Oracle Efficient Private Non-Convex Optimization

Seth Neel [*1] **Aaron Roth** [*2] **Giuseppe Vietri** [*3] **Zhiwei Steven Wu** [*3]

Abstract

One of the most effective algorithms for differentially private learning and optimization is *objective perturbation*. This technique augments a given optimization problem (e.g. deriving from an ERM problem) with a random linear term, and then exactly solves it. However prior analyses of this approach crucially rely on the convexity and smoothness of the objective function. We give two algorithms that extend this approach substantially. The first algorithm requires nothing except boundedness of the loss function, and operates over a discrete domain. We achieve this by introducing a novel "normalization" step into the objective perturbation algorithm, which provides enough stability to satisfy differential privacy even without convexity. The second algorithm operates over a continuous domain and its privacy analysis requires only that the loss function be bounded and Lipschitz in its continuous parameter. We complement our theoretical results with an empirical evaluation of the non-convex case, in which we use an integer program solver as our optimization oracle. We find that for the problem of learning linear classifiers, directly optimizing for 0/1 loss using our approach can out-perform the more standard approach of privately optimizing a convex-surrogate loss function on the Adult dataset.

1. Introduction

Consider the general problem of optimizing a function $L : \mathcal{L}^n \times \mathcal{W} \mapsto \mathbb{R}$ defined with respect to a dataset $\mathcal{D} \in \mathcal{L}^n$ and a parameter $w \in \mathcal{W}$: $\arg\min_w L(\mathcal{D}, w)$. This general class of problems includes classical empirical risk minimization, amongst others, and is a basic problem in learning and optimization. We say that such a function L is 1-sensitive in the dataset $\mathcal{D}$ if changing one datapoint in $\mathcal{D}$ can change the value of $L(\mathcal{D}, w)$ by at most 1, for any parameter value w. Suppose that we want to solve an optimization problem like this subject to the constraint of differential privacy. The *exponential mechanism* provides a powerful, general-purpose, and often error-optimal method to solve this problem (McSherry & Talwar, 2007). It requires no assumptions on the function other than that it is 1-sensitive (this is a minimal assumption for privacy: more generally, its guarantees are parameterized by the sensitivity of the function). Unfortunately, the exponential mechanism is generally infeasible to run: its implementation (and the implementation of related mechanisms, like "Report-Noisy-Max" (Dwork & Roth, 2014)) requires the ability to enumerate the parameter range $\mathcal{W}$, making it infeasible in most learning settings, despite its use in proving general information theoretic bounds in private PAC learning (Kasiviswanathan et al., 2011). When $L(\mathcal{D}, w)$ is continuous, convex, and satisfies second order conditions like strong convexity or smoothness, the situation is better: there are a number of algorithms available, including simple output perturbation (Chaudhuri et al., 2011) and objective perturbation (Chaudhuri et al., 2011; Kifer et al., 2012; Iyengar et al., 2019). This partly mirrors the situation in non-private data analysis, in which convex optimization problems can be solved quickly and efficiently, and most non-convex problems are NP-hard in the worst case.

In the non-private case, however, the worst-case complexity of optimization problems does not tell the whole story. For many non-convex optimization problems, such as integer programming, there are fast heuristics that not only reliably succeed in optimizing functions deriving from real inputs, but can also certify their own success. In such settings, can we leverage these heuristics to obtain practical private optimization algorithms? In this paper, we give two novel analyses of *objective perturbation* algorithms that extend their applicability to 1-sensitive non-convex problems (and more generally, bounded sensitivity functions). We also get new results for *convex* problems, without the need for second order conditions like smoothness or strong convexity. Our first algorithm operates over a discrete parameter space $\mathcal{W}$,

[*]Equal contribution [1]Wharton Statistics Department, University of Pennsylvania, Philadelphia, Pennsylvania, USA [2]Department of Computer and Information Sciences, University of Pennsylvania, Philadelphia, Pennsylvania, USA [3]Department of Computer Science and Engineering, University of Minnesota, Minneapolis, Minnesota, USA. Correspondence to: Seth Neel <sethneel93@gmail.com >, Aaron Roth <aaroth@cis.upenn.edu>, Giuseppe Vietri <vietr002@umn.edu>, Zhiwei Steven Wu <zstevenwu@cmu.edu>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

and requires no further assumptions beyond 1-sensitivity for either its privacy or accuracy analysis — i.e. it is comparable in generality to the exponential mechanism. The second algorithm operates over a continuous parameter space $\mathcal{W}$, and requires only that $L(\mathcal{D}, w)$ be Lipschitz-continuous in its second argument. Its privacy analysis does not require convexity. Its accuracy analysis does — but does not require any 2nd order conditions. We implement our first algorithm to directly optimize classification error over a discrete set of linear functions on the Adult dataset, and find that it substantially outperforms private logistic regression.

1.1. Related work

Objective perturbation was first introduced by Chaudhuri et al. (2011), and analyzed for the special case of strongly convex functions. Its analysis was subsequently improved and generalized (Kifer et al., 2012; Iyengar et al., 2019) to apply to smooth convex functions, and to tolerate a small degree of error in the optimization procedure. We introduce a variant of objective perturbation that crucially uses a novel normalization step. Our paper is the first to give an analysis of a variant of objective perturbation without the assumption of convexity, and the first to give an accuracy analysis without making second order assumptions on the objective function even in the convex case. Chaudhuri et al. (2011) also introduced the related technique of *output perturbation* which perturbs the exact optimizer of a strongly convex function. The work most closely related to our first algorithm is Neel et al. (2018), who also give a similar "oracle efficient" algorithm for non-convex differentially private optimization: i.e. reductions from non-private optimization to private optimization. Their algorithm ("Report Separator Perturbed Noisy Max", or RSPM) relies on an implicit perturbation of the optimization objective by augmenting the dataset $\mathcal{D}$ with a random collection of examples drawn from a *separator set*. This is a non-standard piece of combinatorial structure, and so Neel et al. (2018)'s approach only works in limited settings (we do not know of any settings in which it applies beyond the handful specifically identified in Neel et al. (2018)). The algorithms which we introduce in this paper are substantially more general: because they directly perturb the objective, they do not rely on the existence of a small separator set for the class of functions in question. One of the contributions of our paper is the first experimental analysis of RSPM, in section 5. Neel et al. (2018) also give a generic method to transform an algorithm (like ours) whose privacy analysis depends on the success of the optimization oracle, to an algorithm whose privacy analysis does not depend on this, whenever the optimization heuristic can certify its success (integer program solvers have this property). Their method applies to the algorithms we develop in this paper. Our second algorithm crucially uses an ℓ_1 stability result for the "Follow-the-perturbed-leader" (FTPL) method in Suggala & Netrapalli (2019) in the context of online learning. Recently, Vietri et al. (2020) use the same FTPL method in the context of private synthetic data release, but they do not rely on the stability of FTPL to achieve privacy. Similar to our work, they follow the approach of designinig oracle-efficient algorithms that use heuristics like integer program solvers to solve private problems that are hard in the worst case, which was first proposed by Gaboardi et al. (2014).

2. Preliminaries

We first define a dataset, a loss function with respect to a dataset, and the two types of optimization oracles we will call upon. We then define differential privacy, and state basic properties.

A dataset $\mathcal{D} \subset \mathcal{L}^n$ is defined as a (multi)set of G-Lipschitz loss functions l. (Note that frequently, the dataset will explicitly contain "data points", and the loss functions will be implicitly defined). For w in a parameter space $\mathcal{W} \subset \mathbb{R}^d$, the loss on dataset $\mathcal{D}$ is defined to be

$$L\big(\mathcal{D}, w\big) = \sum_{l \in \mathcal{D}} l(w)$$

We will define two types of perturbed loss functions, and the corresponding oracles which are assumed to be able to optimize each type. These will be used in our discrete objective perturbation algorithm in Section 3 and our sampling based objective perturbation algorithm in Section 4 respectively.

Given a vector $\eta \in \mathbb{R}^d$, we define the perturbed loss to be:

$$\bar{L}(\mathcal{D}, w, \eta) = \frac{L(\mathcal{D}, w) - \big\langle \eta, w \big\rangle}{n}$$

where $n = |\mathcal{D}|$ is the size of the dataset $\mathcal{D}$. This is simply the loss function augmented with a linear term.

Let π be the normalization function formally defined in Section 3, which informally maps a d-dimensional vector with l_2 norm at most D to a unit vector in $\mathbb{R}^{d+1}$. Given a vector $\eta \in \mathbb{R}^{d+1}$ We define the perturbed normalized loss to be:

$$\bar{L}_\pi(\mathcal{D}, w, \eta) = \frac{L(\mathcal{D}, w) - \big\langle \eta, \pi\big(w\big) \big\rangle}{n}$$

Definition 2.1 (Approximate Linear Optimization Oracle)**.** Given as input a dataset $\mathcal{D} \in \mathcal{L}^n$ and a d-dimensional vector η, an α-approximate linear optimization oracle $\mathcal{O}_\alpha$ returns $w^* = \mathcal{O}_\alpha(\mathcal{D}, \eta) \in \mathcal{W}$ such that

$$\bar{L}(\mathcal{D}, w^*, \eta) \leq \inf_{w \in \mathcal{W}} \bar{L}(\mathcal{D}, w, \eta) + \alpha$$

When $\alpha = 0$ we say $\mathcal{O}$ is a linear optimization oracle.

Definition 2.2 (Approximate Normalized Linear Optimization Oracle)**.** Given as input a dataset $\mathcal{D} \in \mathcal{L}^n$ and a $(d+1)$-dimensional vector η, an α-approximate normalized linear optimization oracle $\mathcal{O}_{\alpha,\pi}$ returns $w^* = \mathcal{O}_{\alpha,\pi}(\mathcal{D}, \eta) \in \mathcal{W}$ such that

$$\bar{L}_\pi(\mathcal{D}, w^*, \eta) \leq \inf_{w \in \mathcal{W}} \bar{L}_\pi(\mathcal{D}, w, \eta) + \alpha$$

When $\alpha = 0$ we say $\mathcal{O}_\pi$ is a normalized linear optimization oracle. We remark that while it seems less natural to assume an oracle for the normalized perturbed loss which involves the non-linearity $\pi(w)$, in the supplement we show how we can linearize this term by introducing an auxiliary variable and introducing a convex constraint. This is ultimately how we implement this oracle in our experiments.

Definition 2.3. A randomized algorithm $\mathcal{M} : \mathcal{L}^n \to \mathcal{W}$ is an (α, β)-minimizer for $\mathcal{W}$ if for every dataset $\mathcal{D} \in \mathcal{L}^n$, with probability $1 - \beta$, it outputs $\mathcal{M}(\mathcal{D}) = w$ such that:

$$\frac{1}{n} L(\mathcal{D}, w) \leq \inf_{w^* \in \mathcal{W}} \frac{1}{n} L(\mathcal{D}, w^*) + \alpha$$

Certain optimization routines will have guarantees only for discrete parameter spaces:

Definition 2.4 (Discrete parameter spaces)**.** A τ-separated discrete parameter space $\mathcal{W}_\tau \subseteq \mathbb{R}^d$ is a discrete set such that for any pair of distinct vectors $w_1, w_2 \in \mathcal{W}_\tau$ we have $\|w_1 - w_2\|_2 \geq \tau$.

Finally we define differential privacy.

We call two *data sets* $\mathcal{D}, \mathcal{D}' \in \mathcal{L}^n$ *neighbors* (written as $\mathcal{D} \sim \mathcal{D}'$) if $\mathcal{D}$ can be derived from $\mathcal{D}'$ by replacing a single loss function $l_i \in \mathcal{D}'$ with some other element of $\mathcal{L}$.

Definition 2.5 (Differential Privacy (Dwork et al., 2006b;a))**.** Fix $\epsilon, \delta \geq 0$. A randomized algorithm $A : \mathcal{L}^* \to \mathcal{O}$ is (ϵ, δ)-differentially private (DP) if for every pair of neighboring data sets $\mathcal{D} \sim \mathcal{D}' \in \mathcal{L}^*$, and for every event $\Omega \subseteq \mathcal{O}$:

$$\Pr[A(\mathcal{D}) \in \Omega] \leq \exp(\epsilon) \Pr[A(\mathcal{D}') \in \Omega] + \delta.$$

The Laplace distribution centered at 0 with scale b is the distribution with probability density function $\text{Lap}(z|b) = \frac{1}{2b} e^{-\frac{|z|}{b}}$. We also make use of the exponential distribution which has density function $\text{Exp}(z|b) = \frac{1}{b} e^{-\frac{z}{b}}$ if $z \geq 0$ and $\text{Exp}(z|b) = 0$ otherwise.

3. Objective perturbation over a discrete decision space

In this section we give an objective perturbation algorithm that is (ϵ, δ)-differentially private for any non-convex Lipschitz objective over a discrete decision space $\mathcal{W}_\tau$. We assume that each $l \in \mathcal{L}$ is G-Lipschitz over $\mathcal{W}_\tau$ w.r.t. ℓ_2 norm: that is for any $w, w' \in \mathcal{W}_\tau$, $|l(w) - l(w')| \leq G\|w - w'\|_2$. Note that if l takes values in $[0, 1]$, then we know l is also $1/\tau$-Lipschitz due to the τ-separation in $\mathcal{W}_\tau$.

The Normalization Trick. The key technical innovation in this section of the paper is the modification of the standard objective perturbation algorithm by introducing a normalization step: rather than minimizing the perturbed loss, we minimize the perturbed normalized loss. Let D be a bound on the maximum ℓ_2 norm of any vector in $\mathcal{W}_\tau$. We will make use of a normalization onto the unit sphere in one higher dimension. The normalization function $\pi : \mathbb{R}^d \to \mathbb{R}^{d+1}$ is defined as:

$$\pi(w) = \left(w_1, \ldots, w_d, D\sqrt{1 - \|w\|_2^2 / D^2} \right) \frac{1}{D}$$

Note that $\|\pi(w)\|_2 = 1$ for all $w \in \mathcal{W}_\tau$, and also that for any $w, w' \in \mathcal{W}_\tau$,

$$\|\pi(w) - \pi(w')\|_2^2 \geq \frac{1}{D^2} \|w - w'\|_2^2, \tag{1}$$

since $\|\pi(w) - \pi(w')\|_2^2 = \frac{1}{D^2}(\|w - w'\|_2^2 + D^2(\sqrt{1 - \|w\|_2^2/D^2} - \sqrt{1 - \|w'\|_2^2/D^2})^2) \geq \frac{1}{D^2}\|w - w'\|_2^2$. This shows that normalizing into the $(d+1)$-dimensional sphere can't force points too much closer together than they start. The intuition behind the privacy proof is that the linear perturbation term provides stability; specifically we will argue that for any value of the noise η than induces a particular minimizer $\hat{w}$ on a dataset $\mathcal{D}$, there is a nearby value η' that would induce $\hat{w}$ on any adjacent dataset $\mathcal{D}'$. The argument proceeds by contradiction: suppose that there existed some $v \neq \hat{w}$ that was the minimizer on $\mathcal{D}'$. Then since $\mathcal{D}$ and $\mathcal{D}'$ only differ in one data point, the difference between the normalized losses of v and $\hat{w}$ on $\mathcal{D}'$ can be broken into three terms: the difference between their scores on $\mathcal{D}$ and the original perturbation term η, the difference between their scores on the two data points that differ between $\mathcal{D}, \mathcal{D}'$, and the inner product between their normalized difference $\pi(\hat{w}) - \pi(v)$ with $\eta' - \eta$. The first term is positive by virtue of $\hat{w}$ being the minimizer on the original dataset $\mathcal{D}$. The second term can be lower bounded using Lipschitzness of $\mathcal{L}$. The third term is lower bounded using the fact that $\eta' - \eta$ is chosen to maximize the inner product $\langle \eta' - \eta, \pi(\hat{w}) - \pi(v) \rangle$ by making the change in noise $\eta' - \eta$ move in the direction of $\pi(\hat{w})$ We can only guarantee this has a greater inner product with $\hat{w}$ than v if $\|\pi(\hat{w})\|_2 = \|\pi(v)\|_2$, which is the rationale behind the normalization trick. Then the whole expression can be shown to be lower bounded by 0, contradicting the fact that v is the unique minimizer of the normalized loss on $\mathcal{D}'$.

We now prove that OPDisc is differentially private, illustrating the importance of the normalization trick. We then state an accuracy bound, which follows from a simple tail bound on the random linear perturbation term.

Algorithm 1 Objective Perturbation over Discrete Space OPDisc

input $\mathcal{D} = \{l_i\}_{i=1}^n$, oracle $\mathcal{O}_\pi$ over $\mathcal{W}_\tau$, privacy parameters ϵ, δ

$\sigma \leftarrow \frac{7GD^2\sqrt{\ln 1/\delta}}{\tau\epsilon}$

Draw random vector $\eta \sim \mathcal{N}\big(0, \sigma^2\big)^{d+1}$ and use the projected oracle to solve:

$$\hat{w} \leftarrow \mathcal{O}_\pi(\mathcal{D}, \eta) \in \arg\min_{w \in \mathcal{W}_\tau} \bar{L}_\pi(\mathcal{D}, \eta, w)$$

output $\hat{w}$

Theorem 1. *Algorithm 1 is (ϵ, δ)-differentially private.*

Proof. For any realized noise vector η, we write $\hat{w} = \mathcal{O}_\pi(\mathcal{D}, \eta)$ as the output. Now consider the set of mappings $\mathcal{G} : W_\tau \times \mathbb{R}^{d+1} \to \mathbb{R}^{d+1}$. If we can show:

- $\exists g \in \mathcal{G}$ s.t. $\hat{w} = \mathcal{O}_\pi(\mathcal{D}', g(\hat{w}, \eta))$ (Lemma 4)
- $\Pr\big[\eta\big] \approx \Pr\big[g(\hat{w}, \eta)\big]$ (Lemma 3)
- W.p.1, $\arg\min_{w \in W_\tau} \bar{\mathcal{L}}(\mathcal{D}, w, \eta)$ is unique, (Lemma 2)

then the probability of outputting any particular w on input $\mathcal{D}$ is close to the corresponding probability, on input $\mathcal{D}'$ as desired. Lemma 3 follows from simple properties of the Gaussian distribution, and Lemma 2 from discreteness of W_τ, which are established in the Appendix. We focus on proving Lemma 4, which is the central part of the proof.

Lemma 2. *Fix any τ-separated vector space $\mathcal{W}_\tau$. For every dataset $\mathcal{D}$ there is a subset $B \subset \mathbb{R}^{d+1}$ such that $Pr\big[\eta \in B\big] = 0$ and for any $\eta \in \mathbb{R}^{d+1} \setminus B$:*

$$\exists \text{ a unique minimizer } \hat{w} \in \arg\min_{w \in \mathcal{W}_\tau} L\big(\mathcal{D}, w\big) - \big\langle \eta, \pi\big(w\big)\big\rangle$$

Denote the set of of noise vectors that induce output w on dataset $\mathcal{D}$ by $\mathcal{E}\big(\mathcal{D}, w\big) = \{\eta : \mathcal{O}_\pi(\mathcal{D}, \eta) = w\}$. Define our mapping $g \in \mathcal{G}$ by:

$$g(\hat{w}, \eta) \stackrel{\text{def}}{=} g_{\hat{w}}(\eta) = \eta + \frac{2}{\tau} GD^2 \pi\big(\hat{w}\big)$$

Note that the vector $\eta' - \eta = g_{\hat{w}}(\eta) - \eta$ is parallel to $\pi(\hat{w})$. Lemma 3 shows that with high probability over the draw of η, $\Pr\big[\eta\big] \approx \Pr\big[g_{\hat{w}}(\eta)\big]$.

Lemma 3. *Let $\eta \sim \mathcal{N}(0, \sigma^2)^{d+1}$, $\sigma \leftarrow \frac{7G^2D^2\sqrt{\log(1/\delta)}}{\tau\epsilon}$, and $w \in \mathcal{W}_\tau$. Then there exists a set $C \subset \mathbb{R}^{d+1}$ such that $Pr\big[\eta \in C^c\big] \geq 1 - \delta$, and for all $r \in C^c$ if p denotes the probability density function of η:*

$$\frac{p(r)}{p(g_w(r))} \leq e^\epsilon$$

Lemma 4. *Fix any $\hat{w}$ and any pair of neighboring datasets $\mathcal{D}, \mathcal{D}'$. Let $\eta \in \mathcal{E}\big(\mathcal{D}, \hat{w}\big)$ be such that $\hat{w}$ is the unique minimizer $\hat{w} \in \inf_w L(\mathcal{D}, w) - \big\langle \eta, \pi\big(w\big)\big\rangle$. Then $g_{\hat{w}}(\eta) \in \mathcal{E}\big(\mathcal{D}', \hat{w}\big)$. Hence:*

$$\mathbb{I}\{\eta \in \mathcal{E}\big(\mathcal{D}, \hat{w}\big)\} \leq \mathbb{I}\{g_{\hat{w}}(\eta) \in \mathcal{E}\big(\mathcal{D}', \hat{w}\big)\}$$

Proof. Let $c = \frac{4}{\tau} GD^2$. Suppose that $v \neq \hat{w}$ is the output on neighboring dataset $\mathcal{D}'$ when the noise vector is $g_{\hat{w}}(\eta)$. We will derive a contradiction. Since v is the unique minimizer on $\mathcal{D}'$:

$$\Big(L\big(\mathcal{D}', v\big) - \big\langle g_{\hat{w}}(\eta), \pi\big(v\big)\big\rangle\Big)$$
$$-\Big(L\big(\mathcal{D}', \hat{w}\big) - \big\langle g_{\hat{w}}(\eta), \pi\big(\hat{w}\big)\big\rangle\Big) < 0$$

Let i be the index where $\mathcal{D}$ and $\mathcal{D}'$ are different, such that $l_i \in \mathcal{D}$ and $l_i^{'} \in \mathcal{D}'$. Then $L\big(\mathcal{D}', w\big) = L\big(\mathcal{D}, w\big) - l_i(w) + l_i^{'}(w)$. Now, write the loss function in terms of $\mathcal{D}$ and rearranging terms:

$$\Big[\Big(L\big(\mathcal{D}, v\big) - \big\langle \eta, \pi\big(v\big)\big\rangle\Big) - \Big(L\big(\mathcal{D}, \hat{w}\big) - \big\langle \eta, \pi\big(\hat{w}\big)\big\rangle\Big)\Big]$$
$$+\big(l_i(\hat{w}) - l_i(v)\big) - \big(l_i^{'}(\hat{w}) - l_i^{'}(v)\big)$$
$$+\big\langle c\pi\big(\hat{w}\big), \pi\big(\hat{w}\big)\big\rangle - \big\langle c\pi\big(\hat{w}\big), \pi\big(v\big)\big\rangle < 0$$

Since $\hat{w}$ is a unique minimizer for $\mathcal{D}$ and η then term in the square bracket is positive. Hence:

$$\big(l_i(\hat{w}) - l_i(v)\big) - \big(l_i^{'}(\hat{w}) - l_i^{'}(v)\big)$$
$$+\big\langle c\pi\big(\hat{w}\big), \pi\big(\hat{w}\big) - \pi\big(v\big)\big\rangle < 0$$

Since l_i, l_i' are G-Lipschitz functions $\big(l_i(\hat{w}) - l_i(v)\big) - \big(l_i^{'}(\hat{w}) - l_i^{'}(v)\big) \geq -2G\|\hat{w} - v\|_2$. Now comes the importance of the normalization trick: because $||\pi(v)||_2 = ||\pi(\hat{w})||_2 = 1$, $\big\langle c\pi\big(\hat{w}\big), \pi\big(\hat{w}\big) - \pi\big(v\big)\big\rangle = \frac{c}{2}\|\pi\big(\hat{w}\big) - \pi\big(v\big)\|_2^2$, by expanding $\|\pi\big(\hat{w}\big) - \pi\big(v\big)\|_2^2$. Note that without the normalization, this last term could be negative, breaking the contradiction argument. Substituting this becomes:

$$-2G\|\hat{w} - v\|_2 + \frac{c}{2}\|\pi\big(\hat{w}\big) - \pi\big(v\big)\|_2^2 < 0$$

For the next step we use inequality (1). We also apply the assumption that for two vectors $\hat{w} \neq v$ the following inequality holds $\|\hat{w} - v\|_2 \geq \tau$.

$$\frac{c}{2D^2}\|\hat{w} - v\|_2^2 < 2G\|\hat{w} - v\|_2 \quad \text{(Inequality (1))}$$
$$\frac{c}{2D^2}\|\hat{w} - v\|_2 < 2G \quad \text{(Divide both sides by } \|\hat{w} - v\|_2\text{)}$$
$$c\|\hat{w} - v\|_2 < 4GD^2$$
$$c\tau < 4GD^2 \quad \text{(By assumption } \|\hat{w} - v\|_2 \geq \tau\text{)}$$
$$c < \frac{4GD^2}{\tau} \quad \text{(Divide both sides by } \tau\text{)}$$

This contradicts $c = \frac{4GD^2}{\tau}$. □

Putting the Lemmas together:

$$\Pr\big[\mathcal{O}_\pi(\mathcal{D},\eta) \in S\big] = \Pr\big[\eta \in \bigcup_{\hat{w}} \mathcal{E}\big(\mathcal{D},\hat{w}\big)\big]$$

$$= \int_{\mathbb{R}^{d+1}} p(\eta)\mathbb{I}\{\eta \in \bigcup_{\hat{w}} \mathcal{E}\big(\mathcal{D},\hat{w}\big)\}d\eta$$

$$= \int_{(\mathbb{R}^{d+1}\setminus B)\setminus C} p(\eta)\mathbb{I}\{\eta \in \bigcup_{\hat{w}} \mathcal{E}\big(\mathcal{D},\hat{w}\big)\}d\eta + \int_C p(\eta)\mathbb{I}\{\eta \in \bigcup_{\hat{w}} \mathcal{E}\big(\mathcal{D},\hat{w}\big)\}d\eta \tag{2}$$

$$\leq \int_{(\mathbb{R}^{d+1}\setminus C)\setminus B} p(\eta)\mathbb{I}\{\eta \in \bigcup_{\hat{w}} \mathcal{E}\big(\mathcal{D},\hat{w}\big)\}d\eta + \delta \tag{3}$$

$$= \sum_{\hat{w}\in S} \int_{\mathbb{R}^{d+1}\setminus(C\cup B)} p(\eta)\mathbb{I}\{\eta \in \mathcal{E}\big(\mathcal{D},\hat{w}\big)\}d\eta + \delta$$

$$\leq \sum_{\hat{w}\in S} \int_{\mathbb{R}^{d+1}\setminus(C\cup B)} p(\eta)\mathbb{I}\{g_{\hat{w}}(\eta) \in \mathcal{E}\big(\mathcal{D}',\hat{w}\big)\}d\eta + \delta \tag{4}$$

$$\leq \sum_{\hat{w}\in S} \int_{\mathbb{R}^{d+1}\setminus(C\cup B)} e^{\epsilon} p(g_{\hat{w}}(\eta))\mathbb{I}\{g_{\hat{w}}(\eta) \in \mathcal{E}\big(\mathcal{D}',\hat{w}\big)\}d\eta + \delta \tag{5}$$

$$= \sum_{\hat{w}\in S} \int_{\mathbb{R}^{d+1}\setminus(g_{\hat{w}}(C)\cup g_{\hat{w}}(B))} e^{\epsilon} p(\eta)\mathbb{I}\{\eta \in \mathcal{E}\big(\mathcal{D}',\hat{w}\big)\}\left|\frac{\partial g_{\hat{w}}}{\partial \eta}\right| d\eta + \delta \tag{6}$$

$$\leq e^{\epsilon} \sum_{\hat{w}\in S} \int_{\mathbb{R}^{d+1}} p(\eta)\mathbb{I}\{\eta \in \mathcal{E}\big(\mathcal{D}',\hat{w}\big)\}d\eta + \delta$$

$$= e^{\epsilon}\Pr\big[\eta \in \bigcup_{\hat{w}} \mathcal{E}\big(\mathcal{D}',\hat{w}\big)\big]$$

$$= e^{\epsilon}\Pr\big[\mathcal{O}_\pi(\mathcal{D}',\eta) \in S\big] + \delta$$

where equality (2) follows from Lemma 2. Then inequality (3) holds because C is chosen such that $\Pr\big[\eta \in C\big] < \delta$. The inequality (4) is from lemma 4 and inequality (5) is from the bounded ration lemma 3. Lastly, equality (6) follows because the mapping $\eta \to g_{\hat{w}}(\eta)$ is one-to-one. Also note that $\left|\frac{\partial g_{\hat{w}}}{\partial \eta}\right| = 1$ This completes the proof. □

We now state the accuracy guarantee, which follows from a standard Gaussian tail bound. Then in Subsection 3.1 we compare this guarantee to the accuracy guarantee for the competing RSPM method for learning discrete hyperplanes, in order to shed some light on the accuracy guarantee in practice.

Theorem 5 (Utility). *Algorithm 1 is an (α,β)-minimizer for $\mathcal{W}^*_\tau$ with*

$$\alpha = \frac{14GD^2\sqrt{2(d+1)\ln(4/\beta)\ln(1/\delta)}}{n\tau\epsilon}$$

3.1. Comparing OPDisc and RSPM

While both OPDisc and the RSPM algorithm of Neel et al. (2018) require discrete parameter spaces, OPDisc is substantially more general in that it only requires the loss functions be Lipschitz, whereas RSPM assumes the loss functions are bounded in $\{0,1\}$ (and hence $1/\tau$ Lipschitz over $\mathcal{W}_\tau$) and assumes the existence of a small separator set (defined in the supplement). Nevertheless, we might hope that in addition to greater generality, OPDisc has comparable or superior accuracy for natural classes of learning problems. We show this is indeed the case for the fundamental task of privately learning discrete hyperplanes, where it is better by a linear factor in the dimension. We define the RSPM algorithm, for which we must define the notion of a separator set, in the supplement.

Theorem 6 (RSPM Utility (Neel et al., 2018)). *Let $\mathcal{W}^*_\tau$ be a discrete parameter space with a separator set of size m. The Gaussian RSPM algorithm is an oracle-efficient (α,β)-minimizer for $\mathcal{W}^*_\tau$ for:*

$$\alpha = O\left(\frac{m\sqrt{m\ln(2m/\beta)\ln(1/\delta)}}{\epsilon n}\right)$$

Let I_τ be a τ discretization of $[-1,1]^d$, e.g. $I_\tau = [-1,-1+\tau,\dots 0,\tau,2\tau,\dots 1]^d$. Let W_τ be the subset of vectors in this discretization that lie within the unit Euclidean ball: $W_\tau = I_\tau \cap S(1)^d$. W_τ is τ-separated since any two distinct w, w' differ in at least one coordinate by at least τ. Moreover W_τ admits a separator set of size $m = \frac{2(d-1)}{\tau}$ (see the Appendix of Neel et al. (2018). Since the loss functions $l_i(w) = \mathbf{1}\{w\cdot x_i \geq 1\} \in \{0,1\}$ and W_τ is τ-separated, the loss functions l_i are $\frac{1}{\tau}$-Lipschitz. By Theorem 6, RSPM has accuracy bound:

$$\alpha_{\text{RSPM}} = O\left(\frac{d\sqrt{d\log(d/\beta\tau)\log(1/\delta)}}{\tau\sqrt{\tau}\epsilon n}\right)$$

By Theorem 5 OPDisc has accuracy bound:

$$\alpha_{\text{OPDisc}} = O\left(\frac{\sqrt{d\log(1/\beta)\log(1/\delta)}}{n\tau^2\epsilon}\right)$$

Thus, in this case, OPDisc has an accuracy bound that is different by a factor of roughly $d\sqrt{\tau}$. However, the bound of OPDisc is better only when τ is greater than $1/d^2$, pressing the question of how to set this parameter. The trade-off is

that setting τ too large makes the algorithm OPDisc add too much noise to the objective, and our accuracy guarantee degrades very fast. On the other hand, if τ is too large, then we can miss the optimal solution to a large extent. However, for practical scenarios, setting the value of τ to be much larger than $\frac{1}{d^2}$ gives a discretized decision space such that the optimal answer is not too far from the optimal on the corresponding continuous decision space. For instance, in our experiments, we set τ equals to one.

4. Objective perturbation for Lipschitz functions

We now present an objective perturbation algorithm (paired with an additional output perturbation step), which applies to arbitrary parameter spaces. The privacy guarantee holds for (possibly non-convex) Lipschitz loss functions, while the accuracy guarantee applies only if the loss functions are convex and bounded. Even in the convex case, this is a substantially more general statement than was previously known for objective perturbation: we don't require any second order conditions like strong convexity or smoothness (or even differentiability). Our guarantees also hold with access only to an α-approximate optimization oracle.

We present the full algorithm in Algorithm 2. It 1) uses the approximate linear oracle (in Definition 2.1) to solve polynomially many perturbed optimization objectives, each with an independent random perturbation, and 2) perturbs the average of these solutions with Laplace noise.

Before we proceed to our analysis, let us first introduce some relevant parameters. Let $\mathcal{W}$ have ℓ_∞ diameter D_∞, and ℓ_2 diameter D_2. We assume that the loss functions $l_i \in \mathcal{L}$ are G-Lipschitz with respect to ℓ_1 norm, and assume the loss functions are scaled to take values in $[0, 1]$. Our utility analysis requires convexity in the loss functions, and essentially follows from the high-probability bounds on the linear perturbation terms in the first stage and the output perturbation in the second stage.

Algorithm 2 Objective Perturbation Sampling OPSamp

input Approximate optimization oracle $\mathcal{O}_\alpha$, a dataset $\mathcal{D} = \{l_i\}_{i=1}^n$, privacy parameters ϵ, δ.
$\gamma \leftarrow \frac{\sqrt{\epsilon}}{\sqrt{n}} d^{5/4}\sqrt{D_2}$; $m \longleftarrow \frac{\ln(2d/\delta)}{2\gamma^2}$
for $k = 1$ to m **do**
$\eta \leftarrow \sqrt{\frac{D_2\sqrt{2d}\epsilon}{250G^2d^2D_\infty^2(1+\log(2/\beta))n}}$
Sample a random vector $\sigma^k \sim \mathrm{Exp}(\eta)^d$.
$w_k \leftarrow \mathcal{O}_\alpha\left(\mathcal{D}, \sigma^k\right)$
end for
$\lambda \leftarrow 4D_\infty\gamma + 250\eta G d^2 D_\infty^2 + \frac{\alpha}{10G}$
$\mu \sim \mathrm{Lap}(\lambda/\epsilon)^d$
output $\frac{1}{m}\sum_{k=1}^m w_k + \mu$

Theorem 7 (Utility)**.** *Assuming the loss functions are convex, Algorithm 2 is an (α', β)-minimizer for $\frac{1}{n}\mathcal{L}(w, \mathcal{D})$ with*

$$\alpha' = O\left(\frac{d^{5/4}GD_\infty\sqrt{D_2\log(1/\beta)}}{\sqrt{\epsilon n}} + \frac{\alpha\log(1/\beta)}{\epsilon}\right)$$

where α is the approximation error of the oracle $\mathcal{O}_\alpha$.

The privacy analysis of this algorithm crucially depends on a stability lemma proven by Suggala & Netrapalli (2019) in the context of online learning, and does not require convexity.[1]

Lemma 8 (Stability lemma (Suggala & Netrapalli, 2019))**.** *For any pair of neighboring data sets $\mathcal{D}, \mathcal{D}'$. Let $\mathcal{O}_\alpha(\mathcal{D}, \sigma)$ and $\mathcal{O}_\alpha(\mathcal{D}', \sigma)$ be the output of an approximate oracle on datasets $\mathcal{D}$ and $\mathcal{D}'$ respectively. Then,*

$$\mathbb{E}_\sigma\left[||\mathcal{O}_\alpha(\mathcal{D}, \sigma) - \mathcal{O}_\alpha(\mathcal{D}', \sigma)||_1\right] \leq 250\eta G d^2 D_\infty^2 + \frac{\alpha}{10G}$$

From now on, let $\Sigma = \{\sigma^i : i \in [m]\}$ be a sequence of of m i.i.d d-dimensional noise vectors and $\mathcal{W}(\mathcal{D}, \Sigma) = \frac{1}{m}\sum_i \mathcal{O}_\alpha(\mathcal{D}, \sigma^i)$ is the average output of m calls to an α-approximate oracle.

Lemma 9. *If $m = \frac{\ln(2d/\delta)}{2\gamma^2}$, for $0 \leq \gamma \leq 1$, then, with probability $1 - \delta/2$:*

$$\|\mathcal{W}(\mathcal{D}, \Sigma) - \mathbb{E}_\sigma[\mathcal{O}_\alpha(\mathcal{D}, \sigma)]\|_1 \leq 2D_\infty\gamma$$

where the randomness is taken over the different runs of $\mathcal{O}_\alpha$.

The next lemma combines Lemma 8 and Lemma 9 to get high probability sensitivity bound for the average output of the approximate oracle.

Lemma 10 (High Probability ℓ_1-sensitivity)**.** *For any pair of neighboring datasets $\mathcal{D}, \mathcal{D}'$, let $\mathcal{W}(\mathcal{D}, \Sigma)$, $\mathcal{W}(\mathcal{D}', \Sigma)$ be the sample average after $m = \frac{\ln(2d/\delta)}{\gamma^2}$ calls to an α-approximate oracle. Then, with probability $1 - \delta$ over the random draws of Σ,*

$$||\mathcal{W}(\mathcal{D}, \Sigma) - \mathcal{W}(\mathcal{D}', \Sigma)||_1 \leq 4D_\infty\gamma + 250\eta G d^2 D_\infty^2 + \frac{\alpha}{10G} \tag{7}$$

Theorem 11. *Algorithm 2 is (ϵ, δ)-differentially private.*

Proof sketch. Given a pair of neighboring data sets $\mathcal{D}, \mathcal{D}'$, we will condition on the set of noise vectors Σ satisfy the ℓ_1-sensitivity bound (7), which occurs with probability at least $1 - \delta$. Then the privacy guarantee follows from the use of Laplace mechanism. □

[1] Compared to the bound in Suggala & Netrapalli (2019), our bound has an additional factor of 2 since our neighboring relationship in Definition 2.5 is defined via replacement whereas in Suggala & Netrapalli (2019) the stability is defined in terms of adding another loss function.

5. Experiments

For our experiments, we consider the problem of privately learning a linear threshold function to solve a binary classification task. Given a labeled data set $\{(x_i, y_i)\}_{i=1}^n$ where each $x_i \in \mathbb{R}^d$ and $y_i \in \{-1, 1\}$, the classification problem is to find a hyperplane that best separates the positive from the negative samples. A common approach is to optimize a convex surrogate loss function that approximates the classification loss. We use this approach (private logistic regression) as our baseline. In comparison, using our algorithm OPDisc, we instead try and directly optimize $0/1$ classification error over a discrete parameter space, using an integer program solver. Although this can be computationally expensive, we find that it is feasible for relatively small datasets (we use a balanced subset of the Adult dataset with roughly $n = 15,000$ and $d = 23$ features, after one-hot encodings of categorical features). In this setting, we find that OPDisc can substantially outperform private logistic regression. We remark that "small data" is the regime in which applying differential privacy is most challenging, and we view our approach as a promising way forward in this important setting.

Data description and pre-processing We use the Adult dataset (Lichman, 2013), a common benchmark dataset derived from Census data. The classification task is to predict whether an individual earns over 50K per year. The dataset has $n = 48842$ records and 14 features that are a mix of both categorical and continuous attributes. The Adult dataset is unbalanced: only 7841 individuals have the $\geq 50k$ (positive) label. To arrive at a balanced dataset (so that constant functions achieve 50% error), we take all positive individuals, and an equal number of negative individuals selected at random, for a total dataset size of $n = 15682$. We encode categorical features with one-hot encodings, which increases the dimensionality of the dataset. We found it challenging to run our algorithm with more than 30 features, and so we take a subset of 7 features from the Adult dataset that are represented by $d = 23$ real-valued features after one-hot encoding. We chose the subset of features to optimize the accuracy of our logistic regression baseline.

Baseline: private logistic regression (LR). We use as our baseline private logistic regression which optimizes over the space of continuous halfspaces with the goal of minimizing the logistic loss function, given by $l_i(w) = \log(1 + \exp(-y\langle w, x_i\rangle))$. We implement a differentially private stochastic gradient descent (privateSGD) algorithm from Bassily et al. (2014); Abadi et al. (2016), keeping track of privacy loss using the moment accountant method as implemented in the TensorFlow Privacy Library. The algorithm involves three parameters: gradient clip norm, mini-batch size, and learning rate. For each target privacy parameters (ϵ, δ), we run a grid search to identify the triplet of parameters that give the highest accuracy. To lower the variance of the accuracy, we also take average over all the iterates in the run of privateSGD.

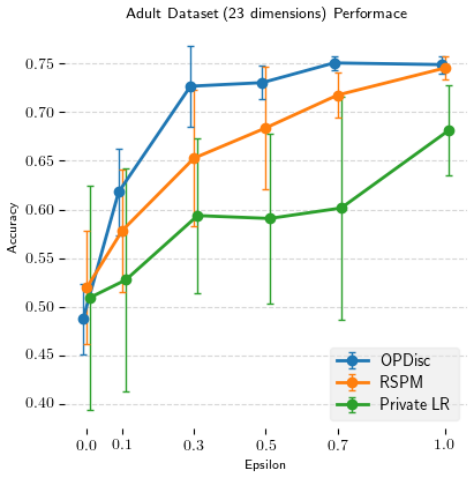

Figure 1. Accuracy versus ϵ.

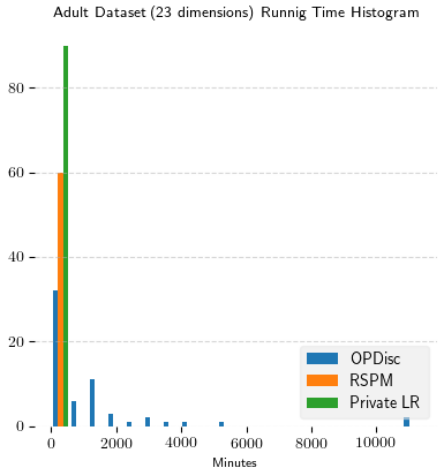

Figure 2. Distribution of run time.

Figure 3. Accuracy and runtime evaluation of OPDisc, RSPM, and Private Logistic Regression (LR) on the Adult data set with size $n = 15682$ and $d = 23$ features. The value of $\delta = 1/n^2$ for all methods in all runs.

Implementation details for OPDisc and RSPM For both OPDisc and RSPM, we encode each record $(x_i, y_i) \in \mathcal{D}$ as a $0/1$ loss function: $l_i(w) = \mathbb{I}[y_i \neq \text{sgn}(\langle x_i, w\rangle)]$. For both algorithms, we have separation parameter $\tau = 1$ and constrains the weight vectors to have ℓ_2 norm bounded

by $\sqrt{d}$. In OPDisc, each coordinate w_j can take values in the discrete set $\{-B, -B+1, \ldots, B-1, B\}$ with $B = \lfloor \sqrt{d} \rfloor$, and we constrain the $\|w\|_2$ to be at most $\sqrt{d}$. In RSPM, we optimize over the set $\{-1, 0, 1\}^d$. OPDisc requires an approximate projected linear optimization oracle (Definition 2.2) and RSPM requires a linear optimization oracle (Definition 2.1). In the appendix, we show that the optimization problems can be cast as mixed-integer programs (MIPs), allowing us to implement the oracles via the Gurobi MIP solver. The Gurobi solver was able to solve each of the integer programs we passed it.

Empirical evaluation. We evaluate our algorithms by their $(0/1)$ classification accuracy. The Figure 1(a) plots the accuracy of OPDisc and our baseline (y-axis) as a function of the privacy parameter ϵ (x-axis), averaged over 15 runs. We fix $\delta = 1/n^2$ for all three algorithms across all runs. The error bars report the empirical standard deviation. We see that both OPDisc and RSPM improve dramatically over the logistic regression baseline. This shows that in small-data settings, it is possible to improve over the error/privacy tradeoff given by standard convex-surrogate approaches by appealing to non-convex optimization heuristics. OPDisc also obtains consistently better error than RSPM. The algorithm OPDisc also has a significantly lower variance in its error compared to the other two algorithms. The Figure 2(a) gives a histogram of the run-time of our three methods for our experiment. For both OPDisc and RSPM, the running time is dominated by an integer-program solver. We see that while our method frequently completes quite quickly (often even beating our logistic regression baseline!), it has high variance, and occasionally requires a long time to run. However, we were always able to solve the necessary optimization problem, eventually.

6. Acknowledgments

Giuseppe Vietri has been supported by the GAANN fellowship from the U.S. Department of Education.

References

Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., and Zhang, L. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 308–318. ACM, 2016.

Bassily, R., Smith, A. D., and Thakurta, A. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*, pp. 464–473, 2014. doi: 10.1109/FOCS.2014.56. URL https://doi.org/10.1109/FOCS.2014.56.

Chaudhuri, K., Monteleoni, C., and Sarwate, A. D. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12(Mar):1069–1109, 2011.

Dwork, C. and Roth, A. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.

Dwork, C., Kenthapadi, K., McSherry, F., Mironov, I., and Naor, M. Our data, ourselves: Privacy via distributed noise generation. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 486–503. Springer, 2006a.

Dwork, C., McSherry, F., Nissim, K., and Smith, A. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the Third Conference on Theory of Cryptography*, TCC'06, pp. 265–284, Berlin, Heidelberg, 2006b. Springer-Verlag. ISBN 3-540-32731-2, 978-3-540-32731-8. doi: 10.1007/11681878_14. URL http://dx.doi.org/10.1007/11681878_14.

Gaboardi, M., Arias, E. J. G., Hsu, J., Roth, A., and Wu, Z. S. Dual query: Practical private query release for high dimensional data. In *International Conference on Machine Learning*, pp. 1170–1178, 2014.

Iyengar, R., Near, J. P., Song, D., Thakkar, O., Thakurta, A., and Wang, L. Towards practical differentially private convex optimization. In *Towards Practical Differentially Private Convex Optimization*, pp. 0. IEEE, 2019.

Kasiviswanathan, S. P., Lee, H. K., Nissim, K., Raskhodnikova, S., and Smith, A. What can we learn privately? *SIAM Journal on Computing*, 40(3):793–826, 2011.

Kifer, D., Smith, A., and Thakurta, A. Private convex empirical risk minimization and high-dimensional regression. In *Conference on Learning Theory*, pp. 25–1, 2012.

Lichman, M. UCI machine learning repository, 2013. URL http://archive.ics.uci.edu/ml.

McSherry, F. and Talwar, K. Mechanism design via differential privacy. In *FOCS*, volume 7, pp. 94–103, 2007.

Neel, S., Roth, A., and Wu, Z. S. How to use heuristics for differential privacy. *arXiv preprint arXiv:1811.07765*, 2018.

Suggala, A. S. and Netrapalli, P. Online non-convex learning: Following the perturbed leader is optimal. *arXiv preprint arXiv:1903.08110*, 2019.

Vietri, G., Tian, G., Bun, M., Steinke, T., and Wu, Z. S. New oracle-efficient algorithms for private synthetic data release. In *International Conference on Machine Learning*, 2020.