DIFFERENTIAL PRIVACY: GETTING MORE FOR LESS

CYNTHIA DWORK

ABSTRACT

The key to the success of differential privacy, now the gold standard for privacy-preserving data analysis, is the ability to quantify and reason about cumulative privacy loss over many differentially private interactions. When upper bounds on privacy loss are loose, the deployment of the algorithms is by definition conservative. Under high levels of composition, much potential utility is lost. We survey two general approaches to getting more utility: privacy amplification methods, which are algorithmic, and definitional methods, which admit a wider class of algorithms and lead to tighter analyses of existing algorithms.

MATHEMATICS SUBJECT CLASSIFICATION 2020

Primary 68P27; Secondary 68W20, 68Q32, 68T05

KEYWORDS

Differential privacy, private data analysis, privacy-preserving data analysis, machine learning, private machine learning

ICM INTERNATIONAL CONGRESS OF MATHEMATICIANS 2022 JULY 6—14

Proc. Int. Cong. Math. 2022, Vol. 6, pp. 4740–4761
DOI 10.4171/ICM2022/196

1. INTRODUCTION

The Fundamental Law of Information Recovery states, informally, that "overly accurate" estimates of "too many" statistics completely destroy privacy ([11] *et sequelae*; see [21] for a survey). Differential privacy is a mathematically rigorous definition of privacy tailored to analysis of large datasets and equipped with a formal measure of privacy loss [12,15–18]. Differentially private algorithms take as input a parameter, typically called ε, that caps the permitted privacy loss in any execution of the algorithm and offers a concrete privacy/utility tradeoff. The key to differential privacy's success is the ability to reason about cumulative privacy loss as the data are analyzed and reanalyzed, that is, we can understand its behavior under *composition*. This permits modular construction of differentially private algorithms from simple differentially private building blocks; in other words, differential privacy is *programmable*. The art of differentially private algorithm design is to obtain as much utility as possible, while minimizing cumulative privacy loss.

A statistic is a quantity computed from a sample. Statistics "feel" private for the same reasoning that statistics works as a discipline, meaning that we expect we will obtain approximately the same outcome independent of the actual sample chosen, provided that proper sampling procedures are followed and the sample is sufficiently large. In this sense, the statistic is not "about" the members of the sample, but instead it is a quantity that describes the population as a whole. Relatedly, statistics feel private because of the privacy of the sample: "I could have opted out," "no one knows that I am in the sample." Differential privacy adheres to this intuition, maintaining the "I could have opted out" semantics even when the computations are carried out on the entire population, as in a census.

Roughly speaking, differential privacy ensures that the outcome of any analysis on a dataset x is distributed very similarly to the outcome on any neighboring dataset y that differs from x in just one row. That is, differentially private algorithms are randomized, giving rise to a probability distribution $\mathcal{A}(x)$ when run on dataset x; the definition requires that the *max divergence* between these two distributions $\mathcal{A}(x), \mathcal{A}(y)$ (the maximum log odds ratio for any event $S \in \text{Range}(\mathcal{A})$, also known as the maximum *privacy loss*; Definition 2.2 below) is bounded by a privacy parameter ε. This absolute guarantee on the maximum privacy loss is now known as *pure* differential privacy.

This absolute bound on privacy loss is a conservative estimate of the privacy offered by any given execution of the algorithm: for $\varepsilon \leq 1$, the expected privacy loss of an arbitrary ε-differentially private algorithm is bounded by ε^2 [19,27]. Under very high levels of composition, which is the norm in machine learning and in continual, industrial-scale tracking, as is common with cell phone location and usage data, any looseness in the bounds is severely compounded. Thus, analytical techniques that more tightly capture the behavior of the privacy loss random variable quickly lead to improved utility through a more informed selection of the parameters.

Beginning with [13], a large body of work examines what can be achieved by providing high probability, rather than absolute, bounds on the privacy loss. Relaxing the protection goal – that is, relaxing the definition of privacy-preserving analysis – provides opportunities

for more refined analyses, and the choice of relaxation may rightfully be influenced by the analytical tools enabled by the definition. In many "workhorse" cases we are interested in better understanding of very large numbers of invocations of the *same* algorithm, applied to the *same* data. Statistical queries ("What fraction of the people in the dataset satisfy property P?"), and gradient descent, central to data analytics and modern machine learning, respectively, are exemplars of this phenomenon [1–3, 6, 10, 15, 24–26, 34, 37].

This informal note reviews definitions and sketches some exciting recent directions in *privacy amplification*. In Section 2 we establish the notation used throughout, and motivate the definition of pure differential privacy. Section 3 describes some basic differentially private primitives. Section 4 discusses four relaxations and provides some comparisons between them. In Section 5 we provide intuition for three amplification techniques. Section 6 describes some applications.

2. PURE DIFFERENTIAL PRIVACY

Let $\mathcal{U}$ denote a unverse of data records, where each $u \in \mathcal{U}$ corresponds to a possible value of the data of an individual. *Datasets* are multisets of draws from $\mathcal{U}$, and *adjacent* datasets differ in the data of just one individual. There are two notions of adjacency. In *replace-one* adjacency the two sets have the same cardinality and agree on all but (possibly) one element; in *add/remove* adjacency one set is contained in the other, and the larger has the data of just one additional individual. The distinction rarely matters, beyond a factor of two in the privacy loss bounds.

(Useful) differentially private algorithms are necessarily randomized [15]. For an algorithm $\mathcal{A}$ and dataset x, we let $\mathcal{A}(x)$ denote the probability distribution on outcomes of the randomized $\mathcal{A}$ operating on x. All probabilities and expectations are taken over the coin flips of the algorithms.

Definition 2.1 (Differential privacy [15]). For $\varepsilon \geq 0$, Algorithm $\mathcal{A}$ is ε-differentially private if, for all adjacent datasets $x, y \in \mathcal{U}^*$ and for any event C in the range of $\mathcal{A}$.

$$\Pr\big[\mathcal{A}(x) \in C\big] \leq e^{\varepsilon} \Pr\big[\mathcal{A}(y) \in C\big] \tag{2.1}$$

where the probabilities are taken over the randomness of $\mathcal{A}$.

Note that the definition is symmetric in x and y and is therefore equivalent to the condition $|\log \frac{\Pr[\mathcal{A}(x)\in C]}{\Pr[\mathcal{A}(y)\in C]}| \leq \varepsilon$. Note also that differential privacy is a worst-case notion: the probabilities are over the randomness of the algorithms, not the choice of the datasets.

Composition refers to running multiple differentially private algorithms on the same dataset, and publishing the outputs at each step.[1] It is easily verified that for any algorithms $\mathcal{A}$ and $\mathcal{A}'$ that are ε and ε' differentially private, respectively, the composition that on input x first runs $\mathcal{A}(x)$ and then runs $\mathcal{A}'(x)$, publishing both outputs, is $(\varepsilon + \varepsilon')$-differentially

1 Composition can be defined more broadly, for example, to cover analyses carried out independently on overlapping datasets; see [20].

private [15]. Moreover, this is true even if $\mathcal{A}'$ is chosen adaptively, after having seen the output of $\mathcal{A}(x)$. Thus, differential privacy is closed under composition: composition does not destroy privacy, but it does, eventually, erode it.

Definition 2.2 (Privacy loss random variable). Let $\mathcal{A}$ be an algorithm and let x, y be adjacent datasets. For all $\xi \in \text{Range}(\mathcal{A})$, the *privacy loss of an outcome* ξ *with respect to* y *when running on* x, denoted $L_{x,y,\xi}$ is the ratio

$$L_{x,y,\xi} = \log \frac{\Pr[\mathcal{A}(x) = \xi]}{\Pr[\mathcal{A}(y) = \xi]}. \tag{2.2}$$

For continuous output spaces, the probabilities above are replaced by the probability density functions.

This definition is not symmetric in x and y, and any event that can occur with nonzero probability when running $\mathcal{A}(x)$ but cannot appear when running $\mathcal{A}(y)$ has infinite privacy loss: if an adversary observes such an event, then it knows, for certain, that the input dataset is not y.

Fix any adjacent x and y, and consider an execution of $\mathcal{A}(x)$ resulting in an output ξ. The loss $L_{x,y,\xi}$ might be positive – which is the case when ξ is more likely under $\mathcal{A}(x)$ than under $\mathcal{A}(y)$ – or it may be negative. The fact that privacy loss can be negative leads to cancellation when we run multiple algorithms: the cumulative privacy loss random variable exhibits a martingale-like behavior, and is tightly concentrated around its expectation. This phenomenon is captured by the *Advanced Composition Theorem* [20], stated in Section 4.

Differential privacy enjoys several other properties, in particular (1) it is "future-proof," meaning it is closed under postprocessing; no amount of computation after the fact, and no auxiliary information obtained from other sources, can increase the realized privacy loss; (2) bounds on privacy loss for *groups* degrades gracefully with the size of the group, and an ε-differentially private algorithm is automatically $k\varepsilon$-differentially private for groups of size k.

3. TWO PRIMITIVES

We briefly describe two primitives, or building blocks, that yield pure differentially private algorithms, which we will need for this note.

Definition 3.1 (Counting queries, statistical queries). A counting query is constructed from a predicate, that is, a mapping $q : \mathcal{U} \to \{0, 1\}$. When applied to a dataset, the counting query is asking how many individuals in the dataset are mapped to 1 ("satisfy q"). The associated statistical query is the fraction of members of the dataset that are mapped to 1.

Randomized Response is a generalization of a technique introduced by Warner to conduct surveys about embarrassing or illegal behavior; it provides plausible deniability, allowing individuals to self-report in a randomized way that is biased towards the truth [36]. In this mechanism, the curator/researcher/analyst does not see the private data: individuals

privatize their information before releasing it to a not-necessarily trusted analyst. Randomized response forms the lion's share of differential privacy in industrial use, where it is viewed as "shifting the trust boundary to the client," absolving the server of risk (and, it is perhaps hoped, liability) of privacy violation even if the server is compromised.

In its simplest form, for $u \in \mathcal{U} = \{0, 1\}$, Randomized Response is the following algorithm [36]:

$\mathcal{RR}(\mathbf{u} \in \{\mathbf{0}, \mathbf{1}\},\ \mathbf{p} \in [\mathbf{0}, \mathbf{1}])$
$b \leftarrow \text{Ber}(p)$
If $b = 1$ **then** output a random draw from Ber$(1/2)$;
Else output u.

Here, the notation Ber(p) denotes the Bernoulli distribution with parameter p. This algorithm, which ensures ε-differential privacy whenever $p \geq 2/(1 + e^{\varepsilon})$ [15], operates on a dataset of size $n = 1$. Given a collection $v = \{v_1, \ldots, v_n\}$, where v_i is obtained by running $\mathcal{RR}(u_i, p)$ (either because individual i ran this algorithm on its own data before sending the result to the server, or because the server has all the data $\{u_1, \ldots, u_n\}$ and has calculated the v_i's as intermediate results), the fraction of the number of u_i that satisfy property q can be estimated by "reverse-engineering" the reported statistics as follows. Let $T = \sum_i v_i$. Approximately $pn/2$ of the observed ones in v come from random draws from Ber$(1/2)$, and so (most of) the remaining 1's come from the truthful responses. Thus, the fraction of positive x_i's is approximately $\frac{T-(pn/2)}{(1-p)n}$. The expected error is $\Theta(\frac{1}{\varepsilon\sqrt{n}})$; when rescaled for a counting query, we get $\Theta(\frac{\sqrt{n}}{\varepsilon})$.

Definition 3.2 (L_1- and L_2-sensitivity). Let $f : \mathcal{U}^* \to \mathbb{R}^d$ be an arbitrary function. The L_1-sensitivity of f is the maximum, over all adjacent datasets x, y, of $\|f(x) - f(y)\|_1$. The L_2-sensitivity of f is the maximum, over all adjacent datasets x, y, of $\|f(x) - f(y)\|_2$.

Sensitivity is a property of the function, and does not depend on the specific dataset to which the function is to be applied.

We will also make use of the 0-centered Laplace distribution, Lap(b), which has density function $\mu(x) = \frac{1}{2b}e^{-|x|/b}$.

Theorem 3.1 (Laplace mechanism [15]). *Let $f : \mathcal{U}^* \to \mathbb{R}^d$ be an arbitrary function, and let Δ_1 denote its L_1-sensitivity. Then the mechanism that, on input dataset x and privacy parameter ε, computes $f(x)$ and adds an independent draw from* Lap(Δ_1/ε) *to each coordinate, satisfies ε-differential privacy. That is,*

$$\mathcal{A}(\varepsilon, x) = f(x) + \left(\text{Lap}\left(\frac{\Delta_1}{\varepsilon}\right)\right)^d \tag{3.1}$$

is ε-differentially private.

The L_1-sensitivity of a counting query is 1; the expected error for this mechanism is $O(\frac{1}{\varepsilon})$, a substantial improvement over randomized response. Indeed, for counting queries,

the error introduced for privacy by the Laplace mechanism is less than the sampling error. In this sense, privacy is "for free."

A natural question is whether addition of Gaussian noise $\mathcal{N}(0, \sigma^2 \mathbb{I}_d)$ also yields differential privacy. It does not, even in one dimension. Let the probability densities of the distributions $\mathcal{N}(0, \sigma^2)$ and $\mathcal{N}(1, \sigma^2)$ at any $t \in \mathbb{R}$ be denoted $\mu_0(t)$ and $\mu_1(t)$, respectively. Then there is no fixed bound on the ratio $\frac{\mu_1(t)}{\mu_0(t)}$ as $|t|$ grows. On the other hand, by choosing σ to be sufficiently large as a function of the sensitivity and ε, we can control the likelihood of *failing* to satisfy privacy loss bounded by ε. This motivates the first of the relaxations of pure differential privacy discussed in the next section.

4. RELAXATIONS OF PURE DIFFERENTIAL PRIVACY

In this section we will discuss several relaxations of differential privacy. The differences between the various notions come down to how they treat very low probability events. A detailed discussion appears in [7].

4.1. Approximate differential privacy

Definition 4.1 (Approximate (or (ε, δ)) differential privacy [13]). For $\varepsilon, \delta \geq 0$, Algorithm $\mathcal{A}$ is (ε, δ)-differentially private if, for all adjacent datasets $x, y \in \mathcal{U}^*$ and for any event C in the range of $\mathcal{A}$,

$$\Pr\big[\mathcal{A}(x) \in C\big] \leq e^{\varepsilon} \Pr\big[\mathcal{A}(y) \in C\big] + \delta, \tag{4.1}$$

where the probabilities are taken over the randomness of $\mathcal{A}$.

When $\delta = 0$, this recovers the definition of pure differential privacy. When $\delta > 0$, even if it is negligibly small, this relaxation provides what amounts to a switch of quantification order: pure differential privacy ensures that on every execution of $\mathcal{A}(x)$ the observed outcome will be essentially equally likely on *all* adjacent datasets y simultaneously. In approximate differential privacy, for any specific y adjacent to x, it is extremely unlikely *ex ante* that the observed value $\mathcal{A}(x)$ will be one that is much more or much less likely when the dataset is x rather than when the dataset is y, but given an output $\xi \leftarrow \mathcal{A}(x)$ it might be possible to find *some* y such that ξ is, say, much more likely to be produced on x than it is on y. For this y, $L_{x,y,\xi}$ would be very large.

Although (ε, δ)-differential privacy satisfies a simple composition theorem analogous to simple composition for pure differential privacy – the epsilons and the deltas add up – it, like pure differential privacy, enjoys the benefits of the Advanced Composition Theorem of Dwork, Rothblum, and Vadhan [20], stated next.

Theorem 4.1 (Advanced composition [20]). *For all $\varepsilon, \delta, \delta' \geq 0$, the class of (ε, δ) differentially private mechanisms satisfies $(\varepsilon', k\delta + \delta')$-differential privacy under adaptive k-fold composition for*

$$\varepsilon' = \sqrt{2k \ln(1/\delta')} \cdot \varepsilon + k\varepsilon(e^{\varepsilon} - 1). \tag{4.2}$$

In Advanced Composition, the dependence on the degree k of composition is of order $\sqrt{k}$, rather than linear in k. Observe that the theorem yields a host of bounds; for each value of $\delta' > 0$, one obtains a corresponding value of ε', and *vice versa*.

Remark 4.2. The coefficient $\varepsilon(e^{\varepsilon} - 1)$ can be improved by a factor of 2 [19,27]; tight bounds are obtained in [27] for the homogeneous case (same epsilons and deltas). The optimal composition bound in the nonhomogeneous case is very hard (#-P hard) to compute [32].

While the Gaussian mechanism, described next, cannot offer pure differential privacy, it yields approximate differential privacy.

Theorem 4.3 (Gaussian mechanism [13]). *Let $f : \mathcal{U}^* \to \mathbb{R}^d$ be an arbitrary function, and let Δ_2 denote its L_2-sensitivity. Then the mechanism that, on input dataset x and privacy parameter ε, computes $f(x)$ and adds an independent draw from $\mathcal{N}(0, c^2(\Delta_2/\varepsilon)^2\mathbb{I}_d)$ to each coordinate, satisfies (ε, δ)-differential privacy whenever $c^2 \geq 2\ln(2/\delta)$. That is, for any such c,*

$$\mathcal{A}(\varepsilon, x) = f(x) + \left(\mathcal{N}\left(0, c^2(\Delta_2/\varepsilon)^2\mathbb{I}_d\right)\right)^d \tag{4.3}$$

is (ε, δ)-differentially private.[2]

Approximate differential privacy can also provide an appealing bridge to *robust statistics* through the so-called *Propose-Test-Release* paradigm [14]. Focusing here on the one-dimensional case, in this framework, one runs a differentially private algorithm to test whether the dataset x is far, in Hamming distance, from all datasets y for which $|f(x) - f(y)|$ is larger than some fixed Δ. If so, the algorithm releases $f(x) + \mathrm{Lap}(\frac{\Delta}{\varepsilon})$, and otherwise it releases a special output $\perp$.[3] Such an algorithm has a risk of a false positive, which would lead to an inadequate parameter for the Laplace draw, giving rise to the additive δ term. The Propose-Test-Release paradigm is useful when we expect the statistic of interest to be insensitive to small changes on datasets seen in practice, despite the worst-case sensitivity being high.

4.2. (t)Concentrated and Rényi differential privacy

Consider a very undesirable randomized algorithm that draws $b \sim \mathrm{Ber}(10^{-6})$ and proceeds to either release the entire dataset, if $b = 1$; or else outputs the empty string, if $b = 0$. This algorithm is $(0, 10^{-6})$-differentially private, but this does not sound like a good idea. For these and other reasons, we think in terms of choosing δ to be cryptographically small. The variants considered in this section are *strengthenings* of approximate differential privacy that preclude such "death and destruction" behavior.

An investigation of the Gaussian mechanism reveals that it never results in catastrophic – that is, infinite – privacy loss. In fact, the distribution of the privacy loss random

2 This bound on c is not tight.

3 The Hamming distance between two datasets is the number of elements on which they differ, so Hamming distance to a set is a sensitivity-1 query.

variable for the Gaussian mechanism is itself a Gaussian! Roughly speaking, the probability of privacy loss exceeding its expectation by $k\varepsilon$ falls exponentially in $k^2/2$. Dwork and Rothblum proposed this as a new relaxation of pure differential privacy, which they called *Concentrated Differential Privacy* (CDP). Concentrated differential privacy requires that the privacy loss random variable be sub-Gaussian [19]. The compelling motivation is that the Gaussian mechanism with a scale σ^2 that is independent of δ "cuts corners" in a way that has no privacy cost under high levels of composition.

To gain a little insight, suppose we will have T applications of the Gaussian mechanism, and we want an overall guarantee of (ε, δ)-differential privacy. Then, using the advanced composition theorem, one can choose $\varepsilon_0 \approx \varepsilon/\sqrt{T \ln(1/\delta)}$ and $\delta_0 = \delta/T$ for the base mechanism. Speaking intuitively, this ensures that each invocation of the Gaussian mechanism is likely to have privacy loss whose absolute value is bounded by ε_0. So long as these bounds hold simultaneously, we can apply the Azuma–Hoeffding bound as in the proof of the Advanced Composition Theorem to bound the cumulative privacy loss. However, even if we allow some small violations of these individual ε_0 bounds, the cumulative loss will still (likely) exhibit sufficient cancellation, when T is large, and this is what is exploited in concentrated differential privacy. In practical terms, it gets the $\sqrt{\log(1/\delta)}$ term out of σ in the Gaussian mechanism, greatly improving accuracy when δ is small.

Bun and Steinke [8] continued this line of investigation, proposing a relaxation of Concentrated Differential Privacy with similar intuition and closure under postprocessing (unlike [19]). Their definition, based on *Rényi divergence*, defined next, is the variant of differential privacy deployed by US Census Bureau for the 2020 Decennial Census.

Definition 4.2 (Rényi divergence between distributions). The Rényi divergence of order $\alpha \in (1, \infty)$ between distributions P and Q over a sample space Ω (with $P \ll Q$)[4] is defined to be

$$D_\alpha(P \| Q) = \frac{1}{\alpha - 1} \ln \int \left(\frac{P(z)}{Q(z)} \right)^\alpha Q(z)\, dz. \tag{4.4}$$

We follow the convention that $0/0 = 1$. Also, if $P \not\ll Q$, we define the divergence to be infinite. Rényi divergence of order $\alpha = 1$ and $\alpha = \infty$ is defined by continuity.

Definition 4.3 (Zero-concentrated differential privacy (zCDP) [8]). A randomized algorithm $\mathcal{A} : \mathcal{U}^n \to \mathcal{Y}$ satisfies (ξ, ρ)-zero-concentrated differential privacy if for all adjacent $x, x' \in \mathcal{U}^n$ and all $\alpha \in (1, \infty)$,

$$D_\alpha\big(\mathcal{A}(x) \| \mathcal{A}(x')\big) \le \xi + \rho\alpha. \tag{4.5}$$

It is common to take $\xi = 0$, which results in a single-parameter formulation, ρ-zCDP, that is easy to work with.

The next notion is a further relaxation that constrains only some fixed lower-order set of divergences instead of all $\alpha \in (1, \infty)$.

4 $P(S) = 0$ whenever $Q(S) = 0$ for all measurable sets S; that is, P is absolutely continuous with respect to Q.

Definition 4.4 (Rényi differential privacy [30]). A mechanism $\mathcal{A}$ satisfies (α, ε)-*Rényi differential privacy* (RDP) if for all adjacent datasets x, x', $D_\alpha(\mathcal{A}(x)\|\mathcal{A}(x')) \leq \varepsilon$.

For $\alpha' \in [1, \alpha)$, one has $D_{\alpha'}(P\|Q) < D_\alpha(P\|Q)$ [30], thus if $\mathcal{A}$ satisfies (α, ε)-RDP then it satisfies (α', ε) for all $\alpha' \in (1, \alpha]$.

We introduce one final variation, truncated concentrated differential privacy (tCDP), which lies between zCDP and RDP [7].

Definition 4.5 (Truncated concentrated differential privacy [7]). Let $\rho > 0$ and $\omega > 1$. A randomized algorithm $\mathcal{A} : \mathcal{U}^n \to \mathcal{Y}$ satisfies ω-truncated ρ-concentrated differential privacy (or (ρ, ω)-tCDP) if, for all adjacent datasets $x, x' \in \mathcal{U}^n$, $\forall \alpha \in (1, \omega)$,

$$D_\alpha\big(\mathcal{A}(x)\|\mathcal{A}(x')\big) \leq \rho\alpha. \tag{4.6}$$

Setting $\omega = \infty$ exactly recovers the definition of ρ-zCDP.

4.3. Some relationships among the relaxations

Following the discussion in [7], for adjacent datasets x, x', we examine the random variable $Z = f(\mathcal{A}(x))$, where $f(\zeta) = \ln(\Pr[\mathcal{A}(x) = \zeta]/\Pr[\mathcal{A}(x') = \zeta])$; this is simply the privacy loss random variable $L_{x,x',\xi}$.

- Pure ε-differential privacy requires $Z \leq \varepsilon$.
- ρ-zCDP requires that Z is sub-Gaussian: the tail behavior of Z should be like that of $\mathcal{N}(\rho, 2\rho)$, with $\Pr[Z > t + \rho] \leq e^{-t^2/(4\rho)}$ for all $t \geq 0$.
- (ρ, ω)-tCDP also requires Z to be sub-Gaussian near the origin, but only subexponential in its tails. That is, $\Pr[Z > t + \rho] \leq e^{-t^2/(4\rho)}$ for all $t \in [0, 2\rho(\omega - 1)]$, and for $t > 2\rho(\omega - 1)$, we have $\Pr[Z > t + \rho] \leq e^{(\omega-1)^2\rho} \cdot e^{-(\omega-1)t}$.
- (ω, ε)-Rényi differential privacy requires $\Pr[Z > t + \varepsilon] \leq e^{-(\omega-1)t}$.
- Up to constant factors (on δ), (ε, δ)-differential privacy requires $\Pr[Z > \varepsilon] \leq \delta$.

See [30] for further discussion.

All the variants introduced in this subsection enjoy pleasant composition bounds (recall that (ε, δ)-differential privacy is governed by the Advanced Composition Theorem (Theorem 4.1)):

Theorem 4.4 (Composition bounds [7,8,30]).

(1) *Let $\mathcal{A} : \mathcal{U}^n \to \mathcal{Y}$ and $\mathcal{A}' : \mathcal{U}^n \to \mathcal{Y}$ satisfy (ξ, ρ)-zCDP and (ξ', ρ')-zCDP, respectively. Then their composition satisfies $(\xi + \xi', \rho + \rho')$-zCDP* [8].

(2) *Let $\mathcal{A} : \mathcal{U}^n \to \mathcal{Y}$ and $\mathcal{A}' : \mathcal{U}^n \to \mathcal{Y}$ satisfy (ρ, ω)-tCDP and (ρ', ω')-tCDP, respectively. Then their composition satisfies $(\rho + \rho', \min\{\omega, \omega'\})$-tCDP* [7].

(3) *Let $\mathcal{A} : \mathcal{U}^n \to \mathcal{Y}$ and $\mathcal{A}' : \mathcal{U}^n \to \mathcal{Y}$ satisfy (α, ε)-RDP and (α, ε')-RDP, respectively. Then their composition satisfies $(\alpha, \varepsilon + \varepsilon')$-RDP* [30].

Moreover, by analogues of the data processing inequalities [35], zCDP, tCDP, and RDP are closed under postprocessing.

Remark 4.5 (Group privacy for the relaxations). As noted above, pure differential privacy yields privacy for groups of size k with a factor k increase in the privacy loss bound. For (ε, δ)-differential privacy, the second term deteriorates markedly, and we obtain $(k\varepsilon, ke^{(k-1)}\delta)$-differential privacy. Also ρ-zCDP yields ρk^2-zCDP for groups of size k [8]; (ρ, ω)-tCDP yields $(k^2\rho, \omega/k)$-tCDP for groups of size $k \leq \omega$, and this is tight. tCDP also provides group privacy that degrades gracefully for larger groups, but the degradation is worse than for smaller groups [7]. The situation for RDP is more complex; see [30] and more recent results in [31].

Remark 4.6 (Canonical noise distributions). Different variants of differential privacy have different "canonical" noise distribution for low-sensitivity, real-valued queries. In the case of pure differential privacy, this is the Laplace distribution; for zCDP, it is the Gaussian. Bun *et al.* suggest that for (ε, δ)-differential privacy this could be the Laplace distribution with standard deviation Δ/ε, but with its support truncated to the interval $[\pm O(\Delta \log(1/\delta)/\varepsilon)]$ [7] (although, a truncated Gaussian also works so perhaps "canonical" is wide of the mark for this case). For tCDP, Bun *et al.* suggest the sinh-*normal distribution* for parameters $\sigma, A > 0$,

$$Z \leftarrow A \cdot \operatorname{arcsinh}\left(\frac{\sigma}{A} \cdot \mathcal{N}(0,1)\right). \tag{4.7}$$

This is just the Gaussian $\mathcal{N}(0, \sigma^2)$ with exponentially faster tail decay. The tails of this distribution decay doubly exponentially, rather than just in a sub-Gaussian manner, and the value of A determines where the transition from linear to logarithmic occurs.

With this sinh-normal noise distribution it is sometimes possible to obtain significantly more accurate results. We give one example here.

In a *histogram* query, the universe $\mathcal{U}$ is partitioned into some number k of disjoint cells, and the query is asking, for a dataset x, how many elements of x lie in each of the cells. Although the number of cells is very large, the sensitivity of the query is only 1, as adding or removing a single individual can change the count of at most one cell, and that change is bounded by 1. Histogram queries are the workhorse of official statistics, and the question of how accurately one can privately answer histogram queries is well studied. For pure differential privacy, we can, with high probability, achieve error $\Theta(\frac{\log k}{\varepsilon})$ by adding independent Laplace draws to the count for each cell. For (ε, δ)-differential privacy, this can be improved to $\Theta(\frac{\log(1/\delta)}{\varepsilon})$ (truncated Laplace noise). For z-CDP, we get $\Theta(\sqrt{\frac{\log k}{\rho}})$ (Gaussian noise). For tCDP, using noise from the sinh-normal distribution, we obtain error $O(\omega \log\log k)$.

5. PRIVACY AMPLIFICATON TECHNIQUES

5.1. Amplification by subsampling

Consider a statistical query specified by a predicate q. In *subsampling*, one first chooses a random subset $S \subseteq x$ of the dataset, for example, by selecting each element for inclusion with a fixed probability p, and then outputs an ε-differentially private estimate of the statistical query performed on S. What can we say about the privacy of this algorithm? Consider a pair of adjacent datasets x, y, and let u be an element in x but not in y. The probability that u is selected to S is only p. If it is not included then, speaking intuitively, it will incur no privacy loss; if it is included, then its privacy loss is bounded by ε. This informal argument suggests a privacy loss of $p\varepsilon < \varepsilon$, which is *almost* the right answer (it is off by roughly a factor of at most 2). Moreover, there is nothing special about statistical queries. There is one caveat, however, and this harkens back to our earlier discussion of why statistics *feel* private: privacy amplification requires secrecy of the subsample.

Privacy via subsampling was formalized by Beimel, Brenner, Kasiviswanathan, and Nissim [4], who describe it as implicit in [28].

Theorem 5.1 (Privacy via subsampling [4, 28]). *Let $\mathcal{A}$ be an ε^*-differentially private algorithm. Construct an algorithm $\mathcal{B}$ that, on input a dataset $x = \{x_1, \ldots, x_n\}$, creates a new dataset y by including each x_i independently with probability*

$$\frac{e^{\varepsilon} - 1}{e^{\varepsilon^*} - e^{\varepsilon - \varepsilon^*} - 1} \tag{5.1}$$

and then runs $\mathcal{A}(y)$. Then $\mathcal{B}$ is ε-differentially private.

Remark 5.2. Similar results to those obtained in Theorem 5.1 hold for (ε, δ)-DP, in part because of the δ "escape hatch." However, amplification by subsampling is not so easy for concentrated or Rényi differential privacy. This was the motivation for tCDP; see [7] for precise bounds. The special case of the subsampled Gaussian mechanism is treated in [1, 31].

5.2. Amplification by shuffling

Randomized Response (Section 3) is used in the distributed setting (your cell phone), where clients apply privacy-preserving randomization before sending the (randomized) information to the server. No attempt is made to disassociate a response from the client (you) that sent it. In the *shuffling model*, the assorted responses are assumed to be randomly permuted (somehow, by someone), severing individuals from their randomized information. These responses can then be analyzed without further application of differential privacy, as the analysis is simply a form of post-processing. The *shuffle model* emerged from a series of works, inspired by the very thoughtful PROCHLO paper of Bittau et al. [5] exploring the practical considerations of privacy-preserving computations in internet-scale systems, and formalized by Erlingsson, Feldman, Mironov, Talwar, and Thakurta [22] and, slightly differently, by Cheu, Smith, Ullman, Zeber, and Zhiyaev [9]. In the next section we will describe a simple algorithm for statistical queries due to Cheu et al. [9] in the shuffle model, and we focus here on the definition used in that work.

There are n users, each with a datum $x_i \in \mathcal{U}$. The term *dataset* now refers to the union of the x_i, $i \in [n]$, although in this model the dataset is held in distributed form, with user i holding data u_i. Protocols in this model consist of three parts:

(1) A *randomizer* $\mathcal{R} : \mathcal{U} \to \mathcal{Y}^m$ maps individual data points to an m-tuple of values in an arbitrary range $\mathcal{Y}$;

(2) A *shuffler* $\mathcal{S} : \mathcal{Y}^* \to \mathcal{Y}^*$ applies a random permutation to the set of all messages in its first argument. In our context, if each of n individuals applies a randomizer that yields m elements of $\mathcal{Y}$, the shuffler will apply a random permutation to the set of nm messages. This breaks up, into m unlinked messages, the m-tuple produced by any given individual, in addition to severing the individual-to-message(s) connection;

(3) An analyzer $\mathcal{A} : \mathcal{Y}^* \to \mathcal{Z}$ attempts to estimate some function $f(x_1, \ldots, x_n)$ from these messages.

The algorithmic task is to define $\mathcal{R}$ and $\mathcal{A}$ so that the former ensures differential privacy, while in conjunction with the latter it permits accurate estimation.

Definition 5.1 (Differential privacy in the shuffled model [9]). A protocol $P = (\mathcal{R}, \mathcal{S}, \mathcal{A})$ is (ε, δ)-differentially private if the algorithm $\mathcal{S}(\mathcal{R}(x_1), \ldots, \mathcal{R}(x_n))$ is (ε, δ)-differentially private.

In a slightly different formulation, Erlingsson et al. showed that shuffling improves ordinary randomized response by a factor of $\Omega(\sqrt{\frac{\log(1/\delta)}{n}})$ [22]; simplifications and the optimal dependence on ε appear in [23]. In Section 6 we will see a special case of this improvement, due to [9]. This is an enormous gain for internet-scale n. Deploying shufflers would go a long way toward remedying the cumulative privacy erosion of continual facially differentially private (but with very large ϵ) monitoring, e.g., of phones, browser activity, and activities within app usage.

5.3. Amplification by secrecy of the journey

The intuition behind this approach, which in the literature is referred to as *privacy amplification by iteration* [24], is that privacy is enhanced when the intermediate steps of the algorithm are kept secret. The motivating scenario is private gradient descent. Standard privacy analyses call for each round to satisfy some privacy guarantee, and then to apply composition to the sequence of all rounds. This will work even though it permits the intermediate results to be public, but this is also potentially wasteful: the analyst only needs the result of the final iteration.

The power of keeping intermediate state private was exploited in privacy via subsampling, which relies on the privacy of the subsample, and in the subsample-and-aggregate framework of Nissim, Raskhodnikova, and Smith [33], which provides a method of achieving differential privacy even if the statistic to be computed has large, or difficult to analyze,

sensitivity. In subsample-and-aggregate, the algorithm partitions the dataset into disjoint subsamples, computes the statistic on each subsample without privacy, and then applies a privacy-preserving aggregation mechanism to combine the results. The intuition is that each data point is contained in only one cell of the partitioning, and therefore can affect only one input to the aggregator.

6. APPLICATIONS

In the standard setup for gradient descent in machine learning, we have a dataset $x = \{x_1, \ldots, x_n\}$, where each $x_i \in \mathbb{R}^p$, a convex body $\mathcal{K} \in \mathbb{R}^d$, a starting point $\omega \in \mathcal{K}$, a loss function $L : \mathbb{R}^d \times (\mathbb{R}^p)^n \to \mathbb{R}$. Typically, $L(\omega, x) = \sum_i \ell(\omega, x_i)$ for a convex, Lipschitz, loss $\ell : \mathbb{R}^d \times \mathbb{R}^p \to \mathbb{R}$.

When the $x_i \sim \mathcal{D}$, for a distribution $\mathcal{D}$ on the underlying population, the goal is often convex risk minimization: to find an approximate minimizer of $L(\xi, \mathcal{D})$. The empirical risk is the difference between the population minimizer ξ^* and the empirical minimizer $\hat{\xi}$, and differential privacy adds to the nonprivate empirical risk, since release of $\hat{\xi}$ would be disclosive.

Fix a number of iterations T, and *step sizes* η_t indexed by the iteration number $t \in [T]$. For any $z \in \mathbb{R}^d$, let $\Pi_{\mathcal{K}}(z)$ denote the projection of z onto the convex body $\mathcal{K}$. After setting ω_0 to the starting point ω, each iteration of projected gradient descent has the form

$$\omega_{t+1} \leftarrow \Pi_{\mathcal{K}}\big(\omega_t - \eta_t \cdot \nabla_\omega L(\omega_t, x)\big) = \Pi_{\mathcal{K}}\left(\omega_t - \eta_t \cdot \sum_{i \in [n]} \nabla_\omega \ell(\omega_t, x_i)\right) \tag{6.1}$$

This is easily made differentially private by introducing appropriately scaled Gaussian noise before multiplication by the stepsize and projection,

$$\omega_{t+1} \leftarrow \Pi_{\mathcal{K}}\left(\omega_t - \eta_t \cdot \left[\sum_{i \in [n]} \nabla_\omega \ell(\omega_t, x_i) + \mathcal{N}\big(0, \sigma^2 \mathbb{I}_d\big)\right]\right). \tag{6.2}$$

Ensuring that σ is sufficiently large to provide privacy for the gradient computation

$$\sum_{i \in [n]} \nabla_\omega \ell(\omega_t, x_i) \tag{6.3}$$

suffices, as multiplication by η_t and projection onto $\mathcal{K}$ are postprocessing steps (provided the value of η_t is independent of x, which is typical). Note that this algorithm does not require that the loss function ℓ be differentiable, and may be run with any subgradient of ℓ.

There are many variations of the basic approach (for example, full-batch, mini-batch, stochastic), as well as of the problem statement, e.g., assumptions on the functions $f(\cdot, x_i)$, and we will not compare the bounds obtained by the algorithms we discuss, instead focusing on the privacy arguments.

6.1. Privacy via subsampling in gradient descent

For a large dataset, the computational cost of each iteration of full-batch gradient descent (Equation (6.2)) would be prohibitive. Bassily, Smith, and Thakurta [3], in an ele-

gant feat of differential privacy sleight of hand, proposed instead a variant in which at each iteration an element z is chosen uniformly from the dataset $x = \{x_1, \ldots, x_n\}$, and only z will be used in the gradient computation at that iteration:

$$z \sim \{x_1, \ldots, x_n\}, \tag{6.4}$$

$$\omega_{t+1} \leftarrow \Pi_{\mathcal{K}}\left(\omega_t - \eta_t \cdot \left[n\nabla_\omega \ell(\omega_t, z) + \mathcal{N}\left(0, \sigma^2 \mathbb{I}_d\right)\right]\right). \tag{6.5}$$

Observe that the scale and expectation of the simple computation $n\nabla_\omega \ell(\omega_t, z)$ and the expensive computation $\sum_{i\in[n]} \nabla_\omega \ell(\omega_t, x_i)$ are respectively identical.

The computational advantage is enormous: savings of a factor of n! On the other hand, the sensitivity of the computation has increased by a factor of n. This is because the contribution, for the selected x_i, to the gradient is n times what it was in the original computation. This is counteracted by privacy amplification via subsampling results for (ε, δ)-differential privacy. On a dataset of size n, the subsampled dataset of size 1 corresponds to a selection probability of $q = 1/n$.

Using this and advanced composition, Bassily et al. show that (ε, δ)-differential privacy can be achieved when

$$\sigma^2 = O\left(\frac{n^2 \log(n/\delta)\log(1/\delta)}{\varepsilon^2}\right) \tag{6.6}$$

even when running for n^2 rounds,[5] where the constants include the diameter of $\mathcal{K}$ and the Lipshitz bounds on f.

6.2. Privacy amplification via shuffling

Recall the randomized response primitive for Boolean inputs: each individual chooses $b \sim \text{Ber}(p)$ and, if $b = 1$, answers with a random draw from $\text{Ber}(1/2)$ and otherwise answers truthfully. As noted earlier, $p \geq 2/(1 + e^\varepsilon)$ suffices to ensure ε-differential privacy. Recall further that in the shuffling model individuals randomize their responses and then these responses are randomly shuffled into a pool of messages.

Consider the randomizer defined by running the randomized response with (tiny) randomization parameter $p = \frac{\kappa \ln(1/\delta)}{n\varepsilon^2}$, for some constant κ. With such a small value of p, most participants will report truthfully, but a handful will respond randomly. The analyzer simply sums the randomized values to obtain an approximate total count of ones.

Cheu et al. proposed and analyzed this algorithm with the following intuition [9]. We can think of the initial Bernoulli draw as partitioning the participants into a small set H of *noise makers*, and its complement, whose members respond truthfully. The noise makers create their noise in the second Bernoulli draw; the sum of their $\text{Ber}(1/2)$ draws follows a binomial distribution $B(|H|, 1/2)$. Concentration bounds for the first Bernoulli control the

5 Bassily et al. note, "Even nonprivate first-order algorithms – i.e., those based on gradient measurements – must learn information about the gradient at $\Omega(n^2)$ points to get risk bounds that are independent of n (this follows from "oracle complexity" bounds showing that $1/\sqrt{T}$ convergence rate is optimal...)".

size of H. The noise is then added to the sum of the truthful responses from those not in H. For sufficiently large $|H|$, this yields (ε, δ)-differential privacy.

The algorithm has expected error of order $O(\frac{1}{\varepsilon}\sqrt{\ln \frac{1}{\delta}})$ for counting queries (cf. $\Theta(1/\varepsilon)$ for the Laplace mechanism and $\Theta(\sqrt{n}/\varepsilon)$ for randomized response). Moreover, it is *succinct*: each participant sends only a single bit to the shuffler.

6.3. Privacy of the journey

While composition theorems permit modular "analysis by parts" of complex differentially private algorithms, there is one sense in which they may be overly conservative: they provide guarantees for cumulative privacy loss at every step of the computation, even when the intermediate results are not released. Gradient descent is a case in point: why use privacy parameters that permit every intermediate ω_t to be released, when the data analyst only cares about the final value ω_T? In other words, the destination, and not the journey, is the sole object of interest in gradient descent. Can we exploit this? For the case of gradient descent, the answer is positive, and two lovely lines of work reach the same conclusions via very different proof techniques.

Consider a variant of noisy stochastic gradient descent in which dataset elements are processed sequentially: x_1, then x_2, and so on, for $T = n$ rounds:

$$\omega_{t+1} \leftarrow \Pi_{\mathcal{K}}\left(\omega_t - \eta_t \cdot \left[n\nabla_\omega \ell(\omega_t, x_{t+1}) + \mathcal{N}\left(0, \sigma^2 \mathbb{I}_d\right)\right]\right). \tag{6.7}$$

Consider the element x_1, and observe that its impact on the evolving computation does not end with the first iteration, even though that is the only step in which it appears as an argument to ℓ. This is because x_1 has an impact on the choice of ω_1, which in turn affects ω_2, and so on. Nonetheless, from the perspective of x_1, everything after the first iteration is just postprocessing, so adding noise scaled to the sensitivity of the gradient descent step during the first iteration suffices to protect privacy.

However, keeping the journey secret leads to the following speculation. Suppose we add only half the requisite noise, $\mathcal{N}(0, (\sigma/2)^2\mathbb{I}_d)$ in the first iteration, and another half after the second iteration, when the algorithm operates on x_2. After the second step, we will have added *two* Gaussian samples: the first sample during the first iteration, and the second during the second iteration. This is equivalent to a draw from $\mathcal{N}(0, \sigma^2\mathbb{I}_d)$. So perhaps x_1 is completely protected.

This intuition was made rigorous by Feldman, Mironov, Talwar, and Thakurta, who introduced a powerful notion that interpolates between a metric distance on the output space $\mathcal{K} \subset \mathbb{R}^d$ and the information-theoretic Rényi divergence D_α on distributions of outputs on neighboring datasets, together with two key lemmata that manipulate this quantity [24].

Definition 6.1 (Contractive mapping). A function $\psi : \mathbb{R}^d \to \mathbb{R}^d$ is *contractive* if it is 1-Lipschitz.

In this note, the contractive functions of interest are those computed at each iteration of the noisy gradient descent algorithm. Later, we will make use of the following facts:

Proposition 6.1 ([24]). *For suitable learning rates, the steps at the heart of projected gradient descent are contractions:*

(1) *For convex $\mathcal{K} \in \mathbb{R}^d$, the projection $\Pi_{\mathcal{K}}(x) = \arg\min_{y \in \mathcal{K}} \|x - y\|_2$ is a contraction.*

(2) *Let $f : \mathbb{R}^d \to \mathbb{R}$ be convex and β-smooth. For $\eta \leq 2/\beta$, the function $\psi(w) = w - \eta \nabla_w f(w)$ is a contraction.*

Given a starting point $\mathcal{X}_0$, a sequence of contractive maps $\Psi_1, \ldots, \Psi_T$, and sequence of noise distributions $\{\zeta_t\}$, we define a *Contractive Noisy Iteration* (CNI) as

$$\mathcal{X}_{t+1} = \Psi_{t+1}(X_t) + Z_{t+1}, \tag{6.8}$$

where Z_{t+1} is drawn independently from ζ_t, and denote the output of this process as

$$\mathrm{CNI}_T\big(\mathcal{X}_0, \{\Psi_t\}, \{\zeta_t\}\big) \tag{6.9}$$

Let us consider what happens when each Ψ_t is the identity map and each $\zeta_t = \mathcal{N}(0, \sigma^2 \mathbb{I}_d)$. In this case, $\mathcal{X}_{t+1} = \Psi_{t+1}(\mathcal{X}_t) + \mathcal{N}(0, \sigma^2 \mathbb{I}_d) = \mathcal{X}_t + \mathcal{N}(0, \sigma^2 \mathbb{I}_d)$, whence by induction and the fact that the sum of T mean-centered Gaussians of noise scale σ has distribution $\mathcal{N}(0, T\sigma^2 \mathbb{I}_d)$, we obtain $X_T = X_0 + \mathcal{N}(0, T\sigma^2 \mathbb{I}_d)$.

With this in mind we note that if $\|\mathcal{X}_0 - \mathcal{X}_0'\|_2 \leq 1$, then, letting

$$X_T = \mathrm{CNI}_T\big(\mathcal{X}_0, \{I\}, \mathcal{N}\big(0, \sigma^2 \mathbb{I}_d\big)\big), \tag{6.10}$$

$$X_T' = \mathrm{CNI}_T\big(\mathcal{X}_0', \{I\}, \mathcal{N}\big(0, \sigma^2 \mathbb{I}_d\big)\big), \tag{6.11}$$

we have

$$D_\alpha(\mathbb{P}_{\mathcal{X}_T} \| \mathbb{P}_{\mathcal{X}_T'}) \leq \frac{\alpha}{2T\sigma^2}.$$

Feldman et al. show that the identity case is the worst case. Although they consider arbitrary noise distributions ζ, we will confine our attention to Gaussian noise. They make heavy use of the following fact.

Fact 6.1. *For all $x \in \mathbb{R}^d$,*

$$D_\alpha\big(\mathcal{N}\big(0, \sigma^2 \mathbb{I}_d\big) \| \mathcal{N}\big(x, \sigma^2 \mathbb{I}_d\big)\big) = \frac{\alpha \|x\|^2}{2\sigma^2}. \tag{6.12}$$

It is helpful to keep the application in mind: we have two adjacent datasets $x, x' \in (\mathbb{R}^p)^n$. We imagine running noisy gradient descent on them in parallel, starting from a common point $\omega_0 = \omega_0' \in \mathbb{R}^d$. We are interested in the probability distributions on ω_t and ω_t' for $t \in [T]$. Due to the addition of noise, the values of ω_t and ω_t' are random variables, denoted $\mathcal{X}_t$ and $\mathcal{X}_t'$, respectively. We now wish to bound the α-Rényi divergence of the distributions of these two random variables.

Definition 6.2. For distributions P, Q over $\mathbb{R}^d$, the ∞-Wasserstein distance $\mathcal{W}_\infty(P, Q)$ is the smallest real number given by

$$\mathcal{W}_\infty(P, Q) = \inf_{\gamma \in \Gamma(P,Q)} \operatorname*{ess\,sup}_{(x,y)\sim\gamma} \|x - y\|_2, \tag{6.13}$$

where $(x, y) \sim \gamma$ means that the essential supremum is taken relative to measure γ; here Γ is the collection of couplings of P and Q.

The next quantity, *shifted Rényi divergence*, defined in [24], is a hybrid distance notion that interpolates between metric distances between points in $\mathbb{R}^d$ and distributional divergences.

Definition 6.3 (Shifted Rényi divergence [24]). Let P, Q be distributions defined on a Banach space $(\mathcal{Z}, \|\cdot\|)$. For parameters $z \geq 0$ and $\alpha \geq 1$, the *z-shifted Rényi divergence* between P and Q is defined as

$$D_\alpha^{(z)}(P\|Q) = \inf_{P' : \mathcal{W}_\infty(P,P') \leq z} D_\alpha(P'\|Q). \tag{6.14}$$

To understand this, consider the Wasserstein ball of radius z around P. Then P' minimizes, among all distributions in this ball, the α-Rényi divergence to Q. Note that the larger the ball, the smaller the shifted divergence, since increasing the radius only adds to the collection of candidates from which to choose P'. Moreover, when the radius is so large that the ball includes Q, the shifted divergence is zero, since the divergence will be minimized at $P' = Q$.

Lemma 6.1 ([24]). *For all $s > 0$ simultaneously,*

$$D_\alpha^{(z-s)}\big(P + \mathcal{N}\big(0, \sigma^2 \mathbb{I}_d\big)\|Q + \mathcal{N}\big(0, \sigma^2 \mathbb{I}_d\big)\big) \leq D_\alpha^{(z)}(P\|Q) + \frac{\alpha s^2}{2\sigma^2}. \tag{6.15}$$

In other words, letting $\tilde{P} = P + \mathcal{N}(0, \sigma^2 \mathbb{I}_d)$ and $\tilde{Q} = Q + \mathcal{N}(0, \sigma^2 \mathbb{I}_d)$, we reduce the shift amount (the superscript (z) is decreased to $(z - s)$), which corresponds to a stronger requirement (smaller Wasserstein ball), paying a divergence price of $\frac{\alpha s^2}{2\sigma^2}$. Figuratively, we are drawing a ball of smaller radius $(z - s < z)$ around a distribution $\tilde{P}$ that is close to P, and finding the distribution $\tilde{P}'$ within that ball that is closest to $\tilde{Q} = Q + \mathcal{N}(0, \sigma^2 \mathbb{I}_d)$. The noise distribution is fixed; the flexibility over the choice of s should be thought of as providing an opportunity for creative divergence accounting.

Notation. In the sequel, we sometimes abuse notation by writing $\mathcal{X}$ instead of $\mathbb{P}_{\mathcal{X}}$, identifying the random variable with its distribution.

We next observe that contraction reduces shifted divergence.

Lemma 6.2 ([24]). *Let Ψ, Ψ' be contractive maps on $(\mathcal{Z}, \|\cdot\|)$. If $\sup_x \|\Psi(x) - \Psi'(x)\| \leq s$ then for random variables $\mathcal{X}$, and $\mathcal{X}'$ over $\mathcal{Z}$,*

$$D_\alpha^{(z+s)}\big(\Psi(\mathcal{X})\|\Psi'(\mathcal{X}')\big) \leq D_\alpha^{(z)}(\mathcal{X}\|\mathcal{X}'). \tag{6.16}$$

Theorem 6.3 ([24], as stated informally in [2, PROPOSITION 2.17]). *Let $\mathcal{X}_T$ and $\mathcal{X}'_T$ denote the outputs of CNI$(\mathcal{X}_0, \{\phi_t\}, \{\xi_t\})$ and CNI$(\mathcal{X}_0, \{\phi'_t\}, \{\xi_t\})$ where $\xi_t = \mathcal{N}(0, \sigma_t^2 \mathbb{I}_d)$. Denote $s_t = \sup_x \|\phi_t(x) - \phi'_t(x)\|$, and consider any sequence $a_1, \ldots, a_T$ such that $z_t = \sum_{i=1}^t (s_i - a_i)$ is nonnegative for all t and satisfies $z_T = 0$. Then*

$$D_\alpha(\mathbb{P}_{\mathcal{X}_T}\|\mathbb{P}_{\mathcal{X}'_T}) \leq \frac{\alpha}{2} \sum_{t=1}^T \frac{a_t^2}{\sigma_t^2}. \tag{6.17}$$

We can now sketch the analysis in [24] of projected stochastic gradient descent for a fixed choice σ of noise scale. Our starting assumption $\mathcal{X}_0 = \mathcal{X}'_0$ ensures that $\mathcal{W}_\infty(\mathcal{X}_0, \mathcal{X}'_0) \leq 1$ which in turn is equivalent to $D_\alpha^{(1)}(\mathcal{X}_0 \| \mathcal{X}'_0) = 0$. Our desired conclusion is a bound on $D_\alpha(\mathcal{X}_T \| \mathcal{X}'_T) = D_\alpha^{(0)}(\mathcal{X}_T \| \mathcal{X}'_T)$. Adding Gaussian noise allows us to reduce the shift amount (Lemma 6.1), while recording a divergence cost: the greater the shift reduction, the higher the privacy cost. Taking gradient descent steps moves us towards our computational goal but increases the shift amount (Lemma 6.2).

Projected noisy stochastic gradient descent
Input: Dataset $x = \{x_1, \ldots, x_n\}$; $f : \mathcal{K} \times \mathcal{U} \to \mathbb{R}$ a convex function in the first parameter; learning rate η; starting point $\omega_0 \in \mathcal{K}$; noise parameter σ.
For $t \in \{0, \ldots, n-1\}$:
$v_{t+1} \leftarrow \omega_t - \eta(\nabla_\omega f(\omega_t, x_{t+1}) + Z)$, where $Z \sim \mathcal{N}(0, \sigma^2 \mathbb{I}_d)$,
$\omega_{t+1} \leftarrow \Pi_{\mathcal{K}}(v_{t+1})$
End For
Return the final iterate ω_n.

Assuming f is convex and β-smooth in its first parameter, the gradient step is contractive whenever $\eta \leq 2/\beta$.

Let $x, x' \in \mathcal{U}^n$ be adjacent, and let t be the unique index in which they differ. For dataset x, we can define the contractive noisy iteration by the initial point ω_0, the sequence of functions $g_i(\omega) = \Pi_{\mathcal{K}}(\omega) - \eta \nabla f(\Pi_{\mathcal{K}}(\omega), x_i)$ and sequence of noise distributions $\zeta_i \sim \mathcal{N}(0, (\eta\sigma)^2 \mathbb{I}_d)$. The CNI is defined analogously for dataset x', but with $g'_t(\omega) = \Pi_{\mathcal{K}}(\omega) - \eta \nabla f(\Pi_{\mathcal{K}}(\omega), x'_t)$. By assumption, $f(\omega, z)$ is L-Lipschitz for every $\omega \in \mathcal{K}$ and $z \in \mathcal{U}$, and therefore

$$\sup_\omega \left\| g_t(\omega) - g'_t(\omega) \right\|_2 \leq 2\eta L. \tag{6.18}$$

We choose $a_1, \ldots, a_{t-1} = 0$, that is, paying no divergence costs for the first $t-1$ noise additions and obtaining no shift reductions, and $a_t, \ldots, a_n = \frac{2\eta L}{n-t+1}$ for the remaining steps, and noting that the contractive map in the tth iteration increases the shift by $s = 2\eta L$, and there are no further increases because the datasets agree on the remaining steps. A simple induction shows that at every step the shift parameter is nonnegative, while the shift parameter at the end of step n is $z_n = 0$, yielding a bound on divergence at the final step of

$$D_\alpha\left(\mathcal{X}_n \| \mathcal{X}'_n\right) \leq \frac{\alpha}{2\eta^2\sigma^2} \sum_{i=1}^{n} a_i^2 \leq \frac{2\alpha L^2}{\sigma^2 \cdot (n - t + 1)}. \tag{6.19}$$

This yields $(\alpha, \frac{\alpha 2L^2}{\sigma^2(n+1-t)})$-Rényi differential privacy for the tth element. Observe that this bound echoes our earlier discussion of the privacy protection for elements processed early, that is, x_i for small i, and elements processed later. The smaller t in this bound, the larger the denominator, yielding smaller divergence, which captures the privacy loss.

Feldman et al. consider various methods of employing this basic mechanism, or some simple variants, to remedy the reduced protections for x_t when t is large. For example,

suppose (for some reason) we have access to a modest sample $\{y_1, \ldots, y_m\}$ of *nonprivate* data drawn from the same distribution as the members of the dataset. Then one could run the algorithm on the augmented dataset $\{x_1, \ldots, x_n, y_1, \ldots, y_m\}$, keeping the iterates secret and only making public the final $(n+m)$th iterate.

6.3.1. Very large numbers of iterations

Suppose we wish to run a CNI process for more than $T = n$ rounds. Theorem 6.3 above (see [24]) says that $D_\alpha(\mathbb{P}_{\mathcal{X}_T} \| \mathbb{P}_{\mathcal{X}'_T}) \leq \frac{\alpha}{2} \sum_{t=1}^{T} \frac{a_t^2}{\sigma_t^2}$, which goes to infinity as T grows.

In very recent papers, two lines of work show this dependence on T can be avoided. Breakthrough results of Chourasia, Ye, and Shokri (for the nonstochastic case) [10], followed by Ye and Shokri [37] and Ryffel, Bach, and Pointcheval [34], use a diffusion argument to prove that Projected Noisy Stochastic Gradient Descent has a privacy loss that converges as $T \to \infty$, provided the smooth loss functions are also strongly convex. The intuition is that Projected Noisy-SGD is a discretization of a continuous-time algorithm with bounded privacy loss; in particular, it can be viewed as the Stochastic Gradient Langevin Dynamics algorithm, which is a discretization of a continuous-time Markov process whose stationary distribution is equivalent to the differentially private *exponential mechanism* [29].

Using different techniques, Altschuler and Talwar [2] combine the privacy amplification via iteration techniques discussed above with privacy amplification via subsampling for the Gaussian mechanism to also obtain finite privacy loss as T goes to infinity; moreover, they are able to remove the strong convexity assumption.

Theorem 6.4 (Informal statement from [2]). *Let $x = \{x_1, \ldots, x_n\} \in \mathcal{U}^n$, whether each x_i defines a convex L-Lipschitz, and M-smooth loss function $f(\cdot, x_i)$ on a convex region $\mathcal{K} \subset \mathbb{R}^d$ of diameter D. For a large range of parameters, Projected Noisy-SGD, when run for T iterations, satisfies (α, ε)-Rényi differential privacy for*

$$\varepsilon \leq \frac{\alpha L^2}{n^2 \sigma^2} \min\left\{T, \frac{Dn}{L\eta}\right\}, \tag{6.20}$$

and this bound is tight up to a constant factor.

The proof of privacy exploits the diameter on the constraint set, as follows. Noisy-SGD updates on adjacent datasets will eventually diverge to maximally distant points. At that time, which can be shown to be of order $\frac{Dn}{L\eta}$, their *shifted* divergence will be zero! Thus, the proof of privacy only needs to be concerned with the final $T - \bar{T}$ iterations.

ACKNOWLEDGMENTS

We thank Kunal Talwar, Franklyn Wang, and the members of the reading group: Silvia Casacuberta-Puig, Zhun Deng, Elbert Du, Conlan Olson, Manish Raghavan, Pranay Tankala, and Linjun Zhang, for helpful discussions, and David Fuchs and Daniel Lee for helpful comments on earlier versions of this manuscript.

FUNDING

We gratefully acknowledge the support of the Sloan Foundation grant, Towards Practicing Privacy.

REFERENCES

[1] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pp. 308–318, ACM, 2016.

[2] J. M. Altschuler and K. Talwar, Privacy of noisy stochastic gradient descent: more iterations without more privacy loss. 2022, arXiv:2205.13710.

[3] R. Bassily, A. Smith, and A. Thakurta, Private empirical risk minimization: Efficient algorithms and tight error bounds. In *2014 IEEE 55th annual symposium on foundations of computer science*, pp. 464–473, IEEE, 2014.

[4] A. Beimel, H. Brenner, S. P. Kasiviswanathan, and K. Nissim, Bounds on the sample complexity for private learning and private data release. *Mach. Learn.* **94** (2014), no. 3, 401–437.

[5] A. Bittau, Ú. Erlingsson, P. Maniatis, I. Mironov, A. Raghunathan, D. Lie, M. Rudominer, U. Kode, J. Tinnes, and B. Seefeld, Prochlo: strong privacy for analytics in the crowd. In *Proceedings of the 26th ACM symposium on operating systems principles*, pp. 441–459, ACM, 2017.

[6] A. Blum, K. Ligett, and A. Roth, A learning theory approach to non-interactive database privacy. In *Proceedings of the 40th ACM SIGACT symposium on theory of computing*, ACM, 2008.

[7] M. Bun, C. Dwork, G. N. Rothblum, and T. Steinke, Composable and versatile privacy via truncated CDP. In *Proceedings of the 50th annual ACM SIGACT symposium on theory of computing*, pp. 74–86, ACM, 2018.

[8] M. Bun and T. Steinke, Concentrated differential privacy: Simplifications, extensions, and lower bounds. In *Theory of cryptography conference*, pp. 635–658, Springer, 2016.

[9] A. Cheu, A. Smith, J. Ullman, D. Zeber, and M. Zhilyaev, Distributed differential privacy via shuffling. In *Annual international conference on the theory and applications of cryptographic techniques*, pp. 375–403, Springer, 2019.

[10] R. Chourasia, J. Ye, and R. Shokri, Differential privacy dynamics of Langevin diffusion and noisy gradient descent. *Adv. Neural Inf. Process. Syst.* **34** (2021), 14771–14781.

[11] I. Dinur and K. Nissim, Revealing information while preserving privacy. In *Proceedings of the 22nd ACM SIGMOD-SIGACT-SIGART symposium on principles of database systems*, pp. 202–210, ACM, 2003.

[12] C. Dwork, Differential privacy. In *Proceedings of the international colloquium on automata, languages and programming (ICALP)*, pp. 1–12, Springer, 2006.

[13] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor, Our data, ourselves: privacy via distributed noise generation. In *Advances in cryptology – EUROCRYPT 2006, 25th annual international conference on the theory and applications of cryptographic techniques, St. Petersburg, Russia, May 28 – June 1, 2006, proceedings*, pp. 486–503, Springer, 2006.

[14] C. Dwork and J. Lei, Differential privacy and robust statistics. In *Proceedings of the 41st ACM SIGACT symposium on theory of computing*, ACM, 2009.

[15] C. Dwork, F. McSherry, K. Nissim, and A. Smith, Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pp. 265–284, Springer, 2006.

[16] C. Dwork, F. McSherry, K. Nissim, and A. Smith, Calibrating noise to sensitivity in private data analysis. *J. Priv. Confid.* **7** (2016), no. 3, 17–51.

[17] C. Dwork and M. Naor, On the difficulties of disclosure prevention in statistical databases or the case for differential privacy. *J. Priv. Confid.* **2** (2010), no. 1.

[18] C. Dwork and A. Roth, The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.* **9** (2014), no. 3–4, 211–407.

[19] C. Dwork and G. N. Rothblum, Concentrated differential privacy. 2016, arXiv:1603.01887.

[20] C. Dwork, G. N. Rothblum, and S. P. Vadhan, Boosting and differential privacy. In *Proceedings of the 51st IEEE symposium on foundations of computer science*, pp. 51–60, IEEE, 2010.

[21] C. Dwork, A. Smith, T. Steinke, and J. Ullman, Exposed! A survey of attacks on private data. *Annu. Rev. Stat. Appl.* **4** (2017), no. 1, 61–84.

[22] Ú. Erlingsson, V. Feldman, I. Mironov, A. Raghunathan, K. Talwar, and A. Thakurta, Amplification by shuffling: From local to central differential privacy via anonymity. In *Proceedings of the thirtieth annual ACM–SIAM symposium on discrete algorithms*, pp. 2468–2479, SIAM, 2019.

[23] V. Feldman, A. McMillan, and K. Talwar, Hiding among the clones: A simple and nearly optimal analysis of privacy amplification by shuffling. In *2021 IEEE 62nd annual symposium on foundations of computer science (FOCS)*, pp. 954–964, IEEE, 2022.

[24] V. Feldman, I. Mironov, K. Talwar, and A. Thakurta, Privacy amplification by iteration. In *2018 IEEE 59th annual symposium on foundations of computer science (FOCS)*, pp. 521–532, IEEE Computer Society, 2018.

[25] M. Hardt and G. N. Rothblum, A multiplicative weights mechanism for interactive privacy-preserving data analysis. In *Proceedings of the 51st annual IEEE symposium on foundations of computing*, IEEE, 2010.

[26] M. Hardt, G. N. Rothblum, and R. A. Servedio, Private data release via learning thresholds. In *Proceedings of the twenty-third annual ACM–SIAM symposium on discrete algorithms*, pp. 168–187, SIAM, 2012.

[27] P. Kairouz, S. Oh, and P. Viswanath, The composition theorem for differential privacy. *International Conference on Machine Learning*, pp. 1376–1385, PMLR, 2015.

[28] S. P. Kasiviswanathan, H. K. Lee, K. Nissim, S. Raskhodnikova, and A. Smith, What can we learn privately? *SIAM J. Comput.* **40** (2011), no. 3, 793–826.

[29] F. McSherry and K. Talwar, Mechanism design via differential privacy. In *48th annual IEEE symposium on foundations of computer science (FOCS'07)*, pp. 94–103, IEEE, 2007.

[30] I. Mironov, Rényi differential privacy. In *2017 IEEE 30th computer security foundations symposium (CSF)*, pp. 263–275, IEEE, 2017.

[31] I. Mironov, K. Talwar, and L. Zhang, Rényi differential privacy of the sampled gaussian mechanism. 2019, arXiv:1908.10530.

[32] J. Murtagh and S. Vadhan, The complexity of computing the optimal composition of differential privacy. In *Theory of cryptography conference*, pp. 157–175, Springer, 2016.

[33] K. Nissim, S. Raskhodnikova, and A. Smith, Smooth sensitivity and sampling in private data analysis. In *Proceedings of the thirty-ninth annual ACM SIGACT symposium on theory of computing*, pp. 75–84, ACM 2007.

[34] T. Ryffel, F. Bach, and D. Pointcheval, Differential privacy guarantees for stochastic gradient Langevin dynamics. 2022, arXiv:2201.11980.

[35] T. van Erven and P. Harremos, Rényi divergence and Kullback–Leibler divergence. *IEEE Trans. Inf. Theory* **60** (2014), no. 7, 3797–3820.

[36] S. L. Warner, Randomized response: a survey technique for eliminating evasive answer bias. *J. Amer. Statist. Assoc.* **60** (1965), no. 309, 63–69.

[37] J. Ye and R. Shokri, Differentially private learning needs hidden state (or much faster convergence). 2022, arXiv:2203.05363.

CYNTHIA DWORK

Department of Computer Science, Harvard John A. Paulson School of Engineering and Applied Sciences, Harvard University, 150 Western Avenue, Allston, MA 02134, USA, dwork@seas.harvard.edu