Oracle Efficient Private Non-Convex Optimization

Seth Neel* Aaron Roth† Giuseppe Vietri‡ Zhiwei Steven Wu§

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, to date, analyses of this approach crucially rely on the convexity and smoothness of the objective function, limiting its generality. 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. Its privacy and accuracy guarantees hold even without assuming convexity. This gives an oracle-efficient optimization algorithm over arbitrary discrete domains that is comparable in its generality to the exponential mechanism. The second algorithm operates over a continuous domain and requires only that the loss function be bounded and Lipschitz in its continuous parameter. Its privacy analysis does not require convexity. Its accuracy analysis does require convexity, but does not require second order conditions like smoothness. Even without convexity, this algorithm can be generically used as an oracle-efficient optimization algorithm, with accuracy evaluated empirically. 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.

*Wharton Statistics Department, University of Pennsylvania. Email: sethneel93@gmail.com

†Department of Computer and Information Sciences, University of Pennsylvania. This material is based upon work supported by the United States Air Force and DARPA under Contract No FA8750-16-C-0022. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Air Force and DARPA. Email: aaroth@cis.upenn.edu.

‡Department of Computer Science and Engineering, University of Minnesota. Supported by the GAANN fellowship from the U.S. Department of Education. Email: vietr002@umn.edu

§Department of Computer Science and Engineering, University of Minnesota. Supported in part by a Google Faculty Research Award, a J.P. Morgan Faculty Award, a Mozilla research grant, and a Facebook Research Award. Email: zstevenwu@cmu.edu

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 is ubiquitous, and includes combinatorial optimization problems, empirical risk minimization problems, and synthetic data generation problems amongst others. 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 [MT07]. 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). It has indeed been used to solve private learning [KLN$^+$11], combinatorial optimization [GLM$^+$10], and synthetic data generation problems [BLR13] subject to differential privacy, often optimally. Unfortunately, the exponential mechanism is generally infeasible to run: its implementation (and the implementation of related mechanisms, like "Report-Noisy-Max" [DR14]) requires the ability to enumerate the parameter range $\mathcal{W}$, making it infeasible in most learning settings. 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 [CMS11] and objective perturbation [CMS11, KST12, INS$^+$19]. 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 $\mathcal{W}$, 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 [CMS11], and analyzed for the special case of strongly convex functions. Its analysis was subsequently improved and generalized [KST12, INS$^+$19] to apply to smooth convex functions, and to tolerate a small degree of error in the optimization procedure. Our paper is the first to give an analysis 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. [CMS11] 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 [NRW19], 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*. 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. They also can yield improved accuracy bounds in cases where both techniques apply: see Sections 3.1 and 5. [NRW19] 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 recently proven by [SN19] in the context of online learning.

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

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

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:

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

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:

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

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 [DMNS06, DKM$^+$06])**.** 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 $\mathcal{D}'$.

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

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

Output: $\hat{w}$

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.

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)

- $\Pr[\eta] \approx \Pr[g(\hat{w}, \eta)]$ (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 $Pr[\eta \in B] = 0$ and for any $\eta \in \mathbb{R}^{d+1} \setminus B$:*

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

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

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

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, $\Pr[\eta] \approx \Pr[g_{\hat{w}}(\eta)]$.

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 $Pr[\eta \in C^c] \geq 1 - \delta$, 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 $\eta \in \mathcal{E}(\mathcal{D}, \hat{w})$ be such that $\hat{w}$ is the unique minimizer $\hat{w} \in \inf_w L(\mathcal{D}, w) - \langle \eta, \pi(w) \rangle$. Then $g_{\hat{w}}(\eta) \in \mathcal{E}(\mathcal{D}', \hat{w})$. Hence:*

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

Proof. Let $c = \frac{4}{\tau} G D^2$. 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 $\mathcal{D}'$:

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

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 $L(\mathcal{D}', w) = L(\mathcal{D}, w) - l_i(w) + l_i'(w)$. Now, write the loss function in terms of $\mathcal{D}$ and rearranging terms:

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

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

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

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

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

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 $\|\hat{w} - v\|_2 \geq \tau$.

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

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

Putting the Lemmas together:

$$\begin{aligned}
\Pr[\mathcal{O}_\pi(\mathcal{D}, \eta) \in S] &= \Pr[\eta \in \bigcup_{\hat{w}} \mathcal{E}(\mathcal{D}, \hat{w})] \\
&= \int_{\mathbb{R}^{d+1}} p(\eta)\mathbb{I}\{\eta \in \bigcup_{\hat{w}} \mathcal{E}(\mathcal{D}, \hat{w})\}d\eta \\
&= \int_{(\mathbb{R}^{d+1}\setminus B)\setminus C} p(\eta)\mathbb{I}\{\eta \in \bigcup_{\hat{w}} \mathcal{E}(\mathcal{D}, \hat{w})\}d\eta + \int_C p(\eta)\mathbb{I}\{\eta \in \bigcup_{\hat{w}} \mathcal{E}(\mathcal{D}, \hat{w})\}d\eta \qquad (2) \\
&\leq \int_{(\mathbb{R}^{d+1}\setminus C)\setminus B} p(\eta)\mathbb{I}\{\eta \in \bigcup_{\hat{w}} \mathcal{E}(\mathcal{D}, \hat{w})\}d\eta + \delta \qquad (3) \\
&= \sum_{\hat{w}\in S} \int_{\mathbb{R}^{d+1}\setminus(C\cup B)} p(\eta)\mathbb{I}\{\eta \in \mathcal{E}(\mathcal{D}, \hat{w})\}d\eta + \delta \\
&\leq \sum_{\hat{w}\in S} \int_{\mathbb{R}^{d+1}\setminus(C\cup B)} p(\eta)\mathbb{I}\{g_{\hat{w}}(\eta) \in \mathcal{E}(\mathcal{D}', \hat{w})\}d\eta + \delta \qquad (4) \\
&\leq \sum_{\hat{w}\in S} \int_{\mathbb{R}^{d+1}\setminus(C\cup B)} e^\epsilon p(g_{\hat{w}}(\eta))\mathbb{I}\{g_{\hat{w}}(\eta) \in \mathcal{E}(\mathcal{D}', \hat{w})\}d\eta + \delta \qquad (5) \\
&= \sum_{\hat{w}\in S} \int_{\mathbb{R}^{d+1}\setminus(g_{\hat{w}}(C)\cup g_{\hat{w}}(B))} e^\epsilon p(\eta)\mathbb{I}\{\eta \in \mathcal{E}(\mathcal{D}', \hat{w})\}\left|\frac{\partial g_{\hat{w}}}{\partial \eta}\right|d\eta \qquad (6) \\
&\leq e^\epsilon \sum_{\hat{w}\in S} \int_{\mathbb{R}^{d+1}} p(\eta)\mathbb{I}\{\eta \in \mathcal{E}(\mathcal{D}', \hat{w})\}d\eta + \delta \\
&= e^\epsilon \Pr[\eta \in \bigcup_{\hat{w}} \mathcal{E}(\mathcal{D}', \hat{w})] \\
&= e^\epsilon \Pr[\mathcal{O}_\pi(\mathcal{D}', \eta) \in S] + \delta
\end{aligned}$$

where equality (2) follows from Lemma 2. Then inequality (3) holds because C is chosen such that $\Pr[\eta \in C] < \delta$. 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 [NRW19] 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 [NRW19])**.** *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. $I_\tau = [-1, -1+\tau, \ldots 0, \tau, 2\tau, \ldots 1]^d$. 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 [NRW19]. 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. The privacy analysis of this algorithm crucially depends on a stability lemma proven by [SN19] in the context of online learning, and does not require convexity.[1]

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*

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

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

Proof. For $\mu_i \sim \text{Lap}(\frac{\lambda}{\epsilon}), |\mu_i| \sim \text{Exp}(\frac{\lambda}{\epsilon})$. By Theorem 5.1 in [Jan17] which gives upper tail bounds for the sum of independent exponential random variables, we can conclude that $||\mu||_1 \leq r = (1 + \log(2/\beta))\frac{\lambda}{\epsilon}$ with probability $1 - \beta/2$.

Then by G-Lipschitzness with respect to the l_1 norm, with probability $1 - \beta/2$:

$$\frac{1}{n}\mathcal{L}\left(\frac{1}{m}\sum_k^m w_k + \mu, \mathcal{D}\right) \leq \frac{1}{n}\mathcal{L}\left(\frac{1}{m}\sum_k^m w_k, \mathcal{D}\right) + Gr$$

[1] Compared to the bound in [SN19], our bound has an additional factor of 2 since our neighboring relationship in Definition 2.5 is defined via replacement whereas in [SN19] the stability is defined in terms of adding another loss function.

Algorithm 2: OPSamp

Input: Approximate optimization oracle $\mathcal{O}_\alpha$, a dataset $\mathcal{D} = \{l_i\}_{i=1}^n$, privacy parameters ϵ, δ.

$\gamma \longleftarrow \frac{\sqrt{\epsilon}}{\sqrt{n}} d^{5/4}\sqrt{D_2}$;

$m \longleftarrow \frac{\ln(2d/\delta)}{2\gamma^2}$;

for $k = 1$ *to* m **do**

$\eta \longleftarrow \sqrt{\frac{D_2\sqrt{2d}\epsilon}{250G^2d^2D_\infty^2(1+\log(2/\beta))n}}$;

Sample a random vector $\sigma^k \sim \text{Exp}(\eta)^d$;

$$w_k \longleftarrow \mathcal{O}_\alpha\Big(\mathcal{D}, \sigma^k\Big)$$

end

$\lambda \longleftarrow 4D_\infty\gamma + 250\eta G d^2 D_\infty^2 + \frac{\alpha}{10G}$;

$\mu \sim \text{Lap}(\lambda/\epsilon)^d$;

Output: $\frac{1}{m}\sum_{k=1}^m w_k + \mu$

We now focus on $\frac{1}{n}\mathcal{L}(\frac{1}{m}\sum_k^m w_k, \mathcal{D})$. By the convexity of the loss functions, we have:

$$\frac{1}{n}\mathcal{L}\left(\frac{1}{m}\sum_k^m w_k, \mathcal{D}\right) \leq \frac{1}{m}\sum_k^m \frac{1}{n}\mathcal{L}\left(w_k, \mathcal{D}\right)$$

Since each $\frac{1}{n}\mathcal{L}(w_k, \mathcal{D})$ is bounded in $[0,1]$ (since each $l_i \in [0,1]$) and independent, by Hoeffding's Inequality (see Appendix) with probability $1 - \beta/2$:

$$\left|\frac{1}{m}\sum_k^m \frac{1}{n}\mathcal{L}(w_k, \mathcal{D}) - \mathbb{E}_{w^*}\left[\frac{1}{n}\mathcal{L}(w^*, \mathcal{D})\right]\right| \leq \sqrt{\frac{\log(4/\beta)}{2m}}$$

So it suffices to show that $\mathbb{E}_{w^*}[\frac{1}{n}\mathcal{L}(w^*, \mathcal{D})] - \arg\min_{w\in\mathcal{W}} \frac{1}{n}\mathcal{L}(w, \mathcal{D})$ is small. Fix $\tilde{w} = \arg\min_{w\in\mathcal{W}} \frac{1}{n}\mathcal{L}(w, \mathcal{D})$. Now by definition of $\mathcal{O}_\alpha$, for any $w \leftarrow \mathcal{O}_\alpha\Big(\sum_{i=1}^{|\mathcal{D}|} l_i - \sigma\Big)$, we have

$$\frac{1}{n}\mathcal{L}(w, \mathcal{D}) - \frac{1}{n}\langle w, \sigma\rangle \leq \frac{1}{n}\mathcal{L}(\tilde{w}, \mathcal{D}) - \frac{1}{n}\langle \tilde{w}, \sigma\rangle + \alpha,$$

hence

$$\frac{1}{n}\mathcal{L}(w, \mathcal{D}) - \frac{1}{n}\mathcal{L}(\tilde{w}, \mathcal{D}) \leq \frac{1}{n}\langle w - \tilde{w}, \sigma\rangle + \alpha$$

$\langle w - \tilde{w}, \sigma\rangle \leq ||w - \tilde{w}||_2||\sigma||_2 \leq D_2||\sigma||_2$, hence:

$$\mathbb{E}_{w^*}\left[\frac{1}{n}\mathcal{L}(w^*, \mathcal{D})\right] - \arg\min_{w\in\mathcal{W}} \frac{1}{n}\mathcal{L}(w, \mathcal{D}) \leq \frac{1}{n}D_2\mathbb{E}\left[||\sigma||_2\right]$$

Now by Jensen's inequality, $\mathbb{E}[||\sigma||_2] \leq \sqrt{\mathbb{E}[||\sigma||_2^2]} = \frac{\sqrt{2d}}{\eta}$, where the last equality is by the variance of the exponential distribution. Putting it all together, with probability $1 - \beta$:

$$\frac{1}{n}\mathcal{L}\left(\frac{1}{m}\sum_k^m w_k + \mu, \mathcal{D}\right) - \arg\min_{w\in W} \frac{1}{n}\mathcal{L}\left(w + \mu, \mathcal{D}\right) \leq Gr + \gamma\sqrt{\log(4/\beta)/2} + \alpha + \frac{1}{n}D_2\frac{\sqrt{2d}}{\eta},$$

Plugging in the value of r, λ and expanding we get the following long expression:

$$
\begin{aligned}
& G(1+\log(2/\beta))\frac{\lambda}{\epsilon}+\gamma\sqrt{\log(4/\beta)/2}+\alpha+\frac{1}{n}D_2\frac{\sqrt{2d}}{\eta} \\
&= G(1+\log(2/\beta))\frac{\Big(4D_\infty\gamma+250\eta Gd^2D_\infty^2+\frac{\alpha}{10G}\Big)}{\epsilon}+\gamma\sqrt{\log(4/\beta)/2}+\alpha+\frac{1}{n}D_2\frac{\sqrt{2d}}{\eta} \\
&= \gamma\Big(\frac{4GD_\infty(1+\log(2/\beta))}{\epsilon}\Big)+\eta\Big(\frac{250G^2d^2D_\infty^2(1+\log(2/\beta))}{\epsilon}\Big)+\alpha\Big(\frac{(1+\log(2/\beta))}{10\epsilon}\Big)+ \\
&\quad \gamma\Big(\sqrt{\log(4/\beta)/2}\Big)+\frac{1}{\eta}\Big(\frac{1}{n}D_2\sqrt{2d}\Big)+\alpha \\
&= \gamma A+\eta B+\alpha C+\gamma D+\frac{E}{\eta}+\alpha \qquad \text{(Setting placeholders } A,B,C,D,E) \\
&= \gamma(A+D)+\sqrt{BE}+\alpha(C+1) \qquad (\eta=\sqrt{\frac{E}{B}})
\end{aligned}
\tag{7}
$$

The last step of equation 7 comes from replacing in the value of $\eta=\sqrt{\frac{D_2\sqrt{2d}\epsilon}{250G^2d^2D_\infty^2(1+\log(2/\beta))n}}=\sqrt{\frac{E}{B}}$. Replacing back the values of A,B,C,D,E results in:

$$
\begin{aligned}
&= \gamma\Big(\frac{G(1+\log(2/\beta))4D_\infty}{\epsilon}+\sqrt{\log(4/\beta)/2}\Big)+\sqrt{\frac{250G^2d^2D_\infty^2D_2\sqrt{2d}(1+\log(2/\beta))}{\epsilon n}}+ \\
&\quad \alpha\Big(\frac{(1+\log(2/\beta))}{10\epsilon}+1\Big)
\end{aligned}
$$

Finally, note that by the choice of the parameter γ, the first term has order at most that of the second term, which gives our stated bound. □

Privacy analysis Before we prove that algorithm 2 satisfies differential-private in theorem 11, we give some useful lemmas.

Lemma 8 (Stability lemma [SN19])**.** *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,*

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

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 $0\le\gamma\le 1$, then, with probability $1-\delta/2$:*

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

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,*

$$||\mathcal{W}(\mathcal{D}, \Sigma) - \mathcal{W}(\mathcal{D}', \Sigma)||_1 \leq 4D_\infty\gamma + 250\eta G d^2 D_\infty^2 + \frac{\alpha}{10G} \tag{8}$$

Proof. By Lemma 9, If we run the approximate oracle $\frac{\ln 2d/\delta}{2\gamma^2}$ times on each neighboring dataset $\mathcal{D}, \mathcal{D}'$, then by union bound we get that with probability $1-\delta$:

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

Adding both inequalities and applying the triangle inequality

$$\begin{aligned}
&\|\mathcal{W}(\mathcal{D}) - \mathbb{E}[\mathcal{O}_\alpha(\mathcal{D}, \sigma)]\|_1 + \|\mathcal{W}(\mathcal{D}') - \mathbb{E}[\mathcal{O}_\alpha(\mathcal{D}', \sigma)]\|_1 \leq 4D_\infty\gamma \\
&\|\mathcal{W}(\mathcal{D}) - \mathbb{E}[\mathcal{O}_\alpha(\mathcal{D}, \sigma)] - \mathcal{W}(\mathcal{D}') + \mathbb{E}[\mathcal{O}_\alpha(\mathcal{D}', \sigma)]\|_1 \leq 4D_\infty\gamma \\
&\|\mathcal{W}(\mathcal{D}) - \mathcal{W}(\mathcal{D}')\|_1 \leq 4D_\infty\gamma + \|\mathbb{E}[\mathcal{O}_\alpha(\mathcal{D}, \sigma)] - \mathbb{E}[\mathcal{O}_\alpha(\mathcal{D}', \sigma)]\|_1
\end{aligned} \tag{9}$$

Lastly, by Lemma 8,

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

□

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 (8), which occurs with probability at least $1-\delta$. Then the privacy guarantee follows from the use of Laplace mechanism. □

Proof. First we introduce some notation. We denote by $\mathcal{W}(\mathcal{D}, \Sigma)$ the average of m runs of $\mathcal{O}_\alpha$ with dataset $\mathcal{D}$ and sequence of i.i.d noise vectors $\Sigma = \{\sigma_1, \sigma_2, \ldots, \sigma_m\}$, sampled i.i.d from the Exponential distribution. Let $\mathcal{M}(\mathcal{D})$ denote a random variable of algorithm 2's output on dataset $\mathcal{D}$. Given any realization of the objective-perturbation noise term Σ and the output-perturbation noise term μ, by an abuse of notation, we can write the output of algorithm 2 as:

$$\mathcal{M}(\mathcal{D}, \Sigma, \mu) = \mathcal{W}(\mathcal{D}, \Sigma) + \mu \tag{10}$$

Following Lemma 8, we let $\lambda \leftarrow 4D_\infty\gamma + 250\eta G d^2 D_\infty^2 + \frac{\alpha}{10G}$ and define the event B as

$$B = \{\Sigma \in \mathbb{R}^{(m,d)} : \|\mathcal{W}(\mathcal{D}, \Sigma) - \mathcal{W}(\mathcal{D}', \Sigma)\|_1 \leq \lambda\}$$

where λ is the ℓ_1-norm sensitivity bound from Lemma 10. Then, by the same lemma, if Σ is drawn independently from the Exponential distribution then the probability that $\Sigma \notin B$ is less than δ.

Now we are ready for the main argument. Fixing any two neighboring dataset $\mathcal{D}, \mathcal{D}'$ and any event S, we first consider the joint probability $\Pr[\mathcal{M}(\mathcal{D}) \in S \cap B]$, and write it as:

$$\Pr[\mathcal{M}(\mathcal{D}) \in S \cap B] = \Pr[\mathcal{M}(\mathcal{D}) \in S | B] \Pr[B] \tag{11}$$

For the next part of the proof, we let $p(\cdot)$ be the probability density functions of the Exponential distribution. We will upper bound the conditional probability $\Pr[\mathcal{M}(\mathcal{D}) \in S|B]$ using the differential privacy of the Laplace mechanism and the sensitivity of the function $\mathcal{W}(\mathcal{D}, \Sigma)$. First, note that if we fix $\Sigma \in B$, then the probability of $\mathcal{M}$ conditioned on Σ is

$$\Pr[\mathcal{M}(\mathcal{D}) \in S|\Sigma] = \Pr_{\mu \sim \text{Lap}(\lambda/\epsilon)}[\mathcal{W}(\mathcal{D}, \Sigma) + \mu \in S]$$

Furthermore, by Lemma 10, we have $\|\mathcal{W}(\mathcal{D}, \Sigma) - \mathcal{W}(\mathcal{D}', \Sigma)\|_1 \leq \lambda$ for any $\Sigma \in B$. And we know that the Laplace mechanism is ϵ-differentially private. Therefore we have

$$\Pr_{\mu \sim \text{Lap}(\lambda/\epsilon)}[\mathcal{W}(\mathcal{D}, \Sigma) + \mu \in S] \leq \exp(\epsilon)\Pr_{\mu \sim \text{Lap}(\lambda/\epsilon)}[\mathcal{W}(\mathcal{D}', \Sigma) + \mu \in S]$$

Putting the last two inequalities together we can upper bound $\Pr[\mathcal{M}(\mathcal{D}) \in S|B]$ by

$$\begin{aligned}
\Pr[\mathcal{M}(\mathcal{D}) \in S|B] &= \int_{\Sigma \in B} p(\Sigma)\Pr[\mathcal{M}(\mathcal{D}) \in S|\Sigma]d\Sigma \\
&= \int_{\Sigma \in B} p(\Sigma)\Pr_{\mu \sim \text{Lap}(\lambda/\epsilon)^d}[\mathcal{W}(\mathcal{D}, \Sigma) + \mu \in S]d\Sigma \\
&\leq \int_{\Sigma \in B} p(\Sigma)\exp(\epsilon)\Pr_{\mu \sim \text{Lap}(\lambda/\epsilon)^d}[\mathcal{W}(\mathcal{D}', \Sigma) + \mu \in S]d\Sigma \\
&= \exp(\epsilon)\Pr[\mathcal{M}(\mathcal{D}') \in S|B]
\end{aligned} \tag{12}$$

Finally,

$$\begin{aligned}
\Pr[\mathcal{M}(\mathcal{D}) \in S] &= \Pr[\mathcal{M}(\mathcal{D}) \in S \cap B] + \Pr[\mathcal{M}(\mathcal{D}) \in S \cap B^c] \\
&= \Pr[\mathcal{M}(\mathcal{D}) \in S \cap B] + \int_{\Sigma \in B^c} \Pr[\mathcal{M}(\mathcal{D})|\Sigma]\Pr[\Sigma] \\
&\leq \Pr[\mathcal{M}(\mathcal{D}) \in S \cap B] + \delta \\
&= \Pr[\mathcal{M}(\mathcal{D}) \in S|B]\Pr[B] + \delta \quad (eq.\ (11)) \\
&\leq \exp(\epsilon)\Pr[\mathcal{M}(\mathcal{D}') \in S|B]\Pr[B] + \delta \quad (eq.\ (12)) \\
&= \exp(\epsilon)\Pr[\mathcal{M}(\mathcal{D}') \in S \cap B] + \delta \\
&\leq \exp(\epsilon)\Pr[\mathcal{M}(\mathcal{D}') \in S] + \delta
\end{aligned} \tag{13}$$

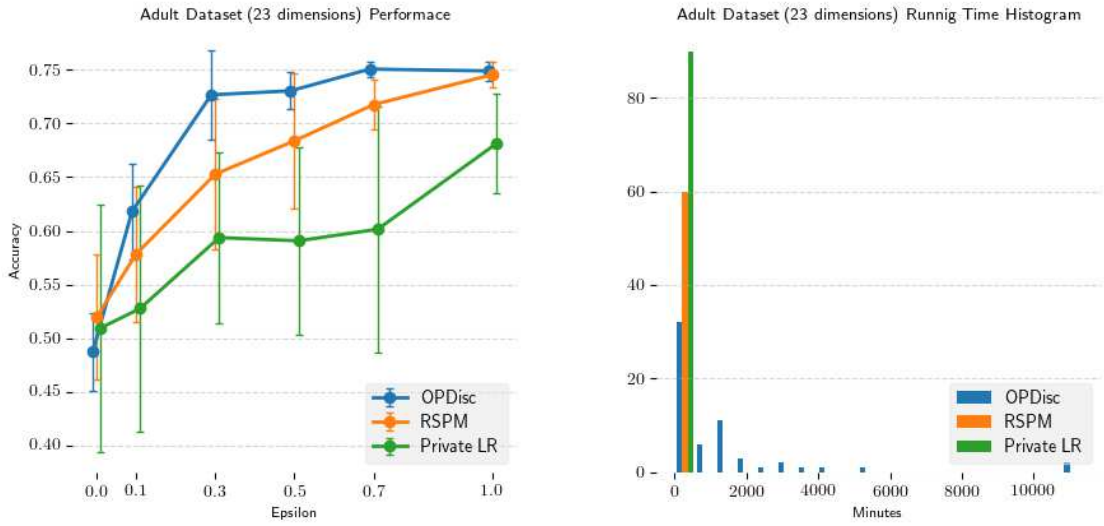
Therefore, $\Pr[\mathcal{M}(\mathcal{D}) \in S] \leq \exp(\epsilon)\Pr[\mathcal{M}(\mathcal{D}') \in S] + \delta$

□

5 Experiments

For our experiments, we consider the problem of privately learning a the 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.



(a) Accuracy versus ϵ.

(b) Distribution of run time.

Figure 1: 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.

Data description and pre-processing We use the Adult dataset [Lic13], 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 difficult 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 $l_i(w) = \log\left(1 + \exp(-y\langle w, x_i\rangle)\right)$. We implement a differentially private stochastic gradient descent (privateSGD) algorithm from [BST14, ACG+16], 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.

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: $l_i(w) = \mathbb{1}[y_i \neq \text{sgn}(\langle x_i, w \rangle)]$. 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. The source code for OPDisc is available via GitHub (`https://github.com/giusevtr/private_objective_perturbation`).

Empirical evaluation. We evaluate our algorithms by their (0/1) classification accuracy. The fig. 1a 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 fig. 1b 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.

References

[ACG+16] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 308–318. ACM, 2016.

[BLR13] Avrim Blum, Katrina Ligett, and Aaron Roth. A learning theory approach to noninteractive database privacy. *Journal of the ACM (JACM)*, 60(2):12, 2013.

[BST14] Raef Bassily, Adam D. Smith, and Abhradeep Thakurta. 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*, pages 464–473, 2014.

[CMS11] Kamalika Chaudhuri, Claire Monteleoni, and Anand D Sarwate. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12(Mar):1069–1109, 2011.

[DKM+06] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 486–503. Springer, 2006.

[DMNS06] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the Third Conference on Theory of Cryptography*, TCC'06, pages 265–284, Berlin, Heidelberg, 2006. Springer-Verlag.

[DR14] Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.

[GKS93] Sally A Goldman, Michael J Kearns, and Robert E Schapire. Exact identification of read-once formulas using fixed points of amplification functions. *SIAM Journal on Computing*, 22(4):705–726, 1993.

[GLM+10] Anupam Gupta, Katrina Ligett, Frank McSherry, Aaron Roth, and Kunal Talwar. Differentially private combinatorial optimization. In *Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms*, pages 1106–1125. Society for Industrial and Applied Mathematics, 2010.

[INS+19] Roger Iyengar, Joseph P Near, Dawn Song, Om Thakkar, Abhradeep Thakurta, and Lun Wang. Towards practical differentially private convex optimization. In *Towards Practical Differentially Private Convex Optimization*, page 0. IEEE, 2019.

[Jan17] Svante Janson. Tail bounds for sums of geometric and exponential variables. *arXiv e-prints*, page arXiv:1709.08157, Sep 2017.

[KLN+11] Shiva Prasad Kasiviswanathan, Homin K Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. What can we learn privately? *SIAM Journal on Computing*, 40(3):793–826, 2011.

[KST12] Daniel Kifer, Adam Smith, and Abhradeep Thakurta. Private convex empirical risk minimization and high-dimensional regression. In *Conference on Learning Theory*, pages 25–1, 2012.

[Lic13] M. Lichman. UCI machine learning repository, 2013.

[MT07] Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *Foundations of Computer Science (FOCS)*, volume 7, pages 94–103, 2007.

[NRW19] Seth Neel, Aaron Roth, and Zhiwei Steven Wu. How to use heuristics for differential privacy. In *Foundations of Computer Science (FOCS)*, 2019.

[SKS16] Vasilis Syrgkanis, Akshay Krishnamurthy, and Robert E. Schapire. Efficient algorithms for adversarial contextual learning. *CoRR*, abs/1602.02454, 2016.

[SN19] Arun Sai Suggala and Praneeth Netrapalli. Online non-convex learning: Following the perturbed leader is optimal. *arXiv preprint arXiv:1903.08110*, 2019.

A Definitions

Definition A.1 ([GKS93, SKS16])**.** A set $U \subseteq \mathcal{L}$ is a *separator set* for a parameter space $\mathcal{W}$ if for every pair of distinct parameters $w, w' \in \mathcal{W}$, there is an $l \in U$ such that:

$$l(w) \neq l(w')$$

If $|U| = m$, then we say that $\mathcal{W}$ has a separator set of size m.

Algorithm 3: Gaussian Report Separator perturbed Minimum [NRW19]

Given: A separator set $U = \{e_1, \ldots, e_m\}$ for class $\mathcal{W}_\tau$ and optimization oracle for $\mathcal{W}_\tau^*$;
Input: $\mathcal{D} = \{l_i\}_{i \in [n]}$
$n \leftarrow |\mathcal{D}|$;
$\sigma \leftarrow \frac{7\sqrt{m \ln 1/\delta}}{\epsilon}$;
Draw i.i.d random vector $\eta \sim \mathcal{N}(0, \sigma^2)^{d+1}$;
Construct a weighted dataset WD of size $n + m$ as follows:

$$WD(\mathcal{D}, \eta) = \{(l_i, 1) : l_i \in \mathcal{D}\} \cup \{(e_i, \eta_i) : e_i \in U\}$$

$$w \in \arg\min_{w^* \in \mathcal{W}_\tau} \sum_{(l_i, p_i) \in WD} p_i l_i(w)$$

Output: w

Definition A.2. A weighted optimization oracle for a class $\mathcal{W}$ is a function $\mathcal{O} : (\mathcal{L} \times \mathbb{R})^* \to \mathcal{W}$ that takes as input a weighted dataset $WD \in (\mathcal{L} \times \mathbb{R})^*$ and outputs $w \in \mathcal{W}$ such that

$$w \in \arg\min_{w^* \in \mathcal{W}} \sum_{(l_i, p_i) \in WD} p_i l_i(w)$$

B Missing Proofs in Section 3

Proof of Lemma 2.

Proof. Since $\mathcal{W}_\tau$ is a discrete space, by a union bound it suffices to show that for any pair $w \neq w' \in \mathcal{W}_\tau$, $\Pr\big[L(\mathcal{D}, w) - \langle \eta, \pi(w) \rangle = L(\mathcal{D}, w') - \langle \eta, \pi(w') \rangle\big] = 0$. Since $w \neq w'$, they must differ in at least one coordinate i. Condition on the realization of all of the coordinates of η but the i^{th}, η_{-i}. Then $L(\mathcal{D}, w) - \langle \eta, \pi(w) \rangle = L(\mathcal{D}, w') - \langle \eta, \pi(w') \rangle$, only if

$$\eta_i = \frac{L(\mathcal{D}, w') - L(\mathcal{D}, w) + \sum_{j \neq i} \eta_j (w_j - w_j)}{(w_i' - w_i)}$$

The expression on the righthand side is well-defined since $w_i \neq w_i'$. But then $\eta_i \sim \mathcal{N}(0, \sigma^2)$ even after conditioning on η_{-i}, and so its probability of taking any fixed value is 0. This proves the claim. □

Proof of Lemma 3.

Proof. Fix any $r \in \mathbb{R}^{d+1}$, $w \in \mathcal{W}_\tau$, and let $v = g_w(r) - r = \frac{2}{\gamma} GD^2 \pi(\hat{w})$. Note that $\|v\|_2 = \frac{2}{\tau} GD^2$. Fix an orthonormal basis of $\mathbb{R}^{d+1}$, where the first basis vector b_1 is parallel to v. Let $r^{[1]}$ be the projection of r onto the direction of b_1. Then by Lemma 17 in [NRW19]:

$$p(r) \leq \exp\left(\frac{1}{2\sigma^2}\Big(\|v\|_2^2 + 2\|v\|_2\|r^{[1]}\|_2\Big)\right) p(g_w(r)) \tag{14}$$

The ratio $p(r)/p(g_w(r))$ is bounded by $\exp(\epsilon)$ in the event that $\|r^{[1]}\|_2 < 2\sigma^2\epsilon/\|v\|_2 - \|v\|_2/2$. $||r^{[1]}||_2 \sim |\lambda|$, where $\lambda \sim \mathcal{N}(0, \sigma^2)$, and so using a tail bound for the χ^2 random variable, $\|r^{[1]}\|_2 < 2\sigma^2\epsilon/\|v\|_2 - \|v\|_2/2$ with probability $1 - \delta$ so long as $\sigma = \frac{c\|v\|_2\sqrt{\ln(1/\delta)}}{2\epsilon}$ for $c \geq 3.5$. Since we have that $\|v\|_2 = \frac{2}{\tau} GD^2$, for us it suffices to set $\sigma = \frac{7GD^2\sqrt{\ln(1/\delta)}}{2\epsilon\tau}$. Let $\Lambda = \sigma^2\epsilon/\|v\|_2 - \|v\|_2/2$ and define the set $C = \{\eta : \|\eta^{[1]}\|_2 > \Lambda\}$. Then since $\Pr[\eta \in C] < \delta$, we are done. □

Proof of Theorem 1.

Proof. Write $\hat{w} = \mathcal{O}_\pi(\mathcal{D}, \eta)$. We first want to show that there exists a mapping $g_{\hat{w}} : \mathbb{R}^{d+1} \to \mathbb{R}^{d+1}$ such that $\hat{w}$ is the parameter vector output on any neighboring dataset $\mathcal{D}'$ when the noise vector is realized as $g_{\hat{w}}(\eta)$: that is, $\hat{w} = \mathcal{O}_\pi(\mathcal{D}', g_{\hat{w}}(\eta))$. Let $S \subset \mathcal{W}_\tau$ be a subset of discrete parameters. If we can show that $\Pr[S] \approx \Pr[g_{\hat{w}}(S)]$, then the probability of outputting any particular w on input $\mathcal{D}$ should be close to the corresponding probability, on input $\mathcal{D}'$ as desired. Denote the set of of noise vectors that induce output w on dataset $\mathcal{D}$ by $\mathcal{E}(\mathcal{D}, w) = \{\eta : \mathcal{O}_\pi(\mathcal{D}, \eta) = w\}$. Define our mapping:

$$g_{\hat{w}}(\eta) = \eta + \frac{2}{\tau} GD^2 \pi(\hat{w})$$

We now use the 3 key Lemmas to finish the privacy proof. Putting it all together:

$$\begin{aligned}
\Pr[\mathcal{O}_\pi(\mathcal{D},\eta)\in S] &= \Pr[\eta\in\bigcup_{\hat{w}}\mathcal{E}(\mathcal{D},\hat{w})]\\
&= \int_{\mathbb{R}^{d+1}} p(\eta)\mathbb{1}\{\eta\in\bigcup_{\hat{w}}\mathcal{E}(\mathcal{D},\hat{w})\}d\eta\\
&= \int_{(\mathbb{R}^{d+1}\setminus B)\setminus C} p(\eta)\mathbb{1}\{\eta\in\bigcup_{\hat{w}}\mathcal{E}(\mathcal{D},\hat{w})\}d\eta + \int_C p(\eta)\mathbb{1}\{\eta\in\bigcup_{\hat{w}}\mathcal{E}(\mathcal{D},\hat{w})\}d\eta\\
&\le \int_{(\mathbb{R}^{d+1}\setminus C)\setminus B} p(\eta)\mathbb{1}\{\eta\in\bigcup_{\hat{w}}\mathcal{E}(\mathcal{D},\hat{w})\}d\eta + \delta \quad \text{(Lemma 2, } \Pr[\eta\in C]<\delta)\\
&= \sum_{\hat{w}\in S}\int_{\mathbb{R}^{d+1}\setminus(C\cup B)} p(\eta)\mathbb{1}\{\eta\in\mathcal{E}(\mathcal{D},\hat{w})\}d\eta + \delta\\
&\le \sum_{\hat{w}\in S}\int_{\mathbb{R}^{d+1}\setminus(C\cup B)} p(\eta)\mathbb{1}\{g_{\hat{w}}(\eta)\in\mathcal{E}(\mathcal{D}',\hat{w})\}d\eta + \delta \quad \text{(Lemma 4)}\\
&\le \sum_{\hat{w}\in S}\int_{\mathbb{R}^{d+1}\setminus(C\cup B)} \exp(\epsilon)p(g_{\hat{w}}(\eta))\mathbb{1}\{g_{\hat{w}}(\eta)\in\mathcal{E}(\mathcal{D}',\hat{w})\}d\eta + \delta \quad \text{(bounded ratio)}\\
&= \sum_{\hat{w}\in S}\int_{\mathbb{R}^{d+1}\setminus(g_{\hat{w}}(C)\cup g_{\hat{w}}(B))} \exp(\epsilon)p(\eta)\mathbb{1}\{\eta\in\mathcal{E}(\mathcal{D}',\hat{w})\}\left|\frac{\partial g_{\hat{w}}}{\partial\eta}\right|d\eta \quad (\eta\to g_{\hat{w}}(\eta))\\
&\le \exp(\epsilon)\sum_{\hat{w}\in S}\int_{\mathbb{R}^{d+1}} p(\eta)\mathbb{1}\{\eta\in\mathcal{E}(\mathcal{D}',\hat{w})\}d\eta + \delta\\
&= \exp(\epsilon)\Pr[\eta\in\bigcup_{\hat{w}}\mathcal{E}(\mathcal{D}',\hat{w})]\\
&= \exp(\epsilon)\Pr[\mathcal{O}_\pi(\mathcal{D}',\eta)\in S] + \delta
\end{aligned}$$

This completes the proof. □

C Missing Proofs in Section 4

Proof of Lemma 9

Proof. If we denote $w(\sigma)=\mathcal{O}_\alpha(\mathcal{D},\sigma)$ as the output of an approximate oracle on dataset $\mathcal{D}$ induced by a realization of the noise vector σ, then $w(\sigma^1),\dots w(\sigma^m)$ are m independent random variables with $-D\le w(\sigma^i)_j\le D$ for all i and for each coordinate $j\le d$.

For any index coordinate j, let $X_i=(w(\sigma^i)_j+D)/2D$, $S=\frac{1}{m}\sum_i^m X_i$ and $\mu_S=\mathbb{E}[S]$. Since $0\le X_i\le 1$, by Chernoff bound we have

$$\begin{aligned}
&\Pr[S>\mu_S+\gamma] < e^{-2m\gamma^2}\\
&\Pr[\frac{1}{m}\sum_i^m (w(\sigma^i)_j+D)/2D > \mu_S+\gamma] < e^{-2m\gamma^2}\\
&\Pr[\frac{1}{m}\sum_i^m w(\sigma^i)_j > 2D\mu_S - D + 2D\gamma] < e^{-2m\gamma^2}\\
&\Pr[\mathcal{W}(\mathcal{D},\Sigma)_j > \mathbb{E}_\sigma[\mathcal{O}_\alpha(\mathcal{D},\sigma)]_j + 2D\gamma] < e^{-2m\gamma^2}
\end{aligned}$$

Plugging in the value of $m = \frac{-\ln(\delta/(2d))}{2\gamma^2}$ we get:

$$\Pr[\mathcal{W}(\mathcal{D},\Sigma)_j - \mathbb{E}_\sigma[\mathcal{O}_\alpha(\mathcal{D},\sigma)]_j > 2D\gamma] < \delta/(2d)$$
$$\Pr[\mathcal{W}(\mathcal{D},\Sigma)_j - \mathbb{E}_\sigma[\mathcal{O}_\alpha(\mathcal{D},\sigma)]_j < -2D\gamma] < \delta/(2d)$$

Thus, by union bound

$$\Pr[\|\mathcal{W}(\mathcal{D},\Sigma) - \mathbb{E}[\mathcal{O}_\alpha(\mathcal{D},\sigma)]\|_1 > 2D\gamma] \leq \sum_{j=1}^{d} \Pr[|\mathcal{W}(\mathcal{D},\Sigma)_j - \mathbb{E}[\mathcal{O}_\alpha(\mathcal{D},\sigma)_j]| > 2D\gamma] < \sum_{j=1}^{d} \delta/(2d) = \delta/2$$

□

D Experiments Details

D.1 Implentation Details

The implementation is written in Python and uses Gurobi as a solver. We run the experiments on a server machine with an 8-core AMD processor and 192 GB of RAM.

D.2 Mixed Integer Programs for OPDisc and RSPM

We use a mixed integer programs (MIP) to encode the optimization problems of OPDisc and RSPM over the space of d-dimentional discrete halfspaces. The input to our algorithm is a dataset $\{(x_i, y_i)\}^n$ where $x_i \in \mathbb{R}^d$, $y \in \{-1, 1\}$ and a noise vector $\eta \in \mathbb{R}^{d+1}$. The discretization parameter is τ and D is the ℓ_2-norm bound of W.

$$\begin{aligned} \min_{w \in W} \quad & \sum_{i=1}^{n} e_i - \sum_{i=1}^{d} \eta_i w_i / D - \eta_{d+1}\lambda/D \\ \text{s.t.} \quad & y_i \sum_{j=1}^{d} w_j x_j + c e_i > 0 \quad \forall i \in [n] \\ & \lambda^2 + \|w\|_2^2 \leq D^2 \\ & e_i \in \{0,1\} \quad \forall i \in [n] \\ & w_j \in \tau\mathbb{Z} \quad \forall j \in [d] \end{aligned} \tag{15}$$

Figure 2: MIP oracle used by OPDisc. The MIP consist of n integral constraints, d linear and 1 quadratic constraint.

In OPDisc, the objective we want to minimize is $L(\mathcal{D}, w) - \langle \eta, \pi(w) \rangle$ which we can rewrite as

$$L(\mathcal{D}, w) - \sum_{i=1}^{d} \eta_i w_i / D - \eta_{d+1} \sqrt{D^2 - \|w\|_2^2}/D \tag{16}$$

The loss term $L(\mathcal{D}, w)$ in the objective is encoded as a sum of n binary variables $e_i \in \{0, 1\}$, such that if $e_i = 0$ only then the constraint $y_i \langle w, x_i \rangle > 0$ must be satisfied. Thus, the sum $\sum_{i=0}^{n} e_i$ is equal

to the number of misclassified samples. For each $i \in [n]$, we enconde the constraint corresponding to e_i in our MIP by the inequality $y_i\langle w, x_i\rangle + ce_i > 0$ where c is a large enough constant with $c > \max_{x,w} \|x\|_2\|w\|_2$. The third term in the objective function 16 is non-linear but we can express it as linear term in the objective by introducing the slack variable λ. Then, in order to force the condition that $\lambda = \sqrt{D^2 - \|w\|_2^2}$ we add the quadratic constraint $\lambda^2 + \|w\|_2^2 \leq D^2$.

$$\begin{aligned} \min_{w \in W} \quad & \sum_{i=1}^{n} e_i - \sum_{i=1}^{d} \eta_i w_i \\ \text{s.t.} \quad & y_i \sum_{j=1}^{d} w_j x_j + ce_i > 0 \quad \forall i \in [n] \\ & e_i \in \{0,1\} \quad \forall i \in [n] \\ & -1 \leq w_j \leq 1 \quad \forall j \in [d] \\ & w_j \in \tau\mathbb{Z} \quad \forall j \in [d] \end{aligned} \tag{17}$$

Figure 3: MIP oracle used by RSPM. The MIP consist of n integral constraints, and d linear constraint.

In RSPM, we are simply optimizing the 0-1 loss over the augmented data set, including the input data set as well as the weighted examples from the separator set.