Private Stochastic Convex Optimization: Optimal Rates in ℓ_1 Geometry

Hilal Asi[*] Vitaly Feldman[†] Tomer Koren[‡] Kunal Talwar[§]

March 3, 2021

Abstract

Stochastic convex optimization over an ℓ_1-bounded domain is ubiquitous in machine learning applications such as LASSO but remains poorly understood when learning with differential privacy. We show that, up to logarithmic factors the optimal excess population loss of any (ε, δ)-differentially private optimizer is $\sqrt{\log(d)/n} + \sqrt{d}/\varepsilon n$. The upper bound is based on a new algorithm that combines the iterative localization approach of Feldman et al. [FKT20] with a new analysis of private regularized mirror descent. It applies to ℓ_p bounded domains for $p \in [1, 2]$ and queries at most $n^{3/2}$ gradients improving over the best previously known algorithm for the ℓ_2 case which needs n^2 gradients. Further, we show that when the loss functions satisfy additional smoothness assumptions, the excess loss is upper bounded (up to logarithmic factors) by $\sqrt{\log(d)/n} + (\log(d)/\varepsilon n)^{2/3}$. This bound is achieved by a new variance-reduced version of the Frank-Wolfe algorithm that requires just a single pass over the data. We also show that the lower bound in this case is the minimum of the two rates mentioned above.

1 Introduction

Convex optimization is one of the most well-studied problems in private data analysis. Existing works have largely studied optimization problems over ℓ_2-bounded domains. However several machine learning applications, such as LASSO and minimization over the probability simplex, involve optimization over ℓ_1-bounded domains. In this work we study the problem of differentially private stochastic convex optimization (DP-SCO) over ℓ_1-bounded domains.

In this problem (DP-SCO), given n i.i.d. samples $z_1, \ldots, z_n$ from a distribution P, we wish to release a private solution $x \in \mathcal{X} \subseteq \mathbb{R}^d$ that minimizes the population loss $F(x) = \mathbb{E}_{z \sim P}[f(x; z)]$ for a convex function f over x. The algorithm's performance is measured using the excess population loss of the solution x, that is $F(x) - \min_{y \in \mathcal{X}} F(y)$. The optimal algorithms and rates for this problem—even without privacy—have a crucial dependence on the geometry of the constraint set $\mathcal{X}$ and in this work we focus on sets with bounded ℓ_1-diameter. Without privacy constraints, there exist standard and efficient algorithms, such as mirror descent and exponentiated gradient decent, that achieve the optimal excess loss $O(\sqrt{\log(d)/n})$ [SSBD14]. The landscape of the problem, however, with privacy constraints is not fully understood yet.

Most prior work on private convex optimization has focused on minimization of the empirical loss $\hat{F}(x) = \frac{1}{n} \sum_{i=1}^{n} f(x; z_i)$ over ℓ_2-bounded domains [CMS11; BST14; BFTT19]. Bassily et al.

[*] Stanford University, part of this work performed while interning at Apple; asi@stanford.edu.

[†] Apple; vitaly.edu@gmail.com.

[‡] School of Computer Science, Tel Aviv University, and Google; tkoren@tauex.tau.ac.il.

[§] Apple; kunal@kunaltalwar.org.

[BST14] show that the optimal excess empirical loss in this setting is $\Theta(\sqrt{d}/\varepsilon n)$ up to log factors. More recently, Bassily et al. [BFTT19] give an asymptotically tight bound of $1/\sqrt{n} + \sqrt{d}/(\varepsilon n)$ on the excess population loss in this setting using noisy gradient descent. Under mild smoothness assumptions, Feldman et al. [FKT20] develop algorithms that achieve the optimal excess population loss using n gradient computations.

In contrast, existing results for private optimization in ℓ_1-geometry do not achieve the optimal rates for the excess population loss [KST12; JT14; TTZ15]. For the empirical loss, Talwar et al. [TTZ15] develop private algorithms with $\widetilde{O}(1/(n\varepsilon)^{2/3})$ excess empirical loss for smooth functions and provide tight lower bounds when the dimension d is sufficiently high. These bounds can be converted into bounds on the excess population loss using standard techniques of uniform convergence of empirical loss to population loss, however these techniques can lead to suboptimal bounds as there are settings where uniform convergence is lower bounded by $\Omega(\sqrt{d/n})$ [Fel16]. Moreover, the algorithm of Talwar et al. [TTZ15] has runtime $O(n^{5/3})$ in the moderate privacy regime ($\varepsilon = \Theta(1)$) which is prohibitive in practice. On the other hand, Jain and Thakurta [JT14] develop algorithms for the population loss, however, their work is limited to generalized linear models and achieves a sub-optimal rate $\widetilde{O}(1/n^{1/3})$.

In this work we develop private algorithms that achieve the optimal excess population loss in ℓ_1-geometry, demonstrating that significant improvements are possible when the functions are smooth, in contrast to ℓ_2-geometry where smoothness does not lead to better bounds. Specifically, for non-smooth functions, we develop an iterative localization algorithm, based on noisy mirror descent which achieves the optimal rate $\sqrt{\log(d)/n} + \sqrt{d}/\varepsilon n$. With additional smoothness assumptions, we show that rates with logarithmic dependence on the dimension are possible using a private variance-reduced Frank-Wolfe algorithm which obtains the rate $\sqrt{\log(d)/n} + (\log(d)/\varepsilon n)^{2/3}$ and runs in linear (in n) time. This shows that privacy is essentially free in this setting even when $d \gg n$ and ε is as small as $n^{-1/4}$. Finally, we show that similar rates are possible for general ℓ_p-geometries for non-smooth functions when $1 \le p \le 2$. Moreover, our algorithms query at most $O(n^{3/2})$ gradients which improves over the best known algorithms for the non-smooth case in ℓ_2-geometry which require n^2 gradients [FKT20].

The following two theorems summarize our upper bounds.

Theorem 1 (non-smooth functions)**.** *Let $\mathcal{X} \subset \mathbb{R}^d$ be a convex body with ℓ_1 diameter less than 1. Let $f(\cdot; z)$ be convex, Lipschitz with respect to $\|\cdot\|_1$ for any $z \in \mathcal{Z}$. There is an (ε, δ)-DP algorithm that takes a dataset $\mathcal{S} \in \mathcal{Z}^n$, queries at most $O(\log n \cdot \min(n^{3/2}\sqrt{\log d}, n^2\varepsilon/\sqrt{d}))$ and outputs a solution $\hat{x}$ that has*

$$\mathbb{E}[F(\hat{x})] \le \min_{x \in \mathcal{X}} F(x) + \widetilde{O}\left(\sqrt{\frac{\log d}{n}} + \frac{\sqrt{d}\log^{3/2} d}{n\varepsilon}\right),$$

where the expectation is over the random choice of $\mathcal{S}$ and the randomness of the algorithm.

Theorem 2 (smooth functions)**.** *Let $\mathcal{X} = \{x \in \mathbb{R}^d : \|x\|_1 \le 1\}$ be the ℓ_1-ball. Let $f(\cdot; z)$ be convex, Lipschitz and smooth with respect to $\|\cdot\|_1$ for any $z \in \mathbb{Z}$. There is an (ε, δ)-DP linear time algorithm that takes a dataset $\mathcal{S} \in \mathcal{Z}^n$ and outputs a solution $\hat{x}$ that has*

$$\mathbb{E}[F(x_K)] \le \min_{x \in \mathcal{X}} F(x) + \widetilde{O}\left(\sqrt{\frac{\log d}{n}} + \left(\frac{\log d}{n\varepsilon}\right)^{2/3}\right),$$

where the expectation is over the random choice of $\mathcal{S}$ and the randomness of the algorithm.

Before proceeding to review our algorithmic techniques, we briefly explain why the approaches used to obtain optimal rates in ℓ_2-geometry [BFTT19; FKT20] do not work in our setting. One of the most natural approaches to proving bounds for private stochastic optimization is to use the generalization properties of differential privacy to derive population loss bounds for a private ERM algorithm. This approach fails to give asymptotically optimal bounds for the ℓ_2 case [BST14], and similarly gives suboptimal bounds for the ℓ_1 case. Broadly, there are two approaches that have been used to get optimal bounds in the ℓ_2 case. An approach due to Bassily et al. [BFTT19] uses stability of SGD on sufficiently smooth losses [HRS16] to get population loss bounds. These stability results rely on contractivity of gradient descent steps. However, as we show in an example that appears in Appendix A, the versions of mirror descent that are relevant to our setting do not have this property. Feldman et al. [FKT20] derive generalization properties of their one pass algorithms from online-to-batch conversion. However, their analysis still relies on contractivity to prove the privacy guarantees of their algorithm. For their iterative localization approach Feldman et al. [FKT20] use stability of the optimal solution to ERM in a different way to determine the scale of the noise added in each phase of the algorithm. In ℓ_1 geometry the norm of the noise added via this approach would overwhelm the signal (we discuss this in detail below).

We overview the key techniques we use to overcome these challenges below.

Mirror descent based Iterative Localization. In the non-smooth setting, we build on the iterative localization framework of Feldman et al. [FKT20]. In this framework in each phase a non-private optimization algorithm is used to solve a regularized version of the optimization problem. Regularization ensures that the output solution has small sensitivity and thus addition of Gaussian noise guarantees privacy. By appropriately choosing the noise and regularization scales, each phase reduces the distance to an approximate minimizer by a multiplicative factor. Thus after a logarithmic number of phases, the current iterate has the desired guarantees. Unfortunately, addition of Gaussian noise (and other output perturbation techniques) results in sub-optimal bounds in ℓ_1-geometry since the ℓ_1-error due to noise grows linearly with d. In contrast, the ℓ_2-error grows as $\sqrt{d}$.

Instead of using output perturbation, we propose to use a private optimization algorithm in each phase. Using stability properties of strongly convex functions, we show that if the output of the private algorithm has sufficiently small empirical excess loss, then it has to be close to an approximate minimizer. Specifically, we reduce the distance to a minimizer by a multiplicative factor (relative to the initial conditions at that phase). We show that a private version of mirror descent for strongly convex empirical risk minimization achieves sufficiently small excess empirical loss giving us an algorithm that achieves the optimal rate for non-smooth loss functions. More generally, this technique reduces the problem of DP-SCO to the problem of DP-ERM with strongly convex objectives. We provide details and analysis of this approach in Section 3.

Dyadic variance-reduced Frank-Wolfe. Our second algorithm is based on recent progress in stochastic optimization. Yurtsever et al. [YSC19] developed (non-private) variance-reduced Frank-Wolfe algorithm that achieves the optimal $\widetilde{O}(1/\sqrt{n})$ excess population loss improving on the standard implementations of Frank-Wolfe that achieve excess population loss of $\widetilde{O}(1/n^{1/3})$. The improvement relies on a novel variance reduction techniques that uses previous samples to improve the gradient estimates at future iterates [FLLZ18]. This frequent reuse of samples is the main challenge in developing a private version of the algorithm.

Inspired by the binary tree technique in the privacy literature [DNPR10; DNRR15], we develop a new binary-tree-based variance reduction technique for the Frank-Wolfe algorithm. At a high level, the algorithm constructs a binary tree and allocates a set of samples to each vertex. The gradient at each vertex is then estimated using the samples of that vertex and the gradients along the path to the root. We assign more samples (larger batch sizes) to vertices that are closer to the root, to account for the fact that they are reused in more steps of the algorithm. This ensures that the privacy budget of samples in any vertex is not exceeded.

Using this privacy-aware design of variance-reduction, we rely on two tools to develop and analyze our algorithm. First, similarly to the private Frank-Wolfe for ERM [TTZ15], we use the exponential mechanism to privatize the updates. A Frank-Wolfe update chooses one of the vertices of the constraint set ($2d$ possibilities including signs for ℓ_1-balls) and therefore the application of the exponential mechanism leads to a logarithmic dependence on the dimension d. This tool together with the careful accounting of privacy losses across the nodes, suffices to get the optimal bounds for the pure ε-DP case ($\delta = 0$). To get the optimal rates for (ε, δ)-DP, we rely on recent amplification by shuffling result for private local randomizers [FMT20]. To amplify privacy, we view our algorithm as a sequence of local randomizers, each operating on a different subset of the tree. Section 4 contains details of this algorithm.

In independent and concurrent work, Bassily et al. [BGN21] study differentially private algorithms for stochastic optimization in ℓ_p-geometry. Similarly to our work, they build on mirror descent and variance-reduced Frank-Wolfe algorithms to design private procedures for DP-SCO albeit without the iterative localization scheme and the binary-tree-based sample allocation technique we propose. As a result, their algorithms achieve sub-optimal rates in some of the parameter regimes: in ℓ_1-geometry, they achieve excess loss of roughly $\log(d)/\varepsilon\sqrt{n}$ in contrast to the $\sqrt{\log(d)}/\sqrt{n} + \log(d)/(\varepsilon n)^{2/3}$ rate of our algorithms. For $1 < p < 2$, their algorithms have excess loss of (up to log factors) $\min(d^{1/4}/\sqrt{n}, \sqrt{d}/(\varepsilon n^{3/4}))$, whereas our algorithms achieve the rate of $\sqrt{d}/\varepsilon n$. On the other hand, Bassily et al. [BGN21] develop a generalized Gaussian mechanism for adding noise in ℓ_p-geometry. Their mechanism improves over the standard Gaussian mechanism and can improve the rates of our algorithms for ℓ_p-geometry (Theorem 5) by a $\sqrt{\log d}$ factor. Moreover, they prove a lower bound for ℓ_p-geometries with $1 < p < 2$ that establishes the optimality of our upper bounds for $1 < p < 2$.

2 Preliminaries

2.1 Stochastic Convex Optimization

We let $\mathcal{S} = (z_1, \ldots, z_n)$ denote datasets where $z_i \in \mathcal{Z}$ are drawn i.i.d. from a distribution P over the domain $\mathcal{Z}$. Let $\mathcal{X} \subseteq \mathbb{R}^d$ be a convex set that denotes the set of parameter for the optimization problem. Given a loss function $f(x; z) : \mathcal{X} \times \mathcal{Z} \to \mathbb{R}$ that is convex in x (for every z), we define the population loss $F(x) = \mathbb{E}_{z \sim P}[f(x; z)]$. The excess population loss of a parameter $x \in \mathcal{X}$ is then $F(x) - \min_{y \in \mathcal{X}} F(y)$. We also consider the empirical loss $\hat{F}(x; S) = \frac{1}{n} \sum_{i=1}^{n} f(x; z_i)$ and the excess empirical loss of $x \in \mathcal{X}$ is $\hat{F}(x; S) - \min_{y \in \mathcal{X}} \hat{F}(y; S)$. For a set $\mathcal{X}$, we will denote its ℓ_p diameter by $\mathsf{diam}_p(\mathcal{X}) = \sup_{x,y \in \mathcal{X}} \|x - y\|_p$.

As we are interested in general geometries, we define the standard properties (e.g., Lipschitz, smooth and strongly convex) with respect to a general norm which are frequently used in the optimization literature [Duc18].

Definition 2.1 (Lipschitz continuity)**.** *A function $f : \mathcal{X} \to \mathbb{R}$ is L-Lipschitz with respect to a norm $\|\cdot\|$ over $\mathcal{X}$ if for every $x, y \in \mathcal{X}$ we have $|f(x) - f(y)| \le L \|x - y\|$.*

A standard result is that L-Lipschitz continuity is equivalent to bounded (sub)-gradients, namely that $\|g\|_* \le L$ for all $x \in \mathcal{X}$ and sub-gradient $g \in \partial f(x)$ where $\|\cdot\|_*$ is the dual norm of $\|\cdot\|$.

Definition 2.2 (smoothness)**.** *A function $f : \mathcal{X} \to \mathbb{R}$ is β-smooth with respect to a norm $\|\cdot\|$ over $\mathcal{X}$ if for every $x, y \in \mathcal{X}$ we have $\|\nabla f(x) - \nabla f(y)\|_* \le \beta \|x - y\|$.*

Definition 2.3 (strong convexity)**.** *A function $f : \mathcal{X} \to \mathbb{R}$ is λ-strongly convex with respect to a norm $\|\cdot\|$ over $\mathcal{X}$ if for any $x, y \in \mathcal{X}$ we have $f(x) + \langle \nabla f(x), y - x\rangle + \frac{\lambda}{2}\|y - x\|^2 \le f(y)$.*

Since we develop private versions of mirror descent, we define the Bregman divergence associated with a differentiable convex function $h : \mathcal{X} \to \mathbb{R}$ to be $D_{\mathrm{h}}(x, y) = h(x) - h(y) - \langle \nabla h(y), x - y\rangle$. We require a definition of strong convexity relative to a function which has been used in several works in the optimization literature [DSSST10; LFN18].

Definition 2.4 (relative strong convexity)**.** *A function $f : \mathcal{X} \to \mathbb{R}$ is λ-strongly convex relative to $h : \mathcal{X} \to \mathbb{R}$ if for any $x, y \in \mathcal{X}$, $f(x) + \langle \nabla f(x), y - x\rangle + \lambda D_{\mathrm{h}}(y, x) \le f(y)$.*

Note that if $h(x)$ is convex, then $h(x)$ is 1-strongly convex relative to $h(x)$ according to this definition. Moreover, the function $f(x) = g(x) + h(x)$ is also 1-strongly convex relative to $h(x)$ for any convex function $g(x)$.

2.2 Differential Privacy

We recall the definition of (ε, δ)-differential privacy.

Definition 2.5 ([DMNS06; DKMMN06])**.** *A randomized algorithm $\mathcal{A}$ is (ε, δ)-differentially private ((ε, δ)-DP) if, for all datasets $\mathcal{S}, \mathcal{S}' \in \mathcal{Z}^n$ that differ in a single data element and for all events $\mathcal{O}$ in the output space of $\mathcal{A}$, we have*

$$\Pr[\mathcal{A}(\mathcal{S}) \in \mathcal{O}] \le e^{\varepsilon} \Pr[\mathcal{A}(\mathcal{S}') \in \mathcal{O}] + \delta.$$

To simplify notation, we sometimes use the notion of (ε, δ)-indistinguishability; two random variables X and Y are (ε, δ)-indistinguishable, denoted $X \approx_{(\varepsilon,\delta)} Y$, if for every $\mathcal{O}$, $\Pr(X \in \mathcal{O}) \le e^{\varepsilon} \Pr[Y \in \mathcal{O}] + \delta$ and $\Pr(Y \in \mathcal{O}) \le e^{\varepsilon} \Pr[X \in \mathcal{O}] + \delta$. When $\delta = 0$, we use the shorter notation ε-DP. We also use the following privacy composition results.

Lemma 2.1 (Basic composition [DR14])**.** *If $\mathcal{A}_1, \dots, A_k$ are randomized algorithms that each is ε-DP, then their composition $(\mathcal{A}_1(\mathcal{S}), \dots, A_k(\mathcal{S}))$ is $k\varepsilon$-DP.*

Lemma 2.2 (Advanced composition [DR14])**.** *If $\mathcal{A}_1, \dots, A_k$ are randomized algorithms that each is (ε, δ)-DP, then their composition $(\mathcal{A}_1(\mathcal{S}), \dots, A_k(\mathcal{S}))$ is $(\sqrt{2k \log(1/\delta')}\varepsilon + k\varepsilon(e^{\varepsilon} - 1), \delta' + k\delta)$-DP.*

3 Algorithms for Non-Smooth Functions

In this section, we develop an algorithm that builds on the iterative localization techniques of Feldman et al. [FKT20] to achieve optimal excess population loss for non-smooth functions over the ℓ_1-ball. Instead of using output perturbation to solve the regularized optimization problems, our algorithm uses general private algorithms for solving strongly convex ERM problems. This essentially reduces the problem of privately minimizing the population loss to that of privately minimizing a strongly convex empirical risk. In Section 3.1 we develop private versions of mirror descent that achieve optimal bounds for strongly convex ERM problems, and in Section 3.2 we use these algorithms in an iterative localization framework to obtain optimal bounds for the population loss.

Algorithm 1 Noisy Mirror Descent

```
Require: Dataset $\mathcal{S} = (z_1, \dots, z_n) \in \mathcal{Z}^n$, convex set $\mathcal{X}$, convex function $h : \mathcal{X} \to \mathbb{R}$, step sizes $\{\eta_k\}_{k=1}^T$, batch size $b$, initial point $x_0$, number of iterations $T$;
for $k = 1$ to $T$ do
    Sample $S_1, \dots, S_b \sim \mathrm{Unif}(\mathcal{S})$
    Set $\hat{g}_k = \frac{1}{b} \sum_{i=1}^b \nabla f(x_k; S_i) + \zeta_i$ where $\zeta_i \sim \mathsf{N}(0, \sigma^2 I_d)$ with $\sigma = 100L\sqrt{d \log(1/\delta)}/b\varepsilon$
    Find $x_{k+1} \coloneqq \mathrm{argmin}_{x \in \mathcal{X}} \{\langle \hat{g}_k, x - x_k \rangle + \frac{1}{\eta_k} D_{\mathrm{h}}(x, x_k)\}$
return $\bar{x}_T = \frac{1}{T} \sum_{k=1}^T x_k$ (convex)
return $\hat{x}_T = \frac{2}{T(T+1)} \sum_{k=1}^T k x_k$ (strongly convex)
```

3.1 Private Algorithms for Strongly Convex ERM

In this section, we consider empirical risk minimization for strongly convex functions and achieve optimal excess empirical loss using noisy mirror descent (Algorithm 1).

Theorem 3. *Let $h : \mathcal{X} \to \mathbb{R}$ be 1-strongly convex with respect to $\|\cdot\|_1$, $x^\star = \mathrm{argmin}_{x \in \mathcal{X}} \hat{F}(x; S)$, and assume $D_{\mathrm{h}}(x^\star, x_0) \le D^2$. Let $f(x; z)$ be convex and L-Lipschitz with respect to $\|\cdot\|_1$ for all $z \in \mathcal{Z}$. Setting $1 \le b$, $T = \frac{n^2}{b^2}$ and $\eta_k = \frac{D}{\sqrt{T}} \frac{1}{\sqrt{L^2 + 2\sigma^2 \log d}}$, Algorithm 1 is (ε, δ)-DP and*

$$\mathbb{E}[\hat{F}(\bar{x}_T; S) - \hat{F}(x^\star; S)] \le LD \cdot O\left(\frac{b}{n} + \frac{\sqrt{d \log d \log \frac{1}{\delta}}}{n\varepsilon}\right).$$

Moreover, if $f(x; z)$ is λ-strongly convex relative to $h(x)$, then setting $\eta_k = \frac{2}{\lambda(k+1)}$

$$\mathbb{E}[\hat{F}(\hat{x}_T; S) - \hat{F}(x^\star; S)] \le O\left(\frac{L^2 b^2}{\lambda n^2} + \frac{L^2 d \log d \log \frac{1}{\delta}}{\lambda n^2 \varepsilon^2}\right).$$

To prove Theorem 3, we need the following standard results for the convergence of stochastic mirror descent for convex and strongly convex functions.

Lemma 3.1 ([Duc18], Corollary 4.2.11)**.** *Assume $h(x)$ is 1-strongly convex with respect to $\|\cdot\|_1$. Let $f(x)$ be a convex function and $x^\star = \mathrm{argmin}_{x \in \mathcal{X}} f(x)$. Consider the stochastic mirror descent update $x_{k+1} = \mathrm{argmin}_{x \in \mathcal{X}} \{\langle g_k, x - x_k \rangle + \frac{1}{\eta_k} D_{\mathrm{h}}(x, x_k)\}$ where $\mathbb{E}[g_k] \in \partial f(x_k)$ with $\mathbb{E}\left[\|g_k\|_\infty^2\right] \le L^2$. If $\eta_k = \eta$ for all k then the average iterate $\bar{x}_T = \frac{1}{T} \sum_{i=1}^T x_i$ has $\mathbb{E}[f(\bar{x}_T) - f(x^\star)] \le \frac{D_{\mathrm{h}}(x^\star, x_1)}{T\eta} + \frac{\eta L^2}{2}$.*

We also need the following result which states the rates of stochastic mirror descent for strongly convex functions. Similar results appear in the optimization literature [LJSB12], though as the statement we require is less common, we provide a proof in Appendix C.1.

Lemma 3.2. *Under the same notation of Lemma 3.1, if $f(x)$ is λ-strongly convex relative to $h(x)$, then setting $\eta_k = \frac{2}{\lambda(k+1)}$ the weighted average $\hat{x}_T = \frac{2}{T(T+1)} \sum_{k=1}^T k x_k$ has $\mathbb{E}[f(\hat{x}_T) - f(x^\star)] \le \frac{L^2}{\lambda(T+1)}$.*

We are now ready to prove Theorem 3.

Proof. The privacy proof follows directly using Moments accountant, that is, Theorem 1 in [ACG-MMTZ16], by noting the the ℓ_2-norm of the gradients is bounded by $\|\nabla f(x; z_i)\|_2 \le \|\nabla f(x; z_i)\|_\infty \sqrt{d} \le$

$L\sqrt{d}$ for all $x \in \mathcal{X}$ and $z \in \mathcal{Z}$. Now we prove the utility of the algorithm. To this end, we have that $\mathbb{E}[\|\hat{g}_k\|_\infty^2] \le 2L^2 + 2\mathbb{E}[\|\zeta_k\|_\infty^2] \le 2L^2 + 4\sigma^2 \log d$. Lemma 3.1 now implies that

$$\begin{aligned}\mathbb{E}[\hat{F}(\bar{x}_T; S) - \hat{F}(x^\star; S)] &\le \frac{D^2}{T\eta} + \eta L^2 + 2\eta\sigma^2 \log d \\ &\le 2D\sqrt{(L^2 + 2\sigma^2 \log d)/T} \\ &\le LD \cdot O\left(\frac{b}{n} + \frac{\sqrt{d \log d \log \frac{1}{\delta}}}{n\varepsilon}\right),\end{aligned}$$

where the second inequality follows from the choice of η. For the second part, Lemma 3.2 implies that

$$\mathbb{E}[\hat{F}(\hat{x}_T; S) - \hat{F}(x^\star; S)] \le \frac{L^2}{\lambda} O\left(\frac{b^2}{n^2} + \frac{d \log d \log \frac{1}{\delta}}{n^2\varepsilon^2}\right). \qquad \square$$

3.2 Private Algorithms for SCO

Building on the noisy mirror descent algorithm of Section 3.1, in this section we develop a localization based algorithm for the population loss that achieves the optimal bounds in ℓ_1 geometry. The algorithm iteratively solves a regularized version of the (empirical) objective function using noisy mirror decent (Algorithm 1). We present the full details in Algorithm 2 which enjoys the following guarantees.

Algorithm 2 Localized Noisy Mirror Descent

```
Require: Dataset $\mathcal{S} = (z_1, \dots, z_n) \in \mathcal{Z}^n$, constraint set $\mathcal{X}$, step size $\eta$, initial point $x_0$;
Set $k = \lceil \log n \rceil$, $p = 1 + 1/\log d$
for $i = 1$ to $k$ do
    Set $\varepsilon_i = 2^{-i}\varepsilon$, $n_i = 2^{-i}n$, $\eta_i = 2^{-4i}\eta$
    Apply Algorithm 1 with $(\varepsilon_i, \delta)$-DP, batch size $b_i = \max(\sqrt{n_i/\log d}, \sqrt{d/\varepsilon_i})$, $T = n_i^2/b_i^2$ and $h_i(x) = \frac{1}{p-1}\|x - x_{i-1}\|_p^2$ for solving the ERM over $\mathcal{X}_i = \{x \in \mathcal{X} : \|x - x_{i-1}\|_p \le 2L\eta_i n_i(p-1)\}$:
        $F_i(x) = \frac{1}{n_i}\sum_{j=1}^{n_i} f(x; z_j) + \frac{1}{\eta_i n_i (p-1)}\|x - x_{i-1}\|_p^2$
    Let $x_i$ be the output of the private algorithm
return the final iterate $x_k$
```

Theorem 4. *Assume* $\mathsf{diam}_1(\mathcal{X}) \le D$ *and* $f(x; z)$ *is convex and* L*-Lipschitz with respect to* $\|\cdot\|_1$ *for all* $z \in \mathcal{Z}$. *If we set*

$$\eta = \frac{D}{L}\min\left\{\sqrt{\log(d)/n}, \varepsilon/\sqrt{d \log d \log \tfrac{1}{\delta}}\right\},$$

then Algorithm 2 uses $O(\log n \cdot \min(n^{3/2}\sqrt{\log d}, n^2\varepsilon/\sqrt{d}))$ *gradients and its output has*

$$\mathbb{E}[F(x_k) - F(x^\star)] = LD \cdot O\left(\frac{\sqrt{\log d}}{\sqrt{n}} + \frac{\sqrt{d \log^3 d \log \frac{1}{\delta}}}{n\varepsilon}\right).$$

We begin with the following lemma which bounds the distance of the private minimizer to the true minimizer at each iteration.

Lemma 3.3. *Let* $\hat{x}_i = \mathrm{argmin}_{x\in\mathcal{X}} F_i(x)$*. Then ,*

$$\mathbb{E}[\|x_i - \hat{x}_i\|_p^2] \le O\left(\frac{L^2\eta_i^2 n_i}{\log d} + L^2\eta_i^2 d\log d\log(1/\delta)/\varepsilon_i^2\right).$$

Proof. First, we prove that $\hat{x}_i \in \mathcal{X}_i$. The definition of $\hat{x}_i$ implies that

$$\frac{1}{n_i}\sum_{j=1}^{n_i} f(\hat{x}_i; z_j) + \frac{1}{\eta_i n_i(p-1)}\left\|\hat{x}_i - x_{i-1}\right\|_p^2 \le \frac{1}{n_i}\sum_{j=1}^{n_i} f(x_{i-1}; z_j).$$

Since $f(x;z)$ is L-Lipschitz, we get

$$\frac{1}{\eta_i n_i(p-1)}\left\|\hat{x}_i - x_{i-1}\right\|_p^2 \le L\left\|\hat{x}_i - x_{i-1}\right\|_1 \le 2L\left\|\hat{x}_i - x_{i-1}\right\|_p$$

where the last inequality follows from the choice of p (since $\|z\|_1 \le d^{1-1/p}\|z\|_p \le 2\|z\|_p$ for all $z \in \mathbb{R}^d$), hence we get $\|\hat{x}_i - x_{i-1}\|_p \le \frac{2L\eta_i n_i}{\log d}$. Thus, we have that $\hat{x}_i \in \mathcal{X}_i = \{x : \|x - x_{i-1}\|_p \le \frac{2L\eta_i n_i}{\log d}\}$.

Now, note that the function $F_i(x)$ is λ_i-strongly convex relative to $h_i(x) = \frac{1}{p-1}\|x - x_{i-1}\|_p^2$ where $\lambda_i = \frac{1}{\eta_i n_i}$. Moreover, the function $r_i(x) = \frac{1}{\eta_i n_i(p-1)}\|x - x_{i-1}\|_p^2$ is $4L$-Lipschitz with respect to $\|\cdot\|_1$ for $x \in \mathcal{X}_i$. Therefore using the bounds of Theorem 3 for noisy mirror descent and observing that $F_i(x)$ is λ_i-strongly convex with respect to $\|\cdot\|_p$,

$$\frac{\lambda_i}{2}\mathbb{E}[\|x_i - \hat{x}_i\|_p^2] \le \mathbb{E}[F_i(x_i) - F_i(\hat{x}_i)] \le O\left(\frac{L^2}{\lambda_i n_i\log d} + \frac{L^2 d\log d\log^2(1/\delta)}{n_i^2\varepsilon_i^2\lambda_i}\right),$$

implying that $\mathbb{E}[\|x_i - \hat{x}_i\|_p^2] \le O\left(\frac{L^2\eta_i^2 n_i}{\log d} + \frac{L^2\eta_i^2 d\log d\log^2(1/\delta)}{\varepsilon_i^2}\right)$. □

The next lemma follows from Shalev-Shwartz et al. [SSSSS09].

Lemma 3.4. *Let* $\hat{x}_i = \mathrm{argmin}_{x\in\mathcal{X}_i} F_i(x)$ *and* $y \in \mathcal{X}$*. If* $f(x;z)$ *is* L*-Lipschitz with respect to* $\|\cdot\|_1$*, then* $\mathbb{E}[F(\hat{x}_i)] - F(y) \le \frac{\mathbb{E}[\|y - x_{i-1}\|_p^2]}{\eta_i n_i(p-1)} + O(L^2\eta_i)$*.*

Proof. The proof follows from Theorems 6 and 7 in [SSSSS09] by noting that the function $r(x;z_j) = f(x;z_j) + \frac{1}{\eta_i n_i(p-1)}\|x - x_{i-1}\|_p^2$ is $\frac{1}{\eta_i n_i}$-strongly convex and $O(L)$-Lipschitz with respect to $\|\cdot\|_1$ over $\mathcal{X}_i$. □

We are now ready to prove Theorem 4.

Proof. First, we prove the claim about runtime and number of queried gradients. Algorithm 1 requires n_i^2/b_i gradients (same runtime) hence since $b_i = \max(\sqrt{n_i/\log d}, \sqrt{d/\varepsilon_i})$ we get that the number of gradients at each stage is at most $\min(n^{3/2}\sqrt{\log d}, n^2\varepsilon/\sqrt{d})$, implying the claim as we have $\log n$ iterates. Next, we prove utility which is similar to the proof of Theorem 4.4 in [FKT20]. Letting $\hat{x}_0 = x^\star$, we have:

$$\mathbb{E}[F(x_k)] - F(x^\star) = \sum_{i=1}^{k}\mathbb{E}[F(\hat{x}_i) - F(\hat{x}_{i-1})] + \mathbb{E}[F(x_k) - F(\hat{x}_k)].$$

First, note that Lemma 3.3 implies

$$\begin{aligned}
\mathbb{E}[F(x_k) - F(\hat{x}_k)] &\le L\mathbb{E}[\|x_k - \hat{x}_k\|_1] \\
&\le L\sqrt{\mathbb{E}[2\left\|x_k - \hat{x}_k\right\|_p^2]} \\
&\le CL^2\eta_k(\sqrt{n_i/\log d} + \sqrt{d\log d\log(1/\delta)}/\varepsilon_k) \\
&\le C2^{-2k}L^2\eta(\sqrt{n/\log d} + \sqrt{d\log d\log(1/\delta)}/\varepsilon) \le CLD/n^2,
\end{aligned}$$

where the last inequality follows since $\eta \le \frac{D}{L}\min(\sqrt{\log(d)/n}, \varepsilon/\sqrt{d\log d\log(1/\delta)})$. Lemmas 3.4 and 3.3 imply

$$\begin{aligned}
\sum_{i=1}^{k}\mathbb{E}[F(\hat{x}_i) - F(\hat{x}_{i-1})] &\le \sum_{i=1}^{k}\frac{\mathbb{E}[\|\hat{x}_{i-1} - x_{i-1}\|_p^2]}{\eta_i n_i(p-1)} + CL^2\eta_i \\
&\le \frac{D^2}{\eta n(p-1)} + \sum_{i=2}^{k}C(L^2\eta_i + \frac{L^2\eta_i d\log d\log(1/\delta)}{n_i\varepsilon_i^2(p-1)}) + C\sum_{i=1}^{k}\frac{L^2\eta}{2^i} \\
&\le \frac{D^2}{\eta n(p-1)} + CL^2\eta + C\sum_{i=2}^{k}2^{-i}\frac{L^2\eta d\log d\log(1/\delta)}{n\varepsilon^2(p-1)} + 2CL^2\eta \\
&\le \frac{D^2}{\eta n(p-1)} + 2C\frac{L^2\eta d\log d\log(1/\delta)}{n\varepsilon^2(p-1)} + 3CL^2\eta.
\end{aligned}$$

The claim now follows by setting the value of η. □

Finally, we can extend Algorithm 2 to work for general ℓ_p geometries for $1 < p \le 2$, resulting in the following theorem. We defer full details to Appendix B.

Theorem 5. *Let $1 < p \le 2$. Assume $\mathsf{diam}_p(\mathcal{X}) \le D$ and $f(x;z)$ is convex and L-Lipschitz with respect to $\|\cdot\|_p$ for all $z \in \mathcal{Z}$. Then there is an (ε,δ)-DP algorithm that uses $O(\log n \cdot \min(n^{3/2}\sqrt{\log d}, n^2\varepsilon/\sqrt{d}))$ and outputs $\hat{x}$ such that*

$$\mathbb{E}[F(\hat{x}) - F(x^\star)] = LD \cdot O\left(\frac{1}{\sqrt{(p-1)n}} + \frac{\sqrt{d\log d\log\frac{1}{\delta}}}{(p-1)n\varepsilon}\right).$$

If $p = 2$ then the output $\hat{x}$ has

$$\mathbb{E}[F(\hat{x}) - F(x^\star)] = LD \cdot O\left(\frac{1}{\sqrt{n}} + \frac{\sqrt{d\log\frac{1}{\delta}}}{n\varepsilon}\right).$$

4 Efficient Algorithms for Smooth Functions

Having established tight bounds for the non-smooth case, in this section we turn to the smooth setting and develop linear-time private Frank-Wolfe algorithms with variance-reduction that achieve the optimal rates. Specifically, our algorithms achieve the rate $\widetilde{O}(1/\sqrt{n\varepsilon})$ for pure ε-DP and $\widetilde{O}\left(1/\sqrt{n} + 1/(n\varepsilon)^{2/3}\right)$ for (ε,δ)-DP. These results imply that the optimal (non-private) statistical rate $\widetilde{O}(1/\sqrt{n})$ is achievable with strong privacy guarantees—whenever $\varepsilon \ge \widetilde{\Omega}(1/n^{1/4})$ for (ε,δ)-DP—even for high dimensional functions with $d \gg n$.

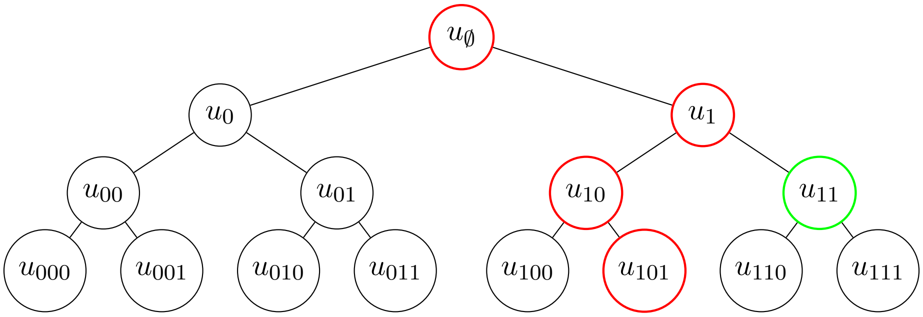

Figure 1. Binary tree at phase $t = 3$ of the algorithm. At the leaf u_{101}, the algorithm has the gradient estimate $v_{t,101}$ which is calculated along the path to the root where every right son applies a correction step to the estimate. Using the gradient estimate $v_{t,101}$, the algorithm applies a Frank-Wolfe step to calculate the next iterate and put its value in the next DFS vertex, namely $u_{t,11}$.

The starting point of our algorithms is the recent non-private Frank-Wolfe algorithm of Yurtsever et al. [YSC19] which uses variance-reduction techniques to achieve the (non-private) optimal rates. Due to the high reuse of samples, a direct approach to privatizing their algorithm would results in sub-optimal bounds. To overcome this, we design a new binary-tree scheme for variance reduction that allows for more noise-efficient private algorithms.

We describe our private Frank-Wolfe procedure in Algorithm 3. We present the algorithm in a more general setting where $\mathcal{X}$ can be an arbitrary convex body with m vertices. The algorithm has T phases (outer iterations) indexed by $1 \le t \le T$ and each phase t has a binary tree of depth t. We will denote vertices by u_s where $s \in \{0,1\}^{\le t}$ is the path to the vertex; i.e., $u_\emptyset$ will denote the root of the tree, u_{01} will denote the right child of u_0. Each vertex u_s is associated with a parameter $x_{t,s}$, a gradient estimate $v_{t,s}$, and a set of samples $S_{t,s}$ of size $2^{-j}b$ where j is the depth of the vertex. Roughly, the idea is to improve the gradient estimate at a vertex (reduce the variance) using the gradient estimates at vertices along the path to the root; since these vertices have more samples, this procedure can result in better gradient estimates.

More precisely, the algorithm traverses through the graph vertices according to the Depth-First-Search (DFS) approach. At each vertex, the algorithm improves the gradient estimate at the current vertex using the estimate at the parent vertex. When the algorithm visits a leaf vertex, it also updates the current iterate using the Frank-Wolfe step with the gradient estimate at the leaf.

For notational convenience, we let $\mathsf{DFS}(t)$ denote the DFS order of the vertices in a binary tree of depth t (root not included), i.e., for $t = 2$ we have $\mathsf{DFS}(t) = \{u_0, u_{00}, u_{01}, u_1, u_{10}, u_{11}\}$. Moreover, for $s \in \{0,1\}^t$ we let $\ell(s)$ denote the integer whose binary representation is s. In the description of the algorithm, we denote iterates by $x_{t,s}$ where t is the phase and $s \in \{0,1\}^t$ is the path from the root. In our proofs, we sometimes use the equivalent notation x_k where $k = 2^{t-1} + \ell(s)$.

We analyze Algorithm 3 for pure and approximate DP.

4.1 Pure Differential Privacy

The following theorem summarizes our guarantees for pure privacy.

Theorem 6. *Assume that $\mathsf{diam}_1(\mathcal{X}) \le D$, $m \le O(d)$ and that $f(x;z)$ is convex, L-Lipschitz and β-smooth with respect to $\|\cdot\|_1$. Assume also that $\frac{L \log m \log^2 n}{n\varepsilon D} \le \beta \le \frac{nL \log m}{\varepsilon D \log^2 n}$. Setting $b = n/\log^2 n$,*

Algorithm 3 Private Variance Reduced Frank-Wolfe

```
Require: Dataset $\mathcal{S} = (z_1, \dots, z_n) \in \mathcal{Z}^n$, constraint set $\mathcal{X} = \mathsf{conv}\{c_1, \dots, c_m\}$, number of phases $T$, batch size $b$, initial point $x_0$;
for $t = 1$ to $T$ do
    Set $x_{t,\emptyset} = x_{t-1,L_{t-1}}$
    Draw $b$ samples to the set $S_{t,\emptyset}$
    $v_{t,\emptyset} \leftarrow \nabla f(x_{t,\emptyset}; S_{t,\emptyset})$
    for $u_s \in \mathsf{DFS}[2^t]$ do
        Let $s = s'c$ where $c \in \{0,1\}$ and $j = |s|$
        if $c = 0$ then
            $v_{t,s} \leftarrow v_{t,s'}$;  $x_{t,s} \leftarrow x_{t,s'}$
        else
            Draw $2^{-j}b$ samples to the set $S_{t,s}$
            $v_{t,s} \leftarrow v_{t,s'} + \nabla f(x_{t,s}; S_{t,s}) - \nabla f(x_{t,s'}; S_{t,s})$
        if $j = t$ then
            Let $s_+$ be the next vertex in the DFS iteration
            $w_{t,s} \leftarrow \mathrm{argmin}_{c_i : 1 \le i \le m} \langle c_i, v_{t,s} \rangle + \zeta_i$ where $\zeta_i \sim \mathsf{Laplace}(\lambda_{t,s})$
            $x_{t,s_+} \leftarrow (1 - \eta_{t,s}) x_{t,s} + \eta_{t,s} w_{t,s}$ where $\eta_{t,s} = \frac{2}{2^{t-1} + \ell(s) + 1}$
return the final iterate $x_K$
```

$\lambda_{t,s} = \frac{2LD2^t}{b\varepsilon}$ and $T = \frac{1}{2}\log\left(\frac{b\varepsilon\beta D}{L\log m}\right)$, Algorithm 3 is ε-DP, queries n gradients, and has

$$\mathbb{E}[F(x_K) - F(x^\star)] \le O\left(D(L + \beta D)\frac{\sqrt{\log d}\log n}{\sqrt{n}} + \frac{\sqrt{\beta L D^3 \log d}\log n}{\sqrt{n\varepsilon}}\right).$$

Moreover, if $\beta \le \frac{L\log m \log^2 n}{n\varepsilon D}$ then setting $T = 1$ and $b = n$, Algorithm 3 is ε-DP, queries n gradients, and has

$$\mathbb{E}[F(x_K) - F(x^\star)] \le DL \cdot O\left(\frac{\sqrt{\log d}}{\sqrt{n}} + \frac{\log d}{n\varepsilon}\right) + O(\beta D^2).$$

To prove the theorem, we begin with the following lemma that gives pure privacy guarantees.

Lemma 4.1. *Assume $2^T \le b$. Setting $\lambda_{t,s} = \frac{2LD2^t}{b\varepsilon}$, Algorithm 3 is ε-DP with $\varepsilon \le 1$. Moreover, $\mathbb{E}[\langle v_{t,s}, w_{t,s}\rangle] \le \mathbb{E}[\min_{w \in \mathcal{X}} \langle v_{t,s}, w\rangle] + O(\frac{LD2^t}{b\varepsilon}\log m)$.*

Proof. The main idea for the privacy proof is that each sample in the set $S_{t,s}$ is used in the calculation of $v_{t,s}$ at most $N_{t,s} = 2^{t-|s|}$ times, hence setting the noise large enough so that each iterate is $\frac{\varepsilon}{N_{t,s}}$-DP, we get that the final mechanism is ε-DP using basic composition. Let us now provide a more formal argument. Let $\mathcal{S} = (z_1, \dots, z_{n-1}, z_n), \mathcal{S}' = (z_1, \dots, z_{n-1}, z'_n)$ be two neighboring datasets with iterates $x = (x_1, \dots, x_K)$ and $x' = (x'_1, \dots, x'_K)$, respectively. We prove that x and x' are ε-indistinguishable, i.e., $x \approx_{(\varepsilon,0)} x'$. Let $S_{t,s}$ be the set (vertex) that contains the last sample (i.e., z_n or z'_n) and let $j = |s|$ denote the depth of this vertex. We will prove privacy given that the n'th sample is in $S_{t,s}$, which will imply our general privacy guarantee as this holds for every choice of t and s.

Note that $|S_{t,s}| = 2^{-j}b$ and that this set is used in the calculation of v_k for at most 2^{t-j} (consecutive) iterates, namely these are leafs that are descendants of the vertex $u_{t,s}$. Let k_0 and k_1 be the first and last iterate such that the set $S_{t,s}$ is used for the calculation of v_k, hence $k_1 - k_0 + 1 \le 2^{t-j}$.

The iterates $(x_1, \ldots, x_{k_0-1})$ and $(x'_1, \ldots, x'_{k_0-1})$ do not depend on the last sample and therefore has the same distribution (hence 0-indistinguishable). Moreover, given that $(x_{k_0}, \ldots, x_{k_1}) \approx_{(\varepsilon,0)} (x'_{k_0}, \ldots, x'_{k_1})$, it is clear that the remaining iterates $(x_{k_1+1}, \ldots, x_K) \approx_{(\varepsilon,0)} (x'_{k_1+1}, \ldots, x'_K)$ by post-processing as they do depend on the last sample only through the previous iterates. It is therefore enough to prove that $(x_{k_0}, \ldots, x_{k_1}) \approx_{(\varepsilon,0)} (x'_{k_0}, \ldots, x'_{k_1})$. To this end, we prove that for each such iterate, $w_k \approx_{(\varepsilon/2^{t-j},0)} w'_k$, hence using post-processing and basic composition the iterates are ε-indistinguishable as $k_1 - k_0 + 1 \le 2^{t-j}$. Note that for every $k_0 \le k \le k_1$ the sensitivity $|\langle c_i, v_k - v'_k \rangle| \le \frac{DL}{2^{-j}b}$. Hence, using privacy guarantees of report noisy max [[]claim 3.9]DworkRo14, we have that $w_k \approx_{(\varepsilon/2^{t-j},0)} w'_k$ since $\lambda_{t,s} = \frac{2LD2^t}{b\varepsilon}$.

Now we prove the second part of the claim. Standard results for the expectation of the maximum of m Laplace random variables imply that $\mathbb{E}[\langle v_{t,s}, w_{t,s} \rangle] \le \min_{1 \le i \le m} \langle v_{t,s}, c_i \rangle + O(\frac{LD2^t}{b\varepsilon} \log m)$. Since $\mathcal{X} = \mathsf{conv}\{c_1, \ldots, c_m\}$, we know that for any $v \in \mathbb{R}^d$, $\operatorname{argmin}_{w \in \mathcal{X}} \langle w, v \rangle \cap \{c_1, \ldots, c_m\} \neq \emptyset$ [TTZ15](Fact 2.3) which proves the claim. □

The next lemma upper bounds the variance of the gradients.

Lemma 4.2. *At the vertex (t, s), we have*

$$\mathbb{E} \left\| v_{t,s} - \nabla F(x_{t,s}) \right\|_\infty \le (L + \beta D) \cdot O\left(\sqrt{\log(d)/b}\right).$$

The claim follows directly from the following lemma.

Lemma 4.3. *Let (s, t) be a vertex and $\sigma^2 = (L^2 + \beta^2 D^2)/b$. For every index $1 \le i \le d$,*

$$\mathbb{E}\left[e^{\lambda(v_{t,s,i} - \nabla F_i(x_{t,s}))}\right] \le e^{O(1)\lambda^2\sigma^2}.$$

Proof. (Lemma 4.2) Lemma 4.3 says that $v_{t,s,i} - \nabla F_i(x_{t,s})$ is $O(\sigma^2)$-sub-Gaussian for every $1 \le i \le d$, hence standard results imply that the maximum of d sub-Gaussian random variables is $\mathbb{E} \left\| v_{t,s} - \nabla F(x_{t,s}) \right\|_\infty \le O(\sigma)\sqrt{\log d}$. The claim follows. □

Proof. (Lemma 4.3) Let us fix i for simplicity and let $B_{t,s} = v_{t,s,i} - \nabla F_i(x_{t,s})$. We prove the claim by induction on the depth of the vertex, i.e., $j = |s|$. If $j = 0$ then $s = \emptyset$ which implies that $v_{t,\emptyset} = \nabla f(x_{t,\emptyset}; S_{t,\emptyset})$ where $S_{t,\emptyset}$ is a sample of size b. Thus we have

$$\begin{aligned}
\mathbb{E}[e^{\lambda B_{t,\emptyset}}] &= \mathbb{E}\left[e^{\lambda(v_{t,\emptyset,i} - \nabla F_i(x_{t,\emptyset}))}\right] \\
&= \mathbb{E}\left[e^{\lambda(\frac{1}{b}\sum_{s \in S_{t,\emptyset}} \nabla f_i(x_{t,\emptyset}; s) - \nabla F_i(x_{t,\emptyset}))}\right] \\
&= \prod_{s \in S_{t,\emptyset}} \mathbb{E}[e^{\frac{\lambda}{b}(\nabla f_i(x_{t,\emptyset}; s) - \nabla F_i(x_{t,\emptyset}))}] \\
&\le e^{\lambda^2 L^2/2b},
\end{aligned}$$

where the last inequality follows since for a random variable $X \in [-L, L]$ and $\mathbb{E}[X] = 0$, we have $\mathbb{E}[e^{\lambda X}] \le e^{\lambda^2 L^2/2}$ ([Duc19], example 3.6). Assume now we have s with $|s| = j > 0$ and let $s = s'c$ where $c \in \{0, 1\}$. If $c = 0$ the claim clearly holds so we assume $c = 1$. Recall that in this case $v_{t,s} = v_{t,s'} + \nabla f(x_{t,s}; S_{t,s}) - \nabla f(x_{t,s'}; S_{t,s})$, hence $B_{t,s} = v_{t,s,i} - \nabla F_i(x_{t,s}) =$

$B_{t,s'} + \nabla f_i(x_{t,s}; S_{t,s}) - \nabla f_i(x_{t,s'}; S_{t,s}) - \nabla F_i(x_{t,s}) + \nabla F_i(x_{t,s'})$ Letting $S_{<t,s} = \cup_{(t_1,s_1)<(t,s)} S_{t_1,s_1}$ be the set of all samples used up to vertex t, s, the law of iterated expectation implies

$$\begin{aligned}
\mathbb{E}[e^{\lambda B_{t,s}}] &= \mathbb{E}[e^{\lambda(B_{t,s'} + \nabla f_i(x_{t,s};S_{t,s}) - \nabla f_i(x_{t,s'};S_{t,s}) - \nabla F_i(x_{t,s}) + \nabla F_i(x_{t,s'}))}] \\
&= \mathbb{E}\left[\mathbb{E}[e^{\lambda(B_{t,s'} + \nabla f_i(x_{t,s};S_{t,s}) - \nabla f_i(x_{t,s'};S_{t,s}) - \nabla F_i(x_{t,s}) + \nabla F_i(x_{t,s'}))}] \mid S_{<(t,s)}\right] \\
&= \mathbb{E}\left[\mathbb{E}[e^{\lambda B_{t,s'}} \mid S_{<(t,s)}] \cdot \mathbb{E}[e^{\lambda(\nabla f_i(x_{t,s};S_{t,s}) - \nabla f_i(x_{t,s'};S_{t,s}) - \nabla F_i(x_{t,s}) + \nabla F_i(x_{t,s'}))} \mid S_{<t,s}]\right] \\
&= \mathbb{E}[e^{\lambda B_{t,s'}}] \cdot \mathbb{E}[e^{\lambda(\nabla f_i(x_{t,s};S_{t,s}) - \nabla f_i(x_{t,s'};S_{t,s}) - \nabla F_i(x_{t,s}) + \nabla F_i(x_{t,s'}))} \mid S_{<t,s}].
\end{aligned}$$

Since $f(\cdot; s)$ is β-smooth with respect to $\|\cdot\|_1$, we have that $|\nabla f_i(x_{t,s}; S_{t,s}) - \nabla f_i(x_{t,s'}; S_{t,s})| \le \beta \left\|x_{t,s} - x_{t,s'}\right\|_1$. Moreover, as $u_{t,s}$ is the right son of $u_{t,s'}$, the number of updates between $x_{t,s}$ and $x_{t,s'}$ is at most the number of leafs visited between these two vertices which is 2^{t-j}. Hence we get that

$$\left\|x_{t,s} - x_{t,s'}\right\|_1 \le D\eta_{t,s'} 2^{t-j} \le D 2^{-j+2},$$

which implies that $|\nabla f_i(x_{t,s}; S_{t,s}) - \nabla f_i(x_{t,s'}; S_{t,s})| \le \beta D 2^{-j+2}$. Since $\mathbb{E}[\nabla f_i(x_{t,s}; S_{t,s}) - \nabla f_i(x_{t,s'}; S_{t,s}) \mid S_{<t,s}] = \nabla F_i(x_{t,s}) - \nabla F_i(x_{t,s'})$, by repeating similar arguments to the case $\ell = 0$, we get that

$$\begin{aligned}
\mathbb{E}[e^{\lambda(\nabla f_i(x_{t,s};S_{t,s}) - \nabla f_i(x_{t,s'};S_{t,s}) - \nabla F_i(x_{t,s}) + \nabla F_i(x_{t,s'}))} \mid S_{<t,s}] &\le e^{O(1)\lambda^2\beta^2 D^2 2^{-2j}/|S_{t,s}|} \\
&\le e^{O(1)\lambda^2\beta^2 D^2 2^{-j}/b}.
\end{aligned}$$

Overall we have that $\mathbb{E}[e^{\lambda B_{t,s}}] \le \mathbb{E}[e^{\lambda B_{t,s'}}] \cdot e^{O(1)\lambda^2\beta^2 D^2 2^{-j}/b}$. Applying this inductively, we get that for every (t, s)

$$\mathbb{E}[e^{\lambda B_{t,s}}] \le e^{O(1)\lambda^2(L^2 + \beta^2 D^2)/b}.$$

□

Using the previous two lemmas, we can prove Theorem 6.

Proof. The setting of the parameters and the condition on β ensures that $2^T \le b$ hence Lemma 4.1 implies the claim about privacy. Now we proceed to prove utility. In this proof, we use the equivalent representation $k = 2^{t-1} + \ell(s)$ for a leaf vertex (t, s) where $\ell(s)$ is the number whose binary representation is s. By smoothness we get,

$$\begin{aligned}
F(x_{k+1}) &\le F(x_k) + \langle \nabla F(x_k), x_{k+1} - x_k \rangle + \beta \left\|x_{k+1} - x_k\right\|_1^2 / 2 \\
&\le F(x_k) + \eta_k \langle \nabla F(x_k), w_k - x_k \rangle + \beta \eta_k^2 D^2 / 2 \\
&= F(x_k) + \eta_k \langle \nabla F(x_k), x^\star - x_k \rangle + \eta_k \langle v_k, w_k - x^\star \rangle \\
&\quad + \eta_k \langle \nabla F(x_k) - v_k, w_k - x^\star \rangle + \beta \eta_k^2 D^2 / 2 \\
&\le F(x_k) + \eta_k (F(x^\star) - F(x_k)) + \eta_k D \left\|\nabla F(x_k) - v_k\right\|_\infty \\
&\quad + \eta_k (\langle v_k, w_k \rangle - \min_{w \in \mathcal{X}} \langle v_k, w \rangle) + \beta \eta_k^2 D^2 / 2.
\end{aligned}$$

Subtracting $F(x^\star)$ from each side, using Lemmas 4.1 and 4.2 and taking expectations, we have

$$\begin{aligned}
\mathbb{E}[F(x_{k+1}) - F(x^\star)] \le (1 - \eta_k)\mathbb{E}[F(x_k) - F(x^\star)] + \eta_k D(L + \beta D)\sqrt{\frac{\log d}{b}} \\
+ \frac{\eta_k^2}{2}\beta D^2 + \eta_k D L \frac{2^t \log m}{b\varepsilon}.
\end{aligned}$$

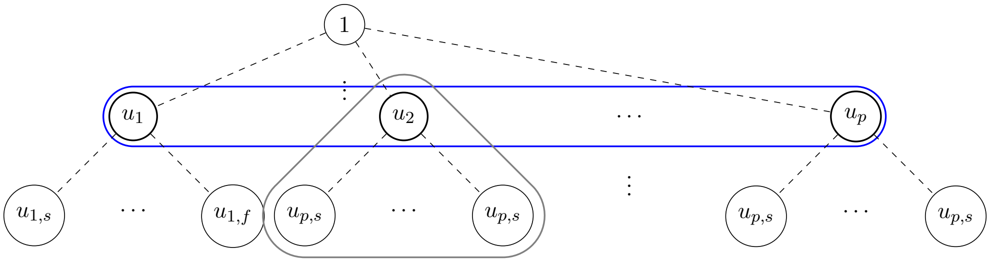

Figure 2. We view Algorithm 3 as a sequence of algorithms $\mathcal{A}_1, \dots, \mathcal{A}_p$, each $\mathcal{A}_i$ operating on the subtree of the vertex u_i using the outputs of the previous algorithms. The gray set denotes the subtree over which $\mathcal{A}_2$ operates; its outputs are the iterates corresponding to the leafs of this subtree. If each $\mathcal{A}_i$ is $(\varepsilon_0, \delta_0)$-DP, then shuffling the samples at nodes of depth j (in blue) amplifies the privacy to roughly $(\varepsilon_0\sqrt{\log(1/\delta)/2^j}, \delta + n\delta_0)$-DP.

Letting $\alpha_k = \eta_k D(L+\beta D)\sqrt{\frac{\log d}{b}} + \frac{\eta_k^2}{2}\beta D^2 + \eta_k DL\frac{2^t \log m}{b\varepsilon}$, we have

$$\begin{aligned}\mathbb{E}[F(x_K) - F(x^\star)] &\le \sum_{k=1}^{K} \alpha_k \prod_{i>k}(1-\eta_i) \\ &= \sum_{k=1}^{K} \alpha_k \frac{(k-1)k}{K(K+1)} \le \sum_{k=1}^{K} \alpha_k \frac{k^2}{K^2}.\end{aligned}$$

Since $t \le T$ and $K = 2^T$, simple algebra now yields

$$\mathbb{E}[F(x_K) - F(x^\star)] \le O\left(D(L+\beta D)\frac{\sqrt{\log d}}{\sqrt{b}} + \frac{\beta D^2}{2^T} + DL\frac{2^T \log m}{b\varepsilon}\right).$$

The number of samples in the algorithm is upper bounded by $T^2 \cdot b$ hence the first part of the claim follows by setting $b = n/\log^2 n$ and $T = \frac{1}{2}\log\left(\frac{b\varepsilon\beta D}{L\log m}\right)$. The condition on β ensures that the term inside the log is greater than 1. The second part follows similarly using $T=1$ and $b=n$. □

4.2 Approximate Differential Privacy

The previous section achieves the optimal non-private rate $1/\sqrt{n}$ only for $\varepsilon = \Theta(1)$. In this section we show that for approximate differential privacy, it is possible to achieve the optimal rates when $\varepsilon \ge \Omega(n^{-1/4})$. The first approach to improve the privacy analysis here is to use advanced composition for approximate DP. Unfortunately, it is not enough by itself and we use amplification by shuffling results to achieve the optimal bounds. The following theorem summarizes the guarantees of Algorithm 3 for approximate privacy.

Theorem 7. *Let $\delta \le 1/n$ and assume that $\mathsf{diam}_1(\mathcal{X}) \le D$, $m \le O(d)$ and that $f(x;z)$ is convex, L-Lipschitz and β-smooth with respect to $\|\cdot\|_1$. Assume $\frac{L\log(n/\delta)\log m \log^2 n}{n\varepsilon D} \le \beta \le \frac{\sqrt{n}L\log(n/\delta)\log m}{\varepsilon D\log n}$ and $\varepsilon \le \frac{(L\log(n/\delta)\log m)^{1/4}\sqrt{\log n}}{(n\beta D)^{1/4}}$. Let $\lambda_{t,s} = \frac{LD2^{T/2}\log(n/\delta)}{b\varepsilon}$, $b = n/\log^2 n$, and $T = \frac{2}{3}\log\left(\frac{b\varepsilon\beta D}{L\log(n/\delta)\log m}\right)$, then Algorithm 3 is (ε,δ)-DP, queries n gradients, and has*

$$\mathbb{E}[F(x_K) - F(x^\star)] \le O\left(D(L+\beta D)\frac{\sqrt{\log d}\log n}{\sqrt{n}}\right) + O\left(\frac{\sqrt{\beta}LD^2\log(1/\delta)\log m\log^2 n}{n\varepsilon}\right)^{2/3}.$$

The following lemma proves privacy in this setting.

Lemma 4.4. *Let $2^T \le b$, $\delta \le 1/n$ and $\varepsilon \le \sqrt{2^{-T}\log(1/\delta)}$. Setting $\lambda_{t,s} = \frac{LD2^{T/2}\log(n/\delta)}{b\varepsilon}$, Algorithm 3 is $(O(\varepsilon), 21\delta)$-DP. Moreover, $\mathbb{E}[\langle v_{t,s}, w_{t,s}\rangle] \le \mathbb{E}[\min_{w\in\mathcal{X}}\langle v_{t,s}, w\rangle]+O(LD2^{T/2}\log(n/\delta)\log(m)/b\varepsilon)$.*

To prove Lemma 4.4, we use the following privacy amplification by shuffling.

Lemma 4.5 ([FMT20], Theorem 3.8)**.** *Let $\mathcal{A}_i : \mathcal{T}^{i-1} \times \mathcal{Z} \to \mathcal{T}$ for $i \in [n]$ be a sequence of algorithm such that $\mathcal{A}_i(w_{1:i-1}, \cdot)$ is $(\varepsilon_0, \delta_0)$-DP for all values of $w_{1:i-1} \in \mathcal{T}^{i-1}$ with $\varepsilon_0 \le O(1)$. Let $\mathcal{A}_S : \mathcal{Z}^n \to \mathcal{T}^n$ be an algorithm that given $z_{1:n} \in \mathcal{Z}^n$, first samples a random permutation π, then sequentially computes $w_i = \mathcal{A}_i(w_{1:i-1}, z_{\pi(i)})$ for $i \in [n]$ and outputs $w_{1:n}$. Then for any δ such that $\varepsilon_0 \le \log(\frac{n}{16\log(2/\delta)})$, the algorithm $\mathcal{A}_s$ is $(\varepsilon, \delta + 20n\delta_0)$ where $\varepsilon \le O(\varepsilon_0\sqrt{\log(1/\delta)/n})$.*

Proof. We use the same notation as Lemma 4.1 where $\mathcal{S} = (z_1, \dots, z_{n-1}, z_n), \mathcal{S}' = (z_1, \dots, z_{n-1}, z_n')$ denote two neighboring datasets with iterates $x = (x_1, \dots, x_K)$ and $x' = (x_1', \dots, x_K')$. Here, we prove privacy after conditioning on the event that the n'th sample is sampled at phase t and depth j. We need to show that the iterates are (ε, δ)-indistinguishable. We only need to prove privacy for the iterates at phase t as the iterates before phase t do not depend on the n'th sample and the iterates after phase t are (ε, δ)-indistinguishable by post-processing.

Let us now focus on the iterates at phase t. Let $u_1, \dots, u_p$ denote the vertices at level j that has samples $S_1, \dots, S_p$ each of size $|S_i| = 2^{-j}b$. We will have two steps in the proof. First, we use advanced composition to show that the iterates that are descendant of a vertex u_i are $(\varepsilon_0, \delta_0)$-DP where roughly $\varepsilon_0 = 2^{j/2}\varepsilon$. As we have $p = 2^j$ vertices at depth j, we then use the amplification by shuffling result to argue that the final privacy guarantee is (ε, δ)-DP (see Fig. 2 for a demonstration of the shuffling in our algorithm).

Let $\mathcal{A}_i$ be the algorithm that outputs the iterates corresponding to the leafs that are descendants of u_i; we denote this output by O_i. Note that the inputs of $\mathcal{A}_i$ are the samples at u_i, which we denote as S_i, and the previous outputs $O_1, \dots, O_{i-1}$. In this notation, we have that $O_i = \mathcal{A}_i(O_1, \dots, O_{i-1}, S_i)$. We let $\mathcal{A}_i$, S_i and O_i denote the above quantities when the input dataset is $\mathcal{S}_i$ and similarly $\mathcal{A}_i'$, S_i' and O_i' for $\mathcal{S}'$. To prove privacy, we need to show that $(O_1, \dots, O_p) \approx_{(\varepsilon,\delta)} (O_1', \dots, O_p')$, that is $(O_1, \dots, O_p)$ and $(O_1', \dots, O_p')$ are (ε, δ)-indistinguishable

To this end, we first describe an equivalent sampling procedure for the sets $S_1, \dots, S_p$. Given r samples, the algorithm basically constructs the sets $S_1, \dots, S_p$ by sampling uniformly at random p sets of size r/p without repetition. Instead, we consider the following sampling procedure. First, we randomly choose a set of size $p(r-1)$ samples that does not include the n'th sample and using this set we randomly choose $r/p - 1$ samples for each set S_i. Then, we shuffle the remaining p samples and add each sample to the corresponding set. It is clear that this sampling procedure is equivalent. We prove privacy conditional on the output of the first stage of the randomization procedure which will imply privacy unconditionally.

Assuming without loss of generality that the samples which remained in the second stage are $z_{n-p+1}, \dots, z_n$, and letting $\pi : [p] \to \{n-p+1, \dots, n\}$ denote the random permutation of the second stage, the algorithms $\mathcal{A}_i$ and $\mathcal{A}_i'$ can be written as a function of the previous outputs and the sample $z_{\pi(i)}$. This is true since the $\mathcal{S}$ and $\mathcal{S}'$ differ in one sample and therefore the first $r/p - 1$ samples in the sets S_i and S_i' are identical. Thus, we can write $O_i = \mathcal{A}_i(O_1, \dots, O_{i-1}, z_{\pi(i)})$.

Using the above notation, we are now ready to prove privacy. First, we show privacy for each i using advanced composition. Similarly to Lemma 4.1, as each iterate k which is a leaf of u_i has sensitivity $|\langle c_i, v_k - v_k'\rangle| \le \frac{DL}{2^{-j}b}$, we have that x_k and x_k' are $\frac{\varepsilon}{2^{T/2-j}\log(n/\delta)}$-indistinguishable since $\lambda_{t,s} = \frac{LD2^{T/2}\log(n/\delta)}{b\varepsilon}$. Since there are 2^{t-j} leafs of u_i, advanced composition (Lemma 2.2) implies that $O_i \approx_{(\varepsilon_0,\delta_0)} O_i'$ where $\varepsilon_0 = \frac{\varepsilon}{2^{T/2-j}\log(n/\delta)}\sqrt{2^{t-j}\log(1/\delta_0)} \le \frac{O(\varepsilon)}{\sqrt{\log(1/\delta)}2^{-j/2}}$ by setting $\delta_0 = \delta/n$.

Finally, we can use the amplification by shuffling result to finish the proof. First, note that we need $\varepsilon_0 \le \log(\frac{2^j}{16\log(2/\delta)})$ to be able to use Lemma 4.5. If $2^j \le O(\log(1/\delta))$ then we do not need the amplification by shuffling result as $\varepsilon_0 \le O(\varepsilon 2^{j/2}/\sqrt{\log(1/\delta)}) \le O(\varepsilon)$. Otherwise 2^j is large enough so that we can use Lemma 4.5. Since each $\mathcal{A}_i$ and $\mathcal{A}'_i$ are $(\varepsilon_0, \delta_0)$-DP and since the second stage shuffles the inputs to each algorithm, Lemma 4.5 now implies that the outputs of the algorithms $\mathcal{A}_i$ and $\mathcal{A}'_i$ are $(\varepsilon_f, \delta + 20n\delta_0)$-DP where $\varepsilon_f \le \frac{\varepsilon_0\sqrt{\log(1/\delta)}}{2^{j/2}} \le O(\varepsilon)$ which proves the claim. □

Theorem 7 now follows using similar arguments to the proof of Theorem 6.

Proof. The assumptions on β ensure that $2^T \le b$ and the assumptions on ε ensure $\varepsilon \le 2^{-T/2}\log(n/\delta)$ hence the privacy follows from Lemma 4.4. The utility analysis is similar to the proof of Theorem 6. Repeating the same arguments in the proof of Theorem 6 while using the new value of $\lambda_{t,s}$, we get

$$\mathbb{E}[F(x_K) - F(x^\star)] \le O\left(D(L+\beta D)\frac{\sqrt{\log d}}{\sqrt{b}} + \frac{\beta D^2}{2^T} + DL\frac{2^{T/2}\log(n/\delta)\log m}{b\varepsilon} \right).$$

As the number of samples is upper bounded by $T^2 \cdot b$, we set $T = \frac{2}{3}\log\left(\frac{b\varepsilon\beta D}{L\log(n/\delta)\log m}\right)$ and $b = n/\log^2 n$ to get the first part of the theorem. Note that the condition on β ensure the term inside the log is greater than 1. □

5 Implications for Strongly Convex Functions

When the function is strongly convex, we use standard reductions to the convex case to achieve better rates [FKT20]. Given a private algorithm $\mathcal{A}$ for the convex case, we use the following algorithm for the strongly convex case (see [FKT20]): run $\mathcal{A}$ for $k = \lceil \log\log n \rceil$ iterates, each initialized at the output of the previous iterates and run for $n_i = 2^{i-2}n/\log n$. Using this reduction with our algorithms for convex functions, we have the following theorems for non-smooth and smooth functions.

Theorem 8. *Assume $\mathsf{diam}_1(\mathcal{X}) \le D$ and $f(x;z)$ is convex, L-Lipschitz, and λ-strongly convex with respect to $\|\cdot\|_1$ for all $z \in \mathcal{Z}$. Then using Algorithm 2 in the above algorithm results in an algorithm that uses $O(\log n \log\log n \cdot \min(n^{3/2}\sqrt{\log d}, n^2\varepsilon/\sqrt{d}))$ gradients and outputs $\hat{x}$ such that*

$$\mathbb{E}[F(\hat{x}) - F(x^\star)] = LD \cdot O\left(\frac{\log d}{n} + \frac{d\log^3 d \log\frac{1}{\delta}}{n^2\varepsilon^2}\right).$$

Theorem 9. *Let $\delta \le 1/n$ and assume that $\mathsf{diam}_1(\mathcal{X}) \le D$, $m \le O(d)$ and that $f(x;z)$ is convex, L-Lipschitz, λ-strongly convex and β-smooth with respect to $\|\cdot\|_1$ where $\beta = O(L/D)$. Then using Algorithm 3 in the above algorithm results in an algorithm that uses $O(n)$ gradients and outputs $\hat{x}$ such that*

$$\mathbb{E}[F(\hat{x}) - F(x^\star)] \le LD \cdot O\left(\frac{\log d \log^2 n}{n}\right) + LD \cdot O\left(\frac{\log(1/\delta)\log m \log^2 n}{n\varepsilon}\right)^{4/3}.$$

The proof follows directly from the proof of Theorem 5.1 in [FKT20], together with the bounds of Section 3 and Section 4.

6 Lower Bounds

We conclude the paper with tight lower bounds. Our lower bounds are for the excess empirical loss but these can be translated to lower bounds for excess population loss using a simple bootstrapping approach [BFTT19].

6.1 Lower Bounds for Non-Smooth Functions

In this section, we prove tight lower bounds for non-smooth functions using bounds for estimating the sign of the mean. In this problem, given a dataset $\mathcal{S} = (z_1, \dots, z_n)$ with mean $\bar{z}$, we aim to design private algorithms that estimate $\text{sign}(\bar{z})$. The following lemma provides a lower bound for this problem. We defer the proof to Appendix D.1.

Lemma 6.1. *Let $\mathcal{S} = (z_1, \dots, z_n)$ where $z_i \in \mathcal{Z} = \{-D/d, D/d\}^d$ and let $\bar{z} = \frac{1}{n}\sum_{i=1}^n z_i$. Then any (ε, δ)-DP algorithm $\mathcal{A} : \mathcal{Z} \to \{-1, +1\}^d$ has*

$$\max_{\mathcal{S} \in \mathcal{Z}^n} \mathbb{E}\left[\sum_{j=1}^d |\bar{z}_j| 1\{\mathcal{A}(\mathcal{S})_j \neq \text{sign}(\bar{z}_j)\}\right] \geq \Omega\left(\frac{D\sqrt{d}}{n\varepsilon \log d}\right).$$

The previous lemma implies our desired lower bound.

Theorem 10. *Let $f(x; z_i) = L\left\|x - z_i\right\|_1$ where $z_i \in \mathcal{Z} = \{-D/d, D/d\}^d$, $\hat{F}(x; \mathcal{S}) = \frac{1}{n}\sum_{i=1}^n \left\|x - z_i\right\|_1$, and $\mathcal{X} = \{x : \left\|x\right\|_1 \leq D\}$. Then any (ε, δ)-DP algorithm $\mathcal{A}$ has*

$$\max_{\mathcal{S} \in \mathcal{Z}^n} \mathbb{E}\left[\hat{F}(\mathcal{A}(\mathcal{S}); \mathcal{S}) - \min_{x \in \mathcal{X}} \hat{F}(x; \mathcal{S})\right] \geq \Omega\left(\frac{LD\sqrt{d}}{n\varepsilon \log d}\right).$$

Proof. First, note that $f(x; z_i)$ is L-Lipschitz with respect to $\left\|\cdot\right\|_1$. Moreover, it is immediate to see that the minimizer of $\hat{F}(\cdot; \mathcal{S})$ is $x^\star = \text{sign}(\bar{z})D/d$ where $\bar{z} = \frac{1}{n}\sum_{i=1}^n z_i$ is the mean. Letting $\hat{x} = \mathcal{A}(\mathcal{S})$, simple algebra yields

$$\hat{F}(\hat{x}; \mathcal{S}) - \hat{F}(x^\star; \mathcal{S}) \geq L\sum_{j=1}^d |\bar{z}_j| 1\{\text{sign}(\hat{x}_j) \neq \text{sign}(\bar{z}_j)\}.$$

The claim now follows from Lemma 6.1 as $\text{sign}(\mathcal{A}(\mathcal{S}))$ is differentially private by post-processing. □

6.2 Lower Bounds for Smooth Functions

In this section we prove tight lower bounds for smooth function. Specifically, we focus on β-smooth functions with $\beta \approx L/D$; such an assumption holds for many applications including LASSO (linear regression). Our results in this section build on the lower bounds of Talwar et al. [TTZ15] which show tight bounds for private Lasso for sufficiently large dimension. We have the following lower bound for smooth functions which we prove in Appendix D.2.

Theorem 11. *Let $\mathcal{X} = \{x \in \mathbb{R}^d : \left\|x\right\|_1 \leq D\}$. There is family of convex functions $f : \mathcal{X} \times \mathcal{Z} \to \mathbb{R}$ that is L-Lipschitz and β-smooth with $\beta \leq L/D$ such that any (ε, δ)-DP algorithm $\mathcal{A}$ with $\delta = n^{-\omega(1)}$ has*

$$\sup_{\mathcal{S} \in \mathcal{Z}^n} \mathbb{E}\left[\hat{F}(\mathcal{A}(\mathcal{S}); \mathcal{S}) - \min_{x \in \mathcal{X}} \hat{F}(x; \mathcal{S})\right] \geq LD \cdot \widetilde{\Omega}\left(\min\left(\frac{1}{(n\varepsilon)^{2/3}}, \frac{\sqrt{d}}{n\varepsilon}\right)\right).$$

The lower bound of Theorem 11 implies the optimality of our upper bounds; if $d \geq \widetilde{O}((n\varepsilon)^{2/3})$ then the lower bound is essentially $1/(n\varepsilon)^{2/3}$ which is achieved by the private Frank-Wolfe algorithm of Section 4, otherwise $d \leq \widetilde{O}((n\varepsilon)^{2/3})$ and the lower bound is $\sqrt{d}/n\varepsilon$ which is the same bound that private mirror descent (Section 3) obtains.

References

[ACGMMTZ16] M. Abadi, A. Chu, I. Goodfellow, B. McMahan, I. Mironov, K. Talwar, and L. Zhang. “Deep Learning with Differential Privacy”. In: *23rd ACM Conference on Computer and Communications Security (ACM CCS)*. 2016, pp. 308–318.

[BFTT19] R. Bassily, V. Feldman, K. Talwar, and A. Thakurta. “Private stochastic convex optimization with optimal rates”. In: *Advances in Neural Information Processing Systems*. Vol. 32. 2019, pp. 11282–11291.

[BGN21] R. Bassily, C. Guzman, and A. Nandi. “Non-Euclidean Differentially Private Stochastic Convex Optimization”. In preparation. 2021.

[BST14] R. Bassily, A. Smith, and A. Thakurta. “Private empirical risk minimization: Efficient algorithms and tight error bounds”. In: *55th Annual Symposium on Foundations of Computer Science*. 2014, pp. 464–473.

[CMS11] K. Chaudhuri, C. Monteleoni, and A. D. Sarwate. “Differentially private empirical risk minimization”. In: *Journal of Machine Learning Research* 12 (2011), pp. 1069–1109.

[DKMMN06] 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)*. 2006.

[DMNS06] C. Dwork, F. McSherry, K. Nissim, and A. Smith. “Calibrating noise to sensitivity in private data analysis”. In: *Proceedings of the Third Theory of Cryptography Conference*. 2006, pp. 265–284.

[DNPR10] C. Dwork, M. Naor, T. Pitassi, and G. N. Rothblum. “Differential privacy under continual observation”. In: *Proceedings of the Forty-Second Annual ACM Symposium on the Theory of Computing*. 2010, pp. 715–724.

[DNRR15] C. Dwork, M. Naor, O. Reingold, and G. N. Rothblum. “Pure differential privacy for rectangle queries via private partitions”. In: *International Conference on the Theory and Application of Cryptology and Information Security*. 2015, pp. 735–751.

[DR14] C. Dwork and A. Roth. “The Algorithmic Foundations of Differential Privacy”. In: *Foundations and Trends in Theoretical Computer Science* 9.3 & 4 (2014), pp. 211–407.

[DSSST10] J. C. Duchi, S. Shalev-Shwartz, Y. Singer, and A. Tewari. “Composite Objective Mirror Descent”. In: *Proceedings of the Twenty Third Annual Conference on Computational Learning Theory*. 2010.

[Duc18] J. C. Duchi. “Introductory Lectures on Stochastic Convex Optimization”. In: *The Mathematics of Data*. IAS/Park City Mathematics Series. American Mathematical Society, 2018.

[Duc19] J. C. Duchi. *Information Theory and Statistics*. Lecture Notes for Statistics 311/EE 377, Stanford University. Accessed May 2019. 2019. URL: http://web.stanford.edu/class/stats311/lecture-notes.pdf.

[Fel16] V. Feldman. "Generalization of ERM in Stochastic Convex Optimization: The Dimension Strikes Back". In: *Advances in Neural Information Processing Systems*. Vol. 29. 2016, pp. 3576–3584.

[FKT20] V. Feldman, T. Koren, and K. Talwar. "Private stochastic convex optimization: optimal rates in linear time". In: *Proceedings of the 52nd Annual ACM on the Theory of Computing*. 2020, pp. 439–449.

[FLLZ18] C. Fang, C. J. Li, Z. Lin, and T. Zhang. "SPIDER: Near-optimal non-convex optimization via stochastic path-integrated differential estimator". In: *Advances in Neural Information Processing Systems*. Vol. 31. 2018, pp. 689–699.

[FMT20] V. Feldman, A. McMillan, and K. Talwar. "Hiding Among the Clones: A Simple and Nearly Optimal Analysis of Privacy Amplification by Shuffling". In: *arXiv:2012.12803 [cs.LG]* (2020).

[HRS16] M. Hardt, B. Recht, and Y. Singer. "Train faster, generalize better: Stability of stochastic gradient descent". In: *ICML*. 2016, pp. 1225–1234. URL: http://jmlr.org/proceedings/papers/v48/hardt16.html.

[JT14] P. Jain and A. Thakurta. "(Near) dimension independent risk bounds for differentially private learning". In: *Proceedings of the 31st International Conference on Machine Learning*. 2014, pp. 476–484.

[KST12] D. Kifer, A. Smith, and A. Thakurta. "Private convex empirical risk minimization and high-dimensional regression". In: *Proceedings of the Twenty Fifth Annual Conference on Computational Learning Theory*. 2012, pp. 25–1.

[LFN18] H. Lu, R. M. Freund, and Y. Nesterov. "Relatively smooth convex optimization by first-order methods, and applications". In: *SIAM Journal on Optimization* 28.1 (2018), pp. 333–354.

[LJSB12] S. Lacoste-Julien, M. Schmidt, and F. Bach. "A simpler approach to obtaining an O (1/t) convergence rate for the projected stochastic subgradient method". In: *arXiv preprint arXiv:1212.2002* (2012).

[SSBD14] S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.

[SSSSS09] S. Shalev-Shwartz, O. Shamir, N. Srebro, and K. Sridharan. "Stochastic Convex Optimization." In: *Proceedings of the Twenty Second Annual Conference on Computational Learning Theory*. 2009.

[SU17] T. Steinke and J. Ullman. "Between Pure and Approximate Differential Privacy". In: *Journal of Privacy and Confidentiality* 7.2 (2017), pp. 3–22.

[TTZ15] K. Talwar, A. Thakurta, and L. Zhang. "Nearly optimal private Lasso". In: *Advances in Neural Information Processing Systems*. Vol. 28. 2015, pp. 3025–3033.

[YSC19] A. Yurtsever, S. Sra, and V. Cevher. "Conditional Gradient Methods via Stochastic Path-Integrated Differential Estimator". In: *Proceedings of the 36th International Conference on Machine Learning*. Vol. 97. 2019, pp. 7282–7291.

A Non-Contractivity of Mirror Descent

In this section, we provide counter examples that show that Mirror Descent is not a contraction in general. To this end, we consider the standard mirror descent algorithm with KL-regularization over the simplex $\Delta_{d-1} = \{x \in \mathbb{R}^d_+ : \|x\|_1 = 1\}$, that is, the following update with $h(x) = \sum_{j=1}^d x_j \log x_j$,

$$x^{k+1} = \operatorname*{argmin}_{x \in \Delta} \left\{ \langle \nabla f(x_k), x \rangle + \frac{1}{\eta_k} D_{\mathrm{h}}(x, x_k) \right\},$$

which yields the update

$$x^{k+1} = \frac{x^k \cdot e^{-\eta \nabla f(x^k)}}{\left\| x^k \cdot e^{-\eta \nabla f(x^k)} \right\|_1}. \tag{1}$$

We let $x_{k+1} = \mathsf{MD}_{\eta,f}(x_k)$ denote the above mirror descent update. The following lemma shows that mirror descent is not contractive even for linear functions.

Lemma A.1. *There exists a linear function $f : \Delta_2 \to \mathbb{R}$ such that for every $0 < \eta \le 1$, there are $x_0, y_0 \in \Delta_2$ such that the mirror descent update $x_1 = \mathsf{MD}_{\eta,f}(x_0)$ and $y_1 = \mathsf{MD}_{\eta,f}(y_0)$ have*

$$\|x_1 - y_1\|_1 \ge (1 + \eta/4) \|x_0 - y_0\|_1, \quad D_{\mathrm{h}}(x_1, y_1) \ge (1 + \eta/4) D_{\mathrm{h}}(x_0, y_0).$$

Proof. We consider a linear function $f(x_1, x_2, x_3) = -x_2 - x_3$, and two starting iterates for $n > 0$ to be chosen presently

$$x_0 = \left(1 - \frac{3}{n}, \frac{1}{n}, \frac{2}{n}\right), \quad y_0 = \left(1 - \frac{3}{n}, \frac{2}{n}, \frac{1}{n}\right).$$

First, notice that for this setting of parameters, we have that:

$$\|x_0 - y_0\|_1 = \frac{2}{n}, \quad D_{\mathrm{h}}(x_0, y_0) = D_{\mathrm{kl}}(x_0, y_0) = \frac{\log 2}{n}.$$

Using mirror descent update (1), we have

$$x_1 = \frac{1}{c} \left(x_{0,1}, x_{0,2}\ e^{\eta}, x_{0,3}\ e^{\eta} \right), \quad y_1 = \frac{1}{c} \left(y_{0,1}, y_{0,2}\ e^{\eta}, y_{0,3}\ e^{\eta} \right),$$

where $c = 1 + \frac{3}{n}(e^{\eta} - 1)$. Setting $n \ge 100(e^{\eta} - 1)/\eta$, we get that $c \le 1 + \eta/20$. Since $x_{0,1} = y_{0,1}$, we get that

$$\begin{aligned} \|x_1 - y_1\|_1 &= \frac{e^{\eta}}{c} \|x_0 - y_0\|_1 \\ &\ge \frac{1 + \eta}{1 + \eta/20} \|x_0 - y_0\|_1 \\ &\ge \|x_0 - y_0\|_1 + \frac{\eta}{4} \|x_0 - y_0\|_1 . \end{aligned}$$

Moreover, for KL-divergence we have

$$\begin{aligned} D_{\mathrm{kl}}(x_1, y_1) &= \frac{e^{\eta}}{c} D_{\mathrm{kl}}(x_0, y_0) \\ &\ge (1 + \eta/4) D_{\mathrm{kl}}(x_0, y_0). \end{aligned}$$

□

Although Lemma A.1 says that mirror descent update is not contractive even for linear functions, it does not preclude the possibility that mirror descent is stable. Indeed, the following lemma shows that mirror descent enjoys similar stability guarantees to SGD for linear functions. Extending this stability result to general convex functions is an interesting open question.

Lemma A.2. *Let $\mathcal{S} = (z_1, \ldots, z_n)$ and $\mathcal{S}' = (z_1', \ldots, z_n)$ be neighboring datasets where $x_i \in \mathbb{R}^d$ and $\|z_i\|_\infty \le L$. Consider the functions $f(x; z) = \langle z, x \rangle$. Let $\{x_k\}_{k=0}^T$ be the iterates of Algorithm 4 on $\mathcal{S}$ with $x_0 = \frac{1}{d} \cdot 1$ for R rounds and $\eta > 0$. Similarly, Let $\{y_k\}_{k=0}^T$ be the iterates of Algorithm 4 on $\mathcal{S}'$ with $y_0 = \frac{1}{d} \cdot 1$ for R rounds and $\eta > 0$. Then after R rounds ($T = Rn$ iterates),*

$$\|x_T - y_T\|_1^2 \le D_{\mathrm{kl}}(x_T, y_T) + D_{\mathrm{kl}}(y_T, x_T) \le 4\eta^2 L^2 R^2.$$

Proof. First, note that

$$\log \frac{x_k}{y_k} = \eta \sum_{i=1}^{k-1} (g_i' - g_i) + C,$$

where C is a constant vector, g_i and g_i' are the (sub)-gradients for $\mathcal{S}$ and $\mathcal{S}'$, respectively. Thus we have that

$$\begin{aligned} D_{\mathrm{kl}}(x_T, y_T) + D_{\mathrm{kl}}(y_T, x_T) &= \langle x_k - y_k, \log \frac{x_k}{y_k} \rangle \\ &= \eta \langle x_k - y_k, \sum_{i=1}^{k-1} (g_i' - g_i) \rangle \\ &\le \eta \sqrt{D_{\mathrm{kl}}(x_T, y_T) + D_{\mathrm{kl}}(y_T, x_T)} \sum_{i=1}^{k-1} \|g_i' - g_i\|_\infty \\ &\le 2\eta L R \sqrt{D_{\mathrm{kl}}(x_T, y_T) + D_{\mathrm{kl}}(y_T, x_T)} \ , \end{aligned}$$

where the first inequality follows from holder's inequality and the strong convexity of KL-divergence with respect to $\|\cdot\|_1$ (this is Pinsker's inequality; see e.g., [Duc19]) and the second inequality follows since the first sample z_1 (or z_1') appears R times. The claim follows. □

Algorithm 4 Stochastic Mirror Descent

```
Require: Dataset $\mathcal{S} = (z_1, \ldots, z_n) \in \mathcal{Z}^n$, step sizes $\eta$, initial point $x_0$, number of rounds $R$;
$k \leftarrow 0$
for $r = 1$ to $R$ do
    Sample a random permutation $\pi : [n] \to [n]$
    for $i = 1$ to $n$ do
        Set $g_k = \nabla f(x_k; z_{\pi(i)})$
        Find $x_{k+1} := \mathrm{argmin}_{x \in \Delta_{d-1}} \{\langle g_k, x - x_k \rangle + \frac{1}{\eta} D_{\mathrm{h}}(x, x_k)\}$ where $h(x) = \sum_{j=1}^d x_j \log x_j$
        $k \leftarrow k + 1$
return $\bar{x}_T = \frac{1}{T} \sum_{k=1}^T x_k$
```

B Rates for General ℓ_p-Geometry

In this section, we extend our algorithms to work for general ℓ_p-geometries for $p > 1$. Here, the optimization is over the domain $\mathcal{X} = \{x \in \mathbb{R}^d : \|x\|_p \le 1\}$ and we consider functions $f : \mathcal{X} \to \mathbb{R}$ that are L-Lipschitz with respect to $\|\cdot\|_p$, that is, $\|g\|_q \le L$ for all x and sub-gradient $g \in \partial f(x)$ where $1/p + 1/q = 1$.

B.1 Algorithms for ERM for $1 \le p \le 2$

To extend Algorithm 1 to work for general geometries, we need to bound the sensitivity of the gradients. Consider $1 \le p \le 2$ then $q > 2$ which implies that $\|g\|_2 \le d^{1/2-1/q}\|g\|_q$, that is, the function is $d^{1/2-1/q}L$ with respect to $\|\cdot\|_2$.

Algorithm 5 Noisy Mirror Descent for General Geometries

```
Require: Dataset $\mathcal{S} = (z_1, \ldots, z_n) \in \mathcal{Z}^n$, $1 < p$ and convex set $\mathcal{X} = \{x \in \mathbb{R}^d : \|x\|_p \le 1\}$, convex function $h : \mathcal{X} \to \mathbb{R}$, step sizes $\{\eta_k\}_{k=1}^T$, batch size $b$, initial point $x_0$, number of iterations $T$;
Find $q \ge 1$ such that $1/q + 1/p = 1$
for $k = 1$ to $T$ do
    Sample $S_1, \ldots, S_b \sim \mathrm{Unif}(\mathcal{S})$
    Set $\hat{g}_k = \frac{1}{b}\sum_{i=1}^{b} \nabla f(x_k; S_i) + \zeta_i$ where $\zeta_i \sim \mathsf{N}(0, \sigma^2 I_d)$ with $\sigma = 100L\sqrt{d^{1-2/q}\log(1/\delta)}/b\varepsilon$
    Find $x_{k+1} := \mathrm{argmin}_{x \in \mathcal{X}}\{\langle \hat{g}_k, x - x_k\rangle + \frac{1}{\eta_k} D_{\mathrm{h}}(x, x_k)\}$
return $\bar{x}_T = \frac{1}{T}\sum_{k=1}^{T} x_k$ (convex)
return $\hat{x}_T = \frac{2}{T(T+1)}\sum_{k=1}^{T} k x_k$ (strongly convex)
```

Theorem 12. *Let $1 < p \le 2$, $h : \mathcal{X} \to \mathbb{R}$ be 1-strongly convex with respect to $\|\cdot\|_p$, $x^\star = \mathrm{argmin}_{x \in \mathcal{X}} \hat{F}(x; S)$, and assume $D_{\mathrm{h}}(x^\star, x_0) \le D^2$. Let $f(x; z)$ be convex and L-Lipschitz with respect to $\|\cdot\|_p$ for all $z \in \mathcal{Z}$. Setting $1 \le b$, $T = \frac{n^2}{b^2}$ and $\eta_k = \frac{D}{\sqrt{T}}\frac{1}{\sqrt{L^2 + 4d^{2/q}\sigma^2 \log d}}$, Algorithm 5 is (ε, δ)-DP and*

$$\mathbb{E}[\hat{F}(\bar{x}_T; S) - \hat{F}(x^\star; S)] \le LD \cdot O\left(\frac{b}{n} + \frac{\sqrt{d \log \frac{1}{\delta}(1 + \log d \cdot 1\{p < 2\})}}{n\varepsilon}\right).$$

Moreover, if $f(x; z)$ is λ-strongly convex relative to $h(x)$, then setting $\eta_k = \frac{2}{\lambda(k+1)}$

$$\mathbb{E}[\hat{F}(\hat{x}_T; S) - \hat{F}(x^\star; S)] \le O\left(\frac{L^2 b^2}{\lambda n^2} + \frac{L^2 d \log \frac{1}{\delta}(1 + \log d \cdot 1\{p < 2\})}{\lambda n^2 \varepsilon^2}\right).$$

Proof. Following the proof of Theorem 3, privacy follows from similar arguments, and for utility we need to upper bound $\mathbb{E}[\|\tilde{g}_k\|_q]$. Note that for $p = q = 2$ we have $\mathbb{E}[\|\tilde{g}_k\|_q^2] \le d$. Otherwise we have

$$\mathbb{E}[\|\tilde{g}_k\|_q^2] \le 2L^2 + 2\mathbb{E}[\|\zeta_k\|_q^2] \le 2L^2 + 2d^{2/q}\mathbb{E}[\|\zeta_k\|_\infty^2] \le 2L^2 + 2d^{2/q}\mathbb{E}[\|\zeta_k\|_\infty^2] \le 2L^2 + 8d^{2/q}\sigma^2 \log d.$$

Now we complete the proof for $p < 2$. The same proof works for $p = 2$. The previous bound implies

$$\begin{aligned}
\mathbb{E}[\hat{F}(\bar{x}_T; S) - \hat{F}(x^\star; S)] &\le \frac{D^2}{T\eta} + \eta L^2 + 4\eta d^{2/q}\sigma^2 \log d \\
&\le 2D\sqrt{(L^2 + 4d^{2/q}\sigma^2 \log d)/T} \\
&\le LD \cdot O\left(\frac{b}{n} + \frac{\sqrt{d \log d \log \frac{1}{\delta}}}{n\varepsilon}\right),
\end{aligned}$$

where the second inequality follows from the choice of η. For the second part, Lemma 3.2 implies that

$$\mathbb{E}[\hat{F}(\hat{x}_T; S) - \hat{F}(x^\star; S)] \le \frac{L^2}{\lambda} O\left(\frac{b^2}{n^2} + \frac{d \log d \log \frac{1}{\delta}}{n^2 \varepsilon^2}\right). \qquad \square$$

B.2 Algorithms for SCO

We extend Algorithm 2 to work for general ℓ_p-geometries by using the general noisy mirror descent (Algorithm 5) to solve the optimization problem at each phase. The following theorem proves our main result for ℓ_p-geometry, that is, Theorem 5.

Theorem 13. *Let $1 < p \le 2$. Assume $\mathsf{diam}_p(\mathcal{X}) \le D$ and $f(x; z)$ is convex and L-Lipschitz with respect to $\|\cdot\|_p$ for all $z \in \mathcal{Z}$. If we set*

$$\eta = \frac{D}{L} \min\left\{1/\sqrt{(p-1)n}, \varepsilon/\sqrt{d \log \tfrac{1}{\delta}(1 + \log d \cdot 1\{p < 2\})}\right\},$$

then Algorithm 6 requires $O(\log n \cdot \min(n^{3/2}\sqrt{\log d}, n^2\varepsilon/\sqrt{d}))$ gradients and its output has

$$\mathbb{E}[F(x_k) - F(x^\star)] = LD \cdot O\left(\frac{1}{\sqrt{(p-1)n}} + \frac{\sqrt{d \log \frac{1}{\delta}(1 + \log d \cdot 1\{p < 2\})}}{(p-1)n\varepsilon}\right).$$

Proof. The proof follows from identical argument to the proof of Theorem 4 using the fact that $h_i(x) = \frac{1}{2(p-1)} \|x - x_{i-1}\|_p^2$ is 1-strongly convex with respect to $\|\cdot\|_p$. $\square$

Algorithm 6 Localized Noisy Mirror Descent

```
Require: Dataset $\mathcal{S} = (z_1, \ldots, z_n) \in \mathcal{Z}^n$, $1 \le p$, constraint set $\mathcal{X}$, step size $\eta$, initial point $x_0$;
Set $k = \lceil \log n \rceil$
for $i = 1$ to $k$ do
    Set $\varepsilon_i = 2^{-i}\varepsilon$, $n_i = 2^{-i}n$, $\eta_i = 2^{-4i}\eta$
    Apply Algorithm 5 with $(\varepsilon_i, \delta)$-DP, batch size $b_i = \max(\sqrt{n_i/\log d}, \sqrt{d/\varepsilon_i})$, $T = n_i^2/b_i^2$
    and $h_i(x) = \frac{1}{2(p-1)} \|x - x_{i-1}\|_p^2$ for solving the ERM over $\mathcal{X}_i = \{x \in \mathcal{X} : \|x - x_{i-1}\|_p \le$
    $2L\eta_i n_i(p-1)\}$:
    $F_i(x) = \frac{1}{n_i} \sum_{j=1}^{n_i} f(x; z_j) + \frac{1}{\eta_i n_i (p-1)} \|x - x_{i-1}\|_p^2$
    Let $x_i$ be the output of the private algorithm
return the final iterate $x_k$
```

C Proofs of Section 3

C.1 Proof of Lemma 3.2

Proof. First, by strong convexity we have

$$\begin{aligned} f(x_k) - f(x^\star) &\le \langle \nabla f(x_k), x_k - x^\star \rangle - \lambda D_{\mathrm{h}}(x^\star, x_k) \\ &= \langle g_k, x_k - x^\star \rangle + \langle \nabla f(x_k) - g_k, x_k - x^\star \rangle - \lambda D_{\mathrm{h}}(x^\star, x_k). \end{aligned} \qquad (2)$$

Let us now focus on the term $\langle g_k, x_k - x^\star \rangle$. The definition of x_{k+1} implies that for all $y \in \mathcal{X}$

$$\langle g_k + \frac{1}{\eta_k}(\nabla h(x_{k+1}) - \nabla h(x_k)), y - x_{k+1} \rangle \ge 0.$$

Substituting $y = x^\star$, we have

$$\begin{aligned}\langle g_k, x_k - x^\star \rangle &= \langle g_k, x_k - x_{k+1} \rangle + \langle g_k, x_{k+1} - x^\star \rangle \\ &\le \langle g_k, x_k - x_{k+1} \rangle + \frac{1}{\eta_k} \langle \nabla h(x_{k+1}) - \nabla h(x_k), x^\star - x_{k+1} \rangle \\ &\overset{(i)}{=} \langle g_k, x_k - x_{k+1} \rangle + \frac{1}{\eta_k} \left(D_{\mathrm{h}}(x^\star, x_k) - D_{\mathrm{h}}(x^\star, x_{k+1}) - D_{\mathrm{h}}(x_{k+1}, x_k) \right) \\ &\overset{(ii)}{\le} \frac{\eta_k}{2} \left\| g_k \right\|_\infty^2 + \frac{1}{2\eta_k} \left\| x_k - x_{k+1} \right\|_1^2 + \frac{1}{\eta_k} \left(D_{\mathrm{h}}(x^\star, x_k) - D_{\mathrm{h}}(x^\star, x_{k+1}) - D_{\mathrm{h}}(x_{k+1}, x_k) \right) \\ &\overset{(iii)}{\le} \frac{\eta_k}{2} \left\| g_k \right\|_\infty^2 + \frac{1}{\eta_k} \left(D_{\mathrm{h}}(x^\star, x_k) - D_{\mathrm{h}}(x^\star, x_{k+1}) \right),\end{aligned}$$

where (i) follows from the definition of bregman divergence, (ii) follows from Fenchel-Young inequality, and (iii) follows since $h(x)$ is 1-strongly convex with respect to $\left\| \cdot \right\|_1$. Substituting into (2),

$$f(x_k) - f(x^\star) \le \frac{\eta_k}{2} \left\| g_k \right\|_\infty^2 + \langle \nabla f(x_k) - g_k, x_k - x^\star \rangle + \frac{1}{\eta_k} \left(D_{\mathrm{h}}(x^\star, x_k) - D_{\mathrm{h}}(x^\star, x_{k+1}) \right) - \lambda D_{\mathrm{h}}(x^\star, x_k).$$

Multiplying by k and summing from $k = 1$ to T, we get

$$\begin{aligned}\sum_{k=1}^{T} k(f(x_k) - f(x^\star)) &\le \frac{1}{2\lambda} \sum_{k=1}^{T} \left\| g_k \right\|_\infty^2 + \langle \nabla f(x_k) - g_k, x_k - x^\star \rangle \\ &\qquad + \frac{\lambda}{2} \left(k(k-1) D_{\mathrm{h}}(x^\star, x_k) - k(k+1) D_{\mathrm{h}}(x^\star, x_{k+1}) \right) \\ &\le \frac{1}{2\lambda} \sum_{k=1}^{T} \left\| g_k \right\|_\infty^2 + \langle \nabla f(x_k) - g_k, x_k - x^\star \rangle.\end{aligned}$$

The claim now follows by taking expectations and using Jensen's inequality. □

D Proofs for Section 6

D.1 Proofs for Lemma 6.1

Without loss of generality, we assume that $D = 1$. Moreover, similarly to the proof of Theorem 11, we prove lower bounds on the sample complexity to achieve a certain error which will imply our lower bound on the utility. For an algorithm $\mathcal{A}$ and data $\mathcal{S} \in \mathcal{Z}^n$, define the error of $\mathcal{A}$:

$$\mathsf{Err}(\mathcal{A}, \mathcal{S}) = \mathbb{E} \left[\sum_{j=1}^{d} |\bar{z}_j| 1\{\mathrm{sign}(\mathcal{A}(\mathcal{S})_j) \neq \mathrm{sign}(\bar{z}_j)\} \right].$$

The error of a $\mathcal{A}$ for datasets of size n is $\mathsf{Err}(\mathcal{A}, n) = \sup_{\mathcal{S} \in \mathcal{Z}^n} \mathsf{Err}(\mathcal{A}, \mathcal{S})$.

We let $n^\star(\alpha, \varepsilon)$ denote the minimal n such that there is an (ε, δ)-DP (with $\delta = n^{-\omega(1)}$) Aansim $\mathcal{A}$ such that $\mathsf{Err}(\mathcal{A}, n^\star(\alpha, \varepsilon)) \le \alpha$. We prove the following lower bound on the sample complexity which implies Lemma 6.1.

Proposition 1. *Let $z_i \in \{-1/d, 1/d\}^d$, $\alpha \le 1$, and $\varepsilon \le 1$. Then*

$$n^\star(\alpha, \varepsilon) \ge \Omega(1) \cdot \frac{\sqrt{d}}{\alpha\varepsilon \log d}.$$

The proof follows directly from the following two lemmas.

Lemma D.1 (Talwar et al. [TTZ15], Theorem 3.2)**.** *Let the assumptions of Proposition 1 hold. Then*

$$n^\star(\alpha = 1/4, \varepsilon = 0.1) \ge \Omega(1) \cdot \frac{\sqrt{d}}{\log d}.$$

The following lemma shows how to extend the above lower bound to arbitrary accuracy and privacy parameters.

Lemma D.2. *Let $\varepsilon_0 \le 0.1$. For $\alpha \le \alpha_0/2$ and $\varepsilon \le \varepsilon_0/2$,*

$$n^\star(\alpha, \varepsilon) \ge \frac{\alpha_0\varepsilon_0}{\alpha\varepsilon} n^\star(\alpha_0, \varepsilon_0).$$

Proof. The proof follows the same arguments as in the proof of Lemma D.5. □

D.2 Proof of Theorem 11

In this section, we prove Theorem 11. We begin by recalling the lower bound of Talwar et al. [TTZ15] and showing how it implies Lemma D.3.

Talwar et al. [TTZ15] consider the family of quadratic functions where $f(x; a_i, b_i) = (a_i^T x - b_i)^2$ where $a_i \in \mathbb{R}^d$ and $b_i \in \mathbb{R}$. We assume $\mathcal{X} = \{x : \|x\|_1 \le D\}$, $\|a_i\|_\infty \le C$, and $|b_i| \le CD$. Note that the function f is L-Lipschitz and β-smooth with $L \le O(C^2 D)$ and $\beta \le O(C^2)$ and there is a choice of a_i, b_i that attains these. Theorem 3.1 in [TTZ15] gives a lower bound of $1/n^{2/3}$ when $C = 1$, $D = 1$, and $d \ge \widetilde{\Omega}(n^{2/3})$. For general values of C and D, noticing that the function value is multiplied by $C^2 D^2$, the following lower bound follows as $LD = C^2 D^2$.

Lemma D.3. *Let $\mathcal{X} = \{x \in \mathbb{R}^d : \|x\|_1 \le D\}$ and $d \ge \widetilde{\Omega}(n^{2/3})$. There is family of convex functions $f : \mathcal{X} \times \mathcal{Z} \to \mathbb{R}$ that is L-Lipschitz and β-smooth with $\beta \le L/D$ such that any $(0.1, \delta)$-DP algorithm $\mathcal{A}$ with $\delta = o(1/n^2)$ has*

$$\sup_{\mathcal{S} \in \mathcal{Z}^n} \mathbb{E}\left[\hat{F}(\mathcal{A}(\mathcal{S}); \mathcal{S}) - \min_{x \in \mathcal{X}} \hat{F}(x; \mathcal{S})\right] \ge \widetilde{\Omega}\left(\frac{LD}{n^{2/3}}\right).$$

Now we proceed to prove Theorem 11 and we assume without loss of generality that $L = 1$ and $D = 1$. We use techniques from [SU17] to extend the lower bound of Lemma D.3 to hold for arbitrary d and ε. To this end, instead of lower bounding the excess loss, it will be convenient to prove lower bounds on the sample size to achieve a certain excess loss α. More precisely, given a dataset $\mathcal{S} \in \mathcal{Z}^n$ and algorithm $\mathcal{A}$, we define its empirical excess loss on $\mathcal{S}$

$$\mathcal{E}(\mathcal{A}, \mathcal{S}) = \mathbb{E}\left[\hat{F}(\mathcal{A}(\mathcal{S}); \mathcal{S}) - \min_{x \in \mathcal{X}} \hat{F}(x; \mathcal{S})\right].$$

We also define its worst-case excess loss over all datasets of size n

$$\mathcal{E}(\mathcal{A}, n) = \sup_{\mathcal{S} \in \mathcal{Z}^n} \mathcal{E}(\mathcal{A}, \mathcal{S}).$$

We let $n^\star(\alpha, \varepsilon)$ be the minimal sample size that is required to achieve excess loss $\mathcal{E}(\mathcal{A}, n^\star(\alpha, \varepsilon)) \le \alpha$ using an (ε, δ)-DP algorithm $\mathcal{A}$ with $\delta = n^{-\omega(1)}$. We prove the following lemma which implies Theorem 11.

Lemma D.4. *Let the assumptions of Theorem 11 hold. Then*

$$n^\star(\alpha,\varepsilon) \ge \begin{cases} \widetilde{\Omega}\left(\frac{1}{\alpha^{3/2}\varepsilon}\right) & \text{if } \alpha = 1/d \\ \widetilde{\Omega}\left(\frac{\sqrt{d}}{\alpha\varepsilon}\right) & \text{if } \alpha \le 1/d \end{cases}$$

The proof of Lemma D.4 basically follows from the following two Lemmas.

Lemma D.5. *For $0 < \alpha \le \alpha_0$ and $0 < \varepsilon \le \varepsilon_0 \le 0.1$,*

$$n^\star(\alpha,\varepsilon) \ge \Omega\left(\frac{\alpha_0\varepsilon_0}{\alpha\varepsilon} n^\star(\alpha_0,\varepsilon_0)\right).$$

Lemma D.6. *We have that*

$$n^\star(\alpha = 1/d, \varepsilon = 0.1) \ge \widetilde{\Omega}\left(d^{3/2}\right).$$

Before proving Lemmas D.5 and D.6, let us finish the proof of Lemma D.4. First, consider the case $\alpha = 1/d$. Lemma D.6 implies that

$$n^\star(\alpha = 1/d, \varepsilon) \ge \Omega\left(\frac{n^\star(\alpha = 1/d, \varepsilon = 0.1)}{\varepsilon}\right) \ge \widetilde{\Omega}\left(d^{3/2}/\varepsilon\right) = \widetilde{\Omega}\left(\frac{1}{\alpha^{3/2}\varepsilon}\right).$$

If $\alpha \le 1/d$, then similarly we have

$$n^\star(\alpha,\varepsilon) \ge \Omega\left(\frac{1}{d\alpha\varepsilon}\right) n^\star(\alpha = 1/d, \varepsilon = 0.1) \ge \widetilde{\Omega}\left(\frac{\sqrt{d}}{\alpha\varepsilon}\right).$$

Hence Lemma D.4 follows. Finally, we provide proofs for the remaining lemmas.

Lemma D.6. This lemma follows directly from Lemma D.3. Indeed, Lemma D.3 implies that if $d \ge \widetilde{\Omega}(n^{2/3})$ and $\varepsilon = 0.1$, the excess loss is lower bounded by $\mathcal{E}(\mathcal{A}, n) \ge \widetilde{\Omega}(1/n^{2/3})$. Stated differently, if $n \le \widetilde{O}(d^{3/2})$ then $\mathcal{E}(\mathcal{A}, n) \ge \widetilde{\Omega}(1/n^{2/3}) \ge \widetilde{\Omega}(1/d)$ which proves the claim. □

Lemma D.5. Given an (ε,δ)-DP algorithm $\mathcal{A}$ with $\mathcal{E}(\mathcal{A}, n) \le \alpha$, we show how to construct $\mathcal{A}'$ that is $(\varepsilon_0, 4\delta\varepsilon_0/\varepsilon)$-DP algorithm that works on datasets of size $n' = \Theta(\frac{\alpha\varepsilon}{\alpha_0\varepsilon_0} n)$ such that $\mathcal{E}(\mathcal{A}', n') \le \alpha_0$. This will prove the claim as we know that $n' \ge n(\alpha_0, \varepsilon_0)$. We now describe the construction of $\mathcal{A}'$. Given $\mathcal{S}' \in \mathcal{Z}^{n'}$ and $k > 0$ to be chosen presently, we define a new dataset $\mathcal{S}$ as follows: the first kn' samples are k copies of $\mathcal{S}'$ and the remaining $n - kn'$ are new samples $z \in \mathcal{Z}$ that have the loss function $f(x; z) = 0$ for all $x \in \mathcal{X}$. Clearly, these functions are convex, 0-Lipschitz, and 0-smooth. We then define $\mathcal{A}'(\mathcal{S}') = \mathcal{A}(\mathcal{S})$. Note that for all x we have that $\hat{F}(x; \mathcal{S}) = \frac{kn'}{n} \hat{F}(x; \mathcal{S}')$, which implies that

$$\begin{aligned} \mathcal{E}(\mathcal{A}', \mathcal{S}') &= \mathbb{E}[\hat{F}(\mathcal{A}(\mathcal{S}); \mathcal{S}') - \min_{x \in \mathcal{X}} \hat{F}(x; \mathcal{S}')] \\ &= \frac{n}{kn'} \mathbb{E}[\hat{F}(\mathcal{A}(\mathcal{S}); \mathcal{S}) - \min_{x \in \mathcal{X}} \hat{F}(x; \mathcal{S})] \\ &= \frac{n}{kn'} \mathcal{E}(\mathcal{A}, \mathcal{S}) \le \frac{n\alpha}{kn'}. \end{aligned}$$

Therefore if $n' \ge n\alpha/k\alpha_0$ we get $\mathcal{E}(\mathcal{A}', \mathcal{S}') \le \alpha_0$. Hence it remains to argue for privacy. Using the group privacy property of private algorithms [SU17](Fact 2.2), the algorithm $\mathcal{A}'$ is $(k\varepsilon, \frac{e^{k\varepsilon}-1}{e^{\varepsilon}-1}\delta)$-DP. Setting $k = \lfloor \log(1 + \varepsilon_0)/\varepsilon \rfloor$ implies the claim as $e^{k\varepsilon} - 1 \le \varepsilon_0$ and $k\varepsilon \le \varepsilon_0$. □