Nearly-Optimal Private LASSO[*]

Kunal Talwar
Google Research
kunal@google.com

Abhradeep Thakurta
(Previously) Yahoo! Labs
guhathakurta.abhradeep@gmail.com

Li Zhang
Google Research
liqzhang@google.com

Abstract

We present a nearly optimal differentially private version of the well known LASSO estimator. Our algorithm provides privacy protection with respect to each training example. The excess risk of our algorithm, compared to the non-private version, is $\widetilde{O}(1/n^{2/3})$, assuming all the input data has bounded ℓ_∞ norm. This is the first differentially private algorithm that achieves such a bound without the polynomial dependence on p under no additional assumptions on the design matrix. In addition, we show that this error bound is nearly optimal amongst all differentially private algorithms.

1 Introduction

A common task in supervised learning is to select the model that best fits the data. This is frequently achieved by selecting a *loss function* that associates a real-valued loss with each datapoint d and model θ and then selecting from a class of admissible models, the model θ that minimizes the average loss over all data points in the training set. This procedure is commonly referred to as *Empirical Risk Minimization*(ERM).

The availability of large datasets containing sensitive information from individuals has motivated the study of learning algorithms that guarantee the privacy of individuals contributing to the database. A rigorous and by-now standard privacy guarantee is via the notion of differential privacy. In this work, we study the design of differentially private algorithms for Empirical Risk Minimization, continuing a long line of work. (See [2] for a survey.)

In particular, we study adding privacy protection to the classical LASSO estimator, which has been widely used and analyzed. We first present a differentially private optimization algorithm for the LASSO estimator. The algorithm is the combination of the classical Frank-Wolfe algorithm [15] and the exponential mechanism for guaranteeing the privacy [21]. We then show that our algorithm achieves nearly optimal risk among all the differentially private algorithms. This lower bound proof relies on recently developed techniques with roots in Cryptography [4, 14],

Consider the training dataset D consisting of n pairs of data $d_i = (x_i, y_i)$ where $x_i \in \mathbb{R}^p$, usually called the feature vector, and $y_i \in \mathbb{R}$, the prediction. The LASSO estimator, or the sparse linear regression, solves for $\theta^* = \mathsf{argmin}_\theta\, \mathcal{L}(\theta; d_i) = \frac{1}{n}\sum_i |x_i \cdot \theta - y_i|^2$ subject to $\|\theta\|_1 \le c$. To simplify presentation, we assume $c = 1$, but our results directly extend to general c. The ℓ_1 constraint tends to induce sparse θ^* so is widely used in the high dimensional setting when $p \gg n$. Here, we will study approximating the LASSO estimation with minimum possible error while protecting the privacy of each individual d_i. Below we define the setting more formally.

[*] Part of this work was done at Microsoft Research Silicon Valley Campus.

Problem definition: Given a data set $D = \{d_1, \cdots, d_n\}$ of n samples from a domain $\mathcal{D}$, a constraint set $\mathcal{C} \subseteq \mathbb{R}^p$, and a loss function $\mathcal{L} : \mathcal{C} \times \mathcal{D} \to \mathbb{R}$, for any model θ, define its excess empirical risk as

$$R(\theta; D) \overset{def}{=} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(\theta; d_i) - \min_{\theta \in \mathcal{C}} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(\theta; d_i). \tag{1}$$

For LASSO, the constraint set is the ℓ_1 ball, and the loss is the quadratic loss function. We define the *risk* of a mechanism $\mathcal{A}$ on a data set D as $R(\mathcal{A}; D) = \mathbb{E}[R(\mathcal{A}(D); D)]$, where the expectation is over the internal randomness of $\mathcal{A}$, and the risk $R(\mathcal{A}) = \max_{D \in \mathcal{D}^n} R(\mathcal{A}; D)$ is the maximum risk over all the possible data sets. Our objective is then to design a mechanism $\mathcal{A}$ which preserves (ϵ, δ)-differential privacy (Definition 1.3) and achieves as low risk as possible. We call the minimum achievable risk as *privacy risk*, defined as $\min_{\mathcal{A}} R(\mathcal{A})$, where the min is over all (ϵ, δ)-differentially private mechanisms $\mathcal{A}$.

There has been much work on studying the privacy risk for the LASSO estimator. However, all the previous results either need to make strong assumption about the input data or have polynomial dependence on the dimension p. First [20] and then [24] studied the LASSO estimator with differential privacy guarantee. They showed that one can avoid the polynomial dependence on p in the excess empirical risk if the data matrix X satisfy the restricted strong convexity and mutual incoherence properities. While such assumptions seem necessary to prove that LASSO recovers the exact support in the worst case, they are often violated in practice, where LASSO still leads to useful models. It is therefore desirable to design and analyze private versions of LASSO in the absence of such assumptions. In this work, we do so by analyzing the loss achieved by the private optimizer, compared to the true optimizer.

We make primarily two contributions in this paper. First we present an algorithm that achieves the privacy risk of $\widetilde{O}(1/n^{2/3})$ for the LASSO problem[1]. Compared to the previous work, we only assume that the input data has bounded ℓ_∞ norm. In addition, the above risk bound only has logarithmic dependence on p, which fits particularly well for LASSO as we usually assume $n \ll p$ when applying LASSO. This bound is achieved by a private version of the Frank-Wolfe algorithm. Assuming that each data point d_i satisfies that $\|d_i\|_\infty \leq 1$, we have

Theorem 1.1. *There exists an (ϵ, δ)-differentially private algorithm $\mathcal{A}$ for LASSO such that*

$$R(\mathcal{A}) = O\left(\frac{\log(np)\sqrt{\log(1/\delta)}}{(n\epsilon)^{2/3}}\right).$$

Our second contribution is to show that, surprisingly, this simple algorithm gives a nearly tight bound. We show that this rather unusual $n^{-2/3}$ dependence is not an artifact of the algorithm or the analysis, but is in fact the right dependence for the LASSO problem: no differentially private algorithm can do better! We prove a lower bound by employing fingerprinting codes based techniques developed in [4, 14].

Theorem 1.2. *For the sparse linear regression problem where $\|x_i\|_\infty \leq 1$, for $\epsilon = 0.1$ and $\delta = o(1/n^2)$, any (ϵ, δ)-differentially private algorithm $\mathcal{A}$ must have*

$$R(\mathcal{A}) = \Omega(1/(n \log n)^{2/3}).$$

Our improved privacy risk crucially depends on the fact that the constraint set is a polytope with few (polynomial in dimensions) vertices. This allows us to use a private version of the Frank-Wolfe algorithm, where at each step, we use the exponential mechanism to select one of the vertices of the polytope. We also present a variant of Frank-Wolfe that uses objective perturbation instead of the exponential mechanism. We show that (Theorem 2.6) we can obtain a risk bound dependent on the *Gaussian width* of the constraint set, which often results in tighter bounds compared to bounds based, e.g., on diameter. While more general, this variant adds much more noise than the Frank-Wolfe based algorithm, as it is effectively publishing the whole gradient at each step. When $\mathcal{C}$ is not a polytope with a small number of vertices, one can still use the exponential mechanism as long as one has a small list of candidate points which contains an approximate optimizer for every direction. For many simple cases, for example the ℓ_q ball with $1 < q < 2$, the bounds attained in this way have

[1] Throughout the paper, we use $\widetilde{O}$ to hide logarithmic factors.

an additional polynomial dependence on the dimension p, instead of the logarithmic dependence in the above result. For example, when $q = 1$, the upper bound from this variant has an extra factor of $p^{1/3}$. Whereas such a dependence is provably needed for $q = 2$, the upper bound jump rather abruptly from the logarithmic dependence for $q = 1$ to a polynomial dependence on p for $q > 1$. We leave open the question of resolving this discontinuity and interpolating more smoothly between the ℓ_1 case and the ℓ_2 case.

Our results enlarge the set of problems for which privacy comes "for free". Given n samples from a distribution, suppose that θ^* is the empirical risk minimizer and θ^{priv} is the differentially private approximate minimizer. Then the non-private ERM algorithm outputs θ^* and incurs expected (on the distribution) loss equal to the $\mathsf{loss}(\theta^*, \text{training-set}) + \text{generalization-error}$, where the *generalization error* term depends on the loss function, $\mathcal{C}$ and on the number of samples n. The differentially private algorithm incurs an additional loss of the privacy risk. If the privacy risk is asymptotically no larger than the generalization error, we can think of privacy as coming for free, since under the assumption of n being large enough to make the generalization error small, we are also making n large enough to make the privacy risk small. In the case when $\mathcal{C}$ is the ℓ_1-ball, and the loss function is the squared loss with $\|x\|_\infty \leq 1$ and $|y| \leq 1$, the best known generalization error bounds dominate the privacy risk when $n = \omega(\log^3 p)$ [1, Theorem 18].

1.1 Related work

There have been much work on private LASSO or more generally private ERM algorithms. The error bounds mainly depend on the shape of the constraint set and the Lipschitz condition of the loss function. Here we will summarize these related results. Related to our results, we distinguish two settings: i) the constraint set is bounded in the ℓ_1-norm and the the loss function is 1-Lipschitz in the ℓ_1-norm. (call it the (ℓ_1/ℓ_∞)-setting). This is directly related to our bounds on LASSO; and ii) the constraint set has bounded ℓ_2 norm and the loss function is 1-Lipschitz in the ℓ_2 norm (the (ℓ_2/ℓ_2)-setting), which is related to our bounds using Gaussian width.

The* (ℓ_1/ℓ_∞)*-setting: The results in this setting include [20, 24, 19, 25]. The first two works make certain assumptions about the instance (*restricted strong convexity* (RSC) and *mutual incoherence*). Under these assumptions, they obtain privacy risk guarantees that depend logarithmically in the dimensions p, and thus allowing the guarantees to be meaningful even when $p \gg n$. In fact their bound of $O(\text{polylog}\, p/n)$ can be better than our *tight* bound of $O(\text{polylog}\, p/n^{2/3})$. However, these assumptions on the data are strong and may not hold in practice. Our guarantees do not require any such data dependent assumptions. The result of [19] captures the scenario when the constraint set $\mathcal{C}$ is the probability simplex and the loss function is a generalized linear model, but provides a *worse* bound of $O(\text{polylog}\, p/n^{1/3})$. For the special case of *linear loss functions*, which are interesting primarily in the online prediction setting, the techniques of [19, 25] provide a bound of $O(\text{polylog}\, p/n)$.

The* (ℓ_2/ℓ_2)*-setting: In all the works on private convex optimization that we are aware of, either the excess risk guarantees depend polynomially on the dimensionality of the problem (p), or assumes special structure to the loss (e.g., generalized linear model [19] or linear losses [25]). Similar dependence is also present in the online version of the problem [18, 26]. [2] recently show that in the private ERM setting, in general this polynomial dependence on p is unavoidable. In our work we show that one can replace this dependence on p with the Gaussian width of the constraint set $\mathcal{C}$, which can be much smaller.

Effect of Gaussian width in risk minimization: Our result on general $\mathcal{C}$ has an dependence on the Gaussian width of $\mathcal{C}$. This geometric concept has previously appeared in other contexts. For example, [1] bounds the the excess generalization error by the Gaussian width of the constraint set $\mathcal{C}$. Recently [5] show that the Gaussian width of a constraint set $\mathcal{C}$ is very closely related to the number of generic linear measurements one needs to perform to recover an underlying model $\theta^* \in \mathcal{C}$. The notion of Gaussian width has also been used by [22, 11] in the context of differentially private query release mechanisms but in the very different context of answering multiple linear queries over a database.

1.2 Background

Differential Privacy: The notion of differential privacy (Definition 1.3) is by now a defacto standard for statistical data privacy [10, 12]. One of the reasons why differential privacy has become so popular is because it provides meaningful guarantees even in the presence of arbitrary auxiliary information. At a semantic level, the privacy guarantee ensures that *an adversary learns almost the same thing about an individual independent of his presence or absence in the data set.* The parameters (ϵ, δ) quantify the amount of information leakage. For reasons beyond the scope of this work, $\epsilon \approx 0.1$ and $\delta = 1/n^{\omega(1)}$ are a good choice of parameters. Here n refers to the number of samples in the data set.

Definition 1.3. *A randomized algorithm $\mathcal{A}$ is (ϵ, δ)-differentially private if, for all neighboring data sets $\mathcal{D}$ and $\mathcal{D}'$ (i.e., they differ in one record, or equivalently, $d_H(D, D') = 1$) and for all events S in the output space of $\mathcal{A}$, we have*

$$\Pr(\mathcal{A}(\mathcal{D}) \in S) \leq e^{\epsilon} \Pr(\mathcal{A}(\mathcal{D}') \in S) + \delta\,.$$

Here $d_H(D, D')$ refers to the Hamming distance.

ℓ_q-norm, $q \geq 1$: For $q \geq 1$, the ℓ_q-norm for any vector $v \in \mathbb{R}^p$ is defined as $\left(\sum\limits_{i=1}^{p} v(i)^q\right)^{1/q}$, where $v(i)$ is the i-th coordinate of the vector v.

L-Lipschitz continuity w.r.t. norm $\|\cdot\|$: A function $\Psi : \mathcal{C} \to \mathbb{R}$ is L-Lispchitz within a set $\mathcal{C}$ w.r.t. a norm $\|\cdot\|$ if the following holds.

$$\forall \theta_1, \theta_2 \in \mathcal{C}, |\Psi(\theta_1) - \Psi(\theta_2)| \leq L \cdot \|\theta_1 - \theta_2\|.$$

Gaussian width of a set $\mathcal{C}$: Let $b \sim \mathcal{N}(0, \mathbb{I}_p)$ be a Gaussian random vector in $\mathbb{R}^p$. The Gaussian width of a set $\mathcal{C}$ is defined as $G_{\mathcal{C}} \overset{def}{=} \mathbb{E}_b \left[\sup\limits_{w \in \mathcal{C}} |\langle b, w \rangle|\right]$.

2 Private Convex Optimization by Frank-Wolfe algorithm

In this section we analyze a differentially private variant of the classical Frank-Wolfe algorithm [15]. We show that for the setting where the constraint set $\mathcal{C}$ is a polytope with k vertices, and the loss function $\mathcal{L}(\theta; d)$ is Lipschitz w.r.t. the ℓ_1-norm, one can obtain an excess privacy risk of roughly $O(\log k/n^{2/3})$. This in particular captures the high-dimensional linear regression setting. One such example is the classical LASSO algorithm[27], which computes $\mathsf{argmin}_{\theta:\|\theta\|_1 \leq 1} \frac{1}{n}\|X\theta - y\|_2^2$. In the usual case of $|x_{ij}|, |y_j| = O(1)$, $\mathcal{L}(\theta) = \frac{1}{n}\|X\theta - y\|_2^2$ is $O(1)$-Lipschitz with respect to ℓ_1-norm, we show that one can achieve the nearly optimal privacy risk of $\widetilde{O}(1/n^{2/3})$.

The Frank-Wolfe algorithm [15] can be regarded as a "greedy" algorithm which moves towards the optimum solution in the first order approximation (see Algorithm 1 for the description). How fast Frank-Wolfe algorithm converges depends on $\mathcal{L}$'s "curvature", defined as follows according to [8, 17]. We remark that a β-smooth function on $\mathcal{C}$ has curvature constant bounded by $\beta\|C\|^2$.

Definition 2.1 (Curvature constant)**.** *For $\mathcal{L} : \mathcal{C} \to \mathbb{R}$, define $\Gamma_{\mathcal{L}}$ as below.*

$$\Gamma_{\mathcal{L}} := \sup_{\theta_1, \theta_2, \in \mathcal{C}, \gamma \in (0,1], \theta_3 = \theta_1 + \gamma(\theta_2 - \theta_1)} \frac{2}{\gamma^2}\left(\mathcal{L}(\theta_3) - \mathcal{L}(\theta_1) - \langle \theta_3 - \theta_1, \bigtriangledown \mathcal{L}(\theta_1)\rangle\right).$$

Remark 1. A useful bound can be derived for a quadratic loss $\mathcal{L}(\theta) = \theta A^T A \theta + \langle b, \theta \rangle$. In this case, by [8], $\Gamma_{\mathcal{L}} \leq \max_{a,b \in A \cdot \mathcal{C}} \|a - b\|_2^2$. When $\mathcal{C}$ is centrally symmetric, we have the bound $\Gamma_{\mathcal{L}} \leq 4 \max_{\theta \in \mathcal{C}} \|A\theta\|_2^2$. For LASSO, $A = \frac{1}{\sqrt{n}} X$.

Define $\theta^* = \underset{\theta \in \mathcal{C}}{\mathsf{argmin}}\, \mathcal{L}(\theta)$. The following theorem bounds the convergence rate of Frank-Wolfe algorithm.

Algorithm 1 Frank-Wolfe algorithm

```
Input: $\mathcal{C} \subseteq \mathbb{R}^p$, $\mathcal{L} : \mathcal{C} \to \mathbb{R}$, $\mu_t$
1: Choose an arbitrary $\theta_1$ from $\mathcal{C}$
2: for $t = 1$ to $T - 1$ do
3:     Compute $\widetilde{\theta}_t = \mathsf{argmin}_{\theta \in \mathcal{C}} \langle \triangledown \mathcal{L}(\theta_t), (\theta - \theta_t) \rangle$
4:     Set $\theta_{t+1} = \theta_t + \mu_t(\widetilde{\theta}_t - \theta_t)$
5: return $\theta_T$.
```

Theorem 2.2 ([8, 17])**.** *If we set $\mu_t = 2/(t+2)$, then $\mathcal{L}(\theta_T) - \mathcal{L}(\theta^*) = O(\Gamma_{\mathcal{L}}/T)$.*

While the Frank-Wolfe algorithm does not necessarily provide faster convergence compared to the gradient-descent based method, it has two major advantages. First, on Line 3, it reduces the problem to solving a minimization of linear function. When $\mathcal{C}$ is defined by small number of vertices, e.g. when $\mathcal{C}$ is an ℓ_1 ball, the minimization can be done by checking $\langle \triangledown \mathcal{L}(\theta_t), x \rangle$ for each vertex x of $\mathcal{C}$. This can be done efficiently. Secondly, each step in Frank-Wolfe takes a convex combination of θ_t and $\widetilde{\theta}_t$, which is on the boundary of $\mathcal{C}$. Hence each intermediate solution is always inside $\mathcal{C}$ (sometimes called *projection free*), and the final outcome θ_T is the convex combination of up to T points on the boundary of $\mathcal{C}$ (or vertices of $\mathcal{C}$ when $\mathcal{C}$ is a polytope). Such outcome might be desired, for example when $\mathcal{C}$ is a polytope, as it corresponds to a sparse solution. Due to these reasons Frank-Wolfe algorithm has found many applications in machine learning [23, 16, 8]. As we shall see below, these properties are also useful for obtaining low risk bounds for their private version.

2.1 Private Frank-Wolfe Algorithm

We now present a private version of the Frank-Wolfe algorithm. The algorithm accesses the private data only through the loss function in step 3 of the algorithm. Thus to achieve privacy, it suffices to replace this step by a private version.

To do so, we apply the exponential mechanism [21] to select an approximate optimizer. In the case when the set $\mathcal{C}$ is a polytope, it suffices to optimize over the vertices of $\mathcal{C}$ due to the following basic fact:

Fact 2.3. *Let $\mathcal{C} \subseteq \mathbb{R}^p$ be the convex hull of a compact set $S \subseteq \mathbb{R}^p$. For any vector $v \in \mathbb{R}^p$, $\arg\min\limits_{\theta \in \mathcal{C}} \langle \theta, v \rangle \cap S \neq \emptyset$.*

Thus it suffices to run the exponential mechanism to select θ_{t+1} from amongst the vertices of $\mathcal{C}$. This leads to a differentially private algorithm with risk logarithmically dependent on $|S|$. When $|S|$ is polynomial in p, it leads to an error bound with $\log p$ dependence. We can bound the error in terms of the ℓ_1-Lipschitz constant, which can be much smaller than the ℓ_2-Lipschitz constant. In particular, as we show in the next section, the private Frank-Wolfe algorithm is nearly optimal for the important high-dimensional sparse linear regression problem.

Algorithm 2 $\mathcal{A}_{\mathsf{Noise-FW(polytope)}}$: Differentially Private Frank-Wolfe Algorithm (Polytope Case)

```
Input: Data set: $\mathcal{D} = \{d_1, \cdots, d_n\}$, loss function: $\mathcal{L}(\theta; D) = \frac{1}{n}\sum\limits_{i=1}^{n} \mathcal{L}(\theta; d_i)$ (with $\ell_1$-Lipschitz
    constant $L_1$ for $\mathcal{L}$), privacy parameters: $(\epsilon, \delta)$, convex set: $\mathcal{C} = conv(S)$ with $\|\mathcal{C}\|_1$ denoting
    $\max_{s \in S} \|s\|_1$.
1: Choose an arbitrary $\theta_1$ from $\mathcal{C}$
2: for $t = 1$ to $T - 1$ do
3:     $\forall s \in S, \alpha_s \leftarrow \langle s, \triangledown \mathcal{L}(\theta_t; D) \rangle + \mathsf{Lap}\left(\frac{L_1 \|\mathcal{C}\|_1 \sqrt{8T \log(1/\delta)}}{n\epsilon}\right)$, where $\mathsf{Lap}(\lambda) \sim \frac{1}{2\lambda} e^{-|x|/\lambda}$.
4:     $\widetilde{\theta}_t \leftarrow \arg\min\limits_{s \in S} \alpha_s$.
5:     $\theta_{t+1} \leftarrow (1 - \mu_t)\theta_t + \mu_t \widetilde{\theta}_t$, where $\mu_t = \frac{2}{t+2}$.
6: Output $\theta^{priv} = \theta_T$.
```

Theorem 2.4 (Privacy guarantee)**.** *Algorithm 2 is (ϵ, δ)-differentially private.*

Since each data item is assumed to have bounded ℓ_∞ norm, for two neighboring databases D and D' and any $\theta \in \mathcal{C}$, $s \in S$, we have that

$$|\langle s, \bigtriangledown\mathcal{L}(\theta; D)\rangle - \langle s, \bigtriangledown\mathcal{L}(\theta; D)\rangle| = O(L_1\|\mathcal{C}\|_1/n)\,.$$

The proof of privacy then follows from a straight-forward application of the exponential mechanism [21] or its noisy maximum version [3, Theorem 5]) and the strong composition theorem [13]. In Theorem 2.5 we prove the utility guarantee for the private Frank-Wolfe algorithm for the convex polytope case. Define $\Gamma_{\mathcal{L}} = \max\limits_{D\in\mathcal{D}} C_{\mathcal{L}}$ over all the possible data sets in $\mathcal{D}$.

Theorem 2.5 (Utility guarantee)**.** *Let L_1, S and $\|\mathcal{C}\|_1$ be defined as in Algorithms 2 (Algorithm $\mathcal{A}_{\mathsf{Noise-FW(polytope)}}$). Let $\Gamma_{\mathcal{L}}$ be an upper bound on the curvature constant (defined in Definition 2.1) for the loss function $\mathcal{L}(\cdot; d)$ that holds for all $d \in \mathcal{D}$. In Algorithm $\mathcal{A}_{\mathsf{Noise-FW(polytope)}}$, if we set $T = \frac{{\Gamma_{\mathcal{L}}}^{2/3}(n\epsilon)^{2/3}}{(L_1\|\mathcal{C}\|_1)^{2/3}}$, then*

$$\mathbb{E}\left[\mathcal{L}(\theta^{priv}; D)\right] - \min_{\theta\in\mathcal{C}} \mathcal{L}(\theta; D) = O\left(\frac{{\Gamma_{\mathcal{L}}}^{1/3}\left(L_1\|\mathcal{C}\|_1\right)^{2/3}\log(n|S|)\sqrt{\log(1/\delta)}}{(n\epsilon)^{2/3}}\right).$$

Here the expectation is over the randomness of the algorithm.

The proof of utility uses known bounds on noisy Frank-Wolfe [17], along with error bounds for the exponential mechanism. The details can be found in the full version.

General $\mathcal{C}$ While a variant of this mechanism can be applied to the case when $\mathcal{C}$ is not a polytope, its error would depend on the size of a cover of the boundary of $\mathcal{C}$, which can be exponential in p, leading to an error bound with polynomial dependence on p. In the full version, we analyze another variant of private Frank-Wolfe that uses objective perturbation to ensure privacy. This variant is well-suited for a general convex set $\mathcal{C}$ and the following result, proven in the Appendix, bounds its excess risk in terms of the Gaussian Width of $\mathcal{C}$. For this mechanism, we only need $\mathcal{C}$ to be bounded in ℓ_2 diameter, but our error now depends on the ℓ_2-Lipschitz constant of the loss functions.

Theorem 2.6. *Suppose that each loss function is L_2-Lipschitz with respect to the ℓ_2 norm, and that $\mathcal{C}$ has ℓ_2 diameter at most $\|\mathcal{C}\|_2$. Let $G_{\mathcal{C}}$ the Gaussian width of the convex set $\mathcal{C} \subseteq \mathbb{R}^p$, and let $\Gamma_{\mathcal{L}}$ be the curvature constant (defined in Definition 2.1) for the loss function $\ell(\theta; d)$ for all $\theta \in \mathcal{C}$ and $d \in \mathcal{D}$. Then there is an (ϵ, δ)-differentially private algorithm $\mathcal{A}_{\mathsf{Noise-FW}}$ with excess empirical risk:*

$$\mathbb{E}\left[\mathcal{L}(\theta^{priv}; D)\right] - \min_{\theta\in\mathcal{C}} \mathcal{L}(\theta; D) = O\left(\frac{{\Gamma_{\mathcal{L}}}^{1/3}\left(L_2 G_{\mathcal{C}}\right)^{2/3}\log^2(n/\delta)}{(n\epsilon)^{2/3}}\right).$$

Here the expectation is over the randomness of the algorithm.

2.2 Private LASSO algorithm

We now apply the private Frank-Wolfe algorithm $\mathcal{A}_{\mathsf{Noise-FW(polytope)}}$ to the important case of the sparse linear regression (or LASSO) problem.

Problem definition: Given a data set $D = \{(x_1, y_1), \cdots, (x_n, y_n)\}$ of n-samples from the domain $D = \{(x, y) : x \in \mathbb{R}^p, y \in [-1, 1], \|x\|_\infty \leq 1\}$, and the convex set $\mathcal{C} = \ell_1^p$. Define the mean squared loss,

$$\mathcal{L}(\theta; D) = \frac{1}{n}\sum_{i\in[n]}\left(\langle x_i, \theta\rangle - y_i\right)^2. \tag{2}$$

The objective is to compute $\theta^{priv} \in \mathcal{C}$ to minimize $\mathcal{L}(\theta; D)$ while preserving privacy with respect to any change of individual (x_i, y_i) pair. The non-private setting of the above problem is a variant of the least squares problem with ℓ_1 regularization, which was started by the work of LASSO [27, 28] and intensively studied in the past years.

Since the ℓ_1 ball is the convex hull of $2p$ vertices, we can apply the private Frank-Wolfe algorithm $\mathcal{A}_{\mathsf{Noise-FW(polytope)}}$. For the above setting, it is easy to check that the ℓ_1-Lipschitz constant is bounded by $O(1)$. Further, by applying the bound on quadratic programming Remark 1, we have that $C_{\mathcal{L}} \leq 4\max_{\theta\in\mathcal{C}} \frac{1}{n}\|X\theta\|_2^2 = O(1)$ since $\mathcal{C}$ is the unit ℓ_1 ball, and $|x_{ij}| \leq 1$. Hence $\Gamma = O(1)$. Now applying Theorem 2.5, we have

Corollary 2.7. *Let $D = \{(x_1, y_1), \cdots, (x_n, y_n)\}$ of n samples from the domain $\mathcal{D} = \{(x, y) : \|x\|_\infty \le 1, |y| \le 1\}$, and the convex set $\mathcal{C}$ equal to the ℓ_1-ball. The output θ^{priv} of Algorithm $\mathcal{A}_{\mathsf{Noise-FW(polytope)}}$ ensures the following.*

$$\mathbb{E}[\mathcal{L}(\theta^{priv}; D) - \min_{\theta \in \mathcal{C}} \mathcal{L}(\theta; D)] = O\left(\frac{\log(np/\delta)}{(n\epsilon)^{2/3}}\right).$$

Remark 2. Compared to the previous work [20, 24], the above upper bound makes no assumption of *restricted strong convexity* or *mutual incoherence*, which might be too strong for realistic settings. Also our results significantly improve bounds of [19], from $\tilde{O}(1/n^{1/3})$ to $\tilde{O}(1/n^{2/3})$, which considered the case of the set $\mathcal{C}$ being the probability simplex and the loss being a generalized linear model.

3 Optimality of Private LASSO

In the following, we shall show that to ensure privacy, the error bound in Corollary 2.7 is nearly optimal in terms of the dominant factor of $1/n^{2/3}$.

Theorem 3.1 (Optimality of private Frank-Wolfe). *Let $\mathcal{C}$ be the ℓ_1-ball and $\mathcal{L}$ be the mean squared loss in equation (2). For every sufficiently large n, for every (ϵ, δ)-differentially private algorithm $\mathcal{A}$, with $\epsilon \le 0.1$ and $\delta = o(1/n^2)$, there exists a data set $D = \{(x_1, y_1), \cdots, (x_n, y_n)\}$ of n samples from the domain $\mathcal{D} = \{(x, y) : \|x\|_\infty \le 1, |y| \le 1\}$ such that*

$$\mathbb{E}[\mathcal{L}(\mathcal{A}(D); D) - \min_{\theta \in \mathcal{C}} \mathcal{L}(\theta; D)] = \widetilde{\Omega}\left(\frac{1}{n^{2/3}}\right).$$

We prove the lower bound by following the fingerprinting codes argument of [4] for lowerbounding the error of (ϵ, δ)-differentially private algorithms. Similar to [4] and [14], we start with the following lemma which is implicit in [4].The matrix X in Theorem 3.2 is the padded Tardos code used in [14, Section 5]. For any matrix X, denote by $X_{(i)}$ the matrix obtained by removing the i-th row of X. Call a column of a matrix a *consensus* column if the entries in the column are either all 1 or all -1. The sign of a consensus column is simply the consensus value of the column. Write $w = m/\log m$ and $p = 1000m^2$. The following theorem follows immediately from the proof of Corollary 16 in [14].

Theorem 3.2. *[Corollary 16 from [14], restated] Let m be a sufficiently large positive integer. There exists a matrix $X \in \{-1, 1\}^{(w+1)\times p}$ with the following property. For each $i \in [1, w+1]$, there are at least $0.999p$ consensus columns W_i in each $X_{(i)}$. In addition, for algorithm $\mathcal{A}$ on input matrix $X_{(i)}$ where $i \in [1, w+1]$, if with probability at least $2/3$, $\mathcal{A}(X_{(i)})$ produces a p-dimensional sign vector which agrees with at least $\frac{3}{4}p$ columns in W_i, then $\mathcal{A}$ is not (ε, δ) differentially private with respect to single row change (to some other row in X).*

Write $\tau = 0.001$. Let $k = \tau w p$. We first form an $k \times p$ matrix Y where the column vectors of Y are mutually orthogonal $\{1, -1\}$ vectors. This is possible as $k \gg p$. Now we construct $w+1$ databases D_i for $1 \le i \le w+1$ as follows. For all the databases, they contain the common set of examples $(z_j, 0)$ (i.e. vector z_j with label 0) for $1 \le j \le k$ where $z_j = (Y_{j1}, \ldots, Y_{jp})$ is the j-th row vector of Y. In addition, each D_i contains w examples $(x_j, 1)$ for $x_j = (X_{j1}, \ldots, X_{jk})$ for $j \ne i$. Then $\mathcal{L}(\theta; D_i)$ is defined as follows (for the ease of notation in this proof, we work with the un-normalized loss. This does not affect the generality of the arguments in any way.)

$$\mathcal{L}(\theta; D_i) = \sum_{j \ne i}(x_j \cdot \theta - 1)^2 + \sum_{j=1}^{k}(y_j \cdot \theta)^2 = \sum_{j \ne i}(x_j \cdot \theta - 1)^2 + k\|\theta\|_2^2.$$

The last equality is due to that the columns of Y are mutually orthogonal $\{-1, 1\}$ vectors. For each D_i, consider $\theta^* \in \left\{-\frac{1}{p}, \frac{1}{p}\right\}^p$ such that the sign of the coordinates of θ^* matches the sign for the consensus columns of $X_{(i)}$. Plugging θ^* in $\mathcal{L}(\theta^*; \hat{D})$ we have the following,

$$\begin{aligned}\mathcal{L}(\theta^*; \hat{D}) &\le \sum_{i=1}^{w}(2\tau)^2 + \frac{k}{p} && \text{[since the number of consensus columns is at least } (1-\tau)p] \\ &= (\tau + 4\tau^2)w. && (3)\end{aligned}$$

We now prove the crucial lemma, which states that if θ is such that $\|\theta\|_1 \leq 1$ and $\mathcal{L}(\theta; D_i)$ is small, then θ has to agree with the sign of most of the consensus columns of $X_{(i)}$.

Lemma 3.3. *Suppose that $\|\theta\|_1 \leq 1$, and $\mathcal{L}(\theta; D_i) < 1.1\tau w$. For $j \in W_i$, denote by s_j the sign of the consensus column j. Then we have*

$$|\{j \in W_i \ : \ \mathrm{sign}(\theta_j) = s_j\}| \geq \frac{3}{4}p\,.$$

Proof. For any $S \subseteq \{1, \ldots, p\}$, denote by $\theta|_S$ the projection of θ to the coordinate subset S. Consider three subsets S_1, S_2, S_3, where

$$\begin{aligned} S_1 &= \{j \in W_i \ : \ \mathrm{sign}(\theta_j) = s_j\}\,, \\ S_2 &= \{j \in W_i \ : \ \mathrm{sign}(\theta_j) \neq s_j\}\,, \\ S_3 &= \{1, \ldots, p\} \setminus W_i\,. \end{aligned}$$

The proof is by contradiction. Assume that $|S_1| < \frac{3}{4}p$.

Further denote $\theta_i = \theta|_{S_i}$ for $i = 1, 2, 3$. Now we will bound $\|\theta_1\|_1$ and $\|\theta_3\|_1$ using the inequality $\|x\|_2 \geq \|x\|_1/\sqrt{d}$ for any d-dimensional vector.

$$\|\theta_3\|_2^2 \geq \|\theta_3\|_1^2/|S_3| \geq \|\theta_3\|_1^2/(\tau p)\,.$$

Hence $k\|\theta_3\|_2^2 \geq w\|\theta_3\|_1^2$. But $k\|\theta_3\|_2^2 \leq k\|\theta\|_2^2 \leq 1.1\tau w$, so that $\|\theta_3\|_1 \leq \sqrt{1.1\tau} \leq 0.04$.

Similarly by the assumption of $|S_1| < \frac{3}{4}p$,

$$\|\theta_1\|_2^2 \geq \|\theta_1\|_1^2/|S_1| \geq 4\|\theta_1\|_1^2/(3p)\,.$$

Again using $k\|\theta\|_2^2 < 1.1\tau w$, we have that $\|\theta_1\|_1 \leq \sqrt{1.1 * 3/4} \leq 0.91$.

Now we have $\langle x_i, \theta\rangle - 1 = \|\theta_1\|_1 - \|\theta_2\|_1 + \beta_i - 1$ where $|\beta_i| \leq \|\theta_3\|_1 \leq 0.04$. By $\|\theta_1\|_1 + \|\theta_2\|_1 + \|\theta_3\|_1 \leq 1$, we have

$$|\langle x_i, \theta\rangle - 1| \geq 1 - \|\theta_1\| - |\beta_i| \geq 1 - 0.91 - 0.04 = 0.05\,.$$

Hence we have that $\mathcal{L}(\theta; D_i) \geq (0.05)^2 w \geq 1.1\tau w$. This leads to a contradiction. Hence we must have $|S_1| \geq \frac{3}{4}p$. □

With Theorem 3.2 and Lemma 3.3, we can now prove Theorem 3.1.

Proof. Suppose that $\mathcal{A}$ is private. And for the datasets we constructed above,

$$\mathbb{E}[\mathcal{L}(\mathcal{A}(D_i); D_i) - \min_\theta \mathcal{L}(\theta; D_i)] \leq cw\,,$$

for sufficiently small constant c. By Markov inequality, we have with probability at least $2/3$, $\mathcal{L}(\mathcal{A}(D_i); D_i) - \min_\theta \mathcal{L}(\theta; D_i) \leq 3cw$. By (3), we have $\min\limits_\theta \mathcal{L}(\theta; D_i) \leq (\tau + 4\tau^2)w$. Hence if we choose c a constant small enough, we have with probability $2/3$,

$$\mathcal{L}(\mathcal{A}(D_i); D_i) < (\tau + 4\tau^2 + 3c)w \leq 1.1\tau w\,. \tag{4}$$

By Lemma 3.3, (4) implies that $\mathcal{A}(D_i)$ agrees with at least $\frac{3}{4}p$ consensus columns in $X_{(i)}$. However by Theorem 3.2, this violates the privacy of $\mathcal{A}$. Hence we have that there exists i, such that

$$\mathbb{E}[\mathcal{L}(\mathcal{A}(D_i); D_i) - \min_\theta \mathcal{L}(\theta; D_i)] > cw\,.$$

Recall that $w = m/\log m$ and $n = w + wp = O(m^3/\log m)$. Hence we have that

$$\mathbb{E}[\mathcal{L}(\mathcal{A}(D_i); D_i) - \min_\theta \mathcal{L}(\theta; D_i)] = \Omega(n^{1/3}/\log^{2/3} n)\,.$$

The proof is completed by converting the above bound to the normalized version of $\Omega(1/(n\log n)^{2/3})$. □

References

[1] P. L. Bartlett and S. Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *The Journal of Machine Learning Research*, 3:463–482, 2003.

[2] R. Bassily, A. Smith, and A. Thakurta. Private empirical risk minimization, revisited. In *FOCS*, 2014.

[3] R. Bhaskar, S. Laxman, A. Smith, and A. Thakurta. Discovering frequent patterns in sensitive data. In *KDD*, New York, NY, USA, 2010.

[4] M. Bun, J. Ullman, and S. Vadhan. Fingerprinting codes and the price of approximate differential privacy. In *STOC*, 2014.

[5] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A. S. Willsky. The convex geometry of linear inverse problems. *Foundations of Computational Mathematics*, 12(6):805–849, 2012.

[6] K. Chaudhuri and C. Monteleoni. Privacy-preserving logistic regression. In *NIPS*, 2008.

[7] K. Chaudhuri, C. Monteleoni, and A. D. Sarwate. Differentially private empirical risk minimization. *JMLR*, 12:1069–1109, 2011.

[8] K. L. Clarkson. Coresets, sparse greedy approximation, and the Frank-Wolfe algorithm. *ACM Transations on Algorithms*, 2010.

[9] J. C. Duchi, M. I. Jordan, and M. J. Wainwright. Local privacy and statistical minimax rates. In *FOCS*, 2013.

[10] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography Conference*, pages 265–284. Springer, 2006.

[11] C. Dwork, A. Nikolov, and K. Talwar. Efficient algorithms for privately releasing marginals via convex relaxations. *arXiv preprint arXiv:1308.1385*, 2013.

[12] C. Dwork and A. Roth. *The Algorithmic Foundations of Differential Privacy*. Foundations and Trends in Theoretical Computer Science. NOW Publishers, 2014.

[13] C. Dwork, G. N. Rothblum, and S. P. Vadhan. Boosting and differential privacy. In *FOCS*, 2010.

[14] C. Dwork, K. Talwar, A. Thakurta, and L. Zhang. Analyze gauss: optimal bounds for privacy-preserving principal component analysis. In *STOC*, 2014.

[15] M. Frank and P. Wolfe. An algorithm for quadratic programming. *Naval research logistics quarterly*, 3(1-2):95–110, 1956.

[16] E. Hazan and S. Kale. Projection-free online learning. In *ICML*, 2012.

[17] M. Jaggi. Revisiting {Frank-Wolfe}: Projection-free sparse convex optimization. In *ICML*, 2013.

[18] P. Jain, P. Kothari, and A. Thakurta. Differentially private online learning. In *COLT*, pages 24.1–24.34, 2012.

[19] P. Jain and A. Thakurta. (near) dimension independent risk bounds for differentially private learning. In *International Conference on Machine Learning (ICML)*, 2014.

[20] D. Kifer, A. Smith, and A. Thakurta. Private convex empirical risk minimization and high-dimensional regression. In *COLT*, pages 25.1–25.40, 2012.

[21] F. McSherry and K. Talwar. Mechanism design via differential privacy. In *FOCS*, pages 94–103. IEEE, 2007.

[22] A. Nikolov, K. Talwar, and L. Zhang. The geometry of differential privacy: The sparse and approximate cases. In *STOC*, 2013.

[23] S. Shalev-Shwartz, N. Srebro, and T. Zhang. Trading accuracy for sparsity in optimization problems with sparsity constraints. *SIAM Journal on Optimization*, 2010.

[24] A. Smith and A. Thakurta. Differentially private feature selection via stability arguments, and the robustness of the Lasso. In *COLT*, 2013.

[25] A. Smith and A. Thakurta. Follow the perturbed leader is differentially private with optimal regret guarantees. *Manuscript in preparation*, 2013.

[26] A. Smith and A. Thakurta. Nearly optimal algorithms for private online learning in full-information and bandit settings. In *NIPS*, 2013.

[27] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 1996.

[28] R. Tibshirani et al. The Lasso method for variable selection in the cox model. *Statistics in medicine*, 16(4):385–395, 1997.

[29] J. Ullman. Private multiplicative weights beyond linear queries. *CoRR*, abs/1407.1571, 2014.