Differentially Private Learning Needs Hidden State (Or Much Faster Convergence)

Anonymous Author(s)
Affiliation
Address
email

Abstract

Differential privacy analysis of randomized learning algorithms typically relies on composition theorems, where the implicit assumption is that the internal state of the iterative algorithm is revealed to the adversary. However, by assuming that the internal state of the algorithm is not revealed, recent works prove a smaller privacy bound for noisy gradient descent (on strongly convex smooth loss functions), compared with composition bounds. In this paper, we significantly improve privacy analysis under the hidden state assumption. We enable taking advantage of privacy amplification by sub-sampling and randomized post-processing, and extend the analysis to "shuffle and partition" and "sample without replacement" stochastic minibatch gradient descent schemes. We prove that, in these settings, our privacy bound is substantially smaller than the composition bounds, notably after many epochs of training (required for learning from high-dimensional data). So, unless the DP algorithm converges fast, our privacy analysis shows that differentially private learning benefits from hidden state privacy analysis. We also empirically show that, given a fixed privacy budget, our analysis enables DP training of image classification models with a better prediction accuracy, compared with prior work.

1 Introduction

Machine learning models leak sensitive information about their training data. To protect user privacy, the widely-used differentially private training algorithm, DP-SGD [1], adds carefully calibrated noise in each gradient update for the model parameters. This added noise guarantees that the models trained on two neighboring datasets are indistinguishable in probability distributions. To quantify the indistinguishability, differential privacy analysis *bounds* the (moment of) likelihood ratio between a pair of models trained on any two neighboring datasets. Improving this privacy analysis is crucial for obtaining a higher (train and test) accuracy of the output model, under a constrained privacy budget.

For a vast majority of learning tasks (such as private image classification), composition-based privacy analysis of DP-SGD [1] is the mainstream analysis method, mainly due to its simplicity and broad applicability. However, private learning via DP-SGD usually suffers from a slow convergence in empirical prediction accuracy [10], under which the total privacy loss grows very large due to composing it over a large number of iterations. This, in turn, results in overestimating the magnitude of required noise for differentially private training, which worsens the privacy-accuracy trade-off.

To alleviate this problem, one popular direction is to design new variants of the DP-SGD algorithm, that converge faster, i.e. require a smaller number of iterations to reach a stable training accuracy. This includes works that derive privacy preserving variants of fast optimization algorithms, such as performing DP-SGD with momentum [22], adaptive gradient clipping [2, 32], adaptive selection of step-size and noise scale [3], and using pre-trained or hand-crafted features [36]. All these approaches aim for a *faster training convergence* for the DP learning algorithm, such that the composition

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

analysis is applied to a smaller number of iterations. This is to reduce the high performance cost of overestimating the privacy loss (and the required noise scale) due to DP composition.

Another line of work focuses on directly improving the privacy analysis of the DP-SGD algorithm. To this end, prior work [18, 5, 15] suggests that hiding the internal state of noisy (S)GD, therefore analyzing the privacy bound for releasing only the last iterate results in a more accurate estimation of privacy loss, which can substantially reduce the performance cost of differential privacy. By applying the last-iterate privacy analysis [18], Feldman et al. [19] design new differentially private algorithms that achieve theoretically optimal excess risk with linear runtime for convex optimization, which is better than the prior privacy utility trade-off for the noisy SGD algorithm [7]. However, the analysis applies only to one single epoch of training, thus it is unclear how such training performs in practical learning settings (which require multiple epochs of training). Chourasia et al. [15] derive a strong privacy guarantee for multiple epoch training with hidden-state noisy GD, under strongly convex loss function. However, the analysis is limited to computationally expensive GD [20], and even if we attempt to directly apply it to stochastic mini-batch training, the analysis fails to model the privacy amplification by post-processing [18, 5] and mini-batch sub-sampling [11, 23, 7, 1, 4]. Thus, the bound under SGD learning becomes very large (as we also show in Figure 1).

Thus, under the hidden-state assumption over many epochs of training, the challenge is to compute a small privacy bound for differentially private stochastic gradient descent algorithms by taking advantage of privacy amplifications due to sub-sampling and randomized updates over mini-batches.

Contributions. In this paper, we tackle this challenge, and bridge the gaps in the prior work on hidden-state (last-iterate) privacy analysis. We model the privacy dynamics of noisy SGD, and show the amplification for hidden-state privacy bounds due to stochastic mini-batches, and demonstrate the empirical utility gain of our bound. **(i)** For multi-epoch noisy *stochastic mini-batch* gradient descent under *"shuffle and partition"* and *iterative "sampling without replacement"*, we prove new converging privacy bounds (for strongly convex smooth loss function) that significantly improves over prior bounds [4, 28, 15]. Our proof relies on our new bounds for the privacy amplification by post-processing (Lemma 4.2) and sub-sampling (Theorems 5.2 and 5.3). **(ii)** We evaluate the privacy accuracy trade-off of practical learning tasks under our privacy dynamics analysis. We empirically show that for learning tasks that require many epochs of training, where DP-SGD [1] has a slow convergence (e.g., logistic regression models trained from scratch), our privacy bound enables achieving a much higher prediction accuracy than what the composition-based analysis brings about. For tasks that converge within a few epochs (e.g., private fine-tuning of pre-trained models), our privacy bound can cause a slightly better privacy accuracy trade-off. [1]

Our results show that, to obtain high accuracy, it is crucial to apply our last-iterate privacy dynamics analysis for learning tasks with slow to moderate convergence. Alternatively, it is not empirically so costly to use composition-based privacy analysis when the training process converges very quickly.

2 Preliminaries

Definition 2.1 (Differential Privacy [16, 17])**.** A randomized algorithm $\mathcal{A}$ is (ε, δ)-differentially private if for any neighboring datasets D, D', and for all possible event S in the output space of $\mathcal{A}$,

$$P\left(\mathcal{A}(D) \in S\right) \leq e^{\varepsilon} \cdot P\left(\mathcal{A}(D') \in S\right) + \delta \tag{1}$$

where we say D, D' are *neighboring* if they are of the same size and differ in at most one data record.

Definition 2.2 ((α, ε)-Rényi DP [27])**.** A randomized algorithm $\mathcal{A}$ is said to satisfy (α, ε)-Rényi differential privacy (or (α, ε)-Rényi DP for short), if for any neighboring datasets D and D',

$$R_\alpha(\mathcal{A}(D) \| \mathcal{A}(D')) \leq \varepsilon, \text{ where } R_\alpha(\mu \| \nu) = \frac{1}{\alpha - 1} \log \mathbb{E}_{\theta \sim \nu}\left[\left(\frac{\mu(\theta)}{\nu(\theta)}\right)^{\alpha}\right] \tag{2}$$

where $\mathcal{A}(D)$ ($\mathcal{A}(D')$) denote the distribution of output given input dataset D (D'), and $R_\alpha(\mu \| \nu)$ is the *Rényi divergence* [33] of order $\alpha > 1$ for two distributions with density $\mu(\theta)$ and $\nu(\theta)$ on $\mathbb{R}^d$.

Definition 2.3 (log-Sobolev Inequality [37])**.** A distribution ν over $\mathbb{R}^d$ satisfies the log-Sobolev inequality (LSI) with constant c if for all smooth function $g : \mathbb{R}^d \to \mathbb{R}$ with $\mathbb{E}_{\theta \sim \nu}\left[g(\theta)^2\right] < \infty$,

$$\mathbb{E}_{\theta \sim \nu}\left[g(\theta)^2 \log\left(g(\theta)^2\right)\right] - \mathbb{E}_{\theta \sim \nu}\left[g(\theta)^2\right] \cdot \log \mathbb{E}_{\theta \sim \nu}\left[g(\theta)^2\right] \leq \frac{2}{c} \mathbb{E}_{\theta \sim \nu}\left[\|\nabla g(\theta)\|^2\right]. \tag{3}$$

[1] As shown after Theorem 5.2, our privacy bound is smaller than composition after $O(1/(\lambda\eta) + n/b)$ epochs.

Algorithm 1 $\mathcal{A}_{\text{Noisy-mBGD}}$: Noisy (Stochastic) Minibatch Gradient Descent

Input: Dataset $D = (\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n)$. Parameter space $\theta \in \mathbb{R}^d$. Loss function $\ell(\theta; \mathbf{x})$. Stepsize η. Noise standard deviation σ. Minibatch size b. Initial parameters θ_0^0 sampled from an arbitrary distribution $p_0(\theta)$.
Batch Generation: obtain minibatches B_k^j of size b, for $j = 0, \cdots, n/b-1$; for epochs $k = 0, \cdots, K-1$.
  ▷ **If "Shuffle and partition"**: randomly partition n data indices into $B^0, \cdots, B^{n/b-1}$, let $B_k^j = B^j$. [2]
  ▷ **If Sample without replacement**: resample b different indices from $\{1, \cdots, n\}$ to obtain every B_k^j.
for epoch $k = 0, 1, \cdots, K-1$ **do**
  for iteration $j = 0, 1, \cdots, n/b-1$ **do**
    $\theta_k^{j+1} = \theta_k^j - \eta \cdot g\left(\theta_k^j; B_k^j\right) + \sqrt{2\eta\sigma^2} \cdot \mathcal{N}(0, \mathbb{I}_d) \quad \text{where} \quad g\left(\theta_k^j; B_k^j\right) = \frac{1}{b}\sum_{i \in B_k^j} \nabla \ell(\theta_k^j; \mathbf{x}_i)$
  $\theta_{k+1}^0 = \theta_k^{n/b}$
Output: θ_K^0

3 An overview of the problem and our approach

We analyze differential privacy loss of the noisy (stochastic) minibatch gradient descent Algorithm 1 under two minibatch generation variants: **(1)** "shuffle and partition" (widely implemented in privacy libraries [42, 21]); **(2)** "sample without replacement" (analyzed extensively in prior results [11, 23, 4, 39] despite its computational cost). Our ultimate goal is to prove a *worst-case* upper bound of the Rényi divergence between distributions of *last-iterate* parameters θ_K^0 and ${\theta'}_K^0$ trained on *any* two neighboring datasets, that is *tighter* than composition and enables a *better* privacy accuracy trade-off.

Our starting point is a tight converging hidden-state privacy dynamics analysis in Chourasia et al. [15] for noisy gradient descent. To extend this GD analysis to the stochastic minibatch setting, a natural solution is to conceptually view the updates that *involve* the (sensitive) differing data record (between neighboring datasets) as gradient descent on a *smaller*, iteratively *changing* dataset of size b (i.e., the size of a minibatch). This simple extension results in the following naive privacy baseline. [3]

Theorem 3.1 (Naive Privacy Dynamics Baseline [15])**.** *If the loss function $\ell(\theta; \mathbf{x})$ is λ-strongly convex and β-smooth, and its gradient has ℓ_2-sensitivity S_g, then Algorithm 1 under "shuffle and partition" and step-size $\eta < \frac{2}{\lambda+\beta}$ satisfies (α, ε)-Rényi DP with $\varepsilon \leq \frac{\alpha S_g^2}{\lambda\sigma^2 b^2}(1 - e^{-\lambda\eta K/2})$.*

However, as also shown in Figure 1, this naive privacy dynamics baseline converges *slowly* to a *huge* constant, which is significantly worse than the bounds derived by composition [28, 1] after a large number of epochs. This is because this naive baseline *fails to* capture the privacy amplification due to the stochasticity of minibatch sub-sampling, and the iterative noisy updates which amplify the privacy of preceding minibatches (referred to as amplification by post-processing). In this paper, however, we show how to compute a much tighter differential privacy bound for SGD, under hidden-state assumption, while taking advantage of these privacy amplifications.

Our approach is as follows. We decompose the distribution of last iterate parameters in Algorithm 1 as a mixture of conditional distributions $p(\theta_K^0) = \sum_B p(B) \cdot p(\theta_K^0 | B)$, given any possible minibatch sequence B. We first analyze the Rényi privacy loss for a fixed minibatch sequence while modeling privacy amplification by randomized post-processing (i.e., how much the randomized gradient update improve the privacy due to updates on preceding mini-batches). We then quantify the privacy amplification by subsampling under stochastic minibatches.

Privacy amplification by randomized post-processing. When a (fixed) minibatch only contains the indices of shared data records between two neighboring datasets, then the deterministic minibatch gradient descent mapping does not cause any privacy loss. Addition of the Gaussian noise that follows this deterministic update, however, serves as randomized post-processing which *decreases* the Rényi divergence between the two processes. To compute this privacy amplification, we pre-

[2] For simplicity of presentation, we assume b divides n. If n/b is not an integer, and if the algorithm ignores the last $n - \lfloor n/b \rfloor \cdot b$ data points, then our privacy dynamics bound holds by replacing n/b with $\lfloor n/b \rfloor$.

[3] A recent concurrent work [34] also follows this direct approach to extend the GD analysis [15] to SGLD setting, and proves a similar bound to our naive privacy dynamics baseline Theorem 3.1. However, there is a slight difference between [34, Corollary 3.3] and our Theorem 3.1, due to the flawed assumption in [34, Lemma 3.4] that the LSI constant proved in Chourasia et al. [15] (that *only* holds for GD process) also holds for SGLD process (that takes the form of a more complicated mixture distribution). See more details in Appendix B.

cisely model the change of parameter distributions with the Fokker-Planck equation (for diffusion process with zero drift).

Hidden-state privacy amplification by sub-sampling. Although privacy amplification by a *single* sub-sampling operation is well-studied [11, 23, 7, 1, 4], to the best of our knowledge, no prior bounds are applicable to *hidden-state* iterative resampling. This is because under the hidden-state assumption, the number of mixing components in the last-iterate parameter distribution grows exponentially with the number of epochs, which makes the mixture distribution very difficult to analyze. In this paper, we first carefully couple the mixture components between the processes associated with neighboring datasets. We use the joint convexity of exponentiated Rényi divergence to bound how much the Rényi divergence between mixture distributions for the pair of processes is smaller than its worst case.

4 Improved privacy dynamics for noisy (fixed) minibatch gradient descent

In this section, we quantify the privacy amplification by randomized post-processing in Algorithm 1, during iterations that do not access the sensitive differing data between neighboring datasets. We then combine it with Rényi DP composition during the remaining iterations to prove a converging privacy bound.

4.1 Privacy amplification by randomized post-processing (additive Gaussian noise)

We first explain the key lemma that proves exponentially decaying Rényi privacy loss under additive Gaussian noise post-processing, when the parameter distributions satisfy the log-Sobolev inequality. This is a new bound for the well-known "privacy amplification by iteration" phenomenon [18, 5].

Lemma 4.1. *Let μ, ν be two distributions on $\mathbb{R}^d$. Let $f : \mathbb{R}^d \to \mathbb{R}^d$ be a measurable mapping on $\mathbb{R}^d$. We denote $\mathcal{N}(0, 2t\sigma^2 \cdot \mathbb{I}_d)$ to be the standard Gaussian distribution on $\mathbb{R}^d$ with covariance matrix $2t\sigma^2 \cdot \mathbb{I}_d$. We denote $p_t(\theta)$ and $p'_t(\theta)$ to be the probability density functions for the distributions $f_\#(\mu) * \mathcal{N}(0, 2t\sigma^2\mathbb{I}_d)$ and $f_\#(\nu) * \mathcal{N}(0, 2t\sigma^2\mathbb{I}_d)$ respectively, where $f_\#(\mu), f_\#(\nu)$ denote the push forward distributions of μ, ν under mapping f. Then if μ and ν satisfy log-Sobolev inequality with constant c, and if the mapping f is L-Lipschitz, then for any order $\alpha > 1$,*

$$\frac{\partial}{\partial t} R_\alpha \left(p_t(\theta) \| p'_t(\theta)\right) \leq -c_t \cdot 2\sigma^2 \cdot \left(\frac{R_\alpha(p_t(\theta)\|p'_t(\theta))}{\alpha} + (\alpha - 1) \cdot \frac{\partial}{\partial \alpha} R_\alpha(p_t(\theta)\|p'_t(\theta)) \right), \quad (4)$$

where $c_t = \left(\frac{L^2}{c} + 2t\sigma^2\right)^{-1}$ is the log-Sobolev inequality constant for distributions $p_t(\theta)$ and $p'_t(\theta)$.

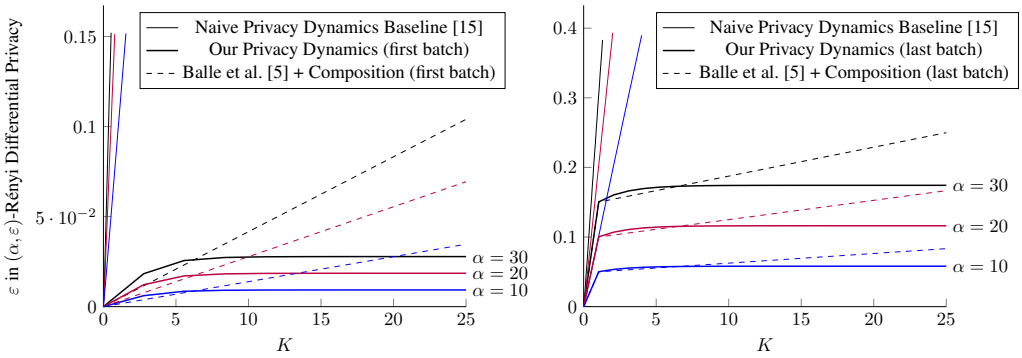
Proof Sketch. The proof starts by modelling Gaussian noise as a diffusion process with *zero* drift. We then bound the rate of Rényi divergence with the LSI constant for the process (following prior works [37, 15]). However, instead of assuming a fixed LSI constant c for all intermediate distributions, we prove a more precise LSI constant c_t that changes with t. Complete proof is in Appendix C.1. □

The partial differential inequality in Lemma 4.1 quantifies the amplification of Rényi privacy loss under additive Gaussian noise. By solving Equation (4) on $t \in [0, \eta]$, we prove the following lemma that models recursive privacy dynamics during one step of noisy minibatch gradient descent.

Lemma 4.2. *Let D, D' be an arbitrary pair of neighboring datasets that differ in the i_0-th data point (i.e. $x_{i_0} \neq x'_{i_0}$). Let B_k^j be a fixed minibatch used (in iteration j of epoch k) in Algorithm 1, which contains b indices sampled from $\{1, \cdots, n\}$. We denote θ_k^j and ${\theta'}_k^j$ as the intermediate parameters in Algorithm 1 on input datasets D and D', respectively. If the distributions of θ_k^j and ${\theta'}_k^j$ satisfy log-Sobolev inequality with a constant c, and if the minibatch GD mapping $f(\theta) = \theta - \eta \cdot \frac{1}{b} \cdot \sum_{i \in B_k^j} \ell(\theta; \mathbf{x}_i)$ is L-Lipschitz for parameters θ, then the following recursive bound for Rényi divergence holds.*

$$\frac{R_\alpha(\theta_k^{j+1} \| {\theta'}_k^{j+1})}{\alpha} \leq \begin{cases} \frac{R_{\alpha'}(\theta_k^j \| {\theta'}_k^j)}{\alpha'} \cdot \left(1 + \frac{c \cdot 2\eta\sigma^2}{L^2}\right)^{-1} & \text{if } i_0 \notin B_k^j \\ \frac{R_\alpha(\theta_k^j \| {\theta'}_k^j)}{\alpha} + \frac{\eta S_g^2}{4\sigma^2 b^2} & \text{if } i_0 \in B_k^j \end{cases} \text{ with } \alpha' = \frac{\alpha - 1}{1 + \frac{c \cdot 2\eta\sigma^2}{L^2}} + 1. \quad (5)$$

Proof Sketch. When $i_0 \notin B_k^j$, we apply Lemma 4.1 on the deterministic minibatch gradient descent update f, and solve Equation (4) to obtain recursive Rényi privacy bound. This proof is similar to



(a) first batch (b) last batch

Figure 1: Rényi privacy loss of noisy (fixed) minibatch gradient descent over K epochs, which goes over a fixed sequence of minibatch partition in each epoch. We show the privacy loss for data points in the first minibatch B^0 and the last minibatch $B^{n/b-1}$ of each pass. We show the ε in the (q, ε)-RDP guarantee derived by our privacy dynamics analysis (bold lines), the naive privacy dynamics baseline (thin lines), and the privacy amplification by mixing and diffusion analysis [5] combined with composition (dashed line). We evaluate under the following setting: RDP order $\alpha \in \{10, 20, 30\}$; λ-strongly convex loss function with $\lambda = 1$; β-smooth loss function with $\beta = 4$; total gradient sensitivity $S_g = 4$; size of the data set $n = 50$; step-size $\eta = 0.02$; noise variance $\sigma^2 = 4$, minibatch size $b = 2$. We use Theorem 4.3 for our privacy dynamics; We use Theorem 3.1 for naive privacy dynamics baseline; We use [5, Theorem 5] to obtain Balle et al. [4] + Composition (details are in Appendix C.6).

that of [15, Theorem 2], however, dealing with the new LSI constant c_t that changes with t introduces an additional technical difficulty. When $i_0 \in B_k^j$, we use composition theorem to prove the additive recursion in the second row of Equation (5). The complete proof is in Appendix C.2. □

Comparison with prior amplification by post-processing bound. Prior to this work, for convex and smooth loss functions, Feldman et al. [18] derive a tight bound for the amplification by post-processing in noisy stochastic gradient descent. In Appendix C.4, we show that our new recursive amplification bound could recover this known *tight* bound for convex smooth loss functions, while using a different proof. For (a more restrictive setting of) *one* epoch of noisy SGD on *strongly convex* loss function, Balle et al. [5] further improve the bound in Feldman et al. [18] via a careful coupling-based approach. However, we do not see an easy way to extend the bound in Balle et al. [4] to *multiple* epochs except by using Rényi DP composition (which would give a linearly worsening Rényi DP bound with the number of epochs). By contrast, our recursive amplification bound easily applies to multiple epochs (under hidden state assumption, and without requiring composition over the epochs), and enables converging privacy dynamics for Algorithm 1 on strongly convex smooth loss functions (as discussed in Section 4.2).

4.2 Improved privacy dynamics for (fixed) minibatch gradient descent

By using the recursive privacy bound Lemma 4.2 and the non-overlapping property of minibatch partitions of the dataset, we prove position-dependent Rényi DP bounds for Algorithm 1 as follows.

Theorem 4.3 (Privacy dynamics under strongly convex smooth loss)**.** *Conditioned on a fixed sequence of partitioned minibatches $B^0, \cdots, B^{n/b-1}$ in Line 3, if the loss function is λ-strongly convex, β-smooth and its gradient has ℓ_2-sensitivity S_g, then running Algorithm 1 for $K \geq 1$ epochs with step-size $\eta < \frac{2}{\lambda+\beta}$, satisfies (α, ε)-Rényi DP for data points in the batch B^{j_0}, with*

$$\varepsilon \leq \varepsilon_0^{\lfloor \frac{n}{2b} \rfloor}(\alpha) \cdot \frac{1 - (1-\eta\lambda)^{2\cdot(K-1)\cdot(n/b - \lfloor \frac{n}{2b} \rfloor)}}{1 - (1-\eta\lambda)^{2\cdot(n/b - \lfloor \frac{n}{2b} \rfloor)}} + \varepsilon_0^{n/b - j_0}(\alpha) \tag{6}$$

where $\varepsilon_0^j(\alpha) = \frac{\alpha\eta S_g^2}{4\sigma^2 b^2} \cdot (1-\eta\lambda)^{2\cdot(j-1)} \cdot \frac{1}{\sum_{s=0}^{j-1}(1-\eta\lambda)^{2s}}$ for any $j = 1, \cdots, \frac{n}{b}$ (we assume $\frac{n}{b} \geq 2$).

Proof Sketch. We first prove that the distribution of parameters θ_k^j satisfies LSI with a constant c_k^j that depends on k, j. Then we plug the LSI constants into Lemma 4.2 and prove a recursive privacy bound for data points in each batch B^{j_0}. Finally, by carefully choosing which recursion to use and solve, we obtain the privacy bound in the theorem statement. The complete proof is in Appendix C.5. □

The above new privacy bound Theorem 4.3 quantifies the privacy amplification during iterations that only access the shared data point between neighboring datasets, and it is significantly smaller than the naive privacy dynamics baseline derived from Chourasia et al. [15] (that does not capture this additional privacy amplification). In Figure 1, we observe that for data points in the first batch B^0, our new privacy dynamics bound is smaller by a multiplicative factor of approximately n/b (where b/n is the batch sampling ratio), and the improvement for data points in the last-batch is also significant.

Bound improvement compared with Balle et al. [5]. In the first epoch (when $K = 1$), our privacy dynamics bound (based on recursive scheme) in Theorem 4.3 is of same order as the privacy amplification bound Balle et al. [5, Theorem 5], as shown in Figure 1. However, for training with multiple epochs, Balle et al. [5] use a coupling-based approach, and we do not see an easy way to extend their analysis except by using Rényi DP composition [27] over the epochs, which gives a linearly accumulating Rényi DP bound for multiple epochs (as K increases). On the contrary, as shown in Figure 1, our improved privacy dynamics bound converges to a constant, thus significantly improves over the Rényi DP composition of Balle et al. [5, Theorem 5].

5 Privacy dynamics for noisy (stochastic) minibatch gradient descent

In this section, we further improve over the position-dependent privacy dynamics analysis in Section 4, by incorporating the effect of amplification of privacy loss due to the stochastic minibatch sampling. We first prove the following lemma by the convexity of f-divergence with $f(x) = x^\alpha$.

Lemma 5.1 (Joint convexity of scaled exponentiation of Rényi divergence)**.** *Let $\mu_1, \cdots, \mu_m$ and $\nu_1, \cdots, \nu_m$ be distributions over R^d. Then for any RDP order $\alpha \geq 1$, and any coefficients $p_1, \cdots, p_m \geq 0$ that satisfy $p_1 + \cdots + p_m = 1$, the following inequality holds.*

$$e^{(\alpha-1)\cdot R_\alpha(\sum_{j=1}^m p_j\mu_j \| \sum_{j=1}^m p_j\nu_j)} \leq \sum_{j=1}^m p_j \cdot e^{(\alpha-1)\cdot R_\alpha(\mu_j\|\nu_j)} \tag{7}$$

We provide a detailed proof for Lemma 5.1 in Appendix D. This Lemma is our main tool for quantifying the additional privacy amplification under stochastic minibatches in the rest of the section.

5.1 Privacy dynamics under shuffle and partition

The "shuffle and partition" batch generation scheme is commonly used for practical DP-SGD implementations in privacy libraries [21, 42]. By using Theorem 4.3 and new bounds for the privacy amplification by shuffling in Algorithm 1, we prove the following Rényi DP guarantee.

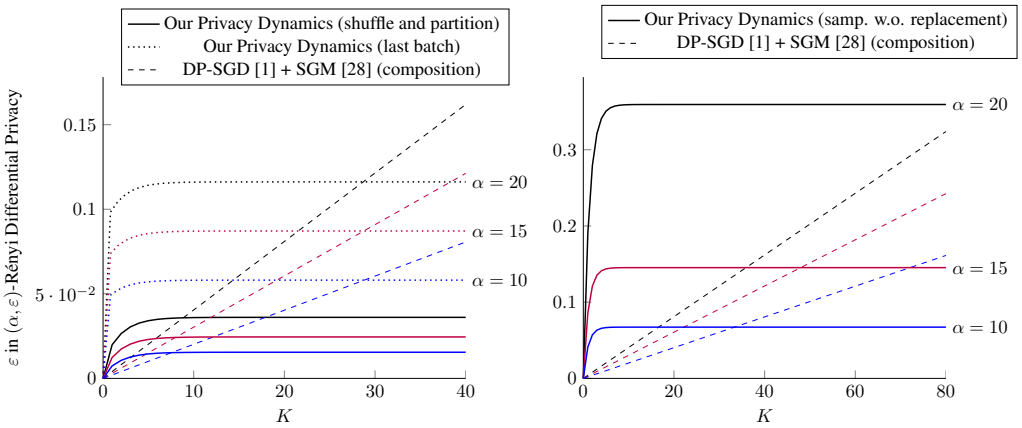
Theorem 5.2 (Privacy dynamics under "shuffle and partition")**.** *If the loss function $\ell(\theta; x)$ is λ-strongly convex, β-smooth, and if its gradient has finite ℓ_2-sensitivity S_g, then for $\frac{n}{b} \geq 2$, running Algorithm 1 for $K \geq 1$ epochs with step-size $\eta < \frac{2}{\lambda+\beta}$, under "shuffle and partition" batch generation scheme, satisfies (α, ε)-Rényi DP with*

$$\varepsilon \leq \varepsilon_0^{\lfloor \frac{n}{2b} \rfloor}(\alpha) \cdot \frac{1-(1-\eta\lambda)^{2\cdot(K-1)\cdot(n/b-\lfloor \frac{n}{2b} \rfloor)}}{1-(1-\eta\lambda)^{2\cdot(n/b-\lfloor \frac{n}{2b} \rfloor)}} + \frac{1}{\alpha-1} \cdot \log\left(\underset{0\leq j_0<n/b}{Avg} e^{(\alpha-1)\varepsilon_0^{n/b-j_0}(\alpha)} \right) \tag{8}$$

where $\varepsilon_0^j(\alpha) = \frac{\alpha\eta S_g^2}{4\sigma^2 b^2} \cdot (1-\eta\lambda)^{2\cdot(j-1)} \cdot \frac{1}{\sum_{s=0}^{j-1}(1-\eta\lambda)^{2s}}$ for any $j = 1, \cdots, \frac{n}{b}$ (we assume $\frac{n}{b} \geq 2$).

The above Rényi DP bound (8) in Theorem 5.2 is always smaller than the bound in Theorem 4.3 given the worst-case fixed minibatch sequence (i.e., when the differing data point is in the last minibatch of each pass). Therefore, Theorem 5.2 quantifies the privacy amplification due to shuffling, when compared to the worst-case Rényi DP bound among all possible minibatch sequences. We illustrate this amplification in Figure 2 (a). Moreover, our Rényi DP bound in Theorem 5.2 is strictly smaller than composition-based privacy bound for DP-SGD [1, 28] after $4 \cdot \frac{1}{\lambda\eta} + 4 \cdot \frac{n}{b}$ epochs. This is because the first term in Equation (178) is strictly smaller than the composition-based privacy bound after $4 \cdot \frac{1}{\lambda\eta}$ epochs, and the second term in Equation (178) is strictly smaller than the composition-based privacy bound after $4 \cdot (n/b)$ epochs (as we explain with more details in Appendix D.1).

[4] When $\alpha - 1$ and $\varepsilon_0^{n/b-j_0}(\alpha)$ are large, the second term in Equation (8) might overflow. In the experiments, we compute this log-sum-exp term with the shifted approximation $\varepsilon_0^1 + \log \underset{0\leq j_0<n/b}{Avg} e^{(\alpha-1)\left(\varepsilon_0^{n/b-j_0}(\alpha) - \varepsilon_0^1(\alpha)\right)}$.



(a) shuffle and partition

(b) (re)sample without replacement

Figure 2: Rényi privacy loss of noisy (stochastic) minibatch gradient descent over K epochs. We show ε in the (q, ε)-RDP guarantee derived by our privacy dynamics bound under "shuffle and partition" (bold lines, left plot), our privacy dynamics bound under sampling without replacement (bold lines, right plot), our privacy dynamics bound for data points in the last batch $B^{n/b-1}$ (thin dashed lines), and the baseline composition-based bound for DP-SGD [1, 28] (thin lines). We evaluate under the following setting: RDP order $\alpha \in \{10, 20, 30\}$; λ-strongly convex loss function with $\lambda = 1$; β-smooth loss function with $\beta = 4$; finite total gradient sensitivity $S_g = 4$; size of the data set $n = 50$; step-size $\eta = 0.02$; noise variance $\sigma^2 = 4$, batch size $b = 2$. The expressions for computing the privacy bounds are: Privacy Dynamics (shuffle and partition): Theorem 5.2 [4]; Privacy Dynamics (samp. w.o. replacement): Theorem 5.3; Privacy Dynamics (last batch): Theorem 4.3 under $j_0 = n/b - 1$; Composition: derived from Section 3.3. of Mironov et al. [28] that approximately equals $\frac{b}{n} \cdot \frac{\alpha \eta S_g^2}{4\sigma^2 b^2} \cdot K$.

5.2 Privacy dynamics under sampling without replacement

We now similarly analyze privacy dynamics for another variant of noisy minibatch gradient descent (Algorithm 1), where we iteratively (re)sample b indices without replacement from $\{1, \cdots, n\}$ to obtain each minibatch. That is, the minibatches used in different iterations (of one epoch) may overlap.

Theorem 5.3 (Recursive amplification by sampling without replacement)**.** *If the loss function $\ell(\theta; x)$ is λ-strongly convex, β-smooth, and if its gradient has finite ℓ_2-sensitivity S_g, then Algorithm 1 under sampling without replacement and stepsize $\eta < \frac{2}{\lambda+\beta}$ satisfies (α, ε)-Rényi DP guarantee with*

$$\varepsilon \leq \frac{1}{\alpha - 1} \log \left(S_K^0(\alpha) \right) \tag{9}$$

where the terms $S_k^j(\alpha)$ for $k = 0, \cdots, K-1$ and $j = 0, \cdots, n/b - 1$ are recursively computed by $S_0^0(\alpha) = 1$; $S_k^{j+1}(\alpha) = \frac{b}{n} \cdot e^{\frac{(\alpha-1)\alpha\eta S_g^2}{4\sigma^2 b^2}} \cdot S_k^j(\alpha) + (1 - \frac{b}{n}) \cdot S_k^j(\alpha)^{(1-\eta\lambda)^2}$; and $S_{k+1}^0(\alpha) = S_k^{n/b}(\alpha)$.

In Figure 2 (b), we compare the above Rényi DP bound Theorem 5.3 with the baseline composition-based privacy bound for DP-SGD [1, 27]. We observe that for RDP order $\alpha = 10, 15$, our privacy dynamics bound significantly improves over the baseline composition-based bound after 50 epochs. However, this improvement is smaller for large RDP order $\alpha = 20$. Moreover, our privacy bound under "sample without replacement" in Figure 2 (b) is larger than our privacy bound under "shuffle and partition" in Figure 2 (a), which suggests room for future improvement of the bound in Theorem 5.3.

6 Evaluating privacy accuracy trade-off under improved privacy dynamics

In this section, we evaluate regularized logistic regression in a multi-class setting (with per-class bias). We compare the optimal accuracy of (ε, δ)-DP models under our privacy dynamics bound, against the best known prior results [36] under DP-SGD [1, 28], for the same privacy budget ε and δ.

Experiment setting. We run experiments on image classification tasks. The training dataset that we take as input is $D = (\mathbf{z}_1, \cdots, \mathbf{z}_n)$, where each data record $\mathbf{z}_i = (\mathbf{x}_i, \mathbf{y}_i)$ consists of the d-dimentional data input feature vector $\mathbf{x}_i \in \mathbb{R}^d$, and the label vector $\mathbf{y}_i \in \{0, 1\}^c$ (in one-hot encoding). To train a

Table 1: **Test accuracy of logistic regression model trained on ScatterNet features under our privacy dynamics bound in Corollary E.3, compared with prior results [36] under DP-SGD** on MNIST, Fashion-MNIST, and CIFAR-10. We fix the privacy budget for all the trained models at $(\varepsilon = 3, \delta = 10^{-5})$, and show the mean and standard deviation across five runs. The hyperparameters are optimized within each setup to faciliate optimal accuracy. Feature clipping worsens the test accuarcy of trained model (compared to gradient clipping). Despite this negative effect, the test accuracy under privacy dynamics (with feature clipping) is consistently higher than that under unregularized DP-SGD (with momentum). Full hyperparameters are in Appendix E.7.

Analysis	DP-SGD [1] + momentum			Dynamics	Non-DP
DP Notion	Add/Remove a record [5]			Change a record	-
Regularizer	No		Yes	Yes	No
Clipping	Gradient	Feature+Gradient	Feature+Gradient	Feature+Gradient	No
MNIST	98.47±0.03%	98.36±0.06%	98.33±0.08%	**98.85±0.05%**	99.3±0.0%
FMNIST	89.64±0.19%	87.71±0.21%	87.76±0.22%	**89.87±0.18%**	91.5±0.0%
CIFAR-10	66.78±0.27%	65.87±0.20%	65.71±0.32%	**69.51±0.23%**	71.1±0.0%
Reference	[36]	-	-	This Paper	[36]

differentially private classifier, we use Algorithm 1 under "shuffle and partition", which is equivalent to practical implementations of DP-SGD [1] (as explained in Appendix E.1). To satisfy the necessary conditions for our privacy dynamics bound, we use regularized logistic regression to ensure strong convexity, use feature clipping to ensure smoothness, and use gradient clipping to enforce finite gradient sensitivity (described in Appendix E.2, E.3 and E.4). The formulas for these privacy bounds, are in Appendix E.5. Our code is provided in the supplemental material.

Hyperparameter tuning. In hyperparameter tuning, multiple accesses to the same private training dataset are required for training models under different hyperparameters, thus contributing to possibly higher privacy cost than the privacy bound for a single run of training algorithm. However, most prior works [1, 36] do not account for this additional privacy cost due to hyperparameter tuning. Therefore, for a fair comparison with Tramèr and Boneh [36], we also do not account the privacy cost due to hyperparameter tuning. Moreover, recent works [24, 30] also have proposed algorithms that reduce the privacy cost for hyperparameter tuning to a constant times the privacy cost of a single run.

6.1 Experiments on ScatterNet features

ScatterNet features. We follow Tramèr and Boneh [36] and similarly train regularized logistic regression models from scratch on hand-crafted ScatterNet [9, 29] features (extracted via data-independent scale invariant feature transformation [25]). This is because, to the best of our knowledge, Tramèr and Boneh [36] provides the best known accuracy for logistic regression models trained from scratch. More details could be found in Section 2.1 and Appendix C.1 in Tramèr and Boneh [36].

Slow convergence of DP-SGD for training from scratch. Tramèr and Boneh [36] show that differentially private linear models trained on hand-crafted ScatterNet features improve the current state-of-the art test accuracy for end-to-end differentially private CNN models (both under DP-SGD analysis). We reproduce their test accuracy results for MNIST, Fashion-MNIST, and CIFAR-10 in the following Table 1. We observe that due to the restriction from the composition-based privacy bound, their training algorithm could only run for a limited number of epochs, e.g. 60 epochs for training on CIFAR-10. However, the accuracy of the trained model continues to grow even after 60 epochs (discussed in detail in Appendix E.6), indicating that insufficient number of allowed running epochs of DP-SGD restricts its accuracy for training differentially private model from scratch.

Improved accuracy under privacy dynamics Analysis. Our tighter privacy bound in Corollary E.3 allows the noisy minibatch gradient to run for significantly more epochs than DP-SGD, within the same fixed privacy budget. In Table 1, we show that our privacy dynamics bound enables in a better privacy accuracy trade-off for (regularized) logistic regression on hand-crafted features, than that in Tramèr and Boneh [36] (which, to the best of our knowledge, is also the best reported test accuracy for differentially private image classification models trained from scratch in prior works).

[5] The "change a record" DP notion is different from the "add/remove a record" DP notion. However, (ε, δ)-DP under "add/remove a record" can be easily translated to $(2\varepsilon, \delta)$-DP under "change a record".

Table 2: **Test accuracy of model obtained by differentially privately fine-tuning the last layer of pretrained models under our privacy dynamics bound in Corollary E.3, compared with DP-SGD results reproduced from Tramèr and Boneh [36].** We fix the privacy budget for all the trained models at $(\varepsilon = 2, \delta = 10^{-5})$, and show the mean and standard deviation across five random runs. The hyperparameters are optimized within each setup to faciliate optimal accuracy. The test accuracy under privacy dynamics is only slightly higher than that under DP-SGD (with momentum). The full hyperparameters are in Appendix E.8.

Analysis	DP-SGD [1] + momentum	Dynamics	Non-DP
DP Notion	Add/Remove a record [5]	Change a record	-
Clipping	Gradient	Feature + Gradient	No
ResNeXt-29 (CIFAR-100)	79.79±0.22%	**80.09±0.08%**	84%
SIMCLR v2 (unlabelled ImageNet)	92.36±0.11%	**92.71±0.11%**	95.3%
Reference	[36]	This paper	[41] and [13]

6.2 Experiments on Transfer Learning

In this section, we empirically analyze the complementing setting of transfer learning, which typically only requires a few epochs to reach a stable accuracy. We use CIFAR-10 as the target private dataset, and differentially privately fine-tune *the last layer* of the source pre-trained model (trained on another dataset), which is equivalent to training a logistic regression model on top of frozen second-to-last layer features of the source pre-trained model.

Source pre-trained model. The problem of privately fine-tuning a source model (trained from a different/public dataset) to a private target training dataset is well-studied [1, 31, 36]. The high quality features learned from pre-trained model usually enable a better privacy accuracy trade-off of the target learning task, than training from scratch. We follow the setups in Tramèr and Boneh [36] (which to the best our knowledge, attain the highest accuracies for fine-tuned models in the literature) and experiment with two non-private pre-trained models: **(1)** a ResNeXt-29 [41] model trained on the CIFAR-100 dataset; **(2)** a SimCLR [13] model trained on unlabeled ImageNet dataset.

Fast convergence of DP-SGD for fine-tuning. In our experiments, we observe that DP-SGD reaches stable accuracy within 50 epochs, for fine-tuning pre-trained models on CIFAR-10. This is much faster than training from scratch that typically requires a few times more epochs to converge.

Similar accuracy under privacy dynamics analysis. Transfer learning tasks need relatively small number of iterations to reach convergence. However, our privacy dynamics bound is only smaller than composition-based bound after $O(\frac{1}{\lambda\eta} + \frac{n}{b})$ epochs (shown after Theorem 5.2), which suggests that composition may be tighter if the training only requires a few epochs. Therefore, in Table 2, we observe that our privacy dynamics analysis only enables slightly better privacy accuracy trade-off.

7 Conclusion

We prove a novel converging last-iterate privacy bound for noisy stochastic mini-batch gradient descent on strongly convex smooth loss functions. Our bound substantially improves the prior privacy dynamics bound for noisy GD [15], by proving novel bounds for the additional privacy amplification (by post-processing and subsampling) during training with stochastic mini-batches. We evaluate the privacy accuracy trade-off for various learning tasks under our improved privacy dynamics bound, and compare it with that under DP-SGD. We show that when the training process has a moderate to slow convergence, our privacy dynamics analysis enables better accuracy than that under composition privacy bound for DP-SGD. This shows that to achieve the optimal privacy accuracy trade-off, differentially private learning needs to be evaluated by a last-iterate privacy bound, unless it has a very fast convergence (under which, in practice, the cost of composition bound is not significant).

Future Work. There is room for improving our privacy amplification bound Theorem 5.2 under smaller batch size b, because under *one* sampling step, the best known amplification rate is as small as $O(\frac{b^2}{n^2})$ [28]. This suggests possibility for a better *hidden-state* privacy amplification bound (without using composition) for multiple iterations of sampling.

References

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

[2] Galen Andrew, Om Thakkar, Brendan McMahan, and Swaroop Ramaswamy. Differentially private learning with adaptive clipping. *Advances in Neural Information Processing Systems*, 34, 2021.

[3] Hilal Asi, John Duchi, Alireza Fallah, Omid Javidbakht, and Kunal Talwar. Private adaptive gradient methods for convex optimization. In *International Conference on Machine Learning*, pages 383–392. PMLR, 2021.

[4] Borja Balle, Gilles Barthe, and Marco Gaboardi. Privacy amplification by subsampling: Tight analyses via couplings and divergences. *Advances in Neural Information Processing Systems*, 31, 2018.

[5] Borja Balle, Gilles Barthe, Marco Gaboardi, and Joseph Geumlek. Privacy amplification by mixing and diffusion mechanisms. *arXiv preprint arXiv:1905.12264*, 2019.

[6] Jean-Baptiste Bardet, Nathaël Gozlan, Florent Malrieu, and Pierre-André Zitt. Functional inequalities for gaussian convolutions of compactly supported measures: Explicit bounds and dimension dependence. *Bernoulli*, 24(1):333–353, 2018.

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

[8] Dankmar Böhning. Multinomial logistic regression algorithm. *Annals of the institute of Statistical Mathematics*, 44(1):197–200, 1992.

[9] Joan Bruna and Stéphane Mallat. Invariant scattering convolution networks. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1872–1886, 2013.

[10] Zhiqi Bu, Hua Wang, Qi Long, and Weijie J Su. On the convergence of deep learning with differential privacy. *arXiv preprint arXiv:2106.07830*, 2021.

[11] Kamalika Chaudhuri and Nina Mishra. When random sampling preserves privacy. In *Annual International Cryptology Conference*, pages 198–213. Springer, 2006.

[12] Hong-Bin Chen, Sinho Chewi, and Jonathan Niles-Weed. Dimension-free log-sobolev inequalities for mixture distributions. *Journal of Functional Analysis*, 281(11):109236, 2021.

[13] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[14] Xiangyi Chen, Steven Z Wu, and Mingyi Hong. Understanding gradient clipping in private sgd: A geometric perspective. *Advances in Neural Information Processing Systems*, 33:13773–13782, 2020.

[15] Rishav Chourasia, Jiayuan Ye, and Reza Shokri. Differential privacy dynamics of langevin diffusion and noisy gradient descent. *Advances in Neural Information Processing Systems*, 34: 14771–14781, 2021.

[16] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer, 2006.

[17] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3-4):211–407, 2014.

[18] Vitaly Feldman, Ilya Mironov, Kunal Talwar, and Abhradeep Thakurta. Privacy amplification by iteration. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 521–532. IEEE, 2018.

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

[20] Jonas Geiping, Micah Goldblum, Phillip E Pope, Michael Moeller, and Tom Goldstein. Stochastic training is not necessary for generalization. *arXiv preprint arXiv:2109.14119*, 2021.

[21] Google et al. Tensorflow privacy, 2021.

[22] Peter Kairouz, Brendan McMahan, Shuang Song, Om Thakkar, Abhradeep Thakurta, and Zheng Xu. Practical and private (deep) learning without sampling or shuffling. *arXiv preprint arXiv:2103.00039*, 2021.

[23] Ninghui Li, Wahbeh Qardaji, and Dong Su. On sampling, anonymization, and differential privacy or, k-anonymization meets differential privacy. In *Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security*, pages 32–33, 2012.

[24] Jingcheng Liu and Kunal Talwar. Private selection from private candidates. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 298–309, 2019.

[25] David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee, 1999.

[26] Siwei Lyu. Interpretation and generalization of score matching. *arXiv preprint arXiv:1205.2629*, 2012.

[27] Ilya Mironov. Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pages 263–275. IEEE, 2017.

[28] Ilya Mironov, Kunal Talwar, and Li Zhang. R\'enyi differential privacy of the sampled gaussian mechanism. *arXiv preprint arXiv:1908.10530*, 2019.

[29] Edouard Oyallon and Stéphane Mallat. Deep roto-translation scattering for object classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2865–2873, 2015.

[30] Nicolas Papernot and Thomas Steinke. Hyperparameter tuning with renyi differential privacy. *arXiv preprint arXiv:2110.03620*, 2021.

[31] Nicolas Papernot, Steve Chien, Shuang Song, Abhradeep Thakurta, and Ulfar Erlingsson. Making the shoe fit: Architectures, initializations, and tuning for learning with privacy. 2019.

[32] Venkatadheeraj Pichapati, Ananda Theertha Suresh, Felix X Yu, Sashank J Reddi, and Sanjiv Kumar. Adaclip: Adaptive clipping for private sgd. *arXiv preprint arXiv:1908.07643*, 2019.

[33] Alfréd Rényi. On measures of entropy and information. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*, volume 4, pages 547–562. University of California Press, 1961.

[34] Théo Ryffel, Francis Bach, and David Pointcheval. Differential privacy guarantees for stochastic gradient langevin dynamics. *arXiv preprint arXiv:2201.11980*, 2022. URL `https://arxiv.org/abs/2201.11980v2`.

[35] Inder Jeet Taneja and Pranesh Kumar. Relative information of type s, csiszár's f-divergence, and information inequalities. *Information Sciences*, 166(1-4):105–125, 2004.

[36] Florian Tramèr and Dan Boneh. Differentially private learning needs better features (or much more data). In *International Conference on Learning Representations*, 2020.

[37] Santosh Vempala and Andre Wibisono. Rapid convergence of the unadjusted langevin algorithm: Isoperimetry suffices. *Advances in neural information processing systems*, 32, 2019.

[38] Feng-Yu Wang and Jian Wang. Functional inequalities for convolution probability measures. In *Annales de l'Institut Henri Poincaré, Probabilités et Statistiques*, volume 52, pages 898–914. Institut Henri Poincaré, 2016.

[39] Yu-Xiang Wang, Borja Balle, and Shiva Prasad Kasiviswanathan. Subsampled rényi differential privacy and analytical moments accountant. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1226–1235. PMLR, 2019.

[40] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

[41] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017.

[42] Ashkan Yousefpour, Igor Shilov, Alexandre Sablayrolles, Davide Testuggine, Karthik Prasad, Mani Malek, John Nguyen, Sayan Ghosh, Akash Bharadwaj, Jessica Zhao, et al. Opacus: User-friendly differential privacy library in pytorch. *arXiv preprint arXiv:2109.12298*, 2021.

[43] Jingzhao Zhang, Tianxing He, Suvrit Sra, and Ali Jadbabaie. Why gradient clipping accelerates training: A theoretical justification for adaptivity. *arXiv preprint arXiv:1905.11881*, 2019.

[44] David Zimmermann. Logarithmic sobolev inequalities for mollified compactly supported measures. *Journal of Functional Analysis*, 265(6):1064–1083, 2013.

[45] David Zimmermann. Elementary proof of logarithmic sobolev inequalities for gaussian convolutions on $\mathbb{R}$. In *Annales Mathématiques Blaise Pascal*, volume 23, pages 129–140, 2016.

Checklist

The checklist follows the references. Please read the checklist guidelines carefully for information on how to answer these questions. For each question, change the default **[TODO]** to [Yes] , [No] , or [N/A] . You are strongly encouraged to include a **justification to your answer**, either by referencing the appropriate section of your paper or providing a brief inline description. For example:

- Did you include the license to the code and datasets? [Yes] See Section **??**.
- Did you include the license to the code and datasets? [No] The code and the data are proprietary.
- Did you include the license to the code and datasets? [N/A]

Please do not modify the questions and only use the provided macros for your answers. Note that the Checklist section does not count towards the page limit. In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See Section 1.
   (b) Did you describe the limitations of your work? [Yes] See Section 7.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Section 4 and 5.
   (b) Did you include complete proofs of all theoretical results? [Yes] See supplemental material.

3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See supplemental material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See supplemental material.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See Table 1 and 2.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See supplemental material.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See supplemental material.
   (b) Did you mention the license of the assets? [Yes] See supplemental material.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] See supplemental material.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] See supplemental material.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

Appendix

Table of Contents

A Symbols **15**

B Discussion about the concurrent work [34, Corollary 3.3] **16**

C Proof for Section 4 **17**

C.1 Proof for Lemma 4.1 18

C.2 Proof for Lemma 4.2 20

C.3 Proof for LSI sequence for noisy minibatch gradient descent 23

C.4 Equivalence Between Lemma 4.2 (Ours) And The Bound in Feldman et al. [18] . 24

C.5 Proof for Theorem 4.3 25

C.6 Explanations for the privacy bound derived from Balle et al. [4] in Figure 1 . . . 28

C.7 Revisiting noisy GD: A tighter bound than [15, Corollary 1] 28

D Proof for Section 5 **31**

D.1 Proof for Theorem 5.2 31

D.2 Proof for Theorem 5.3 33

E Proofs and Explanations for Section 6 **34**

E.1 Pseudocode for Our Implementation 34

E.2 Implementation details for satisfying strong convexity 34

E.3 Implementation Deatils for satisfying smoothness 35

E.4 Implementation details for satisfying finite gradient sensitivity 37

E.5 Exact formulas for composition-based privacy bound and our improved privacy dynamics analysis in evaluations 37

E.6 Detailed Illustration for the Slow Convergence of DP-SGD for training from scratch 38

E.7 Full hyperparameters for Table 1 (and Appendix E.6) 38

E.8 Full hyperparameters for Table 2 39

A Symbols

Algorithms and Definitions

Symbol	Meaning	Where
$\mathbf{x}$	A data vector	Algorithm 1
n	Size of a dataset	Algorithm 1
b	Size of a minibatch	Algorithm 1
K	Total number of epochs in a learning algorithm	Algorithm 1
$\mathbb{I}_d$	Identity matrix with d rows and d columns	Algorithm 1
θ	Model parameters	Algorithm 1
$\ell(\theta, \mathbf{x})$	A loss function of θ parameterized by $\mathbf{x}$	Algorithm 1
D, D'	Neighboring datasets that differ in at most one record	Section 2
S_g	ℓ_2-sensitivity of total gradient $g(\theta; D) = \sum_{\mathbf{x} \in D} \nabla \ell(\theta; \mathbf{x})$ with regard to neighboring datasets. More specifically, $S_g = \max_{D,D',\theta} \|g(\theta, D) - g(\theta, D')\|_2$	Algorithm 1
$\mathcal{N}(\mu, \sigma \cdot \mathbb{I}_d)$	Gaussian distribution over $\mathbb{R}^d$ with mean μ and covariance matrix $\sigma \cdot \mathbb{I}_d$	Algorithm 1
η	Stepsize for each iterative update in the learning algorithm	Algorithm 1
$\{0, 1, \ldots, n\}$	The set of all integers between 0 and n	Algorithm 1
$\mathbf{x}_i$	Data point i of dataset D, with indexing starting at 1	Algorithm 1
θ_k^j, B_k^j	The parameters or generated minibatch after k epochs and j iterations of a learning algorithm, with indexing starting at 0	Algorithm 1
α	Rényi DP order	Section 2
(ε, δ)	Differential Privacy parameters	Section 2
(α, ε)	Rényi Differential Privacy parameters	Section 2
c	A log-Sobolev inequality constant	Section 2
λ	The strong convexity parameter for a loss function	Section 4
β	The smoothness parameter for a loss function	Section 4

Probability and Information Theory

Symbol	Meaning	Where
μ	A distribution over $\mathbb{R}^d$ with density $\mu(\theta)$	Section 4
$p(\theta)$	A probability density function over $\theta \in \mathbb{R}^d$	Section 3
$f_{\#}(\mu)$	The push forward distribution of μ under mapping f on the same domain	Section 4
$\mu * \nu$	The convolution of two distributions μ and ν	Section 4
$R_\alpha(\mu \| \nu)$	Rényi divergence between distributions μ and ν	Section 2
$\mathbb{E}_{\theta\sim\mu}[f(\theta)]$	Expectation of $f(\theta)$ with respect to $\mu(\theta)$	Section 2
$KL(\mu \| \nu)$	Kullback-Leibler divergence between distributions μ and ν	Appendix C

Calculus and Linear Algebra

Symbol	Meaning
t	A real scalar
$\lfloor t \rfloor$	The largest integer that is smaller than or equal than a real number t
$A_{i,j}$	Element i, j of matrix A
$\frac{dy}{dx}$	Derivative of y with respect to x
$\frac{\partial y}{\partial x}, \nabla_x y$	Partial derivative of y with respect to x
$\nabla_x^2 y$	Laplacian of y with respect to x
$\nabla_x y$	Gradient of y with respect to x
$x \cdot y$	Inner products between real vectors x and y
x	ℓ_2 norm of a vector x
$A \otimes B$	Tensor product between two real matrices A and B
x^T	Transpose of a real vector or matrix x
$f \circ g$	Composition of the functions or mappings f and g
$A \preceq B$	The matrix $B - A$ is semi-positive definite.

B Discussion about the concurrent work [34, Corollary 3.3]

Chourasia et al. [15] prove privacy dynamics bound for noisy gradient descent, and Ryffel et al. [34] extend this bound to SGLD, by directly viewing each minibatch update as gradient descent on a smaller dataset of size b (which is the size of a minibatch). This approach is similar to our approach for deriving the naive privacy dynamics baseline Theorem 3.1, and the expression in [34, Corollary 3.3] is very similar to the expression in our Theorem 3.1, except for having n^2 (instead of b^2) in the bound denominator.

However, an inspection of the proof for [34, Lemma 3.4] shows that, this difference between [34, Corollary 3.3] and our Theorem 3.1 is caused because by a flawed assumption in [34, Lemma 3.3]. More specifically, [34, Lemma 3.3] wrongly assume that the LSI constant proved in [15, Lemma 5] (which only holds for a GD process) would also similarly hold for a SGLD process that takes

the form of a more complex mixture distribution. Here, each mixture component is the conditional distribution of last-iterte parameters given a fixed sequence of minibatches.

This assumption is wrong because a given mixture distribution generally satisfies a different LSI constant than each of its component distributions. Moreover, bounding the LSI constant for a mixture distribution is largely an open problem [44, 45, 38, 6, 12]. The current best bound for this problem, to the best of our knowledge, is [12, Theorem 1], which says that the LSI constant for a mixture distribution, depends on the LSI constant for the distribution of each component, and the *worst-case* χ^2 distance between *any* two components' distributions. Therefore, the actual LSI constant for SGLD process would be significantly smaller (related to the number of components in the mixture distribution) than the assumed LSI constant in [34, Lemma 3.4] (which only holds for the conditional parameter distribution given *a fixed minibatch sequence*). After replacing the wrongly assumed LSI constant in [34, Lemma 3.4] with a correct LSI constant for SGLD process (that takes the form of mixture distribution), the privacy dynamics bounds in [34, Corollary 3.3] would be significantly worse (larger).

Due to this flawed assumption, in this paper, we do not compare our improved privacy dynamics theorem Theorem 4.3 with [34, Corollary 3.3]. Instead, we compare Theorem 4.3 with the naive privacy dynamics baseline Theorem 3.1 (which has similar expression as [34, Corollary 3.3]) in Figure 1.

C Proof for Section 4

We first establish a tool lemma for proving Lemma 4.1 and obtaining the partial differential inequality that bounds the growth of differential privacy loss.

Lemma C.1 (Lemma 5 in Vempala and Wibisono [37])**.** *Suppose ν is a distribution that satisfies log-Sobolev inequality with constant $c > 0$, and that has smooth density $\mu(\theta)$. Let $\alpha \geq 1$. For all measure μ that has smooth density $\nu(\theta)$,*

$$\frac{I_\alpha(\mu\|\nu)}{E_\alpha(\mu\|\nu)} \geq \frac{2c}{\alpha^2} \cdot R_\alpha(\mu\|\nu) + \frac{2c}{\alpha^2} \cdot \alpha(\alpha-1)\frac{\partial R_\alpha(\mu\|\nu)}{\partial \alpha} \tag{10}$$

where $I_\alpha(\mu\|\nu) = \mathbb{E}_{\theta\sim\nu}\left[\left(\frac{\mu(\theta)}{\nu(\theta)}\right)^\alpha \cdot \left\|\nabla \log \frac{\mu(\theta)}{\nu(\theta)}\right\|^2\right]$*, and* $E_\alpha(\mu\|\nu) = \mathbb{E}_{\theta\sim\nu}\left[\left(\frac{\mu(\theta)}{\nu(\theta)}\right)^\alpha\right]$

Proof. This Lemma is initially proved in Vempala and Wibisono [37] Lemma 5. We give an alternative proof here, that only uses one step of inequality in Equation (16). We hope this alternative proof helps understand whether there is still room for improving this Lemma (e.g. by improving the inequality in Equation (16)).

We denote ρ to be another distribution with density $\rho(\theta) = \frac{1}{E_\alpha(\mu\|\nu)} \cdot \left(\frac{\mu(\theta)}{\nu(\theta)}\right)^\alpha \cdot \nu(\theta)$. By simple integration, we verify that $\int \rho(\theta)d\theta = \frac{E_\alpha(\mu\|\nu)}{E_\alpha(\mu\|\nu)} = 1$. By definition,

$$\frac{I_\alpha(\mu\|\nu)}{E_\alpha(\mu\|\nu)} = \frac{\mathbb{E}_{\theta\sim\nu}\left[\left(\frac{\mu(\theta)}{\nu(\theta)}\right)^\alpha \cdot \left\|\nabla \log \frac{\mu(\theta)}{\nu(\theta)}\right\|^2\right]}{E_\alpha(\mu\|\nu)} \tag{11}$$

$$= \mathbb{E}_{\theta\sim\nu}\left[\frac{\rho(\theta)}{\nu(\theta)} \cdot \left\|\nabla \log \frac{\mu(\theta)}{\nu(\theta)}\right\|^2\right] \tag{12}$$

$$= \frac{1}{\alpha^2}\mathbb{E}_{\theta\sim\nu}\left[\frac{\rho(\theta)}{\nu(\theta)} \cdot \left\|\nabla \log \left(\frac{\mu(\theta)}{\nu(\theta)}\right)^\alpha\right\|^2\right] \tag{13}$$

$$\text{by } \nabla \log E_\alpha(\mu\|\nu) = 0, \quad = \frac{1}{\alpha^2}\mathbb{E}_{\theta\sim\nu}\left[\frac{\rho(\theta)}{\nu(\theta)} \cdot \left\|\nabla \log \left(\frac{\mu(\theta)}{\nu(\theta)}\right)^\alpha - \nabla \log E_\alpha(\mu\|\nu)\right\|^2\right] \tag{14}$$

$$= \frac{1}{\alpha^2}\mathbb{E}_{\theta\sim\nu}\left[\frac{\rho(\theta)}{\nu(\theta)} \cdot \left\|\nabla \log \frac{\rho(\theta)}{\nu(\theta)}\right\|^2\right] \tag{15}$$

By definition, $\mathbb{E}_{\theta\sim\nu}\left[\frac{\rho(\theta)}{\nu(\theta)}\cdot\left\|\nabla\log\frac{\rho(\theta)}{\nu(\theta)}\right\|^2\right]$ is the relative Fisher information $J(\rho\|\nu)$ of ρ with respect to ν. A celebrated equivalence result (e.g. Section 2.2 of Vempala and Wibisono [37]) says that, if and only if ν satsifies log-Sobolev inequality with constant c, the following relation between the KL divergence and relative Fisher information holds for all ρ:

$$KL(\rho\|\nu)\leq\frac{1}{2c}J(\rho\|\nu) \tag{16}$$

By plugging Equation (16) into Equation (15), we prove

$$\frac{I_\alpha(\mu\|\nu)}{E_\alpha(\mu\|\nu)}\geq\frac{2c}{\alpha^2}\cdot KL(\rho\|\nu) \tag{17}$$

$$\text{by definition,}\quad =\frac{2c}{\alpha^2}\cdot\int_{\mathbb{R}^n}\rho(\theta)\log\frac{\rho(\theta)}{\nu(\theta)}d\theta \tag{18}$$

$$\text{by definition of }\rho,\quad =\frac{2c}{\alpha^2}\cdot\mathbb{E}_{\theta\sim\nu}\left[\frac{1}{E_\alpha(\mu\|\nu)}\cdot\left(\frac{\mu(\theta)}{\nu(\theta)}\right)^\alpha\cdot\left(\alpha\log\frac{\mu(\theta)}{\nu(\theta)}-\log E_\alpha(\mu\|\nu)\right)\right] \tag{19}$$

$$=\frac{2c}{\alpha}\cdot\frac{1}{E_\alpha(\mu\|\nu)}\cdot\mathbb{E}_{\theta\sim\nu}\left[\frac{\partial}{\partial\alpha}\left(\frac{\mu(\theta)}{\nu(\theta)}\right)^\alpha\right]-\frac{2c}{\alpha^2}\cdot\frac{\log E_\alpha(\mu\|\nu)}{E_\alpha(\mu\|\nu)}\cdot\mathbb{E}_{\theta\sim\nu}\left[\left(\frac{\mu(\theta)}{\nu(\theta)}\right)^\alpha\right] \tag{20}$$

$$=\frac{2c}{\alpha}\cdot\frac{1}{E_\alpha(\mu\|\nu)}\cdot\mathbb{E}_{\theta\sim\nu}\left[\frac{\partial}{\partial\alpha}\left(\frac{\mu(\theta)}{\nu(\theta)}\right)^\alpha\right]-\frac{2c}{\alpha^2}\log E_\alpha(\mu\|\nu) \tag{21}$$

By exchanging the order of derivative and expectation (because $\mu(\theta)$ and $\nu(\theta)$ are smooth densities), we prove that Equation (21) is equivalent to the following inequality.

$$\frac{I_\alpha(\mu\|\nu)}{E_\alpha(\mu\|\nu)}\geq\frac{2c}{\alpha}\cdot\frac{\partial}{\partial\alpha}\log\mathbb{E}_\alpha(\mu\|\nu)-\frac{2c}{\alpha^2}\log E_\alpha(\mu\|\nu) \tag{22}$$

By definition, we have $E_\alpha(\mu\|\nu)=e^{(\alpha-1)R_\alpha(\mu\|\nu)}$, therefore we prove

$$\frac{I_\alpha(\mu\|\nu)}{E_\alpha(\mu\|\nu)}\geq\frac{2c}{\alpha}\cdot\left(R_\alpha(\mu\|\nu)+(\alpha-1)\frac{\partial R_\alpha(\mu\|\nu)}{\partial\alpha}\right)-\frac{2c}{\alpha^2}\cdot(\alpha-1)R_\alpha(\mu\|\nu) \tag{23}$$

$$=\frac{2c}{\alpha^2}\cdot R_\alpha(\mu\|\nu)+\frac{2c}{\alpha^2}\cdot\alpha(\alpha-1)\frac{\partial R_\alpha(\mu\|\nu)}{\partial\alpha} \tag{24}$$

□

C.1 Proof for Lemma 4.1

Lemma 4.1. Let μ,ν be two distributions on $\mathbb{R}^d$. Let $f:\mathbb{R}^d\to\mathbb{R}^d$ be a measurable mapping on $\mathbb{R}^d$. We denote $\mathcal{N}(0,2t\sigma^2\cdot\mathbb{I}_d)$ to be the standard Gaussian distribution on $\mathbb{R}^d$ with covariance matrix $2t\sigma^2\cdot\mathbb{I}_d$. We denote $p_t(\theta)$ and $p'_t(\theta)$ to be the probability density functions for the distributions $f_\#(\mu)*\mathcal{N}(0,2t\sigma^2\mathbb{I}_d)$ and $f_\#(\nu)*\mathcal{N}(0,2t\sigma^2\mathbb{I}_d)$ respectively, where $f_\#(\mu),f_\#(\nu)$ denote the push forward distributions of μ,ν under mapping f. Then if μ and ν satisfy log-Sobolev inequality with constant c, and if the mapping f is L-Lipschitz, then for any order $\alpha>1$,

$$\frac{\partial}{\partial t}R_\alpha\left(p_t(\theta)\|p'_t(\theta)\right)\leq-c_t\cdot2\sigma^2\cdot\left(\frac{R_\alpha(p_t(\theta)\|p'_t(\theta))}{\alpha}+(\alpha-1)\cdot\frac{\partial}{\partial\alpha}R_\alpha(p_t(\theta)\|p'_t(\theta))\right), \tag{25}$$

where $c_t=\left(\frac{L^2}{c}+2t\sigma^2\right)^{-1}$ is the log-Sobolev inequality constant for distributions $p_t(\theta)$ and $p'_t(\theta)$.

Proof. By definition, $p_t(\theta)$ and $p'_t(\theta)$ are probability density functions for distributions $f_\#(\mu)*\mathcal{N}(0,2t\sigma^2\mathbb{I}_d)$ and $f_\#(\nu)*\mathcal{N}(0,2t\sigma^2\mathbb{I}_d)$ respectively. Therefore $p_t(\theta)$ and $p'_t(\theta)$ satisfy the following Fokker-Planck equations.

$$\frac{\partial p_t(\theta)}{\partial t}=\sigma^2\Delta p_t(\theta), \tag{26}$$

$$\frac{\partial p'_t(\theta)}{\partial t}=\sigma^2\Delta p'_t(\theta) \tag{27}$$

We denote $E_\alpha(p_t(\theta)\|p'_t(\theta)) = \int \frac{p_t(\theta)^\alpha}{p'_t(\theta)^{\alpha-1}} d\theta$ to be the moment of the likelihood ratio function. Then by definition of Rényi divergence, we prove

$$R_\alpha(p_t(\theta)\|p'_t(\theta)) = \frac{1}{\alpha-1}\log E_\alpha(p_t(\theta)\|p'_t(\theta)) \tag{28}$$

Therefore we compute the rate of Rényi divergence with regard to t as follows.

$$\frac{\partial}{\partial t}R_\alpha(p_t(\theta)\|p'_t(\theta)) = \frac{1}{\alpha-1}\frac{\partial}{\partial t}\log E_\alpha(p_t(\theta)\|p'_t(\theta)) \tag{29}$$

$$= \frac{1}{(\alpha-1)E_\alpha(p_t(\theta)\|p'_t(\theta))}\cdot\frac{\partial}{\partial t}E_\alpha(p_t(\theta)\|p'_t(\theta)) \tag{30}$$

$$\text{By definition of } E_\alpha(p_t(\theta)\|p'_t(\theta)), \quad = \frac{1}{(\alpha-1)E_\alpha(p_t(\theta)\|p'_t(\theta))}\cdot\frac{\partial}{\partial t}\left(\int \frac{p_t(\theta)^\alpha}{p'_t(\theta)^{\alpha-1}}d\theta\right) \tag{31}$$

By exchanging the order of derivative and integration, we prove

$$\frac{\partial}{\partial t}R_\alpha(p_t(\theta)\|p'_t(\theta)) = \frac{1}{(\alpha-1)E_\alpha(p_t(\theta)\|p'_t(\theta))}\cdot\int\frac{\partial}{\partial t}\frac{p_t(\theta)^\alpha}{p'_t(\theta)^{\alpha-1}}d\theta \tag{32}$$

$$= \frac{1}{(\alpha-1)E_\alpha(p_t(\theta)\|p'_t(\theta))}\cdot\int\left(\alpha\cdot\frac{p_t(\theta)^{\alpha-1}}{p'_t(\theta)^{\alpha-1}}\cdot\frac{\partial p_t(\theta)}{\partial t} - (\alpha-1)\cdot\frac{p_t(\theta)^\alpha}{p'_t(\theta)^\alpha}\cdot\frac{\partial p'_t(\theta)}{\partial t}\right)d\theta \tag{33}$$

By the Fokker-Planck equations Equation (26) and Equation (27), we substitute the terms $\frac{\partial p_t(\theta)}{\partial t}$ and $\frac{\partial p'_t(\theta)}{\partial t}$ in the above equation as follows.

$$\frac{\partial}{\partial t}R_\alpha(p_t(\theta)\|p'_t(\theta)) \tag{34}$$

$$= \frac{1}{(\alpha-1)E_\alpha(p_t(\theta)\|p'_t(\theta))}\cdot\int\left(\alpha\sigma^2\cdot\frac{p_t(\theta)^{\alpha-1}}{p'_t(\theta)^{\alpha-1}}\cdot\Delta p_t(\theta) - (\alpha-1)\sigma^2\cdot\frac{p_t(\theta)^\alpha}{p'_t(\theta)^\alpha}\cdot\Delta p'_t(\theta)\right)d\theta \tag{35}$$

By applying Green's first identity in Equation (35), the first intergration term in Equation (35) is changed to

$$\int\alpha\sigma^2\cdot\frac{p_t(\theta)^{\alpha-1}}{p'_t(\theta)^{\alpha-1}}\cdot\Delta p_t(\theta)d\theta = \lim_{r\to\infty}\int_{B_r}\alpha\sigma^2\cdot\frac{p_t(\theta)^{\alpha-1}}{p'_t(\theta)^{\alpha-1}}\cdot\Delta p_t(\theta)d\theta \tag{36}$$

$$= \lim_{r\to\infty}\int_{\partial B_r}\alpha\sigma^2\cdot\frac{p_t(\theta)^{\alpha-1}}{p'_t(\theta)^{\alpha-1}}\cdot\nabla p_t(\theta)\cdot d\mathbf{S} - \int\nabla\left(\alpha\sigma^2\cdot\frac{p_t(\theta)^{\alpha-1}}{p'_t(\theta)^{\alpha-1}}\right)\cdot\nabla p_t(\theta)d\theta, \tag{37}$$

where B_r is the unit ball centered around origin in d-dimensional Euclidean space with radius r. The limits in the first term of Equation (37) becomes zero given the smoothness and fast decay properties of $p_t(\theta)$, and the Lebesgue integrability of $\frac{p_t(\theta)^{\alpha-1}}{p'_t(\theta)^{\alpha-1}}\cdot p_t(\theta)$. Therefore we prove that

$$\int\alpha\sigma^2\cdot\frac{p_t(\theta)^{\alpha-1}}{p'_t(\theta)^{\alpha-1}}\cdot\Delta p_t(\theta)d\theta = -\int\nabla\left(\alpha\sigma^2\cdot\frac{p_t(\theta)^{\alpha-1}}{p'_t(\theta)^{\alpha-1}}\right)\cdot\nabla p_t(\theta)d\theta, \tag{38}$$

Similarly by applying Green's first identity in Equation (35), the second intergration term in Equation (35) is changed to

$$\int -(\alpha-1)\sigma^2\cdot\frac{p_t(\theta)^\alpha}{p'_t(\theta)^\alpha}\cdot\Delta p'_t(\theta)d\theta = \int\nabla\left((\alpha-1)\sigma^2\cdot\frac{p_t(\theta)^\alpha}{p'_t(\theta)^\alpha}\right)\cdot\nabla p'_t(\theta)d\theta, \tag{39}$$

(This techinque for using Green's identity and bounding the rate of entropy change with relative Fisher information-like quantity, has been previously used for KL divergence in Lyu [26], and Rényi divergence in Vempala and Wibisono [37], Chourasia et al. [15]. The result is also closely related to the well known *de Bruijn's indentity* in information theory literature.)

By plugging in Equation (38) and Equation (39) into Equation (35), we prove that

$$\frac{\partial}{\partial t} R_\alpha(p_t(\theta)\|p'_t(\theta)) \tag{40}$$

$$= \frac{\sigma^2}{(\alpha-1)E_\alpha(p_t(\theta)\|p'_t(\theta))} \cdot \int -\alpha \cdot \left\langle \nabla \left(\frac{p_t(\theta)^{\alpha-1}}{p'_t(\theta)^{\alpha-1}}\right), \nabla p_t(\theta) \right\rangle + (\alpha-1) \cdot \left\langle \nabla \left(\frac{p_t(\theta)^{\alpha}}{p'_t(\theta)^{\alpha}}\right), \nabla p'_t(\theta) \right\rangle d\theta \tag{41}$$

$$= \frac{\alpha(\alpha-1)\sigma^2}{(\alpha-1)E_\alpha(p_t(\theta)\|p'_t(\theta))} \cdot \int \frac{p_t(\theta)^{\alpha-2}}{p'_t(\theta)^{\alpha-2}} \cdot \left\langle \nabla \left(\frac{p_t(\theta)}{p'_t(\theta)}\right), -\nabla p_t(\theta) + \frac{p_t(\theta)}{p'_t(\theta)} \nabla p'_t(\theta) \right\rangle d\theta \tag{42}$$

$$= -\frac{\alpha\sigma^2}{E_\alpha(p_t(\theta)\|p'_t(\theta))} \cdot \int \frac{p_t(\theta)^{\alpha-2}}{p'_t(\theta)^{\alpha-2}} \cdot \left\langle \nabla \left(\frac{p_t(\theta)}{p'_t(\theta)}\right), \nabla \left(\frac{p_t(\theta)}{p'_t(\theta)}\right) \right\rangle \cdot p'_t(\theta) d\theta \tag{43}$$

$$= -\alpha\sigma^2 \cdot \frac{I_\alpha(p_t(\theta)\|p'_t(\theta))}{E_\alpha(p_t(\theta)\|p'_t(\theta))}, \tag{44}$$

where $I_\alpha(p_t(\theta)\|p'_t(\theta)) = \int \frac{p_t(\theta)^{\alpha-2}}{p'_t(\theta)^{\alpha-2}} \cdot \left\langle \nabla \left(\frac{p_t(\theta)}{p'_t(\theta)}\right), \nabla \left(\frac{p_t(\theta)}{p'_t(\theta)}\right) \right\rangle \cdot p'_t(\theta) d\theta$. Therefore, if $p_t(\theta)$ and $p'_t(\theta)$ satisfy log-Sobolev inequality with constant c_t, then by Lemma C.1, we obtain the following inequality.

$$\frac{I_\alpha(p_t(\theta)\|p'_t(\theta))}{E_\alpha(p_t(\theta)\|p'_t(\theta))} \geq \frac{2c_t}{\alpha^2} R_\alpha(p_t(\theta)\|p'_t(\theta)) + \frac{2c_t}{\alpha^2} \cdot \alpha(\alpha-1) \frac{\partial}{\partial \alpha} R_\alpha(p_t(\theta)\|p'_t(\theta)) \tag{45}$$

Meanwhile, by Lemma 16 in Vempala and Wibisono [37] and Lemma 17 in Vempala and Wibisono [37], the distributions (with densities $p_t(\theta)$ and $p'_t(\theta)$) indeed satisfy c_t-log-Sobolev inequality with

$$c_t = \left(\frac{L^2}{c} + 2t\sigma^2\right)^{-1} \tag{46}$$

By plugging Equation (46) and Equation (45) into Equation (44), we prove the following bound for the rate of Rényi divergence.

$$\frac{\partial}{\partial t} R_\alpha(p_t(\theta)\|p'_t(\theta)) \leq -\alpha\sigma^2 \cdot \left(\frac{2c_t}{\alpha^2} R_\alpha(p_t(\theta)\|p'_t(\theta)) + \frac{2c_t}{\alpha^2} \cdot \alpha(\alpha-1) \cdot \frac{\partial}{\partial \alpha} R_\alpha(p_t(\theta)\|p'_t(\theta))\right) \tag{47}$$

$$= -c_t \cdot 2\sigma^2 \cdot \left(\frac{R_\alpha(p_t(\theta)\|p'_t(\theta))}{\alpha} + (\alpha-1) \cdot \frac{\partial}{\partial \alpha} R_\alpha(p_t(\theta)\|p'_t(\theta))\right) \tag{48}$$

where $c_t = \left(\frac{L^2}{c} + 2t\sigma^2\right)^{-1}$ is the log-Sobolev inequality constant for distributions $p_t(\theta)$ and $p'_t(\theta)$. □

C.2 Proof for Lemma 4.2

Reduce analysis to point initialization:. Without loss of generality, in this paper, we only analyze recursive Rényi DP bounds for Algorithm 1 under an arbitrary point initialization for initial parameters θ_0^0. This is because, under an arbitrary initialization distribution, the last-iterate parameters θ_K^0 in Algorithm 1 follow a mixture distribution, with each component being the conditional output distribution given fixed initial parameters θ_0^0. Therefore, by the quasi-convexity of Rényi divergence, the largest (worst-case) Rényi DP bound for $R_\alpha(p(\theta_K^0|\theta_0^0)\|p({\theta'}_K^0|\theta_0^0))$ over all possible initial parameters θ_0^0, is also an upper bound for the Rényi privacy loss $R_\alpha(\theta_K^0\|{\theta'}_K^0)$ between (mixture) last-iterate parameters distributions for running Algorithm 1 on two neighboring datasets.

We now proceed to prove the recursive privacy bound Lemma 4.2.

Lemma 4.2. Let D, D' be an arbitrary pair of neighboring datasets that differ in the i_0-th data point (i.e. $x_{i_0} \neq x'_{i_0}$). Let B_k^j be a fixed minibatch used (in iteration j of epoch k) in Algorithm 1,

which contains b indices sampled from $\{1, \cdots, n\}$. We denote θ_k^j and ${\theta'}_k^j$ as the intermediate parameters in Algorithm 1 on input datasets D and D', respectively. If the distributions of θ_k^j and ${\theta'}_k^j$ satisfy log-Sobolev inequality with a constant c, and if the minibatch GD mapping $f(\theta) = \theta - \eta \cdot \frac{1}{b} \cdot \sum_{i \in B_k^j} \ell(\theta; \mathbf{x}_i)$ is L-Lipschitz for parameters θ, then the following recursive bound for Rényi divergence holds.

$$\frac{R_\alpha(\theta_k^{j+1} \| {\theta'}_k^{j+1})}{\alpha} \leq \begin{cases} \frac{R_{\alpha'}(\theta_k^j \| {\theta'}_k^j)}{\alpha'} \cdot \left(1 + \frac{c \cdot 2\eta\sigma^2}{L^2}\right)^{-1} & \text{if } i_0 \notin B_k^j \\ \frac{R_\alpha(\theta_k^j \| {\theta'}_k^j)}{\alpha} + \frac{\eta S_g^2}{4\sigma^2 b^2} & \text{if } i_0 \in B_k^j \end{cases} \text{ with } \alpha' = \frac{\alpha - 1}{1 + \frac{c \cdot 2\eta\sigma^2}{L^2}} + 1. \tag{49}$$

Proof. 1. When $i_0 \notin B_k^j$, the noisy minibatch gradient descent mapping under both dataset D and D' is written in the same way as $f(\theta) + \mathcal{N}(0, 2\eta\sigma^2 \mathbb{I}_d)$, where $f(\theta) = \theta - \frac{\eta}{b} \cdot \sum_{i \in B_k^j} \nabla \ell(\theta; \mathbf{x}_i)$ (this is because $x_i = x_i'$ for $i \in B_k^j$, when $i_0 \notin B_k^j$). Therefore, we could use Lemma 4.1 and solve Equation (4) on the interval $t \in [0, \eta]$ to obtain the recursive privacy bound, where the Rényi divergence at $t = 0$ (i.e. before the update in iteration j of epoch k takes place) satisfies $R_\alpha(p_0(\theta) \| p_0'(\theta)) \leq \varepsilon_k^j(\alpha)$. We denote function $R(\alpha, t) = R_\alpha(p_t(\theta) \| p_t'(\theta))$. Then Equation (4) is equivalent to the following equation.

$$\begin{cases} \frac{\partial}{\partial t} R(\alpha, t) \leq -\left(\frac{L^2}{c} + 2t\sigma^2\right)^{-1} \cdot 2\sigma^2 \cdot \left(\frac{R(\alpha,t)}{\alpha} + (\alpha - 1) \cdot \frac{\partial}{\partial \alpha} R(\alpha, t)\right), \\ R(\alpha, 0) \leq \varepsilon_k^j(\alpha) \end{cases} \tag{50}$$

By substituting $u(\alpha, t) = \frac{R(\alpha,t)}{\alpha}$ into Equation (50), we prove that Equation (50) is equivalent to the following equation.

$$\begin{cases} \frac{\partial}{\partial t} u(\alpha, t) \leq -\left(\frac{L^2}{c} + 2t\sigma^2\right)^{-1} \cdot 2\sigma^2 \cdot \left(u(\alpha, t) + (\alpha - 1) \cdot \frac{\partial}{\partial \alpha} u(\alpha, t)\right), \\ u(\alpha, 0) \leq \frac{\varepsilon_k^j(\alpha)}{\alpha} \end{cases} \tag{51}$$

By change of variable $y = \ln(\alpha - 1)$, we prove that Equation (51) is equivalent to the following equation.

$$\begin{cases} \frac{\partial}{\partial t} u(y, t) \leq -\left(\frac{L^2}{c} + 2t\sigma^2\right)^{-1} \cdot 2\sigma^2 \cdot \left(u(y, t) + \frac{\partial}{\partial y} u(y, t)\right), \\ u(y, 0) \leq \frac{\varepsilon_k^j(e^y + 1)}{e^y + 1} \end{cases} \tag{52}$$

Now we do change of variable $\begin{cases} \tau = t \\ z = y - \int_0^t \left(\frac{L^2}{c} + 2t'\sigma^2\right)^{-1} \cdot 2\sigma^2 dt' + \ln\left(1 + \frac{c \cdot 2\eta\sigma^2}{L^2}\right) \end{cases}$ then by chain rule, we prove the following expressions for the partial derivatives using new variables.

$$\begin{cases} \frac{\partial u}{\partial t} = \frac{\partial u}{\partial \tau} \cdot \frac{\partial \tau}{\partial t} + \frac{\partial u}{\partial z} \cdot \frac{\partial z}{\partial t} = \frac{\partial u}{\partial \tau} - \frac{\partial u}{\partial z} \cdot \left(\frac{L^2}{c} + 2t\sigma^2\right)^{-1} \cdot 2\sigma^2 \\ \frac{\partial u}{\partial y} = \frac{\partial u}{\partial \tau} \cdot \frac{\partial \tau}{\partial y} + \frac{\partial u}{\partial z} \cdot \frac{\partial z}{\partial y} = \frac{\partial u}{\partial z} \end{cases} \tag{53}$$

By plugging Equation (53) into Equation (52), we prove that the partial differential inequality 52 is equivalent to the following inequality under new variables τ and z.

$$\begin{cases} \frac{\partial}{\partial \tau} u(z, \tau) \leq -\left(\frac{L^2}{c} + 2\tau\sigma^2\right)^{-1} \cdot 2\sigma^2 \cdot u(z, \tau), \\ u(z, 0) \leq \frac{\varepsilon_k^j\left(e^{z - \ln\left(1 + \frac{c \cdot 2\eta\sigma^2}{L^2}\right)} + 1\right)}{e^{z - \ln\left(1 + \frac{c \cdot 2\eta\sigma^2}{L^2}\right)} + 1} \end{cases} \tag{54}$$

Now we observe that given any fixed z, Equation (54) is an ordinary differential equation with regard to τ, with a decay term that proportional to $-\left(\frac{L^2}{c}+2\tau\sigma^2\right)^{-1}\cdot 2\sigma^2\cdot u(z,\tau)$. Therefore, we directly solve Equation (54), and prove that

$$\ln(u(z,\tau))-\ln(u(z,0))\leq -\int_0^{\tau}\left(\frac{L^2}{c}+2\tau'\sigma^2\right)^{-1}\cdot 2\sigma^2 d\tau' = -\ln\left(1+\frac{c\cdot 2\tau\sigma^2}{L^2}\right) \tag{55}$$

We take $\tau=\eta$ in Equation (55), then we prove

$$\ln u(z,\eta)-\ln u(z,0)\leq -\ln\left(1+\frac{c\cdot 2\eta\sigma^2}{L^2}\right) \tag{56}$$

By plugging the initial condition for $u(z,0)$ in Equation (54) into Equation (56), we prove that the solution for u at the end of a set $\tau=\eta$ satisfies the following inequality.

$$u(z,\eta)\leq \frac{\varepsilon_k^j\left(e^{z-\ln\left(1+\frac{c\cdot 2\eta\sigma^2}{L^2}\right)}+1\right)}{e^{z-\ln\left(1+\frac{c\cdot 2\eta\sigma^2}{L^2}\right)}+1}\cdot\left(1+\frac{c\cdot 2\eta\sigma^2}{L^2}\right)^{-1} \tag{57}$$

Now we translate the variables $z=z,\tau=\eta$ back to the old variables y and t by definitions, and prove that $t=\tau=\eta$ and $z=y-\int_0^t\left(\frac{L^2}{c}+2t'\sigma^2\right)^{-1}\cdot 2\sigma^2 dt'+\ln\left(1+\frac{c\cdot 2\eta\sigma^2}{L^2}\right)=y$. Therefore, under these variable substitutions, we prove that Equation (56) is equivalent to the following equation.

$$u(y,\eta)\leq \frac{\varepsilon_k^j\left(e^{y-\ln\left(1+\frac{c\cdot 2\eta\sigma^2}{L^2}\right)}+1\right)}{e^{y-\ln\left(1+\frac{c\cdot 2\eta\sigma^2}{L^2}\right)}+1}\cdot\left(1+\frac{c\cdot 2\eta\sigma^2}{L^2}\right)^{-1} \tag{58}$$

Finally, we translate the variable y back to α, under the fixed variable $t=\eta$, by the definition $y=\log(\alpha-1)$. Therefore, we prove that Equation (58) is equivalent to the following solution for $u(\alpha,\eta)$.

$$u(\alpha,\eta)\leq \frac{\varepsilon_k^j\left((\alpha-1)\cdot\left(1+\frac{c\cdot 2\eta\sigma^2}{L^2}\right)^{-1}+1\right)}{(\alpha-1)\cdot\left(1+\frac{c\cdot 2\eta\sigma^2}{L^2}\right)^{-1}+1}\cdot\left(1+\frac{c\cdot 2\eta\sigma^2}{L^2}\right)^{-1} \tag{59}$$

By the definition $u(\alpha,t)=\frac{R(\alpha,t)}{\alpha}$, we prove that Equation (59) is equivalent to the following solution for $R(\alpha,t)$.

$$\frac{R(\alpha,\eta)}{\alpha}\leq \frac{\varepsilon_k^j\left((\alpha-1)\cdot\left(1+\frac{c\cdot 2\eta\sigma^2}{L^2}\right)^{-1}+1\right)}{(\alpha-1)\cdot\left(1+\frac{c\cdot 2\eta\sigma^2}{L^2}\right)^{-1}+1}\cdot\left(1+\frac{c\cdot 2\eta\sigma^2}{L^2}\right)^{-1} \tag{60}$$

Therefore, by using the definition $R(\alpha,\eta)=\varepsilon_k^{j+1}(\alpha)$, we finish the proof for the Lemma statement, that when $i_0\neq B_k^j$,

$$\frac{R_\alpha(\theta_k^{j+1}\|\theta'^{j+1}_k)}{\alpha}\leq \frac{R_{\alpha'}(\theta_k^{j}\|\theta'^{j}_k)}{\alpha'}\cdot\left(1+\frac{c\cdot 2\eta\sigma^2}{L^2}\right)^{-1} \tag{61}$$

where $\alpha'=(\alpha-1)\cdot\left(1+\frac{c\cdot 2\eta\sigma^2}{L^2}\right)^{-1}+1$.

2. When $i_0 \in B_k^j$, by composition theorem for Rényi differential privacy [27], and by the Rényi privacy bound for Gaussian mechanism [27] under ℓ_2-sensitivity S_g/b (for batch averaged gradient) and noise $\mathcal{N}(0, 2\eta\sigma^2\mathbb{I}_d)$, we prove that

$$R_\alpha(\theta_k^{j+1} \| {\theta'}_k^{j+1}) \leq R_\alpha(\theta_k^j, \theta_k^{j+1} \| {\theta'}_k^j, {\theta'}_k^{j+1}) \tag{62}$$

$$\leq R_\alpha(\theta_k^j \| {\theta'}_k^j) + \frac{\eta^2 (S_g/b)^2}{2 \cdot 2\eta\sigma^2} = R_\alpha(\theta_k^j \| {\theta'}_k^j) + \frac{\alpha\eta S_g^2}{4\sigma^2 b^2}. \tag{63}$$

□

C.3 Proof for LSI sequence for noisy minibatch gradient descent

To apply the recurisve privacy bound Lemma 4.2 and prove a converging privacy dynamics, we first need to prove that the distributions of parameters θ_k^j in Algorithm 1 satisfy log-Sobolev inequality with certain constant c_k^j, that depends on k and j. In this section, we prove that noisy minibatch gradient descent on convex smooth loss function, as well as strongly convex smooth loss functions,satusfies LSI with certain sequences of constants as follows.

Lemma C.2 (LSI constant sequence in Algorithm 1 for convex smooth loss)**.** *Suppose that the loss function $\ell(\theta; \mathbf{x})$ in Algorithm 1 is convex and β-smooth. If the step-size $\eta < \frac{2}{\beta}$, then for any $k = 0, \cdots, K-1$ and $j = 0, \cdots, n/b - 1$, the distribution of parameters θ_k^j in Algorithm 1 satisfies c_k^j-log-Sobolev inequality with*

$$c_k^j = \frac{1}{2\eta\sigma^2 \cdot (k \cdot n/b + j)}, \tag{64}$$

and we define $c_0^0 = \frac{1}{0} = +\infty$.

Proof. The minibatch noisy gradient descent update could be written as $\theta_k^{j+1} = f(\theta_k^j) + \mathcal{N}(0, 2\eta\sigma^2\mathbb{I}_d)$, where f is a deterministic mapping on $\mathbb{R}^d$ written as $f(\theta) = \theta - \eta \cdot \frac{\sum_{i \in B} \ell(\theta; \mathbf{x}_i)}{b}$, and B is a minibatch of size b consisting of indices selected from $0, \cdots, n$.

Because the initialization is point distribution around θ_0, therefore θ_0^0 satisfies log-Sobolev inequality with constant $c_0 = \infty$.

By the β-smoothness of $\ell(\theta; \mathbf{x})$, and by $\eta < \frac{2}{\beta}$, we prove that the minibatch gradient mapping $f(\theta)$ is 1-Lipschitz. Further using LSI under Lipchitz mapping (Lemma 16 in Vempala and Wibisono [37]) and under Gaussian convolution (Lemma 17 in Vempala and Wibisono [37]), we prove that

$$\frac{1}{c_k^j} = \frac{1}{c_k^{j-1}} + 2\eta\sigma^2 \tag{65}$$

$$= \frac{1}{c_0^0} + 2\eta\sigma^2 \cdot (k \cdot n/b + j) \tag{66}$$

$$\text{by } c_0^0 = +\infty, \quad = 2\eta\sigma^2 \cdot (k \cdot n/b + j) \tag{67}$$

$$\tag{68}$$

This suffices to prove the LSI sequence in Equation (64). □

Similarly, we prove another LSI sequence for noisy minibatch gradient descent on strongly convex smooth loss function.

Lemma C.3 (LSI constant sequence in Algorithm 1 for strongly convex smooth loss)**.** *Suppose the loss function $\ell(\theta; \mathbf{x})$ in Algorithm 1 is λ-strongly convex and β-smooth. If the step-size $\eta < \frac{2}{\lambda+\beta}$, then for any $k = 0, \cdots, K-1$ and $j = 0, \cdots, n/b - 1$, the distribution of parameters θ_k^j in Algorithm 1 satisfies c_k^j-log-Sobolev inequality with*

$$c_k^j = \frac{1}{2\eta\sigma^2} \cdot \frac{1}{\sum_{s=0}^{k \cdot n/b + j - 1} (1 - \eta\lambda)^{2s}} \tag{69}$$

Proof. The minibatch noisy gradient descent update could be written as $\theta_k^{j+1} = f(\theta_k^j) + \mathcal{N}(0, 2\eta\sigma^2\mathbb{I}_d)$, where f is a deterministic mapping on $\mathbb{R}^d$ written as $f(\theta) = \theta - \eta \cdot \frac{\sum_{i\in B} \ell(\theta;\mathbf{x}_i)}{b}$, and B is a minibatch of size b consisting of indices selected from $0, \cdots, n$.

Because the initialization is point distribution around θ_0, therefore θ_0^0 satisfies log-Sobolev inequality with constant $c_0 = \infty$.

By λ-strong convexity and β-smoothness of $\ell(\theta; \mathbf{x})$, and by $\eta < \frac{2}{\lambda+\beta}$, we prove that the minibatch gradient mapping $f(\theta)$ is $1 - \eta\lambda$-Lipschitz. Further using LSI under Lipchitz mapping (Lemma 16 in Vempala and Wibisono [37]) and under Gaussian convolution (Lemma 17 in Vempala and Wibisono [37]), we prove that

$$\frac{1}{c_k^j} = \frac{(1-\eta\lambda)^2}{c_k^{j-1}} + 2\eta\sigma^2 \tag{70}$$

$$= \frac{(1-\eta\lambda)^{2(k\cdot n/b+j)}}{c_0^0} + 2\eta\sigma^2 \cdot \sum_{s=0}^{k\cdot n/b+j-1} (1-\eta\lambda)^{2s} \tag{71}$$

$$\text{by } c_0^0 = +\infty, \quad = 2\eta\sigma^2 \cdot \sum_{s=0}^{k\cdot n/b+j-1} (1-\eta\lambda)^{2s} \tag{72}$$

This suffices to prove the LSI sequence in Equation (69). □

C.4 Equivalence Between Lemma 4.2 (Ours) And The Bound in Feldman et al. [18]

We now plug the LSI constant sequence derived in Lemma C.2 into the recursive privacy bound Lemma 4.2, and prove the following privacy dynamics theorems for noisy minibatch gradient descent under smooth convex loss functions.

Theorem C.4 (Privacy dynamics under convex smooth loss)**.** *Under fixed minibatches $B^0, \cdots, B^{n/b-1}$, if the loss function $\ell(\theta; x)$ is convex, β-smooth, and if its gradient has finite ℓ_2-sensitivity S_g, then Algorithm 1 with step-size $\eta < \frac{2}{\beta}$ satisfies (α, ε)-Rényi DP for data points in the batch B^{j_0}, with*

$$\varepsilon_j \leq \frac{\alpha\eta S_g^2}{4\sigma^2 b^2} \cdot \frac{b}{n} \cdot (K-1) + \frac{\alpha\eta S_g^2}{4\sigma^2 b^2} \cdot \frac{1}{n/b - j_0} \tag{73}$$

$$\tag{74}$$

Proof. We first observe that for any batch index $j_0 = 0, \cdots, n/b-1$, the privacy bound for data points in batch B^{j_0} in Algorithm 1 (that has K epochs) is equivalent to the privacy bound for data points in the batch B^0 after $K-1$ epochs and $n/b - j_0$ iterations in Algorithm 1. That is, if $R_\alpha(\theta_k^j \| {\theta'}_k^j) \leq \varepsilon_k^j(\alpha)$ is an upper bound for the Rényi divergence between distributions of parameters θ_k^j and ${\theta'}_k^j$ in running Algorithm 1 on neighboring datasets D and D', when the differing data point between D and D' is contained in the minibatch B^0, then the privacy bound ε for data points in the minibatch B^{j_0} of Algorithm 1 satisfies

$$\varepsilon \leq \varepsilon_{K-1}^{n/b-j_0}(\alpha) \leq \varepsilon_{K-1}^0(\alpha) + \varepsilon_0^{n/b-j_0}(\alpha) \tag{75}$$

$$\leq (K-1) \cdot \varepsilon_0^{n/b}(\alpha) + \varepsilon_0^{n/b-j_0}(\alpha), \tag{76}$$

where the last two inequalities are by composition of Rényi DP guarantees.

Therefore, in the remaining proof, we only prove upper bounds for the terms $\varepsilon_0^j(\alpha)$ for $j = 1, \cdots, n/b$, that are required by Equation (76) for bounding ε.

By Lemma C.2, for any $k = 0, \cdots, K-1$ and $j = 0, \cdots, n/b-1$, the distribution of parameters θ_k^j satisfies log-Sobolev inequality with the following constant c_k^j.

$$c_k^j = \frac{1}{2\eta\sigma^2 \cdot (k \cdot n/b + j)} \tag{77}$$

By plugging the LSI constant sequence $\{c_k^j\}$ into Lemma 4.2, and by the noisy minibatch gradient descent mapping, under convex smooth loss with stepsize $\eta < \frac{2}{\beta}$, is 1-Lipschitz, we prove the following recursive bound for $\varepsilon_k^j(\alpha)$, i.e. the Rényi DP bound for data points in the minibatch B^0 of Algorithm 1. For any $k = 0, \cdots, K-1$ and any $j = 0, \cdots, n/b - 1$,

$$\varepsilon_0^0(\alpha) = 0 \tag{78}$$

$$\frac{\varepsilon_k^{j+1}(\alpha)}{\alpha} \leq \begin{cases} \frac{\varepsilon_k^j(\alpha)}{\alpha} + \frac{\eta S_g^2}{4\sigma^2 b^2} & \text{if } j = 0 \\ \frac{\varepsilon_k^j(\alpha')}{\alpha'} \cdot \frac{k \cdot n/b + j}{k \cdot n/b + j + 1} & \text{if } j = 1, \cdots, n/b - 1 \end{cases} \tag{79}$$

$$\text{where } \alpha' = (\alpha - 1) \cdot \left(1 + \frac{c \cdot 2\eta\sigma^2}{L^2}\right)^{-1} + 1 \tag{80}$$

$$\varepsilon_{k+1}^0(\alpha) = \varepsilon_k^{n/b}(\alpha) \tag{81}$$

By solving Equation (79) under $k = 0$, we prove that for any $j = 1, \cdots, n/b$, and for any $\alpha > 1$,

$$\frac{\varepsilon_0^j(\alpha)}{\alpha} \leq \frac{\eta S_g^2}{4\sigma^2 b^2} \cdot \prod_{j'=1}^{j-1} \frac{j'}{j'+1} \tag{82}$$

$$= \frac{\eta S_g^2}{4\sigma^2 b^2} \cdot \frac{1}{j} \tag{83}$$

By plugging Equation (83) into Equation (76), we prove the privacy bound Equation (73) in the theorem statement. □

Theorem C.4 is equivalent to Theorem 23 in Feldman et al. [18] for Algorithm 1 under single-epoch setting (where $K = 1$) with batch-size $b = 1$. For multi-epoch setting with $K = n$ and batch-size $b = 1$, Theorem C.4 is equivalent to Theorem 35 in Feldman et al. [18].

C.5 Proof for Theorem 4.3

We now plug the LSI constant sequence derived in Lemma C.3 into the recursive privacy bound Lemma 4.2, and prove the following privacy dynamics theorems for noisy minibatch gradient descent under strongly convex smooth loss functions.

Theorem 4.3. Conditioned on a fixed sequence of partitioned minibatches $B^0, \cdots, B^{n/b-1}$ in Line 3, if the loss function is λ-strongly convex, β-smooth and its gradient has ℓ_2-sensitivity S_g, then running Algorithm 1 for $K \geq 1$ epochs with step-size $\eta < \frac{2}{\lambda+\beta}$, satisfies (α, ε)-Rényi DP for data points in the batch B^{j_0}, with

$$\varepsilon \leq \varepsilon_0^{\lfloor \frac{n}{2b} \rfloor}(\alpha) \cdot \frac{1 - (1-\eta\lambda)^{2\cdot(K-1)\cdot(n/b - \lfloor \frac{n}{2b} \rfloor)}}{1 - (1-\eta\lambda)^{2\cdot(n/b - \lfloor \frac{n}{2b} \rfloor)}} + \varepsilon_0^{n/b - j_0}(\alpha) \tag{84}$$

where $\varepsilon_0^j(\alpha) = \frac{\alpha\eta S_g^2}{4\sigma^2 b^2} \cdot (1-\eta\lambda)^{2\cdot(j-1)} \cdot \frac{1}{\sum_{s=0}^{j-1}(1-\eta\lambda)^{2s}}$ for any $j = 1, \cdots, \frac{n}{b}$ (we assume $\frac{n}{b} \geq 2$).

Proof. We first observe that for any batch index $j_0 = 0, \cdots, n/b - 1$, the privacy bound for data points in batch B^{j_0} in Algorithm 1 (that has K epochs) is equivalent to the privacy bound for data points in the batch B^0 after $K - 1$ epochs and $n/b - j_0$ iterations in Algorithm 1. That is, if $R_\alpha(\theta_k^j \| \theta'^j_k) \leq \varepsilon_k^j(\alpha)$ is an upper bound for the Rényi divergence between distributions of parameters θ_k^j and θ'^j_k in running Algorithm 1 on neighboring datasets D and D', when the differing data point between D and D' is contained in the minibatch B^0, then the privacy bound ε, for running Algorithm 1 on neighboring datasets D and D' that differs in a point in the minibatch B^{j_0}, satisfies

$$\varepsilon \leq \varepsilon_{K-1}^{n/b-j_0}(\alpha) \leq \varepsilon_{K-1}^0(\alpha) + \varepsilon_0^{n/b-j_0}(\alpha), \tag{85}$$

where the last inequality is by composition of Rényi DP guarantees.

Therefore, in the remaining proof, we only prove upper bounds for the terms $\varepsilon_0^{n/b-1-j_0}(\alpha)$ and $\varepsilon_{K-1}^0(\alpha)$, that are required by Equation (85) for bounding ε.

By Lemma C.3, for any $k = 0, \cdots, K-1$ and $j = 0, \cdots, n/b-1$, the distribution of parameters θ_k^j satisfies log-Sobolev inequality with the following constant c_k^j.

$$c_k^j = \frac{1}{2\eta\sigma^2} \cdot \frac{1}{\sum_{s=0}^{k\cdot n/b+j-1}(1-\eta\lambda)^{2s}} \tag{86}$$

By plugging the LSI constant sequence $\{c_k^j\}$ proved in Lemma C.3 into the proved recursive privacy bound Lemma 4.2, we prove that for running noisy minibatch gradient descent on neighboring datasets that differ in a data point in the first minibatch B^0, the following recursion for the Rényi divergence bound $\varepsilon_k^j(\alpha)$ holds: if the loss function is λ-strongly convex and β-smooth, and if the stepsize $\eta < \frac{2}{\lambda+\beta}$, then for any $k = 0, \cdots, K-1$, any $j = 0, \cdots, n/b-1$, and any $\alpha > 1$,

$$\varepsilon_0^0(\alpha) = 0 \tag{87}$$

$$\frac{\varepsilon_k^{j+1}(\alpha)}{\alpha} \leq \begin{cases} \frac{\varepsilon_k^j(\alpha)}{\alpha} + \frac{\eta S_g^2}{4\sigma^2 b^2} & \text{if } j = 0 \\ \frac{\varepsilon_k^j(\alpha')}{\alpha'} \cdot (1-\eta\lambda)^2 \cdot \frac{\sum_{s=0}^{k\cdot n/b+j-1}(1-\eta\lambda)^{2s}}{\sum_{s=0}^{k\cdot n/b+j}(1-\eta\lambda)^{2s}} & \text{if } j = 1, \cdots, n/b-1 \end{cases} \tag{88}$$

$$\text{where } \alpha' = (\alpha-1) \cdot \left(1 + \frac{c \cdot 2\eta\sigma^2}{L^2}\right)^{-1} + 1 \tag{89}$$

$$\varepsilon_{k+1}^0(\alpha) = \varepsilon_k^{n/b}(\alpha) \tag{90}$$

We now solve the above recursion, thus bounding the privacy loss ε of Algorithm 1 in Equation (85).

1. We first prove a bound for the term $\varepsilon_0^{n/b-j_0}(\alpha)$ in Equation (85). By solving Equation (88) under $k = 0$, we prove that for any $j = 1, \cdots, n/b$, and for any $\alpha > 1$,

$$\frac{\varepsilon_0^j(\alpha)}{\alpha} \leq \frac{\eta S_g^2}{4\sigma^2 b^2} \cdot \prod_{j'=1}^{j-1} \left((1-\eta\lambda)^2 \cdot \frac{\sum_{s=0}^{j'-1}(1-\eta\lambda)^{2s}}{\sum_{s=0}^{j'}(1-\eta\lambda)^{2s}} \right) \tag{91}$$

$$= \frac{\eta S_g^2}{4\sigma^2 b^2} \cdot (1-\eta\lambda)^{2\cdot(j-1)} \cdot \frac{1}{\sum_{s=0}^{j-1}(1-\eta\lambda)^{2s}} \tag{92}$$

2. We now prove a bound for the term $\varepsilon_{K-1}^0(\alpha)$ in Equation (85).

   The term $\varepsilon_k^j(\alpha)$ corresponds to the privacy bound for the data points in the batch B^0 for running Algorithm 1 with k epochs plus j iteration. This is equivalent to the composition $\mathcal{A}_2 \circ \mathcal{A}_1$ of two sub-mechanisms $\mathcal{A}_1$ and $\mathcal{A}_2$, where $\mathcal{A}_1$ corresponds to running Algorithm 1 with $K = k$ epochs, and $\mathcal{A}_2$ corresponds to running Algorithm 1 with only j iterations. Therefore, by the composition theorem for Rényi DP guarantees [27], we prove the following alternative recursive privacy bound for $\varepsilon_k^j(\alpha)$. For any $k = 0, \cdots, K-1$ and any $j = 0, \cdots, n/b-1$,

$$\varepsilon_k^{j+1}(\alpha) \leq \varepsilon_k^0(\alpha) + \varepsilon_0^{j+1}(\alpha) \tag{93}$$

   We now prove an upper bound for $\varepsilon_k^j(\alpha)$, by carefully combining the original recursion Equation (88) and the new alternative recursion Equation (93) (obtained by composition theorem). We use the original recursion Equation (93) for recursively bounding $\varepsilon_k^{j+1}(\alpha)$ during the first half of one epoch, i.e., for $j = 0, \cdots, \lfloor \frac{n}{2b} \rfloor - 1$, and then we use the new alternative recursive bound Equation (88) for the second half of one epoch, i.e., for $j = \lfloor \frac{n}{2b} \rfloor, \cdots, n/b-1$. [6]

[6] This seemingly artificial way of separating one epoch into two halves, and using the recursive bounds separately in each half, is for obtaining a small privacy bound at convergence.

Via this combination, we obtain a new combined recursion for privacy bound as follows. For any $k = 0, \cdots, K-1$,

$$\frac{\varepsilon_k^{j+1}(\alpha)}{\alpha} \leq \begin{cases} \frac{\varepsilon_k^0(\alpha)+\varepsilon_0^{j+1}(\alpha)}{\alpha} & \text{if } j = 0, \cdots, \lfloor \frac{n}{2b} \rfloor - 1 \\ \frac{\varepsilon_k^j(\alpha')}{\alpha'} \cdot (1-\eta\lambda)^2 \cdot \frac{\sum_{s=0}^{k\cdot n/b+j-1}(1-\eta\lambda)^{2s}}{\sum_{s=0}^{k\cdot n/b+j}(1-\eta\lambda)^{2s}} & \text{if } j = \lfloor \frac{n}{2b} \rfloor, \cdots, n/b-1 \end{cases} \tag{94}$$

where $\alpha' = (\alpha-1) \cdot \left(1 + \frac{c\cdot 2\eta\sigma^2}{L^2}\right)^{-1} + 1$.

We now solve this new recursion Equation (94) for one epoch, by accumulating $j = 0, \cdots, n/b-1$. We prove that for any $k = 0, \cdots, K-1$,

$$\frac{\varepsilon_{k+1}^0(\alpha)}{\alpha} = \frac{\varepsilon_k^{n/b}(\alpha)}{\alpha} \tag{95}$$

$$\leq \frac{\varepsilon_k^0(\tilde{\alpha}) + \varepsilon_0^{\lfloor \frac{n}{2b} \rfloor}(\tilde{\alpha})}{\tilde{\alpha}} \cdot \prod_{j'=\lfloor \frac{n}{2b} \rfloor}^{n/b-1} (1-\eta\lambda)^2 \cdot \frac{\sum_{s=0}^{k\cdot n/b+j-1}(1-\eta\lambda)^{2s}}{\sum_{s=0}^{k\cdot n/b+j}(1-\eta\lambda)^{2s}} \tag{96}$$

$$\leq \frac{\left(\varepsilon_k^0(\tilde{\alpha}) + \varepsilon_0^{\lfloor \frac{n}{2b} \rfloor}(\tilde{\alpha})\right)}{\tilde{\alpha}} \cdot (1-\eta\lambda)^{2\cdot(n/b-\lfloor \frac{n}{2b} \rfloor)} \tag{97}$$

where the RDP order $\tilde{\alpha} > 1$ is the $n/b - \lfloor \frac{n}{2b} \rfloor$ fold mapped value of α under repeated mappings $\alpha \leftarrow (\alpha-1) \cdot \left(1 + \frac{c\cdot 2\eta\sigma^2}{L^2}\right)^{-1} + 1$.

We now further solve this new recursion Equation (94) for multiple epochs, by accumulating Equation (97) for $k = 0, 1, \cdots, K-1$. We prove that for any $k = 0, 1, \cdots, K-1$,

$$\frac{\varepsilon_{k+1}^0(\alpha)}{\alpha} \leq \frac{\varepsilon_0^0(\tilde{\alpha})}{\tilde{\alpha}} \cdot (1-\eta\lambda)^{2\cdot(k+1)\cdot(n/b-\lfloor \frac{n}{2b} \rfloor)} + \frac{\varepsilon_0^{\lfloor \frac{n}{2b} \rfloor}(\tilde{\alpha})}{\tilde{\alpha}} \cdot \sum_{k'=1}^{k+1} (1-\eta\lambda)^{2\cdot k'\cdot(n/b-\lfloor \frac{n}{2b} \rfloor)} \tag{98}$$

for some RDP order $\tilde{\alpha} > 1$ that is the $(k+1) \cdot (n/b - \lfloor \frac{n}{2b} \rfloor)$ fold mapped value of α under repeated mapping $\alpha \leftarrow (\alpha-1) \cdot \left(1 + \frac{c\cdot 2\eta\sigma^2}{L^2}\right)^{-1} + 1$.

By further substituting $\varepsilon_0^0(\tilde{\alpha}) = 0$ at the initialization point θ_0 for any $\tilde{\alpha} > 1$ in the above equation, we prove that for any $k = 0, \cdots, K-1$

$$\frac{\varepsilon_{k+1}^0(\alpha)}{\alpha} = \frac{\varepsilon_0^{\lfloor \frac{n}{2b} \rfloor}(\tilde{\alpha})}{\tilde{\alpha}} \cdot (1-\eta\lambda)^{2\cdot(n/b-\lfloor \frac{n}{2b} \rfloor)} \cdot \frac{1-(1-\eta\lambda)^{2\cdot(k+1)\cdot(n/b-\lfloor \frac{n}{2b} \rfloor)}}{1-(1-\eta\lambda)^{2\cdot(n/b-\lfloor \frac{n}{2b} \rfloor)}} \tag{99}$$

for some RDP order $\tilde{\alpha} > 1$.

By setting $k = K-2$ in Equation (99), we prove that for $K \geq 2$

$$\frac{\varepsilon_{K-1}^0(\alpha)}{\alpha} \leq \frac{\varepsilon_0^{\lfloor \frac{n}{2b} \rfloor}(\tilde{\alpha})}{\tilde{\alpha}} \cdot \frac{1-(1-\eta\lambda)^{2\cdot(K-1)\cdot(n/b-\lfloor \frac{n}{2b} \rfloor)}}{1-(1-\eta\lambda)^{2\cdot(n/b-\lfloor \frac{n}{2b} \rfloor)}} \tag{100}$$

for some RDP order $\tilde{\alpha} > 1$. By the format of $\varepsilon_0^j(\alpha)$ for any $\alpha > 1$ in our proof Equation (92) (i.e., $\frac{\varepsilon_0^j(\alpha)}{\alpha}$ is bounded by a constant for all $\alpha > 1$), we prove that Equation (100) is equivalent to the following equation

$$\varepsilon_{K-1}^0(\alpha) \leq \varepsilon_0^{\lfloor \frac{n}{2b} \rfloor}(\alpha) \cdot \frac{1-(1-\eta\lambda)^{2\cdot(K-1)\cdot(n/b-\lfloor \frac{n}{2b} \rfloor)}}{1-(1-\eta\lambda)^{2\cdot(n/b-\lfloor \frac{n}{2b} \rfloor)}} \tag{101}$$

For $K = 1$, by definition, we compute that $\varepsilon_{K-1}^0(\alpha) = 0 =$Right hand side of Equation (101), therefore Equation (101) also holds for $K = 1$.

We now plug our bound Equation (101) into Equation (85), and prove a bound for ε as follows.

$$\varepsilon \leq \varepsilon_{K-1}^{0}(\alpha) + \varepsilon_{0}^{n/b-j_0}(\alpha) \tag{102}$$

$$\leq \varepsilon_{0}^{\lfloor \frac{n}{2b} \rfloor}(\alpha) \cdot \frac{1-(1-\eta\lambda)^{2\cdot(K-1)\cdot(n/b-\lfloor \frac{n}{2b} \rfloor)}}{1-(1-\eta\lambda)^{2\cdot(n/b-\lfloor \frac{n}{2b} \rfloor)}} + \varepsilon_{0}^{n/b-j_0}(\alpha) \tag{103}$$

where the term $\varepsilon_0^j(\alpha)$ is upper bounded by Equation (92). This gives the privacy bound Equation (6) in the theorem statement. □

C.6 Explanations for the privacy bound derived from Balle et al. [4] in Figure 1

For convenience, we first translate [4, Theorem 5] into the symbols used in this paper, as well as under minibatch with size $b > 1$, as follows.

Theorem C.5 (Balle et al. [4]). *Let $\ell(\theta, \mathbf{x})$ be an L-Lipschitz, β-smooth, λ-strongly convex loss function. If $\eta \leq \frac{2}{\beta+\lambda}$, then conditioned on a fixed sequence of partitioned minibatches $B^0, \cdots, B^{n/b-1}$ in Line 3, Algorithm 1 satisfies (α, ε)-Rényi DP for data points in the minibatch j_0, with*

$$\varepsilon = \begin{cases} \alpha \cdot \frac{2(L/b)^2}{2\eta\sigma^2} & j_0 = n/b-1 \\ \alpha \cdot \frac{2(L/b)^2}{(n/b-j_0-1)\cdot 2\eta\sigma^2}\left(1-\frac{2\eta\beta\lambda}{\beta+\lambda}\right)^{\frac{n/b-j_0}{2}} & j_0 = 0, \cdots, n/b-2 \end{cases} \tag{104}$$

The variables from Balle et al. [4] that we replaced are as follows: $L \to L/b$, this is because, in Algorithm 1, we additionally average each per-example gradient with the minibatch size b; $i \to j_0 + 1$, this is because the index used in our paper starts from $j_0 = 0$, while the index used in Balle et al. [4] starts from $i = 1$; $n \to n/b$, this is because under minibatch size $b > 1$, the number of iterations in one epoch in our paper is n/b, while the number of iteration in Balle et al. [4] equals the size of the datasets n; $\sigma^2 \to 2\eta\sigma^2$, this is because in our paper, we follow the noise scaling with variance $2\eta\sigma^2$ that is related to stepsize η, as in SGLD, while Balle et al. [4] use the standard Gaussian noise with variance σ^2.

By further substituting the Lipschitz requirement with its equivalent sensitivity assumption, i.e. $L \to \eta S_g/2$, and by using Rényi DP composition [27] over the epochs, we obtain that the privacy bound in Balle et al. [4, Theorem 5] is $\frac{\alpha\cdot\eta S_g^2}{4\cdot(n/b-1)\cdot b^2\sigma^2} \cdot \left(1-\frac{2\eta\beta\lambda}{\beta+\lambda}\right)^{\frac{n}{2b}} \cdot K$ when the differing data point is in the first batch, and the bound is $\frac{\alpha\cdot\eta S_g^2}{4\cdot(n/b-1)\cdot b^2\sigma^2} \cdot \left(1-\frac{2\eta\beta\lambda}{\beta+\lambda}\right)^{\frac{n}{2b}} \cdot (K-1) + \frac{\alpha\cdot\eta S_g^2}{4b^2\sigma^2}$ when the differing data point is in the last batch.

C.7 Revisiting noisy GD: A tighter bound than [15, Corollary 1]

In this section, we prove a new converging hidden-state privacy bound for the noisy GD algorithm, which is a special case of Algorithm 1 under $b = n$. We then show that our bound is slightly tighter than [15, Corollary 1] and also admits a conceptually simpler proof (our proof relies on the privacy amplification by randomized post-processing results in Section 4). We first state our privacy bound and its proof as follows.

Theorem C.6. *Let $\ell(\theta; \mathbf{x})$ be a λ-strongly convex, and β-smoooth loss function, with a finite total gradient sensitivity S_g, then the noisy gradient descent algorithm with step-size $\eta < \frac{1}{\beta}$, satisfies (α, ε) Rényi Differential Privacy with*

$$\varepsilon \leq \frac{\alpha\eta S_g^2}{2\sigma^2 n^2} \cdot \sum_{k=1}^{K}\left(1-\frac{\eta\lambda}{2}\right)^k. \tag{105}$$

Proof. We first offer a new perspective of viewing a noisy GD update as follows. Recall that one noisy GD update in Algorithm 1 is written as

$$\theta_{k+1}^0 = \theta_k^0 - \eta \cdot g(\theta_k^0; D) + \sqrt{2\eta\sigma^2} \cdot \mathcal{N}(0, \mathbb{I}_d) \text{ where } g(\theta_k^j; D) = \frac{1}{n}\sum_{i=1}^{n} \nabla\ell(\theta_k^j; \mathbf{x}_i) \tag{106}$$

Therefore, this update is equivalent to the following two steps:

$$\theta_k^{\frac{1}{2}} = \theta_k^0 - \eta \cdot g(\theta_k^0; D) + \sqrt{\eta\sigma^2} \cdot \mathcal{N}(0, \mathbb{I}_d) \text{ where } g(\theta_k^0; D) = \frac{1}{n}\sum_{i=1}^{n} \nabla \ell(\theta_k^0; \mathbf{x}_i) \tag{107}$$

$$\theta_{k+1}^0 = \theta_k^{\frac{1}{2}} + \sqrt{\eta\sigma^2} \cdot \mathcal{N}(0, \mathbb{I}_d) \tag{108}$$

That is, we view each noisy GD update as another noisy GD update with smaller noise scale followed by pure additive Gaussian noise.

Moreover, by Lemma C.3, we prove that distributions of θ_k^0 satisfies c_k^0-log Sobolev inequality with

$$c_k^0 = \frac{1}{2\eta\sigma^2} \cdot \frac{1}{\sum_{s=0}^{k-1}(1-\eta\lambda)^{2s}} \tag{109}$$

Therefore, by further using LSI under Lipschitz mapping ([37, Lemma 16]) and under Gaussian convolution ([37, Lemma 17]), we prove that the distribution of $\theta_k^{\frac{1}{2}}$ satisfies $c_k^{\frac{1}{2}}$-log Sobolev inequality with

$$\frac{1}{c_k^{\frac{1}{2}}} = \frac{(1-\eta\lambda)^2}{c_k^0} + \eta\sigma^2 \tag{110}$$

$$= 2\eta\sigma^2 \cdot \sum_{s=1}^{k}(1-\eta\lambda)^{2s} + \eta\sigma^2 \tag{111}$$

Therefore, by applying composition on the conceptual step from $\theta_k^0 \to \theta_k^{\frac{1}{2}}$, we prove that

$$\frac{R_\alpha(\theta_k^{\frac{1}{2}} \| {\theta'}_k^{\frac{1}{2}})}{\alpha} \leq \frac{R_\alpha(\theta_k^0 \| {\theta'}_k^0)}{\alpha} + \frac{\eta S_g^2}{2\sigma^2 n^2} \tag{112}$$

For the remaining conceptual step from $\theta_k^{\frac{1}{2}} \to \theta_{k+1}^0$, we note that pure additive Gaussian noise is equivalent to identity mapping convolved with Gaussian noise. Therefore, we apply $i_0 \notin B_k^j$ case of Lemma 4.2 with $L = 1$ (for identity mapping) and noise standard deviation $\frac{\sigma}{\sqrt{2}}$, and prove that

$$\frac{R_\alpha(\theta_{k+1}^0 \| {\theta'}_{k+1}^0)}{\alpha} \leq \frac{R_{\alpha'}(\theta_k^{\frac{1}{2}} \| {\theta'}_k^{\frac{1}{2}})}{\alpha'} \cdot \left(1 + c_k^{\frac{1}{2}} \cdot \eta\sigma^2\right)^{-1} \text{ with } \alpha' = \frac{\alpha - 1}{1 + c_k^{\frac{1}{2}} \cdot \eta\sigma^2} + 1 \tag{113}$$

Therefore, by combining (112) and (113), we prove the following recursive Rényi DP bound.

$$\frac{R_\alpha(\theta_{k+1}^0 \| {\theta'}_{k+1}^0)}{\alpha} \leq \left(\frac{R_{\alpha'}(\theta_k^0 \| {\theta'}_k^0)}{\alpha'} + \frac{\eta S_g^2}{2\sigma^2 n^2}\right) \cdot \left(1 + c_k^{\frac{1}{2}} \cdot \eta\sigma^2\right)^{-1} \text{ with } \alpha' = \frac{\alpha - 1}{1 + c_k^{\frac{1}{2}} \cdot \eta\sigma^2} + 1 \tag{114}$$

By further plugging in the LSI constant sequence (110), we simplify Equation (114) into the following inequality.

$$\frac{R_\alpha(\theta^0_{k+1}\|\theta'^0_{k+1})}{\alpha} \leq \left(\frac{R_{\alpha'}(\theta^0_k\|\theta'^0_k)}{\alpha'} + \frac{\eta S_g^2}{2\sigma^2 n^2}\right)\cdot\left(1 - \frac{1}{2\sum_{s=0}^{k}(1-\eta\lambda)^{2s}}\right) \tag{115}$$

$$\leq \left(\frac{R_{\alpha'}(\theta^0_k\|\theta'^0_k)}{\alpha'} + \frac{\eta S_g^2}{2\sigma^2 n^2}\right)\cdot\left(1 - \frac{1}{2\sum_{s=0}^{+\infty}(1-\eta\lambda)^{2s}}\right) \tag{116}$$

$$= \left(\frac{R_{\alpha'}(\theta^0_k\|\theta'^0_k)}{\alpha'} + \frac{\eta S_g^2}{2\sigma^2 n^2}\right)\cdot\left(1 - \frac{1-(1-\eta\lambda)^2}{2}\right) \tag{117}$$

$$\leq \left(\frac{R_{\alpha'}(\theta^0_k\|\theta'^0_k)}{\alpha'} + \frac{\eta S_g^2}{2\sigma^2 n^2}\right)\cdot\left(1 - \frac{\eta\lambda}{2}\right), \tag{118}$$

where $\alpha' = (\alpha-1)\cdot\left(1 - \frac{1}{2\sum_{s=0}^{k}(1-\eta\lambda)^{2s}}\right) + 1$, and the last inequality Equation (118) is because of the inequality $\eta\lambda < \frac{\lambda}{\beta} \leq 1$ that is ensured by the condition $\eta < \frac{1}{\beta}$.

Therefore, by solving the recursion Equation (118) from $k = 0, \cdots, K-1$, and by using $R_\alpha(\theta^0_0\|\theta'^0_0) = 0$ for any $\alpha > 1$, we prove the theorem statement in Theorem C.6.

$$\frac{R_\alpha(\theta^0_K\|\theta'^0_K)}{\alpha} \leq \frac{\eta S_g^2}{2\sigma^2 n^2}\cdot\sum_{k=1}^{K}\left(1 - \frac{\eta\lambda}{2}\right)^k \tag{119}$$

□

Comparison with [15, Corollary 1] We now compare our privacy bound Theorem C.6 with [15, Corollary 1] and show that our bound is tighter. For convenience, we now repeat the privacy bound for noisy gradient descent (i.e. when $b = n$ in Algorithm 1) that is proved in [15, Corollary 1] below.

Theorem C.7 (Corollary 1 in Chourasia et al. [15])**.** *Let $\ell(\theta;\mathbf{x})$ be a λ-strongly convex, and β-smoooth loss function, with a finite total gradient sensitivity S_g, then the noisy gradient descent algorithm with start parameter $\theta_0 \sim \mathcal{N}(0, \frac{2\sigma^2}{\lambda}\mathcal{I}_d)$, and step-size $\eta < \frac{1}{\beta}$, satisfies (α,ε) Rényi Differential Privacy with*

$$\varepsilon \leq \frac{\alpha S_g^2}{\lambda\sigma^2 n^2}(1 - e^{-\lambda\eta K/2}). \tag{120}$$

Because $1 - x < e^{-x}$ for $x \neq 0$, we are able to further relax the bound in Theorem C.6 as follows.

$$\varepsilon \leq \frac{\alpha\eta S_g^2}{2\sigma^2 n^2}\cdot\sum_{k=1}^{K}\left(1 - \frac{\eta\lambda}{2}\right)^k \tag{121}$$

$$< \frac{\alpha\eta S_g^2}{2\sigma^2 n^2}\cdot\left(1 - \frac{\eta\lambda}{2}\right)\cdot\sum_{k=0}^{K-1} e^{-\frac{\eta\lambda k}{2}} \tag{122}$$

$$= \frac{\alpha\eta S_g^2}{2\sigma^2 n^2}\cdot\left(1 - \frac{\eta\lambda}{2}\right)\cdot\frac{1 - e^{-\lambda\eta K}}{1 - e^{-\eta\lambda/2}} \tag{123}$$

Because $1 - e^{-x} > x\cdot e^{-x}$ for $x > 0$, we prove that $1 - e^{-\eta\lambda/2} \geq \frac{\lambda\eta}{2}\cdot e^{-\frac{\lambda\eta}{2}}$. By plugging this inequality to Equation (123), we prove that

$$\varepsilon < \frac{\alpha S_g^2}{\lambda\sigma^2 n^2}\cdot\frac{1 - \frac{\eta\lambda}{2}}{e^{-\frac{\lambda\eta}{2}}}\cdot(1 - e^{-\lambda\eta K}) \tag{124}$$

By again using $1 - x < e^{-x}$ for $x \neq 0$ into Equation (124), we prove that

$$\varepsilon < \frac{\alpha S_g^2}{\lambda \sigma^2 n^2} \cdot (1 - e^{-\lambda \eta K}) = \text{RHS of [15, Corollary 1]} \tag{125}$$

This shows that the privacy bound Theorem C.6 enabled by our analysis is strictly tighter than [15, Corollary 1].

D Proof for Section 5

Lemma 5.1. Let $\mu_1, \cdots, \mu_m$ and $\nu_1, \cdots, \nu_m$ be measures over R^d. Then for any $\alpha \geq 1$, and any $p_1, \cdots, p_m \geq 0$ that satisfies $p_1 + \cdots + p_m = 1$,

$$e^{(\alpha-1)\cdot R_\alpha(\sum_{j=1}^m p_j \mu_j \| \sum_{j=1}^m p_j \nu_j)} \leq \sum_{j=1}^m p_j \cdot e^{(\alpha-1)\cdot R_\alpha(\mu_j \| \nu_j)} \tag{126}$$

Proof. By definition of Rényi divergence,

$$e^{(\alpha-1)\cdot R_\alpha(\mu\|\nu)} = \int \left(\frac{\mu(\theta)}{\nu(\theta)}\right)^\alpha \cdot \mu(\theta) d\theta \tag{127}$$

Therefore by definition of f-divergence, $e^{(\alpha-1)\cdot R_\alpha(\mu\|\nu)}$ is f-divergence with $f(x) = x^\alpha$. By the convexity of f when $\alpha \geq 1$, and by applying Theorem 3.1 in Taneja and Kumar [35], we prove that the f-divergence $e^{(\alpha-1)\cdot R_\alpha(\mu\|\nu)}$ is jointly convex in arguments μ, ν. □

D.1 Proof for Theorem 5.2

Theorem 5.2 (Privacy dynamics for "shuffle and partition" minibatch gradient descent). If the loss function $\ell(\theta; x)$ is λ-strongly convex, β-smooth, and if its gradient has finite ℓ_2-sensitivity S_g, then for $K \geq 1$ and $\frac{n}{b} \geq 2$, Algorithm 1 with stepsize $\eta < \frac{2}{\lambda+\eta}$ satisfies (α, ε)-Rényi DP for all data points with

$$\varepsilon \leq \varepsilon_0^{\lfloor \frac{n}{2b} \rfloor}(\alpha) \cdot \frac{1 - (1-\eta\lambda)^{2\cdot(K-1)\cdot(n/b - \lfloor \frac{n}{2b} \rfloor)}}{1 - (1-\eta\lambda)^{2\cdot(n/b - \lfloor \frac{n}{2b} \rfloor)}} + \frac{1}{\alpha - 1} \cdot \log \left(\underset{0 \leq j_0 < n/b}{Avg} e^{(\alpha-1)\varepsilon_0^{n/b - j_0}(\alpha)} \right) \tag{128}$$

where the terms $\varepsilon_0^j(\alpha)$ is upper-bounded for any $j = 1, \cdots, n/b$ as follows.

$$\varepsilon_0^j(\alpha) \leq \frac{\alpha \eta S_g^2}{4\sigma^2 b^2} \cdot (1-\eta\lambda)^{2\cdot(j-1)} \cdot \frac{1}{\sum_{s=0}^{j-1}(1-\eta\lambda)^{2s}} \tag{129}$$

Proof. Our proof relies on the joint convexity of the scaled exponentiated Rényi divergence $e^{(\alpha-1)R_\alpha(\mu\|\nu)}$ in its arguments μ and ν). By further using the batch decomposition in Section 3, we prove

$$\varepsilon = \frac{1}{\alpha-1} \log e^{(\alpha-1)\cdot R_\alpha(\theta_K^0 \| {\theta'}_K^0)} \tag{130}$$

$$\leq \frac{1}{\alpha-1} \log \left(\sum_{B^0, \cdots, B^{n/b-1} \text{in Line 3}} p(B^0, \cdots, B^{n/b-1}) \cdot e^{(\alpha-1)\cdot R_\alpha\left(p(\theta_K^0 | B^0, \cdots, B^{n/b-1}) \| p({\theta'}_K^0 | B^0, \cdots, B^{n/b-1})\right)} \right) \tag{131}$$

Therefore, by plugging the privacy bound of Algorithm 1 proved in Theorem 4.3 into Equation (131), depending on which minibatch contains the differing data point with any index i_0, we further prove

that

$$\varepsilon \leq \frac{1}{\alpha-1}\log\left(\sum_{j_0=0}^{n/b-1} p(i_0 \in B^{j_0}) \cdot e^{(\alpha-1)\cdot\left(\varepsilon_0^{\lfloor \frac{n}{2b}\rfloor}(\alpha)\cdot\frac{1-(1-\eta\lambda)^{2\cdot(K-1)\cdot(n/b-\lfloor\frac{n}{2b}\rfloor)}}{1-(1-\eta\lambda)^{2\cdot(n/b-\lfloor\frac{n}{2b}\rfloor)}}+\varepsilon_0^{n/b-j_0}(\alpha)\right)}\right) \tag{132}$$

$$= \frac{1}{\alpha-1}\log\left(\sum_{j_0=0}^{n/b-1} \frac{b}{n} \cdot e^{(\alpha-1)\cdot\left(\varepsilon_0^{\lfloor \frac{n}{2b}\rfloor}(\alpha)\cdot\frac{1-(1-\eta\lambda)^{2\cdot(K-1)\cdot(n/b-\lfloor\frac{n}{2b}\rfloor)}}{1-(1-\eta\lambda)^{2\cdot(n/b-\lfloor\frac{n}{2b}\rfloor)}}+\varepsilon_0^{n/b-j_0}(\alpha)\right)}\right) \tag{133}$$

$$\tag{134}$$

□

Explanations for the terms in Theorem 5.2. The privacy bound Theorem 5.2 is strictly smaller than the composition-based privacy bound (derived from DP-SGD[1] and SGM [28] analysis) after $4 \cdot \frac{1}{\lambda\eta} + 4 \cdot \frac{n}{b}$ epochs. More specifically, the first term in Equation (178) is strictly smaller than the composition-based privacy bound after $4 \cdot \frac{1}{\lambda\eta}$ epochs, and the second term in Equation (178) is strictly smaller than the composition-based privacy bound after $4 \cdot (n/b)$ epochs (where $n/b \geq 2$). We explain the terms more specifically as follows.

1. The sensitivity for one minibatch gradient update in Algorithm 1 is $\eta \cdot \frac{S_g}{2b}$ (under remove-one neighboring dataset notion), the standard deviation of Gaussian noise added in one update is $\sqrt{2\eta\sigma^2}$, and the sampling probability for each data point is $\frac{b}{n}$. By Abadi et al. [1] and Mironov et al. [28], the composition-based privacy bound for one iteration is larger than $\frac{b^2}{n^2}\frac{\alpha\cdot\eta^2\cdot S_g^2/b^2}{2\cdot 2\eta\sigma^2} = \frac{b^2}{n^2}\frac{\alpha\cdot\eta S_g^2}{4\sigma^2 b^2}$. By Rényi DP composition over n/b iterations, the composition-based privacy bound for one epoch is larger than $\frac{b}{n}\cdot\frac{\alpha\cdot\eta S_g^2}{4\sigma^2 b^2}$.

2. The first term is upper bounded as follows, which is smaller than the composition-based privacy bound after approximately $4 \cdot \frac{1}{\lambda\eta}$ epochs.

$$\varepsilon_0^{\lfloor \frac{n}{2b}\rfloor}(\alpha)\cdot\frac{1-(1-\eta\lambda)^{2\cdot(K-1)\cdot(n/b-\lfloor\frac{n}{2b}\rfloor)}}{1-(1-\eta\lambda)^{2\cdot(n/b-\lfloor\frac{n}{2b}\rfloor)}} \tag{135}$$

$$\leq \frac{\alpha\eta S_g^2}{4\sigma^2 b^2}\cdot(1-\eta\lambda)^{2\cdot(\lfloor\frac{n}{2b}\rfloor-1)}\cdot\frac{1}{\sum_{s=0}^{\lfloor\frac{n}{2b}\rfloor-1}(1-\eta\lambda)^{2s}}\cdot\frac{1-(1-\eta\lambda)^{2\cdot(K-1)\cdot(n/b-\lfloor\frac{n}{2b}\rfloor)}}{1-(1-\eta\lambda)^{2\cdot(n/b-\lfloor\frac{n}{2b}\rfloor)}} \tag{136}$$

$$= \frac{\alpha\eta S_g^2}{4\sigma^2 b^2}\cdot(1-\eta\lambda)^{2\cdot(\lfloor\frac{n}{2b}\rfloor-1)}\cdot\frac{1-(1-\eta\lambda)^2}{1-(1-\eta\lambda)^{2\cdot\lfloor\frac{n}{2b}\rfloor}}\cdot\frac{1-(1-\eta\lambda)^{2\cdot(K-1)\cdot(n/b-\lfloor\frac{n}{2b}\rfloor)}}{1-(1-\eta\lambda)^{2\cdot(n/b-\lfloor\frac{n}{2b}\rfloor)}} \tag{137}$$

$$\leq \frac{\alpha\eta S_g^2}{4\sigma^2 b^2}\cdot\frac{1}{1-(1-\eta\lambda)^{2\cdot(n/b-\lfloor\frac{n}{2b}\rfloor)}} \quad \text{(by } (1-\eta\lambda)^x \text{ is monotonically decreasing for } x \in \mathbb{R}) \tag{138}$$

$$\leq \frac{\alpha\eta S_g^2}{4\sigma^2 b^2}\cdot\frac{1}{2\cdot(n/b-\lfloor\frac{n}{2b}\rfloor)\cdot\lambda\eta} \quad \text{(by } (1-x)^a \leq 1-ax \text{ for } x>0 \text{ and } a \geq 1) \tag{139}$$

$$\leq \frac{\alpha\eta S_g^2}{4\sigma^2 b^2}\cdot\frac{1}{n/b\cdot\lambda\eta} \quad \text{(by } \frac{n}{b} \geq 2) \tag{140}$$

3. The second term is upper bounded by $\varepsilon_0^1(\alpha) = \frac{\alpha\eta S_g^2}{4\sigma^2 b^2}$, which is smaller than the composition-based privacy bound after approximately $4 \cdot \frac{n}{b}$ epochs.

D.2 Proof for Theorem 5.3

Theorem 5.3. If the loss function $\ell(\theta; x)$ is λ-strongly convex, β-smooth, and if its gradient has finite ℓ_2-sensitivity S_g, then Algorithm 1 with stepsize $\eta < \frac{2}{\lambda+\beta}$ satisfies (α, ε)-Rényi DP guarantee with

$$\varepsilon \leq \frac{1}{\alpha - 1} \log \left(S_K^0(\alpha) \right) \tag{141}$$

where the term $S_k^j(\alpha)$ is recursively defined by

$$S_0^0(\alpha) = 1 \tag{142}$$

$$S_k^{j+1}(\alpha) = \frac{b}{n} \cdot e^{\frac{(\alpha-1)\alpha\eta S_g^2}{4\sigma^2 b^2}} \cdot S_k^j(\alpha) + (1 - \frac{b}{n}) \cdot S_k^j(\alpha)^{(1-\eta\lambda)^2}, \text{ for } k = 0, \cdots, K-1 \text{ and } j = 0, \cdots, n/b - 1 \tag{143}$$

$$S_{k+1}^0(\alpha) = S_k^{n/b}(\alpha) \text{ for } k = 0, \cdots, K-1 \tag{144}$$

$$\tag{145}$$

Proof. By definition,

$$e^{(\alpha-1)\varepsilon_k^{j+1}(\alpha)} = \int \left(\frac{p_k^{j+1}(\theta)}{p'^{j+1}_k(\theta)} \right)^\alpha p'^{j+1}_k(\theta) d\theta \tag{146}$$

$$= \int \left(\frac{\mathbb{E}_{B_0^0, \cdots, B_k^j} \left[p_k^{j+1}(\theta | B_0^0, \cdots, B_k^j) \right]}{\mathbb{E}_{B_0^0, \cdots, B_k^j} \left[p'^{j+1}_k(\theta | B_0^0, \cdots, B_k^j) \right]} \right)^\alpha \cdot \mathbb{E}_{B_0^0, \cdots, B_k^j} \left[p'^{j+1}_k(\theta | B_0^0, \cdots, B_k^j) \right] d\theta \tag{147}$$

By the joint convexity of the function $\frac{x^\alpha}{y^{\alpha-1}}$ on $x, y > 0$ (Lemma 20 Balle et al. [5]), and by the minibatch decomposition for the distribution of the last iterate parameters in Section 3, we prove

$$e^{(\alpha-1)\varepsilon_k^{j+1}(\alpha)} \leq \mathbb{E}_{B_0^0, \cdots, B_k^j} \left[\int \left(\frac{p_k^{j+1}(\theta | B_0^0, \cdots, B_k^j)}{p'^{j+1}_k(\theta | B_0^0, \cdots, B_k^j)} \right)^\alpha \cdot p'^{j+1}_k(\theta | B_0^0, \cdots, B_k^j) d\theta \right] \tag{148}$$

We now derive a recursive scheme for computing the right hand side term denoted as $S_k^{j+1}(\alpha)$. By definition,

$$S_k^{j+1}(\alpha) = \mathbb{E}_{B_0^0, \cdots, B_k^j} \left[\int \left(\frac{p_k^{j+1}(\theta | B_0^0, \cdots, B_k^j)}{p'^{j+1}_k(\theta | B_0^0, \cdots, B_k^j)} \right)^\alpha \cdot p'^{j+1}_k(\theta | B_0^0, \cdots, B_k^j) d\theta \right] \tag{149}$$

The term inside expectation corresponds to Rényi privacy loss under fixed minibatches $B_0^0, \cdots, B_k^j$. Therefore, by Lemma 4.2,

$$\int \left(\frac{p_k^{j+1}(\theta | B_0^0, \cdots, B_k^j)}{p'^{j+1}_k(\theta | B_0^0, \cdots, B_k^j)} \right)^\alpha \cdot p'^{j+1}_k(\theta | B_0^0, \cdots, B_k^j) d\theta \tag{150}$$

$$\leq \begin{cases} \int \left(\frac{p_k^j(\theta | B_0^0, \cdots, B_k^j)}{p'^j_k(\theta | B_0^0, \cdots, B_k^j)} \right)^\alpha \cdot p'^j_k(\theta | B_0^0, \cdots, B_k^j) d\theta \cdot e^{\frac{(\alpha-1)\alpha\eta S_g^2}{4\sigma^2 b^2}} & \text{if } i_0 \in B_k^j \\ \left(\int \left(\frac{p_k^j(\theta | B_0^0, \cdots, B_k^j)}{p'^j_k(\theta | B_0^0, \cdots, B_k^j)} \right)^\alpha \cdot p'^j_k(\theta | B_0^0, \cdots, B_k^j) d\theta \right)^{(1-\eta\lambda)^2} & \text{if } i_0 \notin B_k^j \end{cases} \tag{151}$$

By plugging Equation (151), and by the definition of $S_k^{j+1}(\alpha)$, we prove

$$S_k^{j+1}(\alpha) = \mathbb{E}_{B_0^0,\cdots,B_k^j}\left[\int\left(\frac{p_k^{j+1}(\theta|B_0^0,\cdots,B_k^j)}{p'^{j+1}_k(\theta|B_0^0,\cdots,B_k^j)}\right)^\alpha \cdot p'^{j+1}_k(\theta|B_0^0,\cdots,B_k^j)d\theta\right] \tag{152}$$

$$= P(i_0 \in B_k^j)\cdot S_k^j \cdot e^{\frac{(\alpha-1)\alpha\eta S_g^2}{4\sigma^2 b^2}} + P(i_0 \notin B_k)\cdot \tag{153}$$

$$\mathbb{E}_{B_0^0,\cdots,B_k^{j-1}}\left[\left(\int\left(\frac{p_k^j(\theta|B_0^0,\cdots,B_k^j)}{p'^j_k(\theta|B_0^0,\cdots,B_k^j)}\right)^\alpha \cdot p'^j_k(\theta|B_0^0,\cdots,B_k^j)d\theta\right)^{(1-\eta\lambda)^2}\right] \tag{154}$$

By further using the concavity of $x^{(1-\eta\lambda)^2}$, and the definition of $S_k^j(\alpha)$, we prove

$$S_k^{j+1}(\alpha) \le P(i_0 \in B_k^j)\cdot S_k^j \cdot e^{\frac{(\alpha-1)\alpha\eta S_g^2}{4\sigma^2 b^2}} + P(i_0 \notin B_k)\cdot \tag{155}$$

$$\left(\mathbb{E}_{B_0^0,\cdots,B_k^{j-1}}\left[\int\left(\frac{p_k^j(\theta|B_0^0,\cdots,B_k^j)}{p'^j_k(\theta|B_0^0,\cdots,B_k^j)}\right)^\alpha \cdot p'^j_k(\theta|B_0^0,\cdots,B_k^j)d\theta\right]\right)^{(1-\eta\lambda)^2} \tag{156}$$

$$= \frac{b}{n}\cdot e^{\frac{(\alpha-1)\alpha\eta S_g^2}{4\sigma^2 b^2}}\cdot S_k^j + \left(1-\frac{b}{n}\right)\cdot (S_k^j)^{(1-\eta\lambda)^2} \tag{157}$$

Therefore, the full recursive scheme for computing $S_k^j(\alpha)$ is as follows.

1. $S_0^0(\alpha) = 1$
2. $S_k^{j+1}(\alpha) = \frac{b}{n}\cdot e^{\frac{(\alpha-1)\alpha\eta S_g^2}{4\sigma^2 b^2}}\cdot S_k^j(\alpha) + (1-\frac{b}{n})\cdot S_k^j(\alpha)^{(1-\eta\lambda)^2}$, for $k = 0,\cdots,K-1$ and $j = 0,\cdots,n/b-1$.
3. $S_{k+1}^0(\alpha) = S_k^{n/b}(\alpha)$ for $k = 0,\cdots,K-1$.

And Algorithm 1 satisfies (α,ε)-Rényi DP guarantee with

$$\varepsilon \le \frac{1}{\alpha-1}\log\left(S_K^0(\alpha)\right) \tag{158}$$

□

E Proofs and Explanations for Section 6

E.1 Pseudocode for Our Implementation

We provide the pseudocode of our implementation in Algorithm 2. For the convenience of implementation in existing privacy libaries, we introduce the noise multiplier σ_{mul} to substitute σ in Algorithm 1. We comment that the noisy gradient update in Algorithm 2 under noise multiplier σ_{mul}, is equivalent to a noisy gradient update in Algorithm 1 with noise standard deviation $\sigma = \sqrt{\frac{q}{2}}\cdot\frac{1}{b}\cdot\sigma_{mul}\cdot\frac{S_g}{2}$. By plugging this equivalent σ into Theorem 5.2, we prove the privacy dynamics bound in Corollary E.3 for regularized logistic regression.

E.2 Implementation details for satisfying strong convexity

Regularzied Logistic regression (for strong convexity). The loss function for regularized logistic regression in the multi-class setting (with per-class bias) is as follows.

$$\ell_\lambda(\theta;\mathbf{x},\mathbf{y}) = \ell_0(\theta;\mathbf{x},\mathbf{y}) + \frac{\lambda}{2}\|\theta\|_2^2 \tag{159}$$

Algorithm 2 $\mathcal{A}_{\text{implementation}}$: Noisy Minibatch Gradient Descent on regularized logistic regression loss function

Input: Data domain $\mathcal{X}$. Dataset $D = ((\mathbf{x}_1, \mathbf{y}_1), (\mathbf{x}_2, \mathbf{y}_2), \cdots, (\mathbf{x}_n, \mathbf{y}_n))$, where each data point consists of the feature vector $\mathbf{x}_i \in \mathbb{R}^d$ and the label vector $\mathbf{y}_i \in \{0,1\}^c$. The logistic regression loss function $\ell_0(\theta; \mathbf{x}, \mathbf{y})$ defined as Equation (160) with parameter space $\mathbb{R}^{(d+1)\cdot c}$. Stepsize η, noise multiplier σ_{mul}, a (data-independent) parameter initialization distribution $p_0(\theta)$, minibatch size b, feature clipping norm L, and (unregularized) gradient clipping norm $\frac{S_g}{2}$.

Feature Normalization: $\mathbf{x}_1, \cdots, \mathbf{x}_n \leftarrow \text{normalize}(\mathbf{x}_1, \cdots, \mathbf{x}_n)$, where normalize() is an $(\alpha, \varepsilon_{norm})$-Rényi differentially private batch normalization or group normalization scheme described in Tramèr and Boneh [36] (Section 2.3 and Appendix B).

Feature Clipping: $\mathbf{x}_i \leftarrow \frac{\mathbf{x}_i}{\|\mathbf{x}_i\|_2} \cdot \min\{\|\mathbf{x}_i\|_2, L\}$.

Initialization: Sample θ_0^0 from the initialization distribution $p_0(\theta)$.

Batch Generation: shuffle the indices set $\{1, \cdots, n\}$, and partition them into n/b sequential minibatches $B^0, \cdots, B^{n/b-1}$ that are subsets of $\{1, \cdots, n\}$, each with size b.

for $k = 0, 1, \cdots, K-1$ **do**

  for $j = 0, 1, \cdots, n/b - 1$ **do**

    Gradient Clipping: $g_0(\theta_k^j, B^j) = \frac{1}{b} \sum_{x_i \in B^j} \frac{\nabla \ell(\theta_k^j; \mathbf{x}_i)}{\|\nabla \ell_0(\theta_k^j; \mathbf{x}_i)\|_2} \cdot \min\{\|\nabla \ell_0(\theta_k^j; \mathbf{x}_i)\|_2, \frac{S_g}{2}\}$

    Regularization: $g\left(\theta_k^j; B^j\right) = g_0(\theta_k^j; B^j) + \lambda \cdot \theta_k^j$

    $\theta_k^{j+1} = \theta_k^j - \eta \cdot g\left(\theta_k^j; B^j\right) + \eta \cdot \frac{1}{b} \cdot \sigma_{mul} \cdot \frac{S_g}{2} \cdot \mathcal{N}(0, \mathbb{I}_d)$

  $\theta_{k+1}^0 = \theta_k^{n/b}$

Output θ_K^0

where $\ell_0(\theta; \mathbf{x}, \mathbf{y})$ is the following logistic regression loss function.

$$\ell_0(\theta; \mathbf{x}, \mathbf{y}) = -\mathbf{y}^1 \log\left(\frac{e^{\bar{x}^T \cdot \theta_1}}{e^{\bar{x}^T \cdot \theta_1} + \cdots + e^{\bar{x}^T \cdot \theta_c}}\right) - \cdots - \mathbf{y}^c \log\left(\frac{e^{\bar{x}^T \cdot \theta_c}}{e^{\bar{x}^T \cdot \theta_1} + \cdots + e^{\bar{x}^T \cdot \theta_c}}\right) \quad (160)$$

where $\bar{\mathbf{x}} = (\mathbf{x}, 1) \in \mathbb{R}^{d+1}$ denotes the concatenation of the data feature vector $\mathbf{x}$ and 1, and $\mathbf{y} = (\mathbf{y}^1, \cdots, \mathbf{y}^c)$ is the label vector. The parameter vector is $\theta = (\theta_1, \cdots, \theta_c) \in \mathbb{R}^{(d+1)\cdot c}$ that represents the weight and the per-class bias of the linear model. The logistic regression loss function is convex, and therefore the regularized logistic regression loss function is λ-strongly convex.

E.3 Implementation Deatils for satisfying smoothness

Feature Normalization (for faster convergence). For a better convergence of the learning task, we follow Tramèr and Boneh [36], and use feature normalization (including both batch normalization and group normalization). Batch normalization first computes the per-channel mean and variance of the training dataset, in a differentially private way, and then normalizes each channel of each data point in the training dataset. Group normalization [40] separates the channels in a data feature vector into a number of different groups, and normalizes each channel of each data point with the per-point per-group mean and variance (which does not incur additional privacy cost). We refer the reader to [36] (their Section 2.2 and Appendix B) for more details.

Feature Clipping (for bounding the smoothness constant). To ensure that the condition of finite gradient sensitivity S_g is satisified in our experiments, we follow Feldman et al. [18] and normalize the data feature vector in ℓ_2 norm, such that $\|\mathbf{x}\|_2 \leq L$. Under this data feature clipping, we prove that the logistic regression loss function (160) is $(\frac{L^2+1}{2})$-smooth in the following Proposition E.1.

Proposition E.1. *If the data feature vector* $\mathbf{x}$ *has bounded* ℓ_2 *norm, such that* $\|\mathbf{x}\|_2 \leq L$*, then the logistic regression loss function Equation* (160) *is convex , L-Lipschitz and* β*-smooth, for*

$$L = \sqrt{2(L^2+1)} \quad (161)$$

$$\beta = \frac{L^2+1}{2} \quad (162)$$

Proof. We compute the gradient and Hessian matrix for the logistic regression loss function as follows.

The gradient of the logistic regression loss function Equation (160) with respect to θ is:

$$\nabla_\theta \ell_0(\theta; \mathbf{x}, \mathbf{y}) = -\mathbf{y}^1 \cdot \nabla_\theta \log\left(\frac{e^{\bar{\mathbf{x}}^T \cdot \theta_1}}{e^{\bar{\mathbf{x}}^T \cdot \theta_1} + \cdots + e^{\bar{\mathbf{x}}^T \cdot \theta_c}}\right) - \cdots - \mathbf{y}^c \cdot \nabla_\theta \log\left(\frac{e^{\bar{\mathbf{x}}^T \cdot \theta_c}}{e^{\bar{\mathbf{x}}^T \cdot \theta_c} + \cdots + e^{\bar{\mathbf{x}}^T \cdot \theta_c}}\right) \tag{163}$$

$$= \left(\left(\frac{e^{\bar{\mathbf{x}}^T \cdot \theta_1}}{e^{\bar{\mathbf{x}}^T \cdot \theta_1} + \cdots + e^{\bar{\mathbf{x}}^T \cdot \theta_c}} - \mathbf{y}^1\right)\bar{\mathbf{x}}^T, \cdots, \left(\frac{e^{\bar{\mathbf{x}}^T \cdot \theta_1}}{e^{\bar{\mathbf{x}}^T \cdot \theta_c} + \cdots + e^{\bar{\mathbf{x}}^T \cdot \theta_c}} - \mathbf{y}^c\right)\bar{\mathbf{x}}^T\right) \tag{164}$$

$$\tag{165}$$

The Hessian of loss function with respect to θ is:

$$\nabla_\theta^2 \ell_0(\theta; \mathbf{x}, \mathbf{y}) = \begin{pmatrix} H_{11} & \cdots & H_{1c} \\ \vdots & \vdots & \vdots \\ H_{c1} & \cdots & H_{cc} \end{pmatrix} \tag{166}$$

where the submatrices are as follows.

$$\text{For } i = j, \quad H_{ii} = \frac{e^{\bar{\mathbf{x}}^T \cdot \theta_i}}{e^{\bar{\mathbf{x}}^T \cdot \theta_1} + \cdots + e^{\bar{\mathbf{x}}^T \cdot \theta_c}} \cdot \left(1 - \frac{e^{\bar{\mathbf{x}}^T \cdot \theta_i}}{e^{\bar{\mathbf{x}}^T \cdot \theta_1} + \cdots + e^{\bar{\mathbf{x}}^T \cdot \theta_c}}\right) \bar{\mathbf{x}} \cdot \bar{\mathbf{x}}^T \tag{167}$$

$$\text{For } i \neq j, \quad H_{ij} = -\frac{e^{\bar{\mathbf{x}}^T \cdot \theta_i}}{e^{\bar{\mathbf{x}}^T \cdot \theta_1} + \cdots + e^{\bar{\mathbf{x}}^T \cdot \theta_c}} \cdot \frac{e^{\bar{\mathbf{x}}^T \cdot \theta_j}}{e^{\bar{\mathbf{x}}^T \cdot \theta_1} + \cdots + e^{\bar{\mathbf{x}}^T \cdot \theta_c}} \cdot \bar{\mathbf{x}} \cdot \bar{\mathbf{x}}^T \tag{168}$$

$$\tag{169}$$

Therefore, the Hessian matrix of the loss function with respect to θ equals the following Kronecker product.

$$\nabla_\theta^2 \ell_0(\theta; \mathbf{x}, \mathbf{y}) = T \otimes (\bar{\mathbf{x}} \cdot \bar{\mathbf{x}}^T) \tag{170}$$

$$\text{where } T_{ij} = \begin{cases} \frac{e^{\bar{\mathbf{x}}^T \cdot \theta_i}}{e^{\bar{\mathbf{x}}^T \cdot \theta_1} + \cdots + e^{\bar{\mathbf{x}}^T \cdot \theta_c}} \cdot \left(1 - \frac{e^{\bar{\mathbf{x}}^T \cdot \theta_i}}{e^{\bar{\mathbf{x}}^T \cdot \theta_1} + \cdots + e^{\bar{\mathbf{x}}^T \cdot \theta_c}}\right) & i = j \\ -\frac{e^{\bar{\mathbf{x}}^T \cdot \theta_i}}{e^{\bar{\mathbf{x}}^T \cdot \theta_1} + \cdots + e^{\bar{\mathbf{x}}^T \cdot \theta_c}} \cdot \frac{e^{\bar{\mathbf{x}}^T \cdot \theta_j}}{e^{\bar{\mathbf{x}}^T \cdot \theta_1} + \cdots + e^{\bar{\mathbf{x}}^T \cdot \theta_c}} & i \neq j \end{cases} \tag{171}$$

By Böhning [8], T is a positively semi-definite matrix that satisfies

$$T \preceq \frac{1}{2} \cdot \left(I_c - \frac{1}{c} \cdot \mathbf{1}_c \cdot \mathbf{1}_c^T\right), \text{ where } \mathbf{1}_c = (1, \cdots, 1)^T \in \mathbb{R}^c. \tag{172}$$

Therefore, the eigenvalues of T fall in the range $[0, \frac{1}{2}]$. Therefore, because the eigenvalues for the Kronecker product matrix $\nabla_\theta^2 \ell_0(\theta; \mathbf{x}, \mathbf{y}) = T \otimes \left(\bar{\mathbf{x}} \cdot \bar{\mathbf{x}}^T\right)$ consists of the product of eigenvalues for T and $\bar{\mathbf{x}} \cdot \bar{\mathbf{x}}^T$, we prove that the eigenvalues for $\nabla_\theta^2 \ell_0(\theta; \mathbf{x}, \mathbf{y})$ fall in the range of $[0, \frac{1}{2}\|\bar{\mathbf{x}}\|_2^2]$.

By $\|\mathbf{x}\|_2 \leq L$, we prove that $\|\bar{\mathbf{x}}\|_2 \leq \sqrt{L^2 + 1}$, therefore Equation (161) and Equation (162) (in the proposition statement) hold. □

These computations of these Lipschitz smoothness constants could also be cross-checked in the tensorflow privacy tutorial for logistic regression. `https://github.com/tensorflow/privacy/blob/master/tutorials/mnist_lr_tutorial.py`. The feature clipping technique is different from the DP-SGD [1] algorithm that only requires per-example gradient clipping, and is also different from Bu et al. [10] that uses global gradient clipping. The major reason that we use data feature clipping (besides per-example gradient clipping), is for ensuring smoothness of the logistic regression loss function (by Proposition E.1), which is a necessary condition for applying our privacy bound Theorem 4.3.

E.4 Implementation details for satisfying finite gradient sensitivity

Per-example Clipping on Unregularized Gradient (for reducing gradient sensitivity without harming smoothness or strong convexity). Although feature clipping already bounds the gradient sensitivity by $2\sqrt{2(L^2+1)}$ (by Proposition E.1), this bound grows with the feature clipping norm L. This in turn restricts the signal to noise ratio, and does not give good empirical privacy-utility trade-off in our experiments. Therefore, we additionally perform per-example ℓ_2-clipping on the unregularized gradient (detailed pseudocode in Appendix E.1). Under per-example clipping on unregularized gradient, we prove in the following Proposition E.2, that each gradient update in *regularized logistic regression* has finite gradient sensitivity, and preserves strong convexity and smoothness.

Proposition E.2. *Let $\ell_0(\theta; \mathbf{x}, \mathbf{y})$ be the logistic regression loss function defined in Equation* (160). *Let $g_0(\theta; \mathbf{x}, \mathbf{y}) = \frac{\nabla \ell(\theta; \mathbf{x}, \mathbf{y})}{\|\nabla \ell(\theta; \mathbf{x}, \mathbf{y})\|_2} \cdot \min\{\|\nabla \ell(\theta; \mathbf{x}, \mathbf{y})\|_2, \frac{S_g}{2}\}$ be the clipped gradient of (unregularized) loss function $\ell_0(\theta; \mathbf{x}, \mathbf{y})$, under ℓ_2 clipping norm $\frac{S_g}{2}$. If $g(\theta; \mathbf{x}, \mathbf{y}) = g_0(\theta; \mathbf{x}, \mathbf{y}) + \lambda\theta$, and if the data vector $\mathbf{x}$ has bounded ℓ_2 norm, such that $\|\mathbf{x}\|_2 \leq L$, then $g(\theta; \mathbf{x}, y)$ has finite ℓ_2-sensitivity S_g, is continuous, and is almost everywhere differentiable with*

$$\lambda \cdot \mathbb{I}_{(d+1)\cdot c} \preceq \nabla_\theta g(\theta; \mathbf{x}, \mathbf{y}) \preceq (\beta + \lambda) \cdot \mathbb{I}_{(d+1)\cdot c} \tag{173}$$

for any $\theta, \theta' \in \mathbb{R}^{(d+1)\cdot c}$ and $\beta = \frac{L^2+1}{2}$.

We provide complete proof for this proposition below. This construction of clipped unregularized gradient facilitates us to enjoy the benefits of gradient clipping (such as for speeding up convergence [43, 14]) while satsifying the necessary smoothness and strong convexity conditions for applying our privacy dynamics bound.

Proof. By definition Equation (160) for the logistic regression loss function, $\nabla \ell_0(\theta; \mathbf{x}, \mathbf{y})$ is twice continuosly differentiable and convex. By Proposition E.1, $\ell_\lambda(\theta; \mathbf{x}, \mathbf{y})$ is β-smooth with $\beta = \frac{L^2+1}{2}$. Therefore, the Hessian matrix of $\ell_0(\theta; \mathbf{x}, \mathbf{y})$ satisfies the following inequality.

$$0 \cdot \mathbb{I}_{(d+1)\cdot c} \preceq \nabla^2_\theta \ell_0(\theta; \mathbf{x}, \mathbf{y}) \preceq \beta \cdot \mathbb{I}_{(d+1)\cdot c}, \tag{174}$$

where d is the dimension of the input data feature vector $\mathbf{x}$, and c is the number of classes in the label vector $\mathbf{y}$. Moreover, the clipped (unregularized) gradient $g_0(\theta; \mathbf{x}, \mathbf{y})$ (under ℓ_2 clipping norm $\frac{S_g}{2}$) is continuous, and is almost everywhere differentiable as follows.

$$\nabla_\theta g_0(\theta; \mathbf{x}, \mathbf{y}) = \begin{cases} \nabla^2_\theta \ell(\theta; \mathbf{x}, \mathbf{y}) & \text{if } \|\nabla_\theta \ell(\theta; \mathbf{x}, \mathbf{y})\|_2 < \frac{S_g}{2} \\ \frac{S_g}{2} \cdot \frac{1}{\|\nabla_\theta \ell(\theta; \mathbf{x}, \mathbf{y})\|_2} \left(I_{(d+1)\cdot c} - \frac{\nabla \ell(\theta; \mathbf{x}, \mathbf{y})}{\|\nabla \ell(\theta; \mathbf{x}, \mathbf{y})\|_2} \cdot \left(\frac{\nabla \ell(\theta; \mathbf{x}, \mathbf{y})}{\|\nabla \ell(\theta; \mathbf{x}, \mathbf{y})\|_2} \right)^T \right) \cdot \nabla^2_\theta \ell(\theta; \mathbf{x}, \mathbf{y}) & \text{if } \|\nabla_\theta \ell(\theta; \mathbf{x}, \mathbf{y})\|_2 > \frac{S_g}{2} \end{cases} \tag{175}$$

By $\frac{S_g}{2} \cdot \frac{1}{\|\nabla_\theta \ell(\theta; \mathbf{x}, \mathbf{y})\|_2} \leq 1$ for θ s.t. $\|\nabla_\theta \ell(\theta; \mathbf{x}, \mathbf{y})\|_2 > \frac{S_g}{2}$, and by the positive semi-definite property of symmetric matrix $I_{(d+1)\cdot c} - \frac{\nabla \ell(\theta; \mathbf{x}, \mathbf{y})}{\|\nabla \ell(\theta; \mathbf{x}, \mathbf{y})\|_2} \cdot \left(\frac{\nabla \ell(\theta; \mathbf{x}, \mathbf{y})}{\|\nabla \ell(\theta; \mathbf{x}, \mathbf{y})\|_2} \right)^T$, we prove that for any θ such that $\|\nabla_\theta \ell(\theta; \mathbf{x}, \mathbf{y})\|_2 \neq \frac{S_g}{2}$,

$$0 \cdot \mathbb{I}_{(d+1)\cdot c} \preceq \nabla_\theta g_0(\theta; \mathbf{x}, \mathbf{y}) \preceq \beta \cdot \mathbb{I}_{(d+1)\cdot c} \tag{176}$$

Therefore, by plugging this into the definition of $g(\theta; \mathbf{x}, \mathbf{y}) = g_0(\theta; \mathbf{x}, \mathbf{y}) + \lambda\theta$, we prove that $g(\theta; \mathbf{x}, \mathbf{y})$ is continous, and is almost everywhere differentiable with

$$\lambda \cdot \mathbb{I}_{(d+1)\cdot c} \preceq \nabla_\theta g(\theta; \mathbf{x}, \mathbf{y}) \preceq (\beta + \lambda) \cdot \mathbb{I}_{(d+1)\cdot c} \tag{177}$$

□

E.5 Exact formulas for composition-based privacy bound and our improved privacy dynamics analysis in evaluations

Baseline composition-based privacy analysis. We use the DP-SGD analysis with momentum [1] (combined with tighter bound for the privacy amplification by Subsampled Gaussian Mechanism [28])

as the baseline privacy analysis. The original DP-SGD bound only holds for minibatch sampling from a Poisson distribution. However, currently many privacy libraries [21, 42] still apply the DP-SGD analysis while implementing the "shuffle and partition" sampling scheme. Therefore, we consider DP-SGD with momentum as a reasonable comparison baseline for privacy analysis.

Improved privacy dynamics analysis. We use our privacy dynamics bound Theorem 5.2 for noisy minibatch gradient descent Algorithm 1 under "shuffle and partition" as the baseline. By applying Theorem 5.2, we prove the following privacy dynamics theorem for noisy minibatch gradient descent on regularized logistic regression.

Corollary E.3 (Privacy dynamics for noisy minibatch gradient descent on regularized logistic regression)**.** *For the regularized logistic regression loss Equation* (159) *with regularization coefficient λ, if the data feature vector is clipped in ℓ_2-norm by L, and the unregularized gradient is clipped in ℓ_2 norm by $\frac{S_g}{2}$, then for $K \geq 1$ and $\frac{n}{b} \geq 2$, Algorithm 2 with stepsize $\eta < \frac{2}{(L^2+1)/2+2\lambda}$ and noise multiplier σ_{mul} satisfies $(\alpha, \varepsilon_{norm} + \varepsilon)$-Rényi Differential Privacy with*

$$\varepsilon \leq \varepsilon_0^{\lfloor \frac{n}{2b} \rfloor}(\alpha) \cdot \frac{1-(1-\eta\lambda)^{2\cdot(K-1)\cdot(n/b-\lfloor \frac{n}{2b} \rfloor)}}{1-(1-\eta\lambda)^{2\cdot(n/b-\lfloor \frac{n}{2b} \rfloor)}} + \frac{1}{\alpha-1} \cdot \log\left(\underset{0 \leq j_0 < n/b}{Avg} e^{(\alpha-1)\varepsilon_0^{n/b-j_0}(\alpha)} \right) \tag{178}$$

where the terms $\varepsilon_0^j(\alpha)$ is upper-bounded for any $j = 1, \cdots, n/b$ as follows.

$$\varepsilon_0^j(\alpha) \leq \frac{2\alpha}{\sigma_{mul}^2} \cdot (1-\eta\lambda)^{2\cdot(j-1)} \cdot \frac{1}{\sum_{s=0}^{j-1}(1-\eta\lambda)^{2s}} \tag{179}$$

Proof. The proof is by applying Theorem 5.2 with gradient sensitivity S_g, smoothness constant $\frac{L^2+1}{2} + \lambda$, strong convexity constant λ (derived by Proposition E.2)) and noise standard deviation $\sigma = \sqrt{\frac{\eta}{2}} \cdot \frac{1}{b} \cdot \sigma_{mul} \cdot \frac{S_g}{2}$ (explained in Appendix E.1). Moreover, the above privacy bound Equation (178) is strictly smaller than the composition-based privacy bound (derived from Abadi et al. [1] and Mironov et al. [28]) after $4 \cdot \frac{n}{b} + \frac{4}{\lambda\eta}$ epochs, as discussed in detail after Theorem 5.2. □

E.6 Detailed Illustration for the Slow Convergence of DP-SGD for training from scratch

In this section, we plot the train and test accuracy for DP-SGD training on CIFAR-10 dataset (under the same hyperparameters as Tramèr and Boneh [36]). We observe that the logistic regression task on CIFAR10 converges slowly and does not reach stable training accuracy after 60 epochs.

E.7 Full hyperparameters for Table 1 (and Appendix E.6)

Reproduce Tramèr and Boneh [36]. For optimal test accuracy, we follow the hyperparameters for unregularized DP-SGD with momentum and gradient clipping, discribed in Tramèr and Boneh [36] as follows.

1. MNIST: size of training dataset $n = 60000$, stepsize $\eta = 8$, size of minibatch $b = 4096$, batch normalization noise multiplier 8, noise multiplier for gradient update: 3.04, momentum 0.9, gradient clipping norm 0.1, number of epochs 30.
2. Fashion-MNIST: size of training dataset $n = 60000$, stepsize $\eta = 16$, size of minibatch $b = 8192$, group normalization with 27 groups, noise multiplier for gradient update: 4.05, momentum 0.9, gradient clipping norm 0.1, number of epochs 40.
3. CIFAR-10: size of training dataset $n = 50000$, stepsize $\eta = 4$, size of minibatch $b = 8192$, batch normalization noise multiplier 8, noise multiplier for gradient update: 5.67, momentum 0.9, gradient clipping norm 0.1, number of epochs 60.

Column 2,3 of Table 1 uses the same hyperparameters as Tramèr and Boneh [36] (described above) except that they use the same ℓ_2-regularization constant, feature clipping norm and gradient clipping norm as column 4.

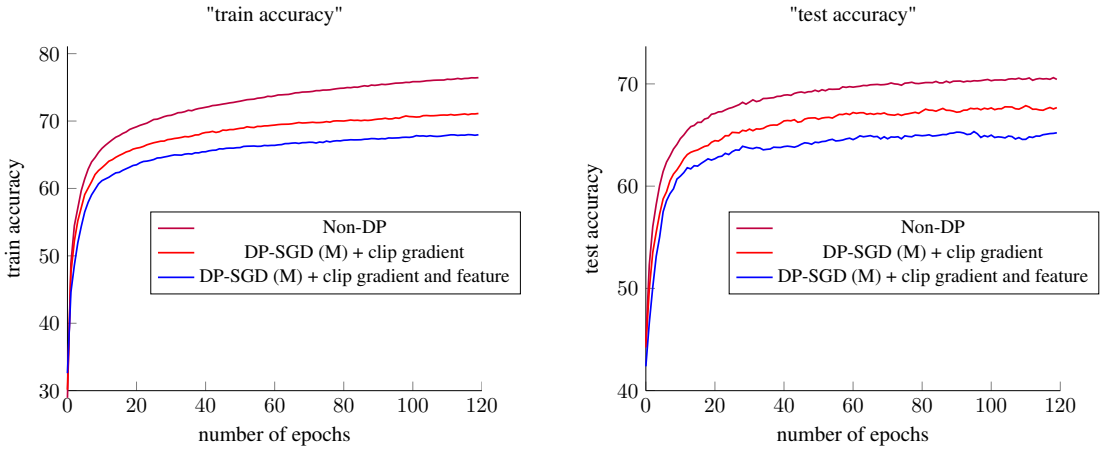


Figure 3: Convergence of DP-SGD (with momentum) in train accuracy (left plot) and test accuracy (right plot) for (regularized) logistic regression with ScatterNet features on CIFAR-10. The DP guarantee for DP-SGD (M) is analyzed as in Abadi et al. [1] and [28]. The algorithm hyperparameters are optimized for each training setting. Full hyperparameters are in Appendix E. For a differential privacy budget of $(\varepsilon = 3, \delta = 10^{-5})$, the algorithm needs to stop after 60 epochs (otherwise the privacy loss will exceeds the given DP budget). However, under all settings, the train and test accuracy for the model continue to grow even after 60 epochs, indicating that insufficent number of allowed running epochs restricts the accuracy for training differentially private model.

Optimal hyperparameters under privacy dynamics bound. For optimal test accuracy under privacy dynamics analysis, we perform hyperparameter tuning for Algorithm 1 via grid search among the following possible values: feature clipping norm $L \in \{1, 1.41, 2, 2.82\}$, gradient clipping norm $= g_{frac} \cdot \sqrt{2 \cdot (L^2 + 1)}$ for $g_{frac} \in \{0.25, 0.5, 0.75, 1\}$, minibatch size $b \in \{512, 1024, 2048, 4096, 8192, 16384\}$, stepsize: $\eta_{frac} \cdot \frac{2}{\frac{L^2+1}{2} + 2 \cdot \lambda}$ for $\eta_{frac} \in \{0.125, 0.25, 0.5, 1\}$, number of epochs: $K \in \{100, 300, 600, 1200\}$, batch normalization noise multiplier $\in \{6, 8\}$, group norm with number of groups $\in \{9, 27, 81\}$, ℓ_2-regularization constant $\lambda \in \{0.01, 0.02, 0.04, 0.08, 0.16\}$. The optimal hyperparameters that we use in Table 1 are as follows.

1. MNIST: stepsize $\eta = 0.75$, size of minibatch $b = 2048$, batch normalization noise multiplier 8, noise multiplier for gradient update: 3.08, momentum 0, feature clipping norm 2, gradient clipping norm 3.16, ℓ_2-regularization 0.08, number of epochs 1200.
2. Fashion-MNIST: stepsize $\eta = 1.92$, size of minibatch $b = 2048$, group normalization with 27 groups, noise multiplier for gradient update: 3.01, momentum 0, feature clipping norm 1, gradient clipping norm 2, ℓ_2-regularization 0.02, number of epochs 1200.
3. CIFAR-10: stepsize $\eta = 0.75$, size of minibatch $b = 2048$, batch normalization noise multiplier 6, noise multiplier for gradient update: 3.23, momentum 0, feature clipping norm 2, gradient clipping norm 1.58, ℓ_2 regularization 0.08, number of epochs 1200.

E.8 Full hyperparameters for Table 2

For both settings, the size of training dataset (CIFAR-10) is $n = 50000$.

Reproduce Tramèr and Boneh [36]. We follow the hyperparameters discribed in Tramèr and Boneh [36] as follows.

1. ResNeXt-39 (CIFAR-10): stepsize $\eta = 8$, size of minibatch $b = 2048$, batch normalization (no privacy cost as it is based on the on CIFAR-100 dataset), noise multiplier for gradient update: 3.32, momentum 0.9, gradient clipping norm 0.1, number of epochs 40.
2. SIMCLR v2 (unlabelled ImageNet): stepsize $\eta = 4$, size of minibatch $b = 1024$, batch normalization (no privacy cost as it is based on the on CIFAR-100 dataset), noise multiplier for gradient update: 2.40, momentum 0.9, gradient clipping norm 0.1, number of epochs 40.

Optimal hyperparameters under privacy dynamics bound. Similar to Appendix E.7, we perform grid search for hyperparameter tuning, and use the following hyperparameters to optimize accuracy.

1. ResNeXt-39 (CIFAR-10): stepsize $\eta = 1.92$, size of minibatch $b = 1024$, batch normalization (no privacy cost as it is based on the on CIFAR-100 dataset), noise multiplier for gradient update: 4.20, momentum 0, feature clipping norm 1, gradient clipping norm 1.5, regularization constant 0.02, number of epochs 600.
2. SIMCLR v2 (unlabelled ImageNet): stepsize $\eta = 0.44$, size of minibatch $b = 1024$, batch normalization (no privacy cost as it is based on the on CIFAR-100 dataset), noise multiplier for gradient update: 4.56, momentum 0, feature clipping norm 2.82, gradient clipping norm 2.11, regularization constant 0.01, number of epochs 600.