Optimal Group Privacy for DP-SGD

Saeed Mahloujifar* Alexandre Sablayrolles† Graham Cormode‡ Somesh Jha§

Abstract

One challenging problem with differentially private machine learning is *privacy accounting*. After years of research, the community has successfully established tight privacy accounting methods for differentially private stochastic gradient descent (DP-SGD). Despite these advances, tight bounds for *group privacy* still remain elusive. Group privacy is an essential aspect of differential privacy that enables many applications. In this work, we develop tight bounds on group privacy for DP-SGD. In this work, we develop tight bounds on group privacy for DP-SGD. Our analysis uses a novel technique to show "dominating pairs of distributions" explicitly tailored for the case of group privacy. Our experiments show that our bounds are significantly better than previously known bounds in certain regimes. Surprisingly, we find that group privacy is significantly affected by sub-sampling. Two sets of hyper-parameters (sampling rate and noise) with the exact same (ϵ, δ) parameters can have significantly different group privacy curves.

1 Introduction

The Differentially Private Stochastic Gradient Descent (DP-SGD) algorithm [1, 25] is the leading method for training machine learning models and conducting a variety of optimization tasks with privacy guarantees. A critical facet that enables DP-SGD for privacy is the notion of (tight) privacy accounting. Privacy accounting addresses a fundamental question: What is the extent of privacy degradation when a differential privacy-protected task is performed repetitively? Since the inception of differential privacy, this question has been studied via "composition theorems" [8, 14]. Subsequent work has focused on tighter composition for privacy, specifically within the framework of DP-SGD. Building upon the foundation laid by Abadi et al. [1], furthered by the introduction of Rényi differential privacy (RDP) by Mironov [20], and enhanced by recent research on the precise analysis of Gaussian differential privacy [7, 11, 29, 32], we can now compute the privacy assurances of DP-SGD with high precision. This analysis has allowed acceptable utility levels for various tasks while simultaneously offering substantial privacy guarantees. Despite these advances, the domain of group privacy has not experienced equivalent progress. Group privacy poses a different question: how does the privacy guarantee deteriorate when the concern is about the impact on a *group* of examples, rather than individual ones? Intriguingly, we still lack better group privacy bounds than the rudimentary black-box bounds that were first introduced with differential privacy [8]. Thus, we ask: Can DP-SGD attain group privacy bounds superior to the black-box bounds for any DP mechanism?

The question of group privacy is significant in various analytical contexts. For instance, group privacy can prove beneficial in scenarios such as federated learning, where a user might contribute multiple data points and we want a "user-level" privacy boundary [13]. Similarly, there might be situations where the

*FAIR at Meta, Corresponding author, saeedm@meta.com
†Mistral AI, work done at Meta
‡Meta
§University of Wisconsin-Madison

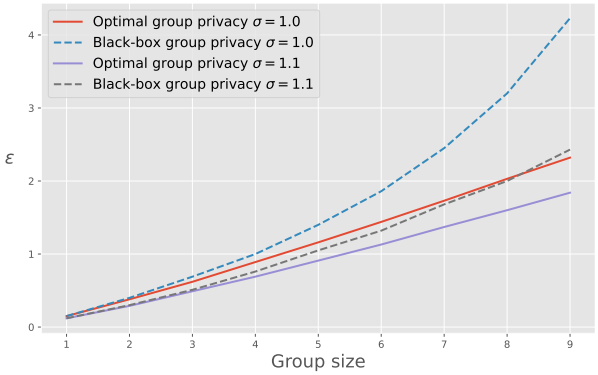

Figure 1: Group composition for 10 steps of DP-SGD with noise multiplier 1.0, sampling rate 0.01, and δ of 10^{-3}. The privacy curve obtained by our bounds is much smoother. We use a large value for δ as calculating the black box group privacy bound for small values of δ is computationally infeasible (unlike our bound).

same or very similar examples are repeated within collected datasets [15]. As demonstrated in recent studies on reconstruction attacks, these repeated points could be considerably more susceptible to privacy breaches [5]. Group privacy bounds enable us to understand the degree of increased vulnerability when examples are replicated. Another practical application of group privacy bounds arises in the context of auditing differential privacy [12, 23, 26]. This process involves injecting a number of examples into the training routine of a machine learning model using DP-SGD, with the aim of calculating a lower limit on privacy and comparing it against the (potentially loose) guaranteed bound. Attaining tighter constraints on group privacy thus helps narrow the gap between the guaranteed privacy levels and the lower limit, facilitating more precise auditing. Last, group privacy bounds contribute to robustness against poisoning attacks. It is well-known that DP affords some protection against manipulations in the training dataset [18]. Enhanced group privacy bounds serve to improve these protections. Hence, optimizing group privacy is not just a theoretical exercise but has profound implications for applications in ML and privacy.

Our contributions: In this work, we provide tight group privacy bounds for DP-SGD. Our bounds are based on our new "domination" theorem that shows the worst possible pair of distributions that might occur while running DP-SGD on two k-neighboring databases. We consider this "domination" result as our main technical contribution. Using the knowledge of worst-case distributions, we give a Monte-Carlo approach to estimate the differential privacy bounds. Our experiments show that our bounds can significantly outperform the previous (black-box) group privacy bounds. We also find a surprising relation between sampling rate and group privacy: in a nutshell, as the sub-sampling rate becomes smaller, the group privacy improves.

2 Preliminaries

We first define a notion of proximity for group privacy.

Definition 2.1 (k-neighboring) *A pair of datasets (D, D') are k-neighboring iff either (D', D) are k-neighboring or $|D \setminus D'| \leq k$ and $D' \setminus D = \emptyset$. We use $D \approx_k D'$ to denote that D and D' are k-neighboring. Note this is symmetric, i.e., $D \approx_k D' \iff D' \approx_k D$.*

Definition 2.2: A mechanism M is (ε, δ, k)-DP if for all k-neighboring datasets D and D' we have

$$\forall S; \qquad \Pr[M(D) \in S] \leq e^{\varepsilon} \Pr[M(D')] + \delta.$$

We also use a more fine-grained notion of privacy, f-DP.

Definition 2.3 (f-DP [7]) *A mechanism M is (f, k)-DP if for all datasets D, D' s.t. $D \approx_k D'$:*

$$\forall S; \qquad \Pr[M(D) \in S] \leq 1 - f(\Pr[M(D') \in S]).$$

Now we state the basic group privacy introduced by Dwork et al. [8]. We give a slightly improved version of the bound, with an almost identical proof to that of [8].

Theorem 2.1 (Black-box group privacy [8]) *If a mechanism is $(\epsilon, \delta, 1)$-DP, then it is also $(k\epsilon, k \cdot \frac{e^{k\epsilon}-1}{e^{\epsilon}-1} \cdot \delta, k)$-DP.*

Proof: We prove this by induction. For $k = 1$, the statement is trivial. Now assume the statement is correct for $k - 1$. Assume $D' = D \cup \{x_1, \ldots, x_k\}$. Let $D'' = D \cup \{x_1\}$. Now since (D, D'') are 1-neighboring and (D', D'') are $(k-1)$-neighboring. Therefore, by the fact that M is (ϵ, δ)-DP we have

$$\Pr[M(D) \in S] \leq e^{\epsilon} \Pr[M(D'') \in S] + \delta.$$

Also, by the induction hypothesis we have

$$\Pr[M(D'') \in S] \leq e^{(k-1)\epsilon} \cdot \Pr[M(D') \in S] + \frac{e^{(k-1)\epsilon} - 1}{e^{\epsilon} - 1}\delta.$$

Combining these two inequalities, we have

$$\Pr[M(D) \in S] \leq e^{(k-1)\epsilon+\epsilon} \Pr[M(D') \in S] + \left(\frac{e^{k\epsilon} - e^{\epsilon}}{e^{\epsilon} - 1} + 1\right)\delta = e^{k\epsilon} \Pr[M(D') \in S] + \frac{e^{k\epsilon} - 1}{e^{\epsilon} - 1}\delta.$$

And, this finishes the proof.

This group-privacy bound is tight when employed as a black-box. In other words, there exists a mechanism M that is (ϵ, δ)-DP which enjoys the exact same group privacy bound as stated in the theorem.

We also note that there is a previously-known black-box group privacy bound for f-DP.

Theorem 2.2 (Black-box group privacy for f-DP [7]) *If a mechanism M is $(f, 1)$-DP, then for all $k \in \mathcal{N}$ it is also (f_k, k)-DP where*

$$f_k(x) = 1 - (1 - f)^k(x).$$

One would expect that the group privacy based on f-DP to be much tighter than the black-box variant for DP. This is simply because knowing that a mechanism is f-DP contains much more information than knowing a mechanism is (ϵ, δ)-DP. However, as we will see in the later sections, this bound is also sub-optimal when sub-sampling is employed. We reiterate that this bound is tight if the only information that we have is the knowledge that the mechanism is f-DP. But when dealing with specific algorithms (e.g., DP-SGD), we can use more information about the inner dynamics of the mechanism to achieve better bounds. In this work, we try to improve these black-box bounds for a specific class of mechanisms: the adaptive composition of multiple sub-sampled Gaussian mechanisms.

DP-SGD and Composition of sub-sampled Gaussian mechanisms. A Gaussian mechanism is simply used to privately calculate the average of a function h over a dataset. By releasing the noisy average $\frac{1}{|D|}\sum_{x \in D} h(x) + \mathcal{N}(0, \sigma^2)$, one would guarantee that the reported average satisfies differential

Algorithm 1 DP-SGD

Require: Private dataset D, Loss function ℓ, Sampling rate p, number of steps n, noise multiplier σ, clipping threshold c

Initialize a model θ.
for $i \leftarrow [1, \ldots, m]$ **do**
  Sample a random batch $B \subseteq D$ using Poisson sampling with probability of p.
  $g = \frac{1}{p \cdot |D|} \sum_{x \in B} \frac{\Delta\ell(\theta, x)}{\max\left(c, \|\Delta\ell(\theta, x)\|\right)}$
  $\tilde{g} = g + c \cdot \mathcal{N}(0, \sigma^2)$
  $\theta = \textbf{update}(\theta, \tilde{g})$
end for
Output θ.

privacy so long as the function h has a bounded range. To further amplify privacy, we can sample a random batch $B \subset D$ using Poisson sampling and report the noisy average (noisy average should be calculated without using the actual size of the batch; see Algorithm 1) over the batch. Then, one can compose many of these mechanisms in an adaptive way and preserve differential privacy, thanks to composition theorems. DP-SGD (Algorithm 1) [1, 25] is the most tangible instantiation of this category of mechanisms and is used for many privacy preserving applications where we need to perform optimization.

Our goal is to analyze the group privacy for DP-SGD. We need several definitions for the analysis. The first is the notion of weighted total variation distance.

Definition 2.4 (Weighted total variation distance) *The weighted variation distance between two distributions X and Y with densities μ and ν for a weight $a > 0$ is:*

$$\mathbf{TV}_a(X, Y) = \int |\mu(x) - a \cdot \nu(x)| dx.$$

We note that this notion is closely related to that of the Hockey-stick divergence [24] and trade-off functions [7]. Hockey-stick divergence is the same integration with a difference that the integral is only taken over the positive values. We prefer weighted TVD because it is more convenient to avoid conditioning the integration. The next claim shows the relevance of the weighted total variation distance in the DP context.

Proposition 2.2: Let (X, Y) be a pair of random variables. Then for all S and $\epsilon > 0$ we have,

$$\Pr[X \in S] \leq e^{\epsilon} \Pr[Y \in S] + \tfrac{1}{2}(\mathbf{TV}_{e^{\epsilon}}(X, Y) + 1 - e^{\epsilon}).$$

Proof: Let S be an arbitrary set. Let μ and ν be the pdf of X, Y respectively. Let $G_\epsilon = \{x; \mu(x) - e^{\epsilon}\nu(x) \geq 0.\}$ and $\bar{G}_\epsilon = \{x; \mu(x) - e^{\epsilon}\nu(x) < 0.\}$

$$\Pr[X \in S] - e^{\epsilon} \Pr[Y \in S] = \int_S \mu(x) - e^{\epsilon}\nu(x) \leq \int_{G_\epsilon} \mu(x) - e^{\epsilon}\nu(x).$$

Let $\delta = \int_{G_\epsilon} \mu(x) - e^{\epsilon}\nu(x)$, we have $\int_{\bar{G}_\epsilon} \mu(x) - e^{\epsilon}\nu(x) = 1 - e^{\epsilon} - \delta$. We also have $\mathbf{TV}_a(X, Y) = \int_{G_\epsilon} \mu(x) - e^{\epsilon}\nu(x) - \int_{\bar{G}_\epsilon} \mu(x) - e^{\epsilon}\nu(x)$. Therefore $\mathbf{TV}_a(X, Y) = 2\delta - 1 + e^{\epsilon}$. Therefore, $\delta = \frac{\mathbf{TV}_a(X,Y) + 1 - e^{\epsilon}}{2}$.

Note that the above proposition is only stated in one direction. However, differential privacy requires the upper bound to hold in both directions. Due to an interesting property of weighted total variation distance, we can also bound the reverse direction without changing the order of distributions in $\mathbf{TV}(X, Y)$.

Corollary 2.2.1: Let (X, Y) be a pair of random variables. Then for all S and $\epsilon > 0$ we have,

$$\Pr[Y \in S] \leq e^{\epsilon} \Pr[X \in S] + \tfrac{1}{2}(e^{\epsilon}\mathbf{TV}_{e^{-\epsilon}}(X, Y) + 1 - e^{\epsilon}).$$

Proof: Observe that $\mathbf{TV}_a(Y, X) = a\mathbf{TV}_{\frac{1}{a}}(X, Y)$. Now we use Proposition 2.2 with X and Y swapped, and apply this observation to finish the proof.

In Section 4, where we explain how to calculate the optimal bounds, we will see why we are interested in preserving the order of pairs. We next define *dominating pairs* of distributions, specific to the case of group-privacy. This notion enables us to use a pair of distributions for privacy accounting and removes the complexity of the choice of dataset.

Definition 2.5 (k-Dominating pair of distributions) *A pair of distributions (X, Y)* dominates *a mechanism M if for any pair of k-neighboring datasets $D \approx_k D'$ with $|D| < |D'|$ and any $a > 0$ we have*

$$\mathbf{TV}_a(X, Y) \geq \mathbf{TV}_a(M(D), M(D')).$$

We say that (X, Y) tightly dominate *M if there are k-neighboring datasets (D, D') such that $M(D) \equiv X$ and $M(D') \equiv Y$.*

Note that domination is defined in an asymmetric way. That is, we fix the order of datasets so that D has fewer data points than D'. The following proposition shows the usefulness of dominating pairs for privacy analysis of a mechanism. This proposition directly follows by applying Proposition 2.2 and Corollary 2.2.1.

Proposition 2.2: A mechanism that is k dominated by (X, Y) is (ϵ, δ, k)-DP for

$$\delta = \tfrac{1}{2}(\max\left(\mathbf{TV}_{e^{\epsilon}}(X, Y), e^{\epsilon}\mathbf{TV}_{e^{-\epsilon}}(X, Y)\right) + 1 - e^{\epsilon})$$

Finally, we state the following lemma that shows we can obtain a dominating pair of distributions for the composition of multiple mechanisms. This has been proved for the case of $k = 1$ in previous work [7, 19, 32]. Here we omit the proof as it is exactly the same.

Lemma 2.2: If a series of mechanisms $M_1, \ldots, M_n$ are k-dominated by pairs of distributions (X_1, Y_1), $\ldots$, (X_n, Y_n) then the adaptive composition of M_i's is k-dominated by

$$(X_1 \times \cdots \times X_n, Y_1 \times \cdots \times Y_n)$$

where $\times$ is the product operation between distributions.

3 Optimal group privacy bounds

Next, we demonstrate a tightly k-dominating pair of distribution for a single step of DP-SGD. Note that, by Lemma 2.2, this will give us a tight dominating pair for multiple steps of DP-SGD as well. We first define two notions that abstract two properties of Gaussian mechanism which we need to prove our result.

Definition 3.1 (Compatible distributions) *We call a triplet of distributions (X, Y, Z) with densities ν_X, ν_Y, and ν_Z* compatible *iff there exists an increasing and continuous transition function g, with $g(0) = 0$ and $\lim_{t\to\infty} g(t) = \infty$ such that $\frac{\nu_Y(x)}{\nu_X(x)} \geq r$ if and only if we have $\frac{\nu_Z(x)}{\nu_X(x)} \geq g(r)$.*

Definition 3.2 (System of nice distributions) *Let $\mathcal{Y} = \{\mathcal{Y}_1, \ldots, \mathcal{Y}_k\}$ be a collection of sets of distributions and let X be a distribution. $(X, \mathcal{Y})$ form a system of nice distributions if the following conditions hold:*

1. *For each $\mathcal{Y}_i$ there exists $Y_i^* \in \mathcal{Y}_i$ such that*
$$\forall a > 0, \forall Y_i \in \mathcal{Y}_i; \mathbf{TV}_a(X, Y_i) \leq \mathbf{TV}_a(X, Y_i^*).$$
2. *for all $i < j \in [k]$ the triplet (X, Y_i^*, Y_j^*) is compatible.*

We now state a lemma that shows when we can get a dominating pair for a mixture of multiple mechanisms. For ease of readability, proofs for most claims in this Section are deferred to Appendix A.

Lemma 3.0: Let $\mathcal{Y} = \{\mathcal{Y}_1, \ldots, \mathcal{Y}_k\}$ be a collection of sets of distributions. Assume that $(X, \mathcal{Y})$ form a system of nice distributions. Let $p_1, \ldots, p_k \in [0, 1]$ with $p_1 + \ldots, p_k = 1$. Let $Y = p_1 \cdot Y_1 + \cdots + p_k \cdot Y_k$ be an arbitrary mixture of distributions with $Y_i \in \mathcal{Y}_i$. Let $Y^* = p_1 \cdot Y_1^* + \cdots + p_k \cdot Y_k^*$. We have
$$\mathbf{TV}_a(X, Y) \leq \mathbf{TV}_a(X, Y^*).$$

This Lemma, which is the key lemma for proving our result, helps us to reduce the complexity owing to the mixture distribution. By knowing the worst-case in each set of distributions, we can identify the worst-case for the mixture as well. Now we turn our attention to the specific case of the Gaussian mechanism and show how to use Lemma 3.0 to obtain the dominating pair. We first show an intuitive result that $\mathbf{TV}_a$ between pairs of isotropic Gaussians is an increasing function of the distance between them.

Proposition 3.0: Let $X \equiv \mathcal{N}(u_1, \sigma^2 \cdot I_d)$ and $Y \equiv \mathcal{N}(u_2, \sigma^2 \cdot I_d)$. Then, for any $a \in \mathbb{R}^+$, $\mathbf{TV}_a(X, Y)$ is only a function of $\|u_1 - u_2\|_2$ and σ^2. Moreover this function is monotonically increasing with respect to $\|u_1 - u_2\|_2$. That is, for any $a \geq \|u_1 - u_2\|_2$ we have

$$\mathbf{TV}_a(X, Y) \leq \mathbf{TV}_a(\mathcal{N}(0, \sigma^2), \mathcal{N}(a, \sigma^2)).$$

Now, we show that a triplet of isotropic Gaussians with collinear means are compatible.

Proposition 3.0: Let $\mu \in R^d$, $c \in R^+$ $X = \mathcal{N}(0^d, \sigma^2 \cdot I_d)$, $Y = (\mu, \sigma^2 \cdot I_d)$ and $Z = (c \cdot \mu, \sigma^2 \cdot I_d)$. Then (X, Y, Z) are compatible.

Finally, we show that collections of sets of isotropic Gaussians, where each set is restricted to have a mean within a ball, form a system of nice distributions.

Proposition 3.0: Let $X = \mathcal{N}(0^d, \sigma^2 \cdot I_d)$ be isotropic Gaussian centered at zero. Also, for $j \in [k]$ let $\mathcal{Y}j = \{\mathcal{N}(\mu, \sigma^2 \cdot I_d); \|\mu\| \leq r_j\}$ for some $r_j \in \mathbb{R}^+$ and let $\mathcal{Y} = \{\mathcal{Y}_1, \ldots, \mathcal{Y}_k\}$. Then $(X, \mathcal{Y})$ forms a nice system of distributions.

Finally, we put things together to prove the following:

Theorem 3.1 (Main result) *Let M be one step of DP-SGD with sub-sampling rate p and clipping threshold 1 and noise σ. Then M is k-dominated by (X, Y) where*
$$X = \mathcal{N}(0, \sigma^2) \quad \text{and} \quad Y = \mathcal{N}(B(k, p), \sigma^2)$$
where $B(k, p)$ is the binomial distribution.

Before proving this theorem we state the following corollary that shows how this extends to multiple steps.

Corollary 3.1.1: DP-SGD with T-steps, noise multiplier σ and sub-sampling rate p is (ϵ, δ, k)-DP, for an arbitrary $\epsilon \in [0,1]$, $k \in [N]$ with

$$\delta = \frac{1}{2} \max \Big(\mathbf{TV}_{e^\epsilon}(X, Y), e^\epsilon \mathbf{TV}_{e^{-\epsilon}}(X, Y) \Big) + 1 - e^\epsilon,$$
$$\text{and} \quad X = \mathcal{N}(0^T, \sigma^2) \quad \text{and} \quad Y = \mathcal{N}(B(k,p)^T, \sigma^2).$$

Proof: From Theorem 3.1 we know that $\mathcal{N}(0, \sigma^2), \mathcal{N}(B(k,p), \sigma^2)$ form a dominating pair for a single step of DP-SGD with aforementioned hyperparameters. Then using Lemma 2.2 we obtain that the pair (X, Y), for X and Y as stated in the theorem, form a k-dominating pair for all T steps of DP-SGD. Therefore, using Proposition 2.2 we finish the proof.

Proof:[Proof of Theorem 3.1] Let us fix D and $D' = D \cup \{x_1, \dots, x_k\}$. Assume we fix the randomness of sub-sampling on all points in D (e.g., assume all of examples in D are sampled). Then, conditioned on this sampling s, the distribution $M(D)|s$ is a Gaussian centered at some μ_0. On the other hand, $M(D')|s$ is a mixture of Gaussians where the center is determined by the choice of sub-sampling on $\{x_1, \dots, x_k\}$. So we can characterize $M(D')$ as a mixture of Gaussians with probability weights $p_{s'}$, and means $\mu_0 + \mu_{s'}$, and with standard deviations σ^2. That is, $M(D') = \sum_{s' \in \{0,1\}^k} p_{s'} \cdot \mathcal{N}(\mu_0 + \mu_{s'}, \sigma^2)$ (note that here the outer sum operator denotes the mixture of distributions). We are interested in upper-bounding $\mathbf{TV}_a(M(D)|s, M(D')|s)$. Since μ_0 is present in the mean of both $M(D)$ and $M(D')$ we can ignore it and upper bound $\mathbf{TV}_a(\mathcal{N}(0, \sigma^2), \sum_{s'} p_{s'} \mathcal{N}(\mu_{s'}, \sigma^2)$. We know that for all s', the norm $\|\mu_{s'}\|$ is bounded by $|s'|_1$ because the clipping threshold is 1. In other words, $\mathcal{N}(\mu_{s'}, \sigma^2) \in \mathcal{Y}_{|s'|_1}$ where $\mathcal{Y}_{|s'|_1}$ is defined as in Proposition 3.0. Now, using Proposition 3.0, we know that the $(X, \mathcal{Y})$ form a system of nice distributions. Therefore, by Lemma 3.0, and the fact that for each $\mathcal{Y}_i$, the distribution $Y_i^* = \mathcal{N}(i, \sigma^2)$ incurs the greatest $\mathbf{TV}_a(X, Y_i^*)$ (according to Proposition 3.0), we have

$$\mathbf{TV}_a(M(D)|s, M(D')|s) \leq \mathbf{TV}_a(\mathcal{N}(0, \sigma^2), \sum p_{s'} \mathcal{N}(|s'|_1, \sigma^2)).$$

Now observe that $\sum p_{s'} \mathcal{N}(|s'|_1, \sigma^2)$ is the same distribution as $\mathcal{N}(B(k,p), \sigma^2)$. This concludes the proof for a fixed choice of sub-sampling s. Finally, note that fixing the sub-sampling is without loss of generality because we have $\mathbf{TV}_a(p_1 X_1 + p_2 X_2, p_1 Y_1 + p_2 Y_2) \leq p_1 \mathbf{TV}_a(X_1, Y_1) + p_2 \mathbf{TV}_a(X_2, Y_2)$ for any X_1, X_2, Y_1 and Y_2.

Remark 1 (Tightness of our bound) *When we say our bound is tight, we mean that there are instantiations of DP-SGD that will exactly incur the same privacy loss as our theorem predicts. Namely, if one runs the auditing attacks to verify DP [23] for these instantiations, the difference between the empirical lower bounds and theoretical upper bounds should be negligible. We also note that the bound of Theorem 3.1 is only tight in the setting where we release all the intermediate steps of DP-SGD. We do not make any claims about the tightness of our bound when we only release the final model (i.e., the weighted sum of all the intermediate gradients) and leave this as an open question. In fact, to the best of our knowledge, it is not understood if the best existing analysis of DP-SGD (PRV accounting and MC accounting [11, 29]) achieve tight bounds even for groups of size 1, when we only release the final model.*

Comparison with group privacy through f-DP. The seminal work of Dong et al. [7] defines the notion of f-DP and its special case, GDP. A mechanism M is f-DP if for all neighboring datasets the trade-off function between $M(D)$ and $M(D')$ is greater than f on all points. This notion contains more information than DP (or RDP) as it embeds the entire privacy curve. The authors propose a simple group privacy bound; a mechanism that is f-DP, will be $1 - (1-f)^k$-DP for groups of size k, where $(1-f)^k$ denotes the k-fold composition of the function $1 - f$. The authors rightfully claim that this group privacy bound cannot be generally improved because it is tight for the pure Gaussian mechanism. However, there

are three main issues with using these group privacy bounds for the sub-sampled Gaussian mechanisms: (1) The bound is not necessarily tight for the case of sub-sampling. (2) Calculating the bounds needs estimation of the entire trade-off function (for extremely small values) which is computationally inefficient. (3) The estimation error of the trade-off function grows exponentially with the number of compositions. On the contrary, our domination result avoids all this issues and enables us to calculate tight group privacy bounds for DP-SGD. Although calculating the f-DP based group privacy bound is infeasible, we can still analytically show that our bound is better. The following proposition formalizes this statement.

Proposition 3.1 (f-DP group privacy) *Let f be the optimal trade-off function for T-steps of DP-SGD with noise multiplier $\sigma > 0$ and sampling rate $p < 1$. Then (ϵ, δ) parameters obtained by group privacy for groups of size $k > 1$ through black-box f-DP group privacy bounds of Theorem 2.2 is strictly worse than that of Theorem 3.1.*

But the crux of the proof lies in the fact that the trade-off function is always convex and the black-box bound will perform operations that involve $f(p \cdot X_1 + (1-p) \cdot X_2)$, while the white box bounds will leverage the knowledge of sub-sampling and achieves $p \cdot f(X_1) + (1-p) \cdot f(X_2)$. This effect will compound over multiple iterations and as long as the sub-sampling rate is below 1, and the group size is larger than one, our bound will be strictly better.

4 Calculating the bound using Monte Carlo approximation

In this section we describe our algorithm for calculating the bound of Corollary 3.1.1. Note that the bound described there does not have a closed form and we need to approximate it. Previous work has explored various ways to calculate these type of bound using Monte Carlo approximation [19, 29] and numerical methods [9, 11, 32]. However, we still need to devise a new method for calculating our bound because previous methods are not general enough to cover the calculation of our bounds out of the box. In this work, we focus on the Monte-Carlo methods for calculating our bound.

Recall that the bound of Corollary 3.1.1 shows how to calculate δ at a given ϵ and the formula involves calculating $\mathbf{TV}_a(X, Y)$ for a pair of distributions X, Y. The procedure for calculating this weighted total variation distance uses two key observations. This first observation is that the formula for calculating the weighted total variation distance can be converted into an expectation form as follows:

$$\begin{aligned}
\mathbf{TV}_a(X, Y) &= \int |\nu_X(x) - a\nu_Y(x)| dx \\
&= 2 \int \max\Big(\nu_X(x) - a\nu_Y(x), 0\Big) dx + a - 1 \\
&= 2\mathbb{E}_{x\sim X}\Big[\max\big(1 - a\frac{\nu_Y(x)}{\nu_X(x)}, 0\big)\Big] + a - 1
\end{aligned}$$

Our second observation is that we can efficiently sample points from X and we can also calculate the ratio between ν_Y and ν_X at any given point x. This is simply by calculating the ratio at each coordinate using the binomial weights and then multiplying all the ratios in different coordinates. Hence, we can use a simple Monte-Carlo approach to approximate this quantity. Algorithm 1 shows our procedure for Monte-Carlo approximation of the δ for a given ϵ.

In a nutshell, the algorithm samples m points $x_1, \ldots, x_m$ from X. Then it calculates the ratio $r_i = e^{\epsilon} \cdot \nu_Y(x_i)/\nu_X(x_i)$ for all x_i. We can do this in n steps by calculating the ratio for each dimension and then multiplying them. Calculating the ratio for a dimension takes time $O(k)$ because the distribution for each dimension is a mixture of k distributions. Note that the algorithm would calculate both

$\mathbf{TV}_{e^\epsilon}(X, Y)$ and $\mathbf{TV}_{e^{-\epsilon}}(X, Y)$ at the same run. This is because of Corollary 3.1.1 that requires both of these quantities to calculate the δ. The running time of this algorithm is $O(knm)$ and the accuracy of approximation δ improves with the number of samples. The following Proposition shows the dependence between the accuracy and number of samples.

Algorithm 2 Compute δ group privacy

Require: Sampling rate p, group size k, number of compositions n, noise multiplier σ, privacy parameter ϵ, number of samples for mean estimation m

function GAUSSIAN_PDF(x, σ)
  return $e^{-\frac{x^2}{2\sigma^2}}$
end function

function BINOM_MIXTURE_PDF$((x, k, \sigma, p))$
  $p \leftarrow 0$
  for $j \leftarrow 0$ to k **do**
    $p_j \leftarrow$ GAUSSIAN_PDF$(x - j, \sigma)$
    $p \leftarrow p + p_j \times$ BINOM_COEFFICIENT(j, k, p)
  end for
  return p
end function

$\delta_1 \leftarrow 0$
$\delta_2 \leftarrow 1 - e^\epsilon$
for $i \leftarrow 1$ to m **do**
  $r_1 \leftarrow e^\epsilon$
  $r_2 \leftarrow e^{-\epsilon}$
  for $j \leftarrow 1$ to n **do**
    $x \sim \mathcal{N}(0, \sigma^2)$
    $\nu \leftarrow$ BINOM_MIXTURE_PDF(x, k, σ, p)
    $\mu \leftarrow$ GAUSSIAN_PDF(x, σ)
    $r_1 \leftarrow r_1 \times \frac{\nu}{\mu}$
    $r_2 \leftarrow r_2 \times \frac{\nu}{\mu}$
  end for
  $\delta_1 \leftarrow \delta_1 + \frac{\max(1-r_1,0)}{m}$

  $\delta_2 \leftarrow \delta_2 + e^\epsilon \cdot \frac{\max(1-r_2,0)}{m}$
end for
output $\max(\delta_1, \delta_2)$

Proposition 4.0: Let M be the composition of n sub-sampled gaussian mechanisms with sampling rate p and noise multiplier σ. Let δ be the output of Algorithm 2 ran on these parameters, with m samples at a given ϵ. Then the mechanism M is $(\epsilon, \delta + \gamma, k)$-DP, with probability at least $e^{1-2e^{-2m\gamma^2}}$, where the probability is taken over the randomness of Algorithm 2.

Proof: Note that Algorithm 2 is essentially finding the mean of the random variable $\max\left(1 - a\frac{\nu_Y(x)}{\nu_X(x)}, 0\right)$ with m samples. This random variable is always between 0 and 1. Using a Chernoff-Hoefding bound, we conclude that the mean estimation has error more than $\gamma e^{-\epsilon}$ with probability at most $p_1 = 2e^{-2me^{-2\epsilon}\gamma^2}$.

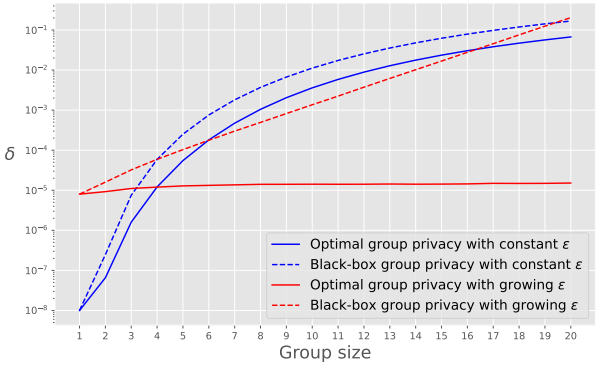

Figure 2: Group composition for 20 steps of DP-SGD with noise multiplier 1.0, sampling rate 0.01. In two of experiments we fix $\epsilon = 2.0$ and calculate the δ. In the other two experiments we grow ϵ linearly and set it to (group size) $\times$ 0.5. Surprisingly, in these experiments δ does not increase much, despite the exponential dependence of δ on the group size in the black box bound.

Therefore, the error of calculating δ_2 is at most γ with probability at least $1-p_1$. The error of calculating δ_1 is also at most γ with probability at least $1 - 2e^{-2m\gamma^2}$. Therefore, the error of $\max(\delta_1, \delta)$ is at most γ with probability at least $1 - 2e^{-2me^{-2\epsilon}\gamma^2} - 2e^{-2m\gamma^2}$. We remark that it is possible to approximate δ with more advanced Monte-Carlo techniques run with fewer samples, similar to what is done in [29]. However, for the purpose of this work, simple Monte-Carlo suffices. One might wonder if we can perform numerical accounting instead of Monte-Carlo. We currently believe that this would be possible for group privacy but it requires a non-black-box view of our proof. We leave this as an interesting open question.

5 Experiments

In this section we describe our experimental setup. We use a range of hyperparameters and group sizes to calculate the group privacy bound. We calculate the group privacy using Algorithm 2. In all experiments, we set m (number of trials) large enough so that with probability at least .99 the error of estimate is at most 1%. Since Algorithm 2 is designed to calculate δ at a given ϵ, we sometimes need to perform a search over ϵ that would give us the desired δ. For the experiments that require calculating ϵ at a given δ, we perform binary search to find the right ϵ.

Comparing with the black-box bound: First we compare our bound with the black-box bound of Theorem 2.1. In this experiment, we fix δ and aim at achieving $(\epsilon, \delta = 10^{-3}, k)$ group privacy for different groups sizes k. We use 10 steps of sub-sampled Gaussian mechanism with sampling probability $p = 0.01$ and noise multiplier $\sigma = 1.0$. Figure 1 shows that our bound can significantly outperform the black-box bound. The growth in ϵ with group size is much closer to linear than what the black-box bound predicts.

Note that for the black-box group privacy, the δ term grows exponentially with the group size. This means we need to calculate the ϵ for a very small value of δ' to be able to get $\delta < 10^{-5}$ after applying the group privacy bound. That is why in Figure 1, we chose $\delta = 10^{-3}$. In contrast, for our bounds there is no such issue and we can calculate ϵ for small values of δ. To illustrate this, we perform another experiment where we fix ϵ and show the growth of δ with ϵ. For this experiment, we use 20 steps of Gaussian mechanism with noise multiplier $\sigma = 2.0$ and sampling rate $p = 0.01$.

The results in Figure 2 (Blue curves) shows that the δ term grows really fast when we grow the group size, both for our bound and the black-box bound. To show the significance of the improvement

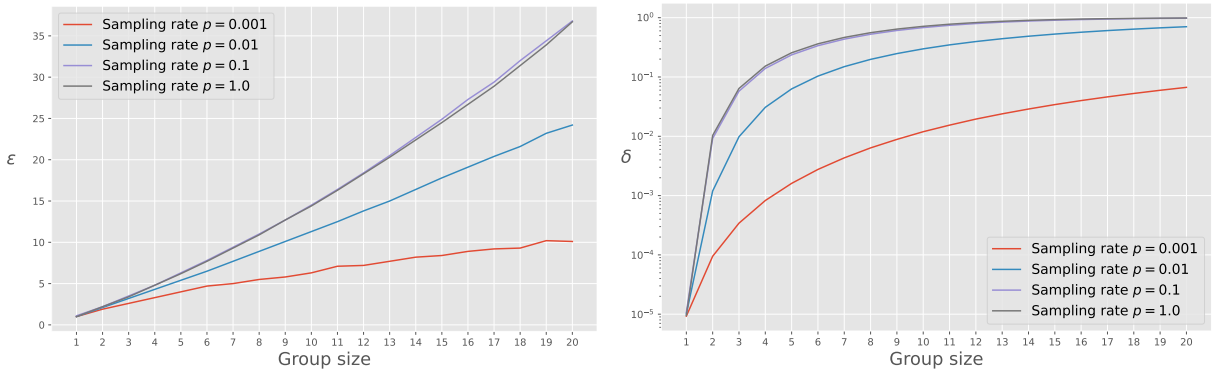

Figure 3: We change sampling rate and corresponding σ to ensure $(1.0, 10^{-5})$-DP when applied for 100 steps. The group privacy is much more graceful for smaller sampling.

in δ, we plot another curve, where we grow ϵ together with group in a linear way. Specifically, we set $\epsilon = k \times 0.5$ where k is the group size. The red curves in Figure 1 shows the comparison of our bound with the black box in this setting. As expected, in the black-box setting the δ still grows exponentially with linearly growing ϵ but the δ calculated using our bound is almost constant.

This experiment suggests a nice approximation of group privacy for sub-sampled Gaussian mechanism with very small sub-sampling rates. In particular, the ϵ seem to grow almost linearly with the group size while keeping the δ constant. Although this does not constitute a privacy guarantee, it is still a good rule of thumb.

Role of sub-sampling rate and step size on group privacy: To better understand the role of sub-sampling on group privacy, we perform experiments by simultaneously varying the sub-sampling rate and noise multiplier so that the (ϵ, δ) terms remain constant $(1.0, 10^{-5})$ (for groups of size 1). Then we calculate the optimal group privacy for each of these settings and compare the curves. Figures 3 show the resulting ϵ and δ terms respectively. In these experiments, we fix the number of steps to 100 and vary the sub-sampling rate to be one of $[0.001, 0.01, 0.1, 1.0]$. Then we find the σ that will satisfy $(1.0, 10^{-5}, 1)$-DP for each sub-sampling value. We then grow the group size for each sub-sampling rate and report the δ at a fixed ϵ in the top figure. We also report ϵ at a fixed δ in the bottom figure.

Our results show that sub-sampling can significantly change the group privacy profile. Lower sub-sampling rates lead to a more graceful degradation of privacy due to group size. We find the effect of sub-sampling on group privacy quite surprising. Note that the black-box group privacy would predict the exact same privacy curve in all scenarios. This finding might suggest an argument for using smaller batch sizes when doing private optimization. Although previous work [6, 27] suggest that larger batch size is better for the trade-off between accuracy and privacy, that dynamic might change if one is interested in the privacy for larger groups.

We also provide further experiments in Appendix B to demonstrate the role of sub-sampling rate at larger scale. We vary the sub-sampling rate and number of steps and observe that the role of sub-sampling diminishes as the number of steps increases. We believe this is mainly because of the behavior of the dominating pairs of distribution for sub-sampled Gaussian mechanism in the limit. We know that these dominating pairs behave similar to the dominating pairs of Gaussian distributions as the number of steps increase [28].

6 Applications and Implications

In this section, we present some of the application and implications of our bounds. Further exploration of these applications is left for future work.

Unit of Privacy: A challenging limitation of DP is that it requires a unit of privacy. The notion of neighboring dataset determines the smallest unit that we would want to protect privacy for. The work of Brown et al. [4] identifies this as one of the main challenges in employing DP for training language models. They question whether we should use words, sentences, paragraphs, documents, or even users as units of privacy. A simple solution to this issue would be to use the smallest unit of privacy imaginable and then apply group privacy to obtain the privacy bounds for larger units. However, the black-box group privacy bounds extremely degrade the privacy parameters, leading to meaningless privacy guarantees for larger privacy units. Our work can change this view and help with selecting small units of privacy, e.g., words or sentences.

Our group privacy bounds are also helpful for calculating user-level privacy [10, 16, 30], in settings that data is collected from multiple users, in a potentially heterogeneous way. In fact, using our general framework of Lemma 3.0, one can obtain even tighter bounds for groups/users that have certain properties. For example, if the gradients in a group are all orthogonal (e.g., they are examples from different classes in a logistic regression setting), $\sqrt{B(k,p)}$ would replace $B(k,p)$ in Theorem 3.1.

Robustness and DP: A large body of work has focused on the study of connections between robustness and DP [17, 21, 31]. Using DP-SGD for training a machine learning model would prevent a training-time attacker (a.k.a. poisoning attack) from changing the behavior of the trained model significantly. The reason relies on the fact that DP would limit the influence of each individual example on the output model. Hence, an adversary who can change a small fraction of the training data will not be able to change the distribution of the trained model more than a certain amount, determined by DP parameters. The (certified) robustness of the final model is determined by applying group privacy bounds, for the groups sizes that are equal to the number of points the adversary can add to the training set. Our improved group privacy bounds can improve the provable consequences of using DP-SGD for certified robustness.

Privacy auditing: A challenge with differential privacy is that verifying its correct implementation is difficult. Recent work has focused on this issue of "privacy auditing" by leveraging attacks that would fail when differential privacy is correctly deployed [12, 22, 26]. Specifically, one would run a membership inference attack to assert the correct implementation of DP; if membership inference succeeds with more than certain probability, then the implementation must be incorrect. We believe our optimal group privacy bounds can add more options for privacy auditing. Instead of individual membership inference attacks, we can focus on group membership inference attacks to verify the correctness of DP implementation. For example, in the context of generative models, it is observed that repetition of single point in the training set can significantly increase its chances of getting regurgitated [5]. In light of our group privacy bounds, an adversary that can distinguish between a model that is trained with 10 copies of a single sample from another model that is not trained on that specific sample, can be used to audit privacy tightly.

Fairness, accuracy, and privacy: Finally, we believe our group privacy bounds can explain some of the observations made about the accuracy and fairness of predictions in private models [2, 3]. Our optimal bounds on group privacy would imply that small groups will have a smaller effect on the models behavior than what was previously thought. This could lead to models that are unfair to small sub-populations. This effect can be exacerbated with certain hyperparameters (e.g. subsampling rate) that will make group privacy stronger. This shows that the choice of hyperparameters should not be

only influenced by the accuracy-privacy trade-off. There could be different hyperparameters that lead to exact same privacy and accuracy while showing different fairness of the model.

References

[1] M. Abadi, A. Chu, I. Goodfellow, B. McMahan, I. Mironov, K. Talwar, and L. Zhang. Deep learning with differential privacy. In *CCS*, 2016.

[2] E. Bagdasaryan, O. Poursaeed, and V. Shmatikov. Differential privacy has disparate impact on model accuracy. *Advances in neural information processing systems*, 32, 2019.

[3] L. Berrada, S. De, J. H. Shen, J. Hayes, R. Stanforth, D. Stutz, P. Kohli, S. L. Smith, and B. Balle. Unlocking accuracy and fairness in differentially private image classification. *arXiv preprint arXiv:2308.10888*, 2023.

[4] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[5] N. Carlini, J. Hayes, M. Nasr, M. Jagielski, V. Sehwag, F. Tramer, B. Balle, D. Ippolito, and E. Wallace. Extracting training data from diffusion models. *arXiv preprint arXiv:2301.13188*, 2023.

[6] S. De, L. Berrada, J. Hayes, S. L. Smith, and B. Balle. Unlocking high-accuracy differentially private image classification through scale. *arXiv preprint arXiv:2204.13650*, 2022.

[7] J. Dong, A. Roth, and W. J. Su. Gaussian differential privacy. *arXiv preprint arXiv:1905.02383*, 2019.

[8] C. Dwork, G. N. Rothblum, and S. Vadhan. Boosting and differential privacy. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 51–60. IEEE, 2010.

[9] B. Ghazi, P. Kamath, R. Kumar, and P. Manurangsi. Faster privacy accounting via evolving discretization. In *International Conference on Machine Learning*, pages 7470–7483. PMLR, 2022.

[10] B. Ghazi, P. Kamath, R. Kumar, P. Manurangsi, R. Meka, and C. Zhang. User-level differential privacy with few examples per user. *arXiv preprint arXiv:2309.12500*, 2023.

[11] S. Gopi, Y. T. Lee, and L. Wutschitz. Numerical composition of differential privacy. *Advances in Neural Information Processing Systems*, 34:11631–11642, 2021.

[12] M. Jagielski, J. Ullman, and A. Oprea. Auditing differentially private machine learning: How private is private SGD? *arXiv preprint arXiv:2006.07709*, 2020.

[13] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

[14] P. Kairouz, S. Oh, and P. Viswanath. The composition theorem for differential privacy. In *International conference on machine learning*, pages 1376–1385. PMLR, 2015.

[15] N. Kandpal, E. Wallace, and C. Raffel. Deduplicating training data mitigates privacy risks in language models. In *International Conference on Machine Learning*, pages 10697–10707. PMLR, 2022.

[16] D. Levy, Z. Sun, K. Amin, S. Kale, A. Kulesza, M. Mohri, and A. T. Suresh. Learning with user-level privacy. *Advances in Neural Information Processing Systems*, 34:12466–12479, 2021.

[17] S. Liu, A. C. Cullen, P. Montague, S. M. Erfani, and B. I. Rubinstein. Enhancing the antidote: improved pointwise certifications against poisoning attacks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 8861–8869, 2023.

[18] Y. Ma, X. Zhu, and J. Hsu. Data poisoning against differentially-private learners: Attacks and defenses. *arXiv preprint arXiv:1903.09860*, 2019.

[19] S. Mahloujifar, A. Sablayrolles, G. Cormode, and S. Jha. Optimal membership inference bounds for adaptive composition of sampled gaussian mechanisms. *arXiv preprint arXiv:2204.06106*, 2022.

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

[21] M. Naseri, J. Hayes, and E. De Cristofaro. Local and central differential privacy for robustness and privacy in federated learning. *arXiv preprint arXiv:2009.03561*, 2020.

[22] M. Nasr, J. Hayes, T. Steinke, B. Balle, F. Tramèr, M. Jagielski, N. Carlini, and A. Terzis. Tight auditing of differentially private machine learning. *arXiv preprint arXiv:2302.07956*, 2023.

[23] M. Nasr, S. Songi, A. Thakurta, N. Papemoti, and N. Carlin. Adversary instantiation: Lower bounds for differentially private machine learning. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 866–882. IEEE, 2021.

[24] I. Sason and S. Verdú. f-divergence inequalities. *IEEE Transactions on Information Theory*, 62(11):5973–6006, 2016.

[25] S. Song, K. Chaudhuri, and A. D. Sarwate. Stochastic gradient descent with differentially private updates. In *2013 IEEE global conference on signal and information processing*, pages 245–248. IEEE, 2013.

[26] T. Steinke, M. Nasr, and M. Jagielski. Privacy auditing with one (1) training run. *arXiv preprint arXiv:2305.08846*, 2023.

[27] X. Tang, A. Panda, V. Sehwag, and P. Mittal. Differentially private image classification by learning priors from random processes. *arXiv preprint arXiv:2306.06076*, 2023.

[28] H. Wang, S. Gao, H. Zhang, M. Shen, and W. J. Su. Analytical composition of differential privacy via the edgeworth accountant. *arXiv preprint arXiv:2206.04236*, 2022.

[29] J. T. Wang, S. Mahloujifar, T. Wu, R. Jia, and P. Mittal. A randomized approach for tight privacy accounting. *arXiv e-prints*, pages arXiv–2304, 2023.

[30] K. Wei, J. Li, M. Ding, C. Ma, H. Su, B. Zhang, and H. V. Poor. User-level privacy-preserving federated learning: Analysis and performance optimization. *IEEE Transactions on Mobile Computing*, 21(9):3388–3401, 2021.

[31] C. Xie, Y. Long, P.-Y. Chen, and B. Li. Uncovering the connection between differential privacy and certified robustness of federated learning against poisoning attacks. *arXiv preprint arXiv:2209.04030*, 2022.

[32] Y. Zhu, J. Dong, and Y.-X. Wang. Optimal accounting of differential privacy via characteristic function. In *International Conference on Artificial Intelligence and Statistics*, pages 4782–4817. PMLR, 2022.

A Deferred proofs

A.1 Proof of Lemma 3.0

Proof: We know that (X, Y_1^*, Y_i^*) are compatible for all $i \in [k]$. Let g_i be the corresponding transition function (see Definition 3.1). Note that g_1 is the identity function. Let t be constants in $[0, 1]$ such that the following holds:

$$\sum_{i\in[k]} \frac{ap_i}{g_i(ap_1/t)} = 1.$$

Note that this t exists because $f(t) = \sum_{i\in[k]} \frac{ap_i}{g_i(ap_1/t)}$ is an increasing and continuous function in t with $\lim_{t\to\infty} = \infty$ and $\lim_{t\to 0} f(t) = 0$. Therefore, there should exist a value of t that makes $f(t) = 1$. Now define $t_i = \frac{ap_i}{g_i(ap_1/t)}$. We are going to use these values to break up the integration. We have,

$$\begin{aligned}
\mathbf{TV}_a(X, Y) &= \int |\nu_X(x) - a\nu_Y(x)|dx \\
&= \int |\nu_X(x) - a\big(\sum_{i\in[k]} p_i \cdot \nu_{Y_i}(x)\big)|dx \\
&= \int |(t_1 + \cdots + t_k) \cdot \nu_X(x) - a\big(\sum_{i\in[k]} p_i \cdot \nu_{Y_i}(x)\big)|dx \\
&\leq \int \Big(\sum_{i\in[k]} |t_i \cdot \nu_X(x) - ap_i \cdot \nu_{Y_i}(x)|\Big)dx \\
&= \sum_{i\in[k]} \int |t_i \cdot \nu_X(x) - ap_i \cdot \nu_{Y_i}(x)|dx \\
&= \sum_{i\in[k]} t_i \cdot \mathbf{TV}_{\frac{ap_i}{t_i}}(X, Y_i) \\
&= \sum_{i\in[k]} t_i \cdot \mathbf{TV}_{g_i(\frac{ap_1}{t})}(X, Y_i) \\
&\leq \sum_{i\in[k]} t_i \mathbf{TV}_{g_i(\frac{ap_1}{t})}(X, Y_i^*)
\end{aligned}$$

Now we have multiple integrations on the absolute values and we need to move back to a single integration. This is where the reason behind the choice of t_i becomes clear. Let

$$s_i(x) = \text{sign}(\nu_X(x) - g_i(ap_1/t)\nu_{Y_i^*}(x)).$$

Based on the definition of g_i in Definition 3.1 we have

$$\forall i \in [k], \forall x; s_i(x) = s_1(x).$$

Using this, based on the fact that $t_i > 0$ we can conclude that

$$\forall j, \forall x; \text{sign}\Big(\sum_{i\in[k]} t_i\big(\nu_X(x) - g_i(ap_1/t)\nu_{Y_i^*}(x)\big)\Big) = s_j(x). \tag{1}$$

Now continuing our calculation of $\mathbf{TV}_a$ we have

$$\begin{aligned}
\mathbf{TV}_a(X,Y) &\le \sum_{i\in[k]} t_i \cdot \mathbf{TV}_{g_i(\frac{ap_1}{t})}(X, Y_i^*) \\
&= \sum_{i\in[k]} \int t_i |\nu_X(x) - g_i(\frac{ap_1}{t}) \cdot \nu_{Y_i^*}(x)| dx \\
&= \int \Big(\sum_{i\in[k]} t_i |\nu_X(x) - g_i(\frac{ap_1}{t}) \cdot \nu_{Y_i^*}(x)| \Big) dx \\
&= \int \Big(\sum_{i\in[k]} t_i \cdot s_i(x) \cdot \big(\nu_X(x) - g_i(\frac{ap_1}{t}) \cdot \nu_{Y_i^*}(x)\big) \Big) dx \\
&= \int \operatorname{sign}\Big(\sum_{i\in[k]} t_i \cdot \big(\nu_X(x) - g_i(\frac{ap_1}{t}) \cdot \nu_{Y_i^*}(x)\big) \Big) \cdot \Big(\sum_{i\in[k]} t_i \cdot \big(\nu_X(x) - g_i(\frac{ap_1}{t}) \cdot \nu_{Y_i^*}(x)\big) \Big) dx \\
&= \int \Big| \sum_{i\in[k]} t_i \cdot \big(\nu_X(x) - g_i(\frac{ap_1}{t}) \cdot \nu_{Y_i^*}(x)\big) \Big| dx \\
&= \int \Big| \nu_X(x) - \sum_{i\in[k]} t_i \cdot g_i(\frac{ap_1}{t}) \cdot \nu_{Y_i^*}(x)\big) \Big| dx \\
&= \int \Big| \nu_X(x) - \sum_{i\in[k]} ap_i \cdot \nu_{Y_i^*}(x)\big) \Big| dx \\
&= \mathbf{TV}_a(X, p_1 \cdot Y_1^* + \cdots + p_k \cdot Y_k^*).
\end{aligned}$$

And this finishes the proof.

A.2 Proof of Proposition 3.0

Proof: The first part follows by the symmetry of isotropic Gaussian. For the second part (monotonicity) we use the definition of $\mathbf{TV}_a$. Without loss of generality we can assume $a \in [0,1]$ as otherwise we can work with $\mathbf{TV}_a(P,Q)/a = \mathbf{TV}_{1/a}(Q,P)$. Let $r = \|u_1 - u_2\|_2$. We can show that the derivative of the integral is always positive. In the following calculations, we use c_1, c_2, c_3 and c_4 to denote positive constants that are independent of r.

First note that $x^* = \frac{r^2 - 2\sigma^2 \ln(a)}{2r}$ is a middle point where $e^{-\frac{x^2}{2\sigma^2}} - ae^{-\frac{(x-r)^2}{2\sigma^2}}$ goes from positive to negative as x increases. By our assumption that $a \in [0,1]$, we have that $x^* > 0$. Recalling that $\operatorname{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z \exp(-t^2) dt$, and that $\operatorname{erf}(\infty) = 1$ so that (by symmetry) $\frac{2}{\sqrt{\pi}} \int_{-\infty}^0 \exp(-t^2) dt = 1$, we can

write

$$\mathbf{TV}_a(P,Q) = c_1\left(\int_{-\infty}^{\infty}\left|e^{-\frac{x^2}{2\sigma^2}} - ae^{-\frac{(x-r)^2}{2\sigma^2}}\right|dx\right)$$

By breaking the integral in an intermediate point we have

$$= c_1\left(\int_{-\infty}^{x^*} e^{-\frac{x^2}{2\sigma^2}} - ae^{-\frac{(x-r)^2}{2\sigma^2}} + \int_{x^*}^{\infty} ae^{-\frac{(x-r)^2}{2\sigma^2}} - e^{-\frac{x^2}{2\sigma^2}}\right)$$

By replacing the integrals with the CDF of Gaussian distribution we have

$$\begin{aligned}
&= c_1\left(1+\operatorname{erf}\left(x^*/\sqrt{2}\sigma\right) - a\operatorname{erf}((x^*-r)/\sqrt{2}\sigma)\right)\\
&\quad + \left(a(1-\operatorname{erf}\left((x^*-r)/\sqrt{2}\sigma\right) + (1-\operatorname{erf}\left(x^*/\sqrt{2}\sigma\right)\right)\\
&= c_2\left(\operatorname{erf}\left(\frac{r^2-\ln(a)\sigma^2}{2\sqrt{2}\sigma r}\right) + 1 - a\operatorname{erf}\left(\frac{-r^2-\ln(a)\sigma^2}{2\sqrt{2}\sigma r}\right) - a\right).
\end{aligned}$$

Now, let $f_1(r) = \operatorname{erf}\left(\frac{r^2-\ln(a)\sigma^2}{2\sqrt{2}\sigma r}\right)$ and $f_2(r) = -a\operatorname{erf}\left(\frac{-r^2-\ln(a)\sigma^2}{2\sqrt{2}\sigma r}\right)$. Taking the derivative with respect to r we have

$$\frac{\partial f_1}{\partial r} = c_3\left(\frac{1}{2\sqrt{2}\sigma} + \frac{\ln(a)\sigma}{2\sqrt{2}r^2}\right)e^{-\left(\frac{r^2-\ln(a)\sigma^2}{2\sqrt{2}\sigma r}\right)^2}$$

$$\frac{\partial f_2}{\partial r} = c_3 a\left(\frac{1}{2\sqrt{2}\sigma} - \frac{\ln(a)\sigma}{2\sqrt{2}r^2}\right)e^{-\left(\frac{-r^2-\ln(a)\sigma^2}{2\sqrt{2}\sigma r}\right)^2}$$

Now note that we have $e^{-\left(\frac{r^2-\ln(a)\sigma^2}{2\sqrt{2}\sigma r}\right)^2} = a^{1/2}\cdot e^{-\left(\frac{-r^2-\ln(a)\sigma^2}{2\sqrt{2}\sigma r}\right)^2}$. Therefore, we have

$$c_4\frac{\partial \mathbf{TV}_a}{\partial r} = e^{-\left(\frac{-r^2-\ln(a)\sigma^2}{2\sqrt{2}\sigma r}\right)^2}\cdot\left(\frac{1+\sqrt{a}}{2\sqrt{2}\sigma} + \frac{\ln(a)\left(\sqrt{a}-1\right)\sigma}{2\sqrt{2}r^2}\right).$$

Now since $a\in[0,1]$, we have $\ln(a)\le 0$ and $\sqrt{a}-1<0$, which means the term $\frac{1+\sqrt{a}}{2\sqrt{2}\sigma} + \frac{\ln(a)(\sqrt{a}-1)\sigma}{2\sqrt{2}r^2}$ is positive. This implies that the whole gradient is positive.

A.3 Proof of Proposition 3.0

Proof: We have

$$\nu_Y(x)/\nu_X(x) = e^{\frac{\|x\|^2-\|x-\mu\|^2}{2\sigma^2}} \quad \text{and} \quad \nu_Z(x)/\nu_X(x) = e^{\frac{\|x\|^2-\|x-c\cdot\mu\|^2}{2\sigma^2}}.$$

For all x we have

$$\begin{aligned}
\|x-c\cdot\mu\|^2 - \|x\|^2 = \sum_{i=1}^{d} c^2\mu_i^2 - 2c\mu_i\cdot x_i &= \Big(\sum_{i=1}^{d} c^2 - c\mu_i^2\Big) + c\Big(\sum \mu_i^2 - 2\mu_i\cdot x_i\Big)\\
&= (c^2-c)\|\mu\| + c(\|x-\mu\|^2 - \|x\|^2).
\end{aligned}$$

Therefore, if $\frac{\nu_Y(x)}{\nu_x(x)} > r$ then we have $\|x\|^2 - \|x-\mu\|^2 > 2\sigma^2\cdot\ln(r)$, which implies $\|x\|^2 - \|x-\mu\|^2 > 2c\cdot\sigma^2\cdot\ln(r) + (c-c^2)\|\mu\|$, which in turn implies $\frac{\nu_z(x)}{\nu(x)} > e^{c\cdot\ln(r)+(c-c^2)\|\mu\|^2}$. Now observe that $g(r) = e^{c\cdot\ln(r)+(c-c^2)\|\mu\|^2}$ is an increasing and continuous function of r. Also observe that all the steps we took are reversible, therefore we have $\nu_Y(x)/\nu_X(x) > r$ if and only if $\nu_Z(x)/\nu_X(x) > g(r)$.

A.4 Proof of Propostion 3.0

Proof: Let $\mu \in R^d$ be an arbitrary unit vector with $\|\mu\| = 1$. By the symmetry of Gaussians, we know that for any constant $c \in \mathbb{R}^+$ we have $\mathbf{TV}_a(\mathcal{N}(0^d, \sigma^2 \cdot I_d)$, Let $\mu \in R^d$ be an arbitrary unit vector with $\|\mu\| = 1$. Again by the symmetry of Gaussians, we know that for any constant $c \in \mathbb{R}^+$ we have $\mathbf{TV}_a(\mathcal{N}(0^d, \sigma^2 \cdot I_d)$, Let $\mu \in R^d$ be an arbitrary unit vector with $\|\mu\| = 1$. Let us define $Y_j^* = \mathcal{N}(r_j \cdot \mu, \sigma^2 \cdot I_d)$. By Lemma 3.0 we know that for all $Y_j \in \mathcal{Y}_j$ we have

$$\mathbf{TV}_a(X, Y_j) \leq \mathbf{TV}_a(X, Y_j^*).$$

On the other hand, by Proposition 3.0 we know that for all $j, j' \in [k]$, the triplet $(X, Y_j^*, Y_{j'}^*)$ are compatible. Hence, $(X, \mathcal{Y})$ form a nice system of distributions.

A.5 Proof of Proposition 3.1

Proof: Let $f_{\sigma,p}$ be the trade-off function associated with a single step of the sub-sampled Gaussian mechanism with noise σ. When $p = 1.0$ we simply write f_σ. We also define $\bar{f}(x) = 1 - f(x)$. We have,

$$\bar{f}_{p,\sigma}(\alpha) = (1-p) \cdot \alpha + p \cdot (\bar{f}_\sigma(\alpha))$$

Now consider applying this function twice, we have

$$\begin{aligned}\bar{f}_{p,\sigma}(\bar{f}_{p,\sigma}(\alpha)) &= (1-p)\bar{f}_{\sigma,p}(\alpha) + p\bar{f}_\sigma(\bar{f}_{p,\sigma}(\alpha)) \\ &= (1-p)^2(\alpha) + p(1-p)\bar{f}_\sigma(\alpha) + p\bar{f}_\sigma\big((1-p)\alpha + p\bar{f}_\sigma(\alpha)\big).\end{aligned}$$

We know that trade-off functions are convex, so using Jensen's inequality we have,

$$\begin{aligned}\bar{f}_{p,\sigma}(\bar{f}_{p,\sigma}(\alpha)) &\geq (1-p)^2(\alpha) + 2p(1-p)\bar{f}_\sigma(\alpha) + (1-p)^2\bar{f}_\sigma\big(\bar{f}_\sigma(\alpha)\big) &(2)\\ &= (1-p)^2\alpha + 2p(1-p)\bar{f}_\sigma(\alpha) + p^2\bar{f}_{\sigma/2}(\alpha) &(3)\end{aligned}$$

Now let $\mu \equiv \mathcal{N}(0, \sigma)$ and $\nu \equiv (1-p)^2\mathcal{N}(0, \sigma) + 2p(1-p)\mathcal{N}(1, \sigma) + p^2\mathcal{N}(2, \sigma)$. The trade-off function between μ and ν is equal to

$$1 - T(\mu, \nu)(\alpha) = (1-p)^2\alpha + 2p(1-p) \cdot f_\sigma(\alpha) + p^2 f_{\sigma/2}(\alpha),$$

which is equal to the right hand side of Equation 2. Using a simple induction, we can show that for all k, defining $\mu = \mathcal{N}(0, \sigma)$ and $\nu = \mathcal{N}(B(k, p), \sigma)$, we can show that the trade-off function between μ and ν is always dominated by the function $f_{\sigma,p}^k$. Also note that this domination is strict as long as $k > 1$ and $0 < p < 1$. This shows that for a single step of DP-SGD, our group privacy bound is strictly better than what is entailed by applying the trade-off function recursively. For more than one step, we can simply use Lemma 2.2 and show that our bound is strictly better for many steps of DP-SGD as well.

B Extra experiments

B.1 Growth of group privacy with step size

In this section, we demonstrate the growth of group privacy parameters with the step size. In each plot, we fix the sampling rate and noise parameters and calculate the group privacy at various step sizes. In general, we observe that the growth of group privacy is faster in the smaller iterations, but it becomes slower as the number of steps further increase.

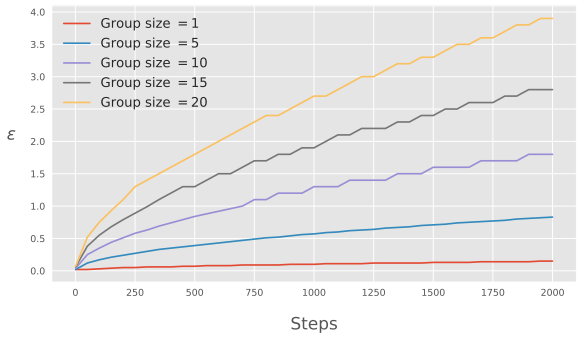

Figure 1: Noise Multiplier=10.00, Sampling rate=0.01.

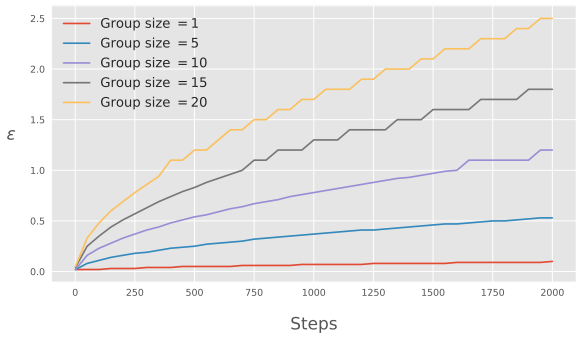

Figure 2: Noise Multiplier=15.00, Sampling rate=0.01.

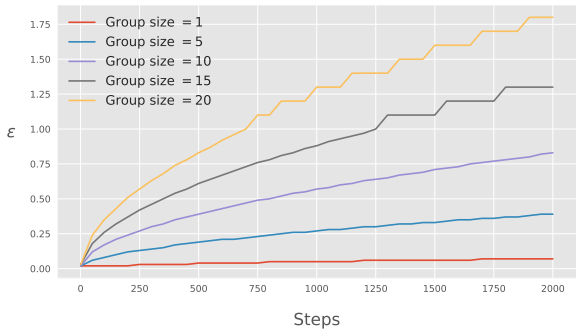

Figure 3: Noise Multiplier=20.00, Sampling rate=0.01.

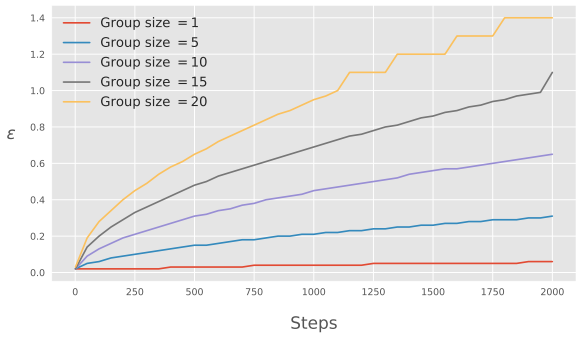

Figure 4: Noise Multiplier=25.00, Sampling rate=0.01.

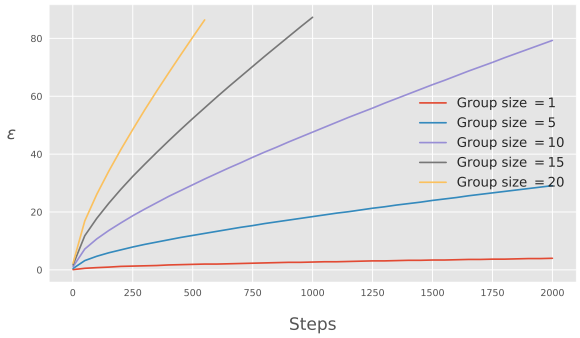

Figure 5: Noise Multiplier=5.00, Sampling rate=0.10.

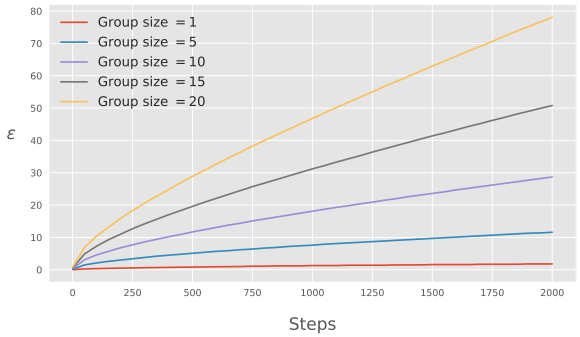

Figure 6: Noise Multiplier=10.00, Sampling rate=0.10.

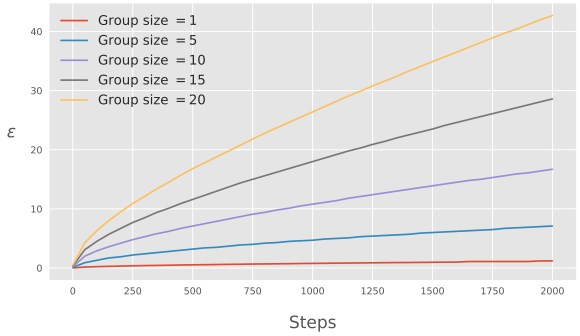

Figure 7: Noise Multiplier=15.00, Sampling rate=0.10.

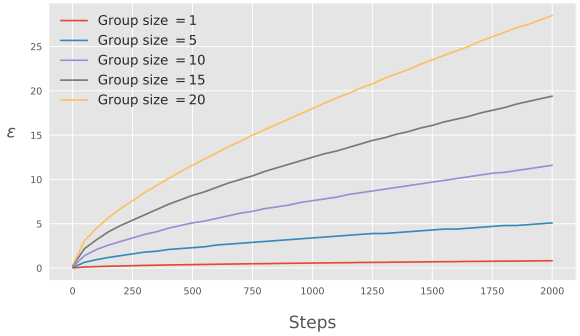

Figure 8: Noise Multiplier=20.00, Sampling rate=0.10.

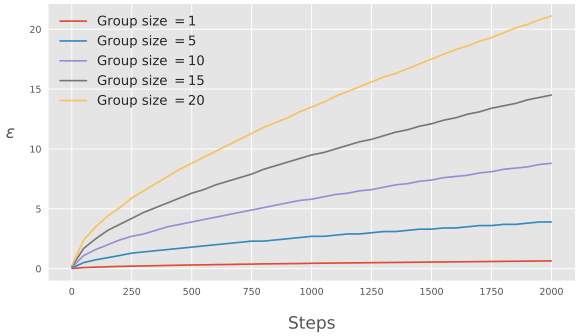

Figure 9: Noise Multiplier=25.00, Sampling rate=0.10.

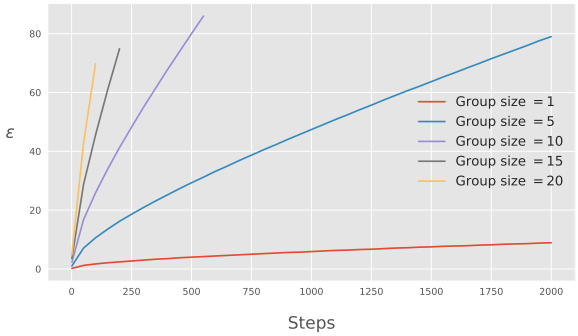

Figure 10: Noise Multiplier=5.00, Sampling rate=0.20.

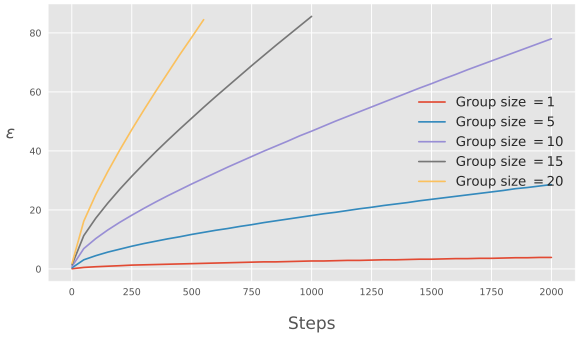

Figure 11: Noise Multiplier=10.00, Sampling rate=0.20.

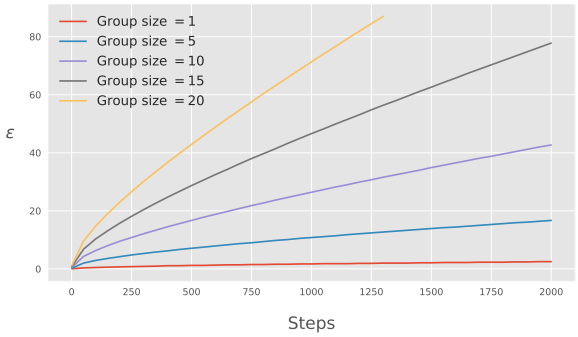

Figure 12: Noise Multiplier=15.00, Sampling rate=0.20.

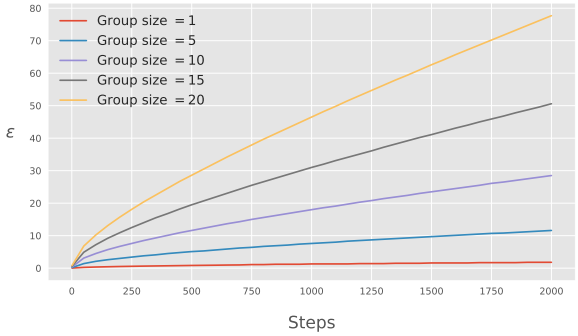

Figure 13: Noise Multiplier=20.00, Sampling rate=0.20.

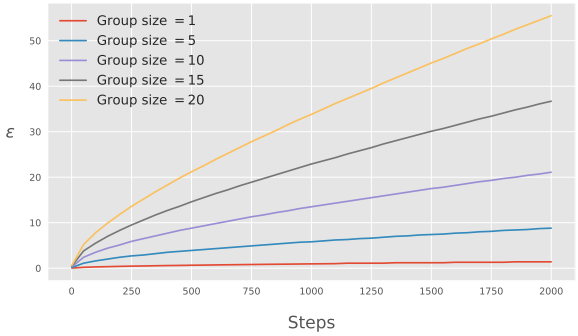

Figure 14: Noise Multiplier=25.00, Sampling rate=0.20.

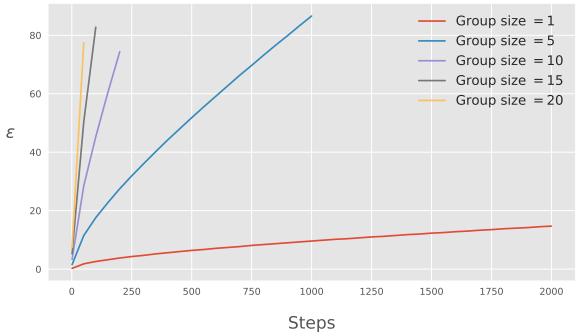

Figure 15: Noise Multiplier=5.00, Sampling rate=0.30.

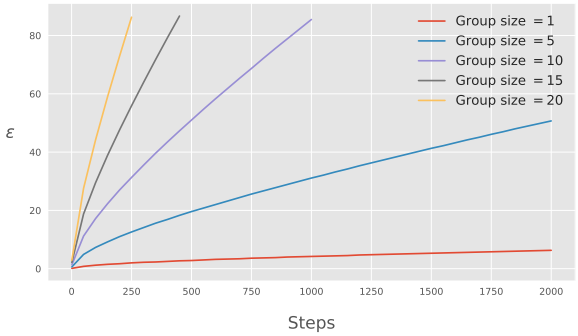

Figure 16: Noise Multiplier=10.00, Sampling rate=0.30.

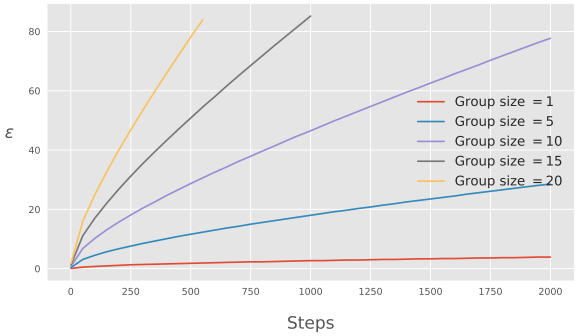

Figure 17: Noise Multiplier=15.00, Sampling rate=0.30.

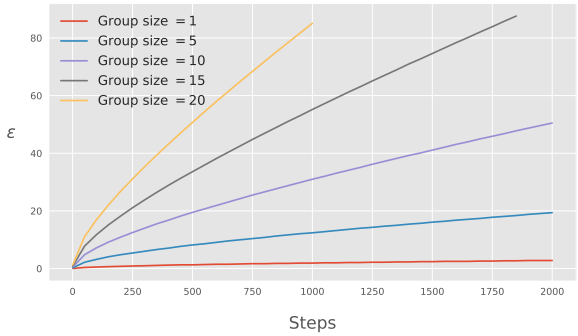

Figure 18: Noise Multiplier=20.00, Sampling rate=0.30.

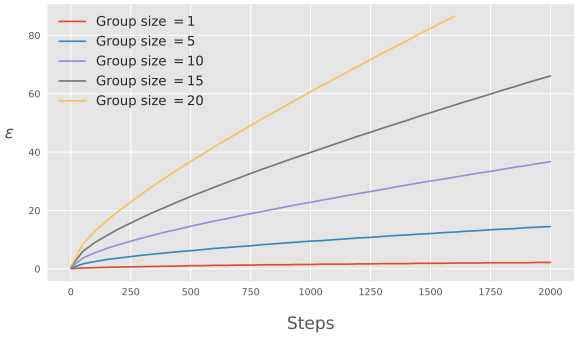

Figure 19: Noise Multiplier=25.00, Sampling rate=0.30.

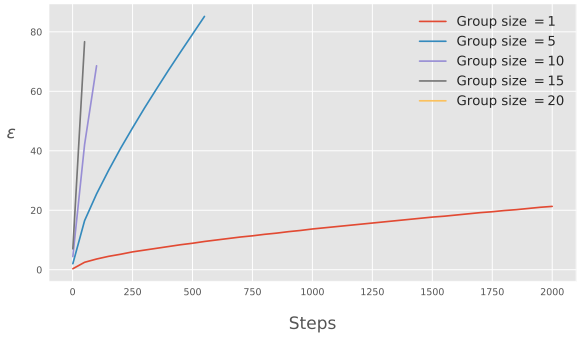

Figure 20: Noise Multiplier=5.00, Sampling rate=0.40.

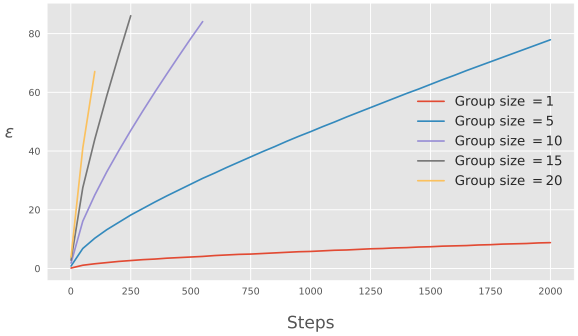

Figure 21: Noise Multiplier=10.00, Sampling rate=0.40.

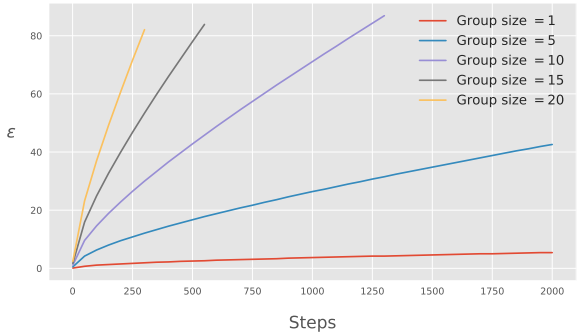

Figure 22: Noise Multiplier=15.00, Sampling rate=0.40.

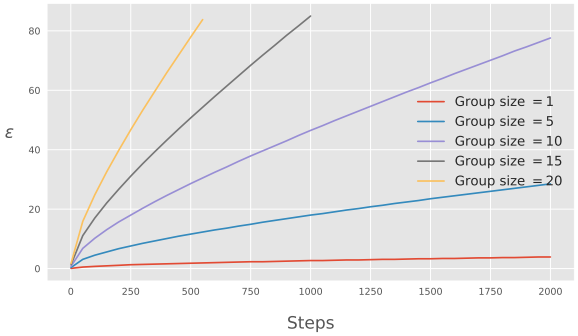

Figure 23: Noise Multiplier=20.00, Sampling rate=0.40.

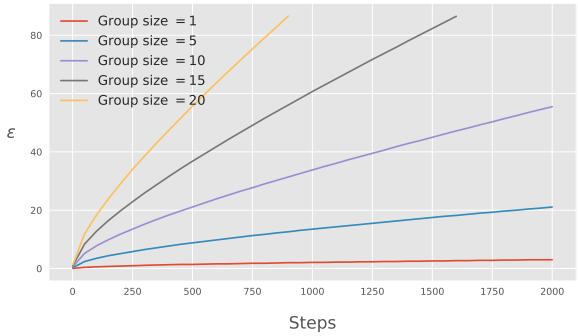

Figure 24: Noise Multiplier=25.00, Sampling rate=0.40.

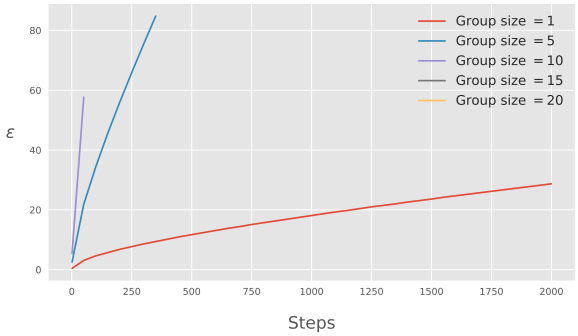

Figure 25: Noise Multiplier=5.00, Sampling rate=0.50.

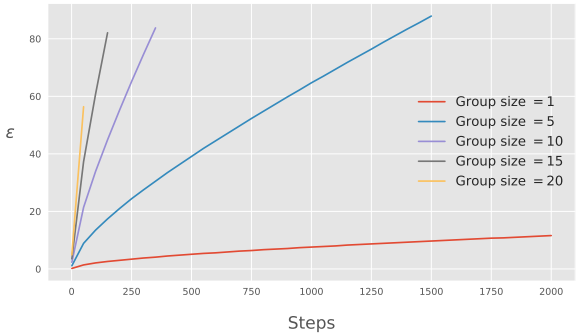

Figure 26: Noise Multiplier=10.00, Sampling rate=0.50.

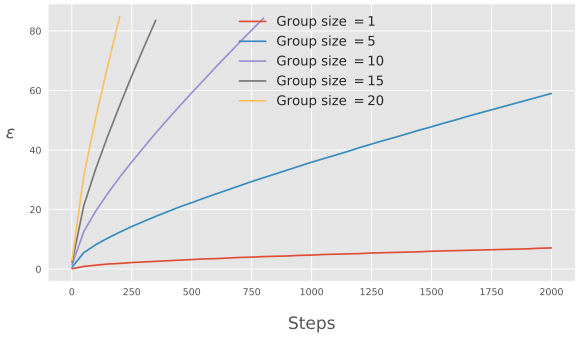

Figure 27: Noise Multiplier=15.00, Sampling rate=0.50.

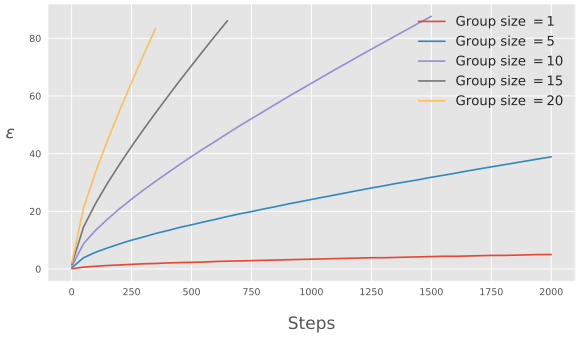

Figure 28: Noise Multiplier=20.00, Sampling rate=0.50.

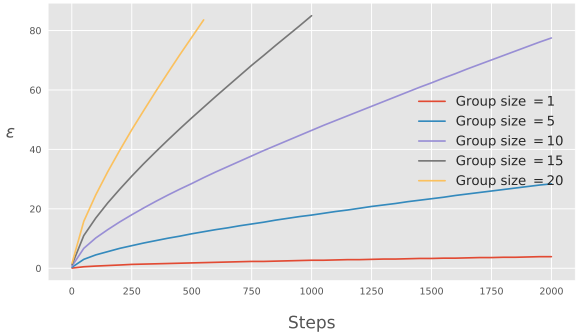

Figure 29: Noise Multiplier=25.00, Sampling rate=0.50.

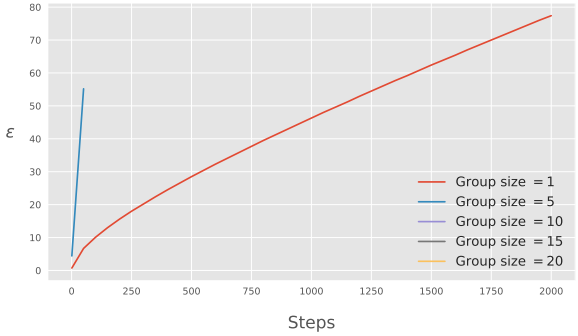

Figure 30: Noise Multiplier=5.00, Sampling rate=1.00.

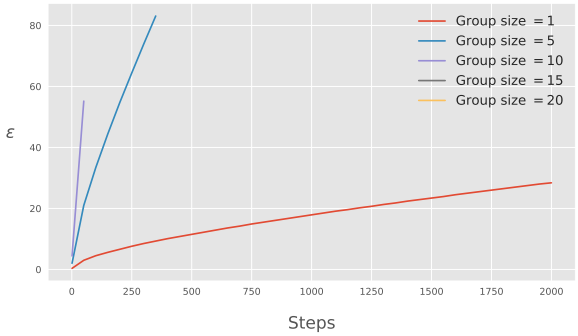

Figure 31: Noise Multiplier=10.00, Sampling rate=1.00.

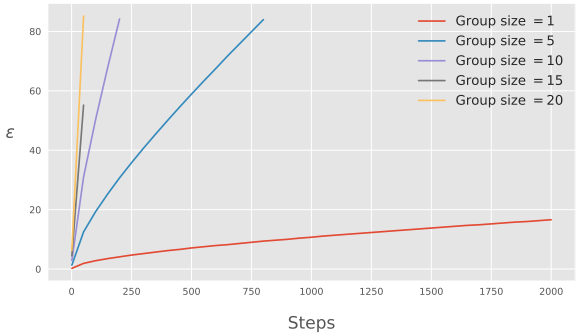

Figure 32: Noise Multiplier=15.00, Sampling rate=1.00.

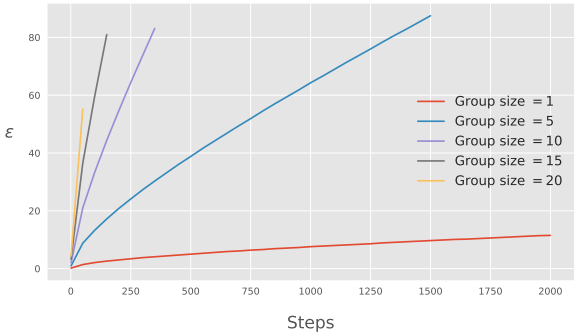

Figure 33: Noise Multiplier=20.00, Sampling rate=1.00.

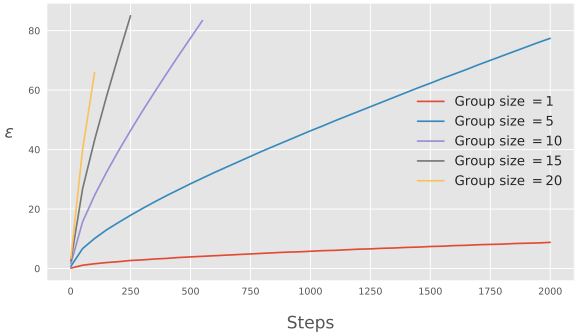

Figure 34: Noise Multiplier=25.00, Sampling rate=1.00.

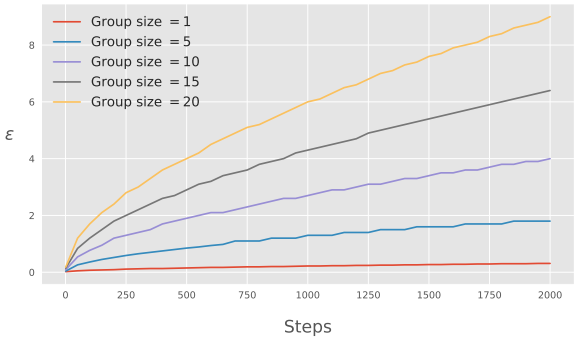

Figure 35: Noise Multiplier=5.00, Sampling rate=0.01.

B.2 Effect of sub-sampling rate on group privacy at various step sizes

In this section, we demonstrate the effect of sub-sampling rate on the group privacy. In the following plots, we set the step size to a fixed value T. We also fix a list of of sampling rates $\{p_i\}_{i\in[10]}$. For each sampling rate, we carefully select a σ_i so that the privacy cost of performing T steps with sampling rate p_i at noise σ_i is exactly the same. Then we calculate the group privacy for all these settings and compare them. Our initial experiments suggested that decreasing the sampling rate will improve group privacy. We ablate this with various step sizes. We try to perform this ablation by plotting figures for a various step sizes. We observe that the effect of sub-sampling rate on group privacy is much more pronounced at smaller step sizes.

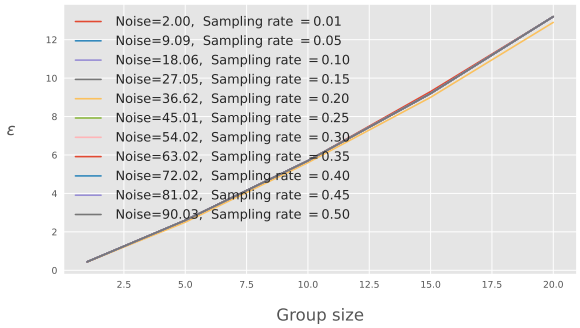

Figure 36: Steps=500.

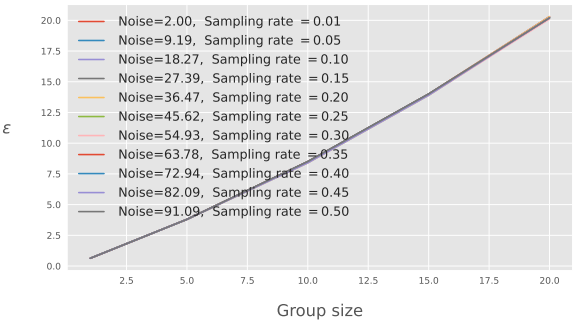

Figure 37: Steps=1000.

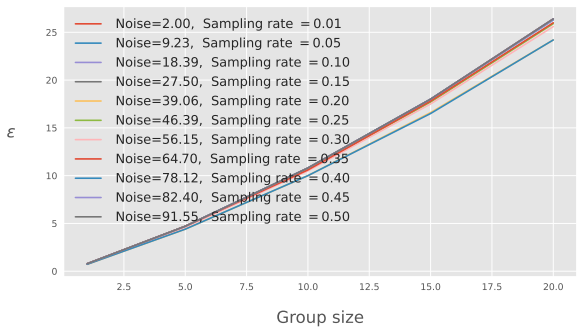

Figure 38: Steps=1500.

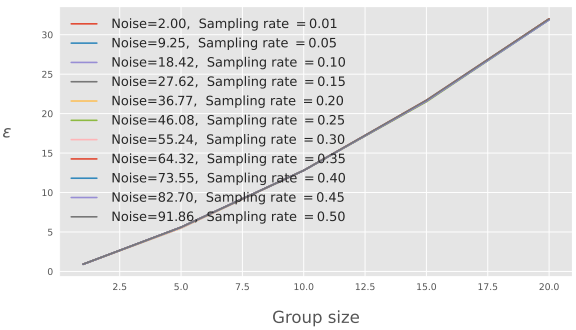

Figure 39: Steps=2000.

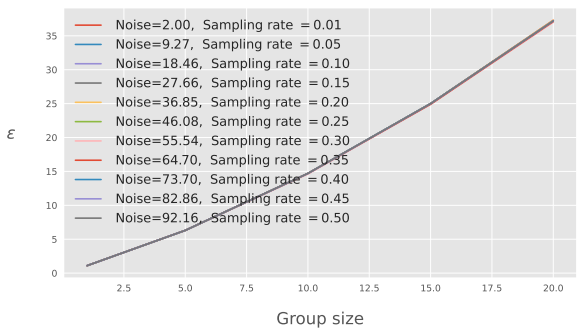

Figure 40: Steps=2500.

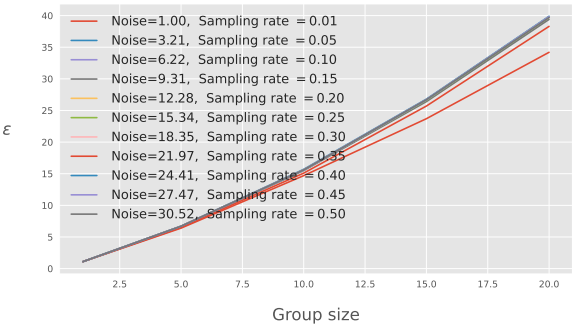

Figure 41: Steps=300.

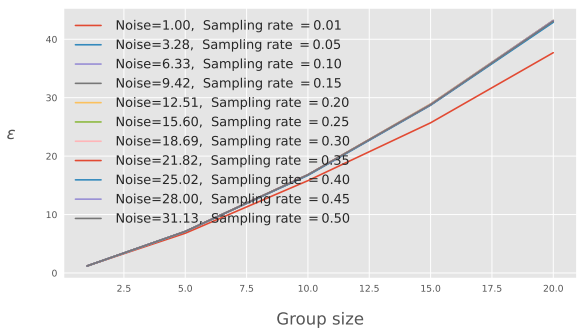

Figure 42: Steps=350.

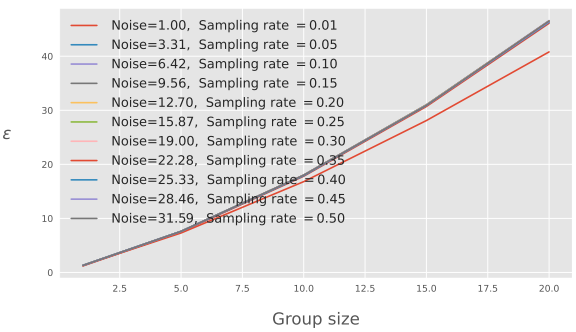

Figure 43: Steps=400.

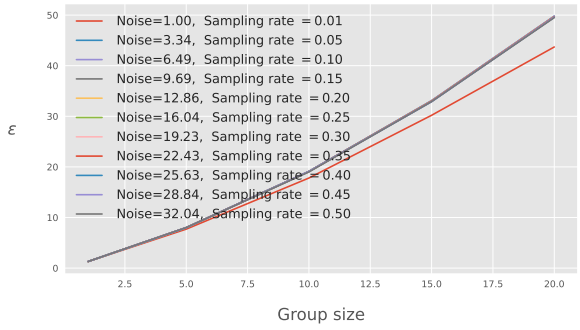

Figure 44: Steps=450.

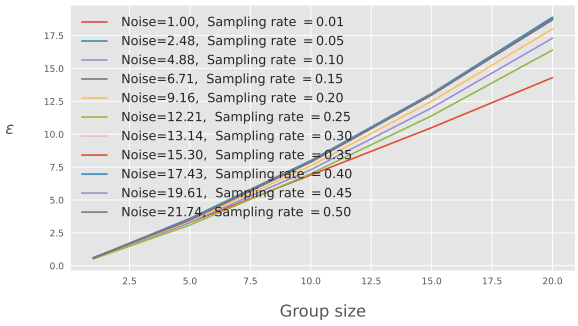

Figure 45: Steps=50.

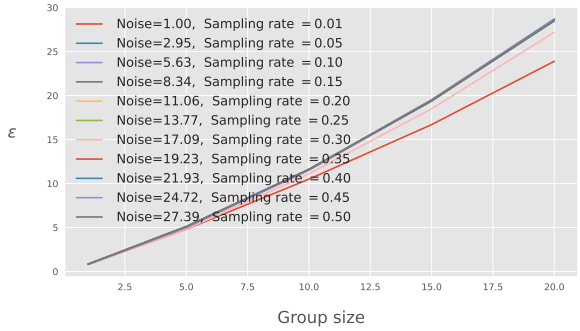

Figure 47: Steps=150.

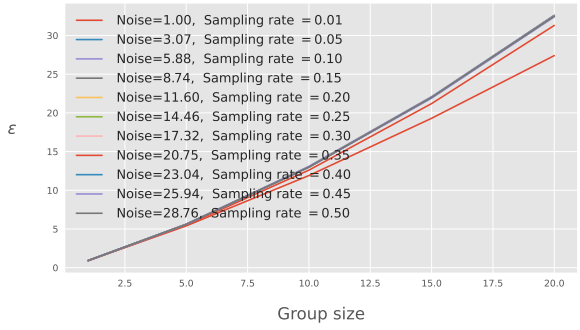

Figure 48: Steps=200.

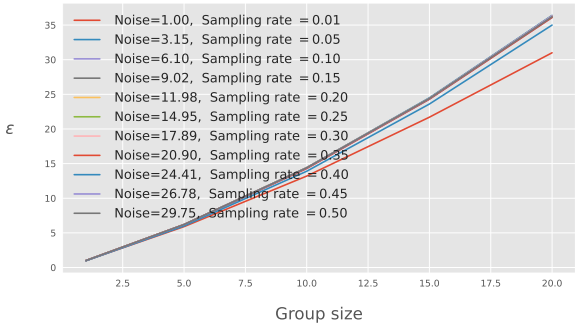

Figure 49: Steps=250.