σ, and the number of training steps τ. However, in *Filter*, different records could have varying values of τ even though they share the same "squared norm budget" B_{norm}. As a result, adjusting the hyperparameters of B_{norm} to align with a specific ε for each record, and vice versa, becomes challenging, particularly in general personalized privacy scenarios like BoundedMixGauss or BoundedPareto.

[11]It computes noisy gradients using the entire training dataset in each iteration, which is different from DP-SGD.

Table 3: Comparison of SCF with BinarySearch in centralized ML.

Priv. Pref. Dist.	Method	Test Acc	Runtime (s)
Group-3: 3 unevenly sized privacy groups (54%-37%-9%) with privacy budgets [1., 2., 3.]	BinarySearch	0.7274	**1.69**
	SCF	0.7240	13.11
Group-100: 100 evenly sized privacy groups with privacy budgets [1., 1.05, ..., 5.95]	BinarySearch	0.8135	52.50
	SCF	0.8134	**13.32**
Individual-1000: per-record privacy budgets drawn from BoundedMixGauss	BinarySearch	0.6858	597.82
	SCF	0.6861	**14.10**

restricts the model's capacity to learn from the dataset effectively. As shown in Table 2, our experimental results demonstrate that our *SCF* outperforms *Filter* in terms of test accuracy.

Computational efficiency and privacy amplification. Due to the requisite computation of per-example gradient norms at each iteration, *Filter* no longer retains the benefits of SGD for improving speed and memory efficiency [12]. In contrast, the Poisson sampling procedure involved in our approach could yield mini-batches of size $\sum_{j\in|D|} q_j < |D|$ in expectation. We report the average time cost for each training step in Table 2, revealing SCF achieves a 2x or 3x speedup compared to Filter in our assessments. Moreover, the absence of the random data subsampling procedure in private GD with Filter also leads to the loss of privacy amplification effect for individual records.

7.1.2 Comparison of SCF with BinarySearch. As discussed in Section 2, BinarySearch faces a significant limitation in terms of efficiency when dealing with records' privacy budgets that cover values within a continuous range, rather than a discrete subset. To illustrate this, we carry out a small-scale experiment where a private model is trained on a subset of 1,000 examples randomly selected from the MNIST dataset using DP-SGD. Three different personalized privacy scenarios are being investigated to evaluate the efficiency and utility of BinarySearch and SCF, as outlined in Table 3. Note that both Group-3 and Group-100 are privacy setups concerned in [3], where all records are split into limited privacy groups and those within one privacy group share the same privacy budget. The Individual-1000 represents the privacy scenario considered in this paper, where the privacy budgets are assigned on an individual basis. We report the test accuracies and the runtimes to obtain the per-record sampling probabilities (averaged over 5 trials). Our experiment results demonstrate that SCF significantly enhances efficiency compared to BinarySeach in the more general scenarios while achieving a comparable model performance. Here we did not evaluate BinarySearch in a parallel manner as this implementation was not discussed in [3] and is not the focus of this paper.

Remark 4. Boenisch et al. [3] also proposed a Scale mechanism that aims at scaling the noise added to each gradient by setting individualized clipping bounds. The optimal clipping bound is still found in a binary-search style. According to Table 16-17 in [3], Scale leads to a comparable overall performance with Sample (regarding runtime and test accuracy). Thus, both Sample and Scale share similar limitations against our method.

7.2 Utility Improvement through Privacy Personalization

In this section, we evaluate the effectiveness of our rPDP-FL algorithm in providing record-level personalized DP while optimizing for model accuracy. Given the absence of alternative implementations achieving record-level PDP in the FL setting, we established three DP-FedAvg-based baseline methods as follows:

- **Minimum**: DP-FedAvg ensuring the most stringent *uniform* $(\varepsilon_{min}, \delta)$-DP guarantees, where $\varepsilon_{min} = \min_{i\in[M], j\in[|D_i|]} \varepsilon_{i,j}$.
- **Dropout**: DP-FedAvg providing moderate *uniform* $(\varepsilon_{mod}, \delta)$-DP guarantees for a subset of individuals whose privacy budgets $\varepsilon_{i,j}$ are larger than a predefined threshold ε_{mod} and dropping out those with privacy budgets below the threshold. Here, ε_{mod} is determined as the empirical sample mean of $\frac{1}{|\mathcal{D}|}\sum_{i,j}\varepsilon_{i,j}$.
- **PrivacyFree**: The vanilla FedAvg [41] without DP guarantees, which serves as a benchmark for assessing the reduction in global model utility.

Note that both *Minimum* and *Dropout* ensure the accumulative privacy costs of all records remain at/below their specified privacy budgets throughout the training process but lead to a significant waste for records with large privacy budgets. All three types of privacy preference distributions are considered for a comprehensive evaluation of all methods' performance.

Unless otherwise specified, the default setup of *ThreeLevels* comprises $\varepsilon_1 = 0.1$, $\varepsilon_2 = 1.0$ and $\varepsilon_3 = 5.0$ with corresponding proportion of 70%, 20%, and 10%. Similarly, the *BoundedMixGauss* distribution is defined as a mixture of $\mathcal{N}_1(0.1, 0.01)$, $\mathcal{N}_2(1.0, 0.05)$ and $\mathcal{N}_3(5.0, 0.5)$ with weights of 0.7, 0.2 and 0.1. For the *BoundedPareto* case, we consider a specific Pareto distribution with a shape value of 1.0 and a lower bound of 0.1. These parameters are chosen to simulate realistic scenarios where the majority of users have strict privacy requirements. Examples of 1,000 records' privacy preferences are illustrated in the first column of Figure 8. In the context of training a larger ResNet-18 model from scratch on the CIFAR-10 dataset, where the total number of trainable parameters exceeds 11M, we opt for more liberal privacy settings to maintain acceptable model utility. Specifically, we set $\varepsilon_1 = 1.0$, $\varepsilon_2 = 3.0$, and $\varepsilon_3 = 10.0$ for both the *ThreeLevels* and *BoundedMixGauss* setups while keeping the ratios the same. For the *BoundedPareto* distribution, the lower bound is adjusted to 1.0.

Hyperparameters. For the Fed-Heart-Disease experiments, we fix the per-client sampling probability λ=1.0 and δ=1e-3; for the other experiments, we fix λ=0.5 and δ=1e-4. The total communication rounds T and the number of local training steps per round τ for different tasks are detailed in Table 1. We explore a variety of constant learning rates (e.g., [0.1, 0.05, 0.01, 0.005, 0.001]) and clipping thresholds (e.g., [0.5, 1.0, 3.0, 5.0]) for the best results.

Model Utility. The evaluation results under various privacy preference distributions, as shown in Figure 8, consistently demonstrate the superiority of our *rPDP-FL* method over both *Minimum* and *Dropout* across diverse FL datasets. Specifically, the advantages over the *Minimum* method suggest that the model benefits from records with larger privacy budgets, while its advantages over the *Dropout* method imply that even records with conservative privacy budgets contribute positively to the training process. These findings underscore the crucial role of personalized privacy integration in

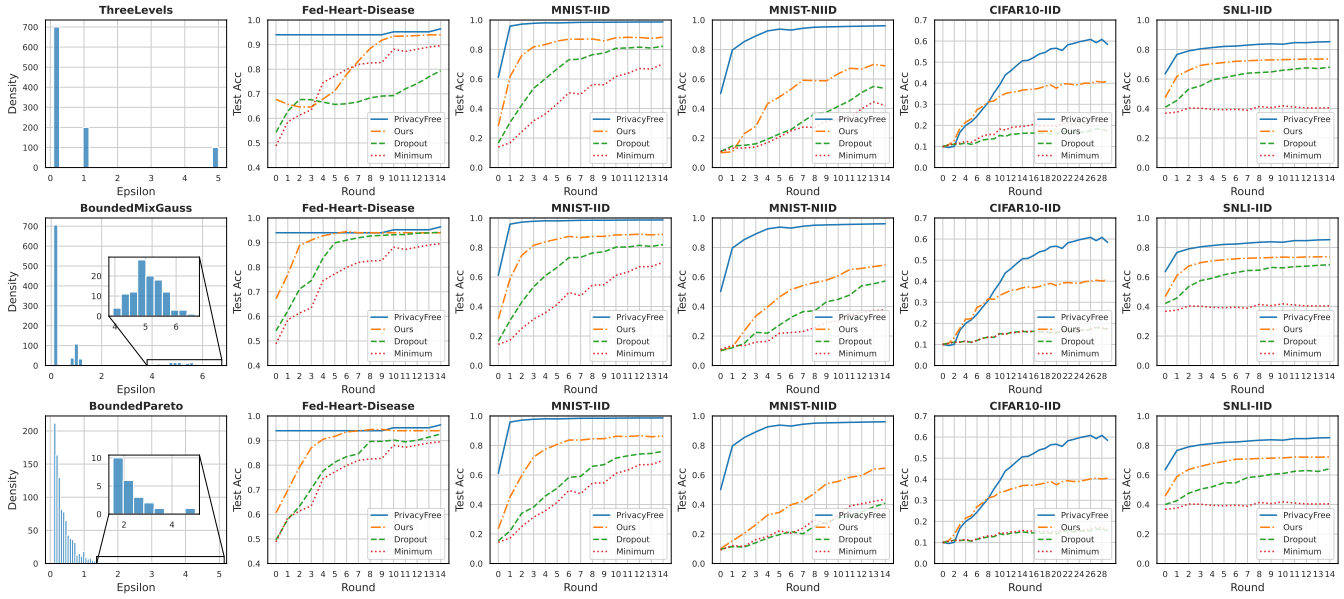

Figure 8: Evaluation of rPDP-FL (labeled as *Ours*) across diverse privacy preference distributions and datasets in federated learning.

achieving a more favorable trade-off between privacy and utility, particularly in scenarios where most individuals exhibit a strong emphasis on privacy concerns.

Upon comparing the results between MNIST-IID and MNIST-NIID, we observed a consistent reduction in test accuracy in the non-IID case across *Minimum*, *Dropout*, and our proposed method. This decline arises from the inherent limitations of the *DP-FedAvg* algorithm under a heterogeneous environment [45]. Despite this deterioration, it's noteworthy that the utility advantages stemming from incorporating privacy personalization remain.

8 CONCLUSION AND FUTURE WORK

In this paper, we studied an unexplored real-world challenge to enable record-level personalized differential privacy in federated learning. Our proposed solution is a novel framework called *rPDP-FL*, which employs a *two-stage hybrid sampling* scheme with non-uniform record-level sampling. We devise an efficient strategy, referred to as *Simulation-CurveFitting* (SCF), to estimate the individual sampling probabilities for all records associated with varying privacy budgets. Our investigation uncovers a valuable insight regarding rPDP-FL, i.e., a one-to-one correspondence between the sampling probabilities of records and their accumulative privacy costs which can be mathematically expressed through a simple exponential function. Empirical demonstrations show that our solutions yield significant performance enhancements compared to baselines that overlook personalized privacy preservation.

As an early exploration into FL with personalized privacy protection, our work lays a foundation for future in-depth investigations and highlights several promising directions, such as:

- **User-level (device-level) PDP in cross-silo (cross-device) FL.** In FL scenarios, individual users (or devices) might possess multiple records or contribute data to various clients simultaneously [25]. Intuitively, a single user may have distinct privacy preferences for their records, which poses significant challenges in establishing user-level personalized privacy protection.
- **Effective learning on non-IID data with data-dependent privacy budgets.** In Appendix A, we offer a brief discussion on a more complex scenario where the privacy budgets of users are directly linked to their raw data (e.g., dependent on their labels). Our findings point out the deficiencies in the current privacy personalization methods to yield substantial utility gain for groups that have significantly smaller privacy budgets and are a minority in the population. We highlight the need to address this challenge effectively in future research endeavors.

ACKNOWLEDGMENTS

We would like to thank the reviewers for their thoughtful comments and efforts toward improving our paper and artifacts. This work was supported in part by the National Natural Science Foundation of China grants (62172423, 62206207, 62102352, U23A20306), National Institutes of Health grants (R01ES033241, R01LM013712), National Science Foundation grants (CNS-2124104, CNS-2125530, IIS-2302968), and National Key Research and Development Program of China grant (2021YFB3101100).

REFERENCES

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep learning with differential privacy. In *CCS*. 308–318.
[2] Borja Balle, Gilles Barthe, and Marco Gaboardi. 2018. Privacy amplification by subsampling: Tight analyses via couplings and divergences. In *NeurIPS*. 6280–6290.
[3] Franziska Boenisch, Christopher Mühl, Adam Dziedzic, Roy Rinberg, and Nicolas Papernot. 2023. Have it your way: Individualized Privacy Assignment for DP-SGD. In *NeurIPS*. 19073–19103.
[4] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In

EMNLP. 632–642.
[5] Rui Chen, Haoran Li, A Kai Qin, Shiva Prasad Kasiviswanathan, and Hongxia Jin. 2016. Private spatial data aggregation in the local setting. In *ICDE*. 289–300.
[6] Rui Chen, Qiyu Wan, Xinyue Zhang, Xiaoqi Qin, Yanzhao Hou, Di Wang, Xin Fu, and Miao Pan. 2023. EEFL: High-Speed Wireless Communications Inspired Energy Efficient Federated Learning over Mobile Devices. In *MobiSys*. 544–556.
[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*. 4171–4186.
[8] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating noise to sensitivity in private data analysis. In *TCC*. 265–284.
[9] Cynthia Dwork, Aaron Roth, et al. 2014. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science* 9, 3-4 (2014), 211–407.
[10] Cynthia Dwork and Guy N Rothblum. 2016. Concentrated differential privacy. *arXiv:1603.01887* (2016).
[11] Hamid Ebadi, David Sands, and Gerardo Schneider. 2015. Differential privacy: Now it's getting personal. *ACM SIGPLAN Notices* 50, 1 (2015), 69–81.
[12] Vitaly Feldman and Tijana Zrnic. 2021. Individual privacy accounting via a Rényi filter. In *NeurIPS*. 28080–28091.
[13] Jie Fu, Yuan Hong, Xinpeng Ling, Leixia Wang, Xun Ran, Zhiyu Sun, Wendy Hui Wang, Zhili Chen, and Yang Cao. 2024. Differentially private federated learning: A systematic review. *arXiv:2405.08299* (2024).
[14] Jie Fu, Qingqing Ye, Haibo Hu, Zhili Chen, Lulu Wang, Kuncan Wang, and Xun Ran. 2024. DPSUR: Accelerating Differentially Private Stochastic Gradient Descent Using Selective Update and Release. *Proc. VLDB Endow.* 17, 6 (2024), 1200–1213.
[15] Dawei Gao, Daoyuan Chen, Zitao Li, Yuexiang Xie, Xuchen Pan, Yaliang Li, Bolin Ding, and Jingren Zhou. 2023. FS-Real: A Real-World Cross-Device Federated Learning Platform. *Proc. VLDB Endow.* 16, 12 (2023), 4046–4049.
[16] Robin C Geyer, Tassilo Klein, and Moin Nabi. 2017. Differentially private federated learning: A client level perspective. *arXiv:1712.07557* (2017).
[17] Antonious Girgis, Deepesh Data, Suhas Diggavi, Peter Kairouz, and Ananda Theertha Suresh. 2021. Shuffled Model of Differential Privacy in Federated Learning. In *AISTATS*. 2521–2529.
[18] Ian J Goodfellow, Mehdi Mirza, Da Xiao, Aaron Courville, and Yoshua Bengio. 2013. An empirical investigation of catastrophic forgetting in gradient-based neural networks. *arXiv:1312.6211* (2013).
[19] Filip Hanzely, Slavomír Hanzely, Samuel Horváth, and Peter Richtárik. 2020. Lower bounds and optimal algorithms for personalized federated learning. In *NeurIPS*. 2304–2315.
[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*. 770–778.
[21] Briland Hitaj, Giuseppe Ateniese, and Fernando Perez-Cruz. 2017. Deep models under the GAN: information leakage from collaborative deep learning. In *CCS*. 603–618.
[22] Zach Jorgensen, Ting Yu, and Graham Cormode. 2015. Conservative or liberal? Personalized differential privacy. In *ICDE*. 1023–1034.
[23] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. 2021. Advances and open problems in federated learning. *Foundations and trends® in machine learning* 14, 1–2 (2021), 1–210.
[24] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. 2020. Scaffold: Stochastic controlled averaging for federated learning. In *ICML*. 5132–5143.
[25] Fumiyuki Kato, Li Xiong, Shun Takagi, Yang Cao, and Masatoshi Yoshikawa. 2023. ULDP-FL: Federated Learning with Across Silo User-Level Differential Privacy. *arXiv:2308.12210* (2023).
[26] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. 2017. Overcoming catastrophic forgetting in neural networks. *PNAS* 114, 13 (2017), 3521–3526.
[27] Alex Krizhevsky, Geoffrey Hinton, et al. 2009. Learning multiple layers of features from tiny images. (2009), 1–58.
[28] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324.
[29] Ninghui Li, Wahbeh Qardaji, and Dong Su. 2012. On sampling, anonymization, and differential privacy or, k-anonymization meets differential privacy. In *ASIACCS*. 32–33.
[30] Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. 2021. Ditto: Fair and robust federated learning through personalization. In *ICML*. 6357–6368.
[31] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. 2020. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems* 2 (2020), 429–450.
[32] Zitao Li, Tianhao Wang, and Ninghui Li. 2023. Differentially Private Vertical Federated Clustering. *Proc. VLDB Endow.* 16, 6 (2023), 1277–1290.
[33] Junxu Liu, Jian Lou, Li Xiong, Jinfei Liu, and Xiaofeng Meng. 2021. Projected federated averaging with heterogeneous differential privacy. *Proceedings of the VLDB Endowment* 15, 4 (2021), 828–840.
[34] Shang Liu, Yang Cao, Takao Murakami, Weiran Liu, Seng Pei Liew, Tsubasa Takahashi, Jinfei Liu, and Masatoshi Yoshikawa. 2024. Federated graph analytics with differential privacy. *arXiv:2405.20576* (2024).
[35] Yixuan Liu, Suyun Zhao, Li Xiong, Yuhan Liu, and Hong Chen. 2023. Echo of neighbors: privacy amplification for personalized private federated learning with shuffle model. In *AAAI*. 2008–2020.
[36] Ziyu Liu, Shengyuan Hu, Zhiwei Steven Wu, and Virginia Smith. 2022. On Privacy and Personalization in Cross-Silo Federated Learning. In *NeurIPS*. 5925–5940.
[37] Andrew Lowy, Ali Ghafelebashi, and Meisam Razaviyayn. 2023. Private Non-Convex Federated Learning Without a Trusted Server. In *AISTATS*. 5749–5786.
[38] Jing Ma, Qiuchen Zhang, Jian Lou, Li Xiong, and Joyce C Ho. 2021. Communication efficient federated generalized tensor factorization for collaborative health data analytics. In *Proceedings of the Web Conference*. 171–182.
[39] Samuel Maddock, Graham Cormode, Tianhao Wang, Carsten Maple, and Somesh Jha. 2022. Federated Boosted Decision Trees with Differential Privacy. In *CCS*. 2249–2263.
[40] Saber Malekmohammadi, Yaoliang Yu, and Yang Cao. 2024. Noise-Aware Algorithm for Heterogeneous Differentially Private Federated Learning. *arXiv:2406.03519* (2024).
[41] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *AISTATS*. 1273–1282.
[42] H Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. 2017. Learning differentially private recurrent language models. *arXiv:1710.06963* (2017).
[43] Ilya Mironov. 2017. Rényi differential privacy. In *CSF*. 263–275.
[44] Ilya Mironov, Kunal Talwar, and Li Zhang. 2019. Rényi differential privacy of the sampled Gaussian mechanism. *arXiv:1908.10530* (2019).
[45] Maxence Noble, Aurélien Bellet, and Aymeric Dieuleveut. 2022. Differentially private federated learning on heterogeneous data. In *AISTATS*. 10110–10145.
[46] Maria Rigaki and Sebastian Garcia. 2023. A survey of privacy attacks in machine learning. *Comput. Surveys* 56, 4 (2023), 1–34.
[47] Ryan M Rogers, Aaron Roth, Jonathan Ullman, and Salil Vadhan. 2016. Privacy odometers and filters: Pay-as-you-go composition. In *NeurIPS*. 1929–1937.
[48] Enrico Scalas, Rudolf Gorenflo, Hugh Luckock, Francesco Mainardi, Maurizio Mantelli, and Marco Raberto. 2004. Anomalous waiting times in high-frequency financial data. *Quantitative Finance* 4, 6 (2004), 695–702.
[49] Enrico Scalas, Rudolf Gorenflo, Hugh Luckock, Francesco Mainardi, Maurizio Mantelli, and Marco Raberto. 2005. On the intertrade waiting-time distribution. *Finance Letters* 3, 1 (2005), 38–43.
[50] Jean Ogier du Terrail, Samy-Safwan Ayed, Edwige Cyffers, Felix Grimberg, Chaoyang He, Regis Loeb, Paul Mangold, Tanguy Marchand, Othmane Marfoq, Erum Mushtaq, et al. 2022. Flamby: Datasets and benchmarks for cross-silo federated learning in realistic healthcare settings. *arXiv:2210.04620* (2022).
[51] Vale Tolpegin, Stacey Truex, Mehmet Emre Gursoy, and Ling Liu. 2020. Data poisoning attacks against federated learning systems. In *ESORICS*. 480–501.
[52] Yu-Xiang Wang, Borja Balle, and Shiva Prasad Kasiviswanathan. 2019. Subsampled Rényi Differential Privacy and Analytical Moments Accountant. In *AISTATS*. 2521–2529.
[53] Zhibo Wang, Mengkai Song, Zhifei Zhang, Yang Song, Qian Wang, and Hairong Qi. 2019. Beyond inferring class representatives: User-level privacy leakage from federated learning. In *INFOCOM*. 2512–2520.
[54] Kang Wei, Jun Li, Ming Ding, Chuan Ma, Howard H Yang, Farhad Farokhi, Shi Jin, Tony QS Quek, and H Vincent Poor. 2020. Federated learning with differential privacy: Algorithms and performance analysis. *TIFS* 15 (2020), 3454–3469.
[55] Ye Wu, Changsong Zhou, Jinghua Xiao, Jürgen Kurths, and Hans Joachim Schellnhuber. 2010. Evidence for a bimodal distribution in human communication. *PANS* 107, 44 (2010), 18803–18808.
[56] Zihang Xiang, Tianhao Wang, Wanyu Lin, and Di Wang. 2023. Practical differentially private and byzantine-resilient federated learning. *Proceedings of the ACM on Management of Data* 1, 2 (2023), 1–26.
[57] Xiaoyu Zhang, Chao Chen, Yi Xie, Xiaofeng Chen, Jun Zhang, and Yang Xiang. 2023. A survey on privacy inference attacks and defenses in cloud-based deep neural network. *Computer Standards & Interfaces* 83 (2023), 103672.
[58] Xiaoyu Zhang, Shen Lin, Chao Chen, and Xiaofeng Chen. 2023. MODA: Model Ownership Deprivation Attack in Asynchronous Federated Learning. *IEEE Transactions on Dependable and Secure Computing* (2023).
[59] Ligeng Zhu, Zhijian Liu, and Song Han. 2019. Deep leakage from gradients. In *NeurIPS*. 14774–14784.
[60] Yuqing Zhu and Yu-Xiang Wang. 2019. Poission subsampled Rényi differential privacy. In *ICML*. 7634–7642.

Table 4: Evaluation of the per-class test accuracy of a CNN model trained on MNIST with data-dependent privacy budgets for 100 iterations.

MNIST	0	1	2	3	4	5	6	7	8	9	Overall
Budget (ε)	0.5	0.75	2.0	2.6	4.1	2.1	2.05	3.0	3.1	6.1	-
rPDP-SGD	93.49	95.9	87.45	**90.83**	**95.1**	**90.37**	**94.74**	**92.13**	**88.62**	**92.64**	**92.55**
DP-SGD (ε=0.5)	93.63	95.81	87.26	87.18	89.67	85.89	93.73	88.71	85.74	82.19	89.08
DP-SGD (ε=3.0)	**94.24**	**96.09**	**89.75**	89.71	92.01	89.37	94.54	90.5	84.78	85.2	90.69
Vanilla SGD	99.1	99.6	97.11	98.33	99.1	98.65	98.27	99.07	95.87	94.1	98.06

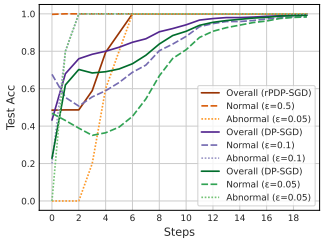

Figure 9: Evaluation of the per-class test accuracy of a logistic regression model trained on the Heart-Disease dataset with data-dependent privacy budgets (ε=0.5 for normal patients and ε=0.05 for abnormal patients) for 20 iterations.

A ADDITIONAL EVALUATION RESULTS

Learning with data-dependent privacy bdugets. In this study, we focus on a general personalized privacy scenario in which individual privacy budgets follow a random distribution, independent of other factors such as the raw data. Nonetheless, there are scenarios where individuals with particular attributes (or labels) might exhibit different privacy concerns. For example, heart disease patients might demand more stringent privacy safeguards when their health records are utilized in training ML models.

To examine the utility improvement through privacy personalization on different groups, we carry out preliminary experiments by training centralized ML models on the Heart-Disease and MNIST datasets. In each experiment, we apply DP-SGD and a variant of DP-SGD that incorporates the SCF strategy for achieving rPDP, which we denoted as *rPDP-SGD*. We allocate distinct privacy budgets to each class. For the Heart-Disease dataset, "normal" patients are assigned a privacy budget of ε=0.5, whereas "abnormal" records are given a more conservative budget of ε=0.05. For the MNIST dataset, we adhere to the setup described in [3].

In Figure 9, we visualize per-class and overall test accuracy (averaged over 10 trials) for the logistic regression model trained on the Heart-Disease dataset. Due to the inherent simplicity of the dataset, both *rPDP-SGD* and baselines (ε=0.05 and 0.1 for all classes) achieve convergence to perfect accuracy (1.0) within 20 iterations. In this experiment, we do not observe a discernible utility gain of rPDP-SGD compared to the baselines for the "abnormal" records (as indicated by the three dotted lines in the figure). This lack of utility improvement could stem from the unbalancedness of data distribution, together with the fact that "abnormal" records are sampled less frequently than "normal" records, causing the model to primarily learn from "normal" records during the initial phases of training.

Table 4 displays the final test accuracy (averaged over 5 trials) for 10 evenly sized classes of the MNIST dataset. It can be observed that for classes with privacy budgets below 3.0 (Classes 3, 5, 6, and 7), their test accuracy of rPDP-SGD significantly surpass those of the other two baselines. For classes with much lower privacy budgets (Classes 0, 1, and 2), rPDP-SGD demonstrates performance on par with that of DP-SGD (ε=3.0). Our findings indicate that the sampling-based method does not yield substantial utility improvements for groups that have significantly smaller privacy budgets and are a minority in the population. As discussed in Section 7.1, current methods such as Filter and BinarySearch also fail to adequately address this issue. This suggests that an efficient and effective solution for this challenge has yet to be developed, leaving it as an open question for further investigation.