Sampling Without Compromising Accuracy in Adaptive Data Analysis

Benjamin Fish BENJAMIN.FISH@MICROSOFT.COM
Microsoft Research
Montréal, H3A 3H3, Canada

Lev Reyzin LREYZIN@UIC.EDU
Department of Mathematics, Statistics, and Computer Science
University of Illinois at Chicago
Chicago, IL 60607, USA

Benjamin I. P. Rubinstein BRUBINSTEIN@UNIMELB.EDU.AU
School of Computing & Information Systems
University of Melbourne
Parkville, 3010 VIC, Australia

Editors: Aryeh Kontorovich and Gergely Neu

Abstract

In this work, we study how to use sampling to speed up mechanisms for answering adaptive queries into datasets without reducing the accuracy of those mechanisms. This is important to do when both the datasets and the number of queries asked are very large. In particular, we describe a mechanism that provides a polynomial speed-up per query over previous mechanisms, without needing to increase the total amount of data required to maintain the same generalization error as before. We prove that this speed-up holds for arbitrary statistical queries. We also provide an even faster method for achieving statistically-meaningful responses wherein the mechanism is only allowed to see a constant number of samples from the data per query. Finally, we show that our general results yield a simple, fast, and unified approach for adaptively optimizing convex and strongly convex functions over a dataset.

Keywords: Adaptive data analysis, differential privacy, sublinear-time algorithms

1. Introduction

The field of data analysis seeks out statistically valid conclusions from data: inferences that generalize to an underlying distribution rather than specialize to the data sample at hand. As a result, classical proofs of statistical efficiency have focused on independence assumptions on data with a predetermined sequence of analyses (Lee et al., 2016). In practice, most data analysis is adaptive or exploratory: previous inferences inform future analysis. This adaptivity is nigh impossible to avoid when multiple scientists contribute work to an area of study using the same or similar data sets. Unfortunately, adaptivity may lead to 'false discovery,' where the dependence on past analysis may create pervasive overfitting—also known as 'the garden of forking paths' or 'p hacking' (Gelman and Loken, 2014).

There has been much recent progress in minimizing the amount of data needed to draw generalizable conclusions, without having to make any assumptions about the type of adaptations used by the data analysis. Meanwhile, bootstrapping and related sampling techniques have enjoyed widespread

	computational complexity			sample complexity		
query type	previous work	this work		previous work		this work
statistical queries (Section 3)	$\tilde{O}\left(\frac{k^{3/2}}{\alpha^2}\right)$	$\tilde{O}\left(\frac{k\log^2(k)}{\alpha^2}\right)$	$\Omega\left(\frac{k}{\alpha^2}\right)$	$\tilde{O}\left(\frac{\sqrt{k}}{\alpha^2}\right)$	$\Omega\left(\frac{\sqrt{k}}{\alpha^2}\right)$	$\tilde{O}\left(\frac{\sqrt{k}}{\alpha^2}\right)$
sampling counting queries (Section 4)	—	$\tilde{O}\left(k\log\left(\frac{k}{\alpha}\right)\right)$	$\Omega(k)$	—	—	$\tilde{O}\left(\frac{\sqrt{k}}{\alpha^2}\right)$

Table 1: Summary of our upper and lower bounds compared to previous work (Bassily et al., 2016) over the course of answering k queries. α is the accuracy rate. Dependence on the probability of failure has been suppressed for ease of reading. For more precise definitions, see Section 2.

and successful use in practice across a variety of settings (Kleiner et al., 2012; Xiao et al., 2016), including in adaptive settings (Golbandi et al., 2011), but they have been largely ignored in this burgeoning field. This is a gap that not only points to an unexplored area of theoretical study, but also opens up the possibility of creating substantially faster algorithms for answering adaptively generated queries.

In this paper, we aim to do just this: we develop strong theoretical results that are significantly faster than previous approaches, taking sublinear time per query, thereby initiating the intersection of sublinear-time algorithm design and adaptive data analysis.

1.1. Motivation and Results

As in previous literature (starting with Dwork et al. (2015b)), a mechanism $\mathcal{M}$ is given an i.i.d. sample S of size n from an unknown distribution D over a space X, and is supplied queries of the form $q : D \to \mathbb{R}$. After each query, the mechanism must respond with an answer a that is close to $q(D)$ up to a parameter α with high probability. Furthermore, each query may be adaptive: The query may depend on the previous queries and answers to those queries.

Our results are summarized in Table 1. We first point out that we can use well-known privacy amplification techniques to get a fast mechanism for answering statistical queries (which asks questions of the form 'What is the expected value of my function on the data?') *without* losing accuracy. Usually, using these privacy amplification techniques results in a loss of accuracy, so it's notable that in this setting, we can speed up responses without the loss in accuracy on the distribution. In Section 3, we show that our method still has $n = \tilde{O}(\sqrt{k}/\alpha^2)$[1] sample complexity as in previous work but takes only $\tilde{O}(k\log^2(k)/\alpha^2)$ time to answer k queries, instead of $\tilde{O}(k^{3/2}/\alpha^2)$ time as in previous approaches (Theorem 5). Moreover, our mechanism to answer a query is simple, and involves subsampling $\ell = \tilde{O}(\log(k)/\alpha^2)$ samples per query. While it is not possible to improve the sample complexity over previous work (Nissim et al., 2018), we decrease the number of samples that need to be examined per query, resulting in faster responses to queries.

We also show that our upper bound on total computational complexity is tight up to poly-log factors when the mechanism gets to ask for evaluations of queries at given sample points. This lower

1. We use the notation $\tilde{O}(f)$ to hide terms that are logarithmic in f.

bound on computational complexity that we provide is larger than the sample complexity $\tilde{O}(\sqrt{k}/\alpha^2)$ for answering statistical queries. Running time then may become a problem for very large and popular datasets, making it valuable to give provably accurate mechanisms that are fast enough to run on very large datasets when the number of queries is large compared to the size of the dataset.

However, an analyst may wish to control the number of samples ℓ examined to compute the response to a query, down to possibly one point, in order to save on time and effort. The above methods cannot handle this case gracefully because when ℓ is sufficiently small, the guarantees on accuracy become trivial—we get only that $\alpha = O(1)$. Instead, we want to have a statistically-meaningful reply even if $\ell = 1$. Indeed, the empirical answer when $\ell = 1$ is $\{0, 1\}$-valued, unlike a response using Laplacian noise.

To address these issues, we consider an 'honest' setting where the mechanism must always yield a plausible reply to each query (Section 4). This is analogous to the honest version (Yang, 2001) of the statistical query (SQ) setting for learning (Blum et al., 1994; Kearns, 1998), or the 1-STAT oracle for optimization (Feldman et al., 2017a). Thus we introduce *sampling counting queries*, which imitate the process of an analyst requesting the value of a query on a single random sample. Equivalently, this enforces a binary randomized response, where a query asks for a coin flip from a coin with unknown bias determined by the query on the dataset.

This explores a different extreme than statistical queries by allowing for queries to be answered much faster than statistical queries can, at the cost of accuracy. So, for example, we can't just round the values of statistical queries in order to answer sampling counting queries. We show how to answer these queries by sampling a single point s from S and then applying a simple differentially private algorithm to $q(s)$ (Theorem 11).

Finally, to demonstrate the applicability of our general results, we use them as a black-box technique to obtain bounds for convex optimization (Section 6). In particular, we introduce a simple procedure for adaptive gradient descent that uses our sampling mechanism for statistical queries to compute gradients in the course of gradient descent. This results in a fast, unified approach for answering both convex and strongly convex optimization queries. For answering k convex optimization queries, we decrease the total number of calls to compute the gradient from $O(kdn^2)$ in (Bassily et al., 2016) to $\tilde{O}(kd/\alpha^2)$ in the convex case and $\tilde{O}(kd/\alpha)$ in the strongly convex case, where d is the dimension of the convex space (Corollaries 15 and 16). (Note, however, Bassily et al. 2016 make slightly different assumptions about the loss function. Roughly speaking, they require that the loss function be bounded, whereas we only require the gradient of the loss function be bounded.) Our results are similar to those given by Feldman et al. (2017b) when using our statistical query mechanism to compute gradients. However, we provide a unified approach and a direct proof using primal gradient descent, unlike Feldman et al. (2017b), who uses the more complex dual gradient method of Devolder et al. (2013) in the strongly convex case.

1.2. Previous Work

Previous work in this area has focused on finding accurate mechanisms with low sample complexity (the size of S) for a variety of queries and settings (Bassily et al., 2016; Dwork et al., 2015a,b; Rogers et al., 2016; Steinke and Ullman, 2015a). Bassily et al. (2016) consider, amongst other queries, *statistical queries*; if the queries are nonadaptive, then only roughly $\log(k)/\alpha^2$ samples are needed to answer k such queries. And if the queries are adaptive but the mechanism simply outputs the empirical estimate of q on S, then the sample complexity is much worse—order k/α^2 instead.

In this paper, we will focus only on computationally-efficient mechanisms. It is not necessarily obvious that it is possible to achieve a smaller sample complexity for an efficient mechanism in the adaptive case, but Bassily et al. (2016), building on the work of Dwork et al. (2015b), provide a mechanism with sample complexity $n = \tilde{O}(\sqrt{k}/\alpha^2)$ to answer k statistical queries. Furthermore, for efficient mechanisms, this bound is tight in k (Steinke and Ullman, 2015b). Bassily et al. (2016) also show how to efficiently answer *convex optimization queries*, which ask for the minimizer of a convex loss function, using a (private) gradient descent algorithm of Bassily et al. (2014).

This literature shows that the key to finding such mechanisms with this improvement over the naïve method is finding stable mechanisms: those whose output does not change too much when the sample is changed by a single element. Much of this literature leverages differential privacy (Bassily et al., 2016; Dwork et al., 2015a,b; Steinke and Ullman, 2015a), which offers a strong notion of stability. Here we use differentially-private mechanisms post sampling, noting that sampling in settings where privacy matters has long been deemed useful (Bassily et al., 2014; Jorgensen et al., 2015; Kasiviswanathan et al., 2008; Kellaris and Papadopoulos, 2013).

In particular, we take advantage of the fact that sampling not only maintains privacy, but actually boosts it. Such a result may be found in (Kasiviswanathan et al., 2008), and since then various sampling regimes have been considered, including by Bun et al. (2015), who show that sampling with replacement boosts privacy, and more recently by Balle et al. (2018), who establish tight bounds.

2. Model and Preliminaries

In the adaptive data analysis setting we consider, a (possibly stateful) mechanism $\mathcal{M}$ that is given an i.i.d. sample S of size n from an unknown distribution D over a finite space X. The mechanism $\mathcal{M}$ must answer queries from a stateful adversary $\mathcal{A}$. These queries are adaptive: $\mathcal{A}$ outputs a query q_i, to which the mechanism returns a response a_i, and the outputs of $\mathcal{A}$ and $\mathcal{M}$ may depend on all queries $q_1, \ldots, q_{i-1}$ and responses $a_1, \ldots, a_{i-1}$.

2.1. Statistical Queries and Optimization Queries

In this work, the first type of query we consider is a *statistical query*, which is specified by a function $q : X \to [0, 1]$ that represents a real-valued statistic for any element $x \in X$. The restriction of q to $[0, 1]$ is for convenience; our results easily generalize to the case where q is merely bounded. We then define the query q on a sample $S \in X^m$ as $q(S) = \frac{1}{|S|} \sum_{x \in S} q(x)$ and on the distribution as $q(D) = \mathbb{E}_{x \sim D}[q(x)]$. This represents the average value of the statistic on the sample and distribution, respectively. We now define the accuracy of $\mathcal{M}$:

Definition 1 *A mechanism $\mathcal{M}$ is* (α, β)-accurate *over distribution D on statistical queries $q_1, \ldots, q_k$, if when $\mathcal{M}$ is given an i.i.d. sample S from D, for its responses $a_1, \ldots, a_k$ we have*

$$\mathbb{P}_{\mathcal{M},\mathcal{A}} \left[\max_i |q_i(D) - a_i| \leq \alpha \right] \geq 1 - \beta.$$

We define (α, β)-accuracy over a sample S analogously. In this work, we not only desire (α, β)-accuracy but we also want to consider the time per query taken by $\mathcal{M}$. We assume we will have oracle access to q, which will compute $q(s)$ for a sample point s in unit time (and also $q(S)$ in at most $O(|S|)$ time). This is not a strong assumption: As long as the queries can be computed efficiently,

then this can add only at most a poly-log factor overhead in n and $|X|$ (as long as we only compute q on a roughly $\log(n)$ size sample, which will turn out to be exactly the case).

We also consider optimization queries, first considered in this adaptive setting by Bassily et al. (2016). In convex optimization, we have a loss function $\mathcal{L} : X^n \times \Theta \to \mathbb{R}$ defined over a convex set $\Theta \subseteq \mathbb{R}^d$ and a sample from X^n drawn from a distribution D, and the goal is to output $x \in \Theta$ that minimizes the expected loss, i.e. such a query is defined as

$$q(D) := \arg\min_{x \in \Theta} \mathbb{E}_{S \sim D^n}[\mathcal{L}(S, x)].$$

Since the loss function $\mathcal{L}$ determines the query, we will abuse notation and use $\mathcal{L}$ to also refer to the optimization query. We measure accuracy of the response a_i by the expected regret: A mechanism is (α, β)-*accurate* on optimization queries each specified by a loss function $\mathcal{L}_i$ with respect to a distribution D if

$$\mathbb{P}_{\mathcal{M},\mathcal{A}}\left[\max_i \mathbb{E}_S\left[\mathcal{L}_i(S, a_i) - \min_{x \in \Theta} \mathcal{L}_i(S, x)\right] \leq \alpha\right] \geq 1 - \beta.$$

We will assume that $\mathcal{L}_i$ is convex in x. We will also consider the special case when $\mathcal{L}_i$ is strongly convex in x. A function $\mathcal{L}$ is H-*strongly convex* if for all x, y in Θ,

$$\mathcal{L}(y) \geq \mathcal{L}(x) + \langle \nabla \mathcal{L}(x), y - x \rangle + \frac{H}{2}\|y - x\|_2^2.$$

2.2. Counting Queries and Sampling Counting Queries

Counting queries ask the question "What proportion of the data satisfies property q?" Counting queries are a simple and important restriction of statistical queries (Blum et al., 2008; Bun et al., 2014; Steinke and Ullman, 2015a) that limits the allowed statistics to binary properties. More formally, a counting query is specified by a function $q : X \to \{0, 1\}$, where $q(S) = \frac{1}{|S|}\sum_{s \in S} q(s)$ and $q(D) = \mathbb{E}_{s \sim D}[q(s)]$. As in the statistical query setting, an answer to a counting query must be close to $q(D)$ (Definition 1).

This means, however, that answers to counting queries will not necessarily be counts themselves, nor meaningful in settings where we require ℓ to be small, i.e. very few samples from the database to answer each query. To this end, we introduce *sampling counting queries*. A sampling counting query (SCQ) is again specified by a function $q : X \to \{0, 1\}$, but this time the mechanism $\mathcal{M}$ must return an answer $a \in \{0, 1\}$. Given these restricted responses, we want such a mechanism to act like what would happen if $\mathcal{A}$ were to take a single random sample point s from D and evaluate $q(s)$. We define queries in this way so that they represent the smallest possible amount of information still useful to an analyst.

Now the average value the mechanism returns (over the coins of the mechanism) should be close to the expected value of q. More precisely:

Definition 2 *A mechanism $\mathcal{M}$ is (α, β)-accurate on distribution D for k sampling counting queries q_i if for all states of $\mathcal{M}$ and $\mathcal{A}$, when $\mathcal{M}$ is given an i.i.d. sample S from D,*

$$\mathbb{P}_{S,\mathcal{M},\mathcal{A}}\left[\max_i |\mathbb{E}_{\mathcal{M}}[\mathcal{M}(q_i)] - q_i(D)| \leq \alpha\right] \geq 1 - \beta.$$

We also define (α, β)-accuracy on a sample S from D analogously. Again, our requirement is that $\mathcal{M}$ be (α, β)-accurate with respect to the unknown distribution D, this time using only around $\log(n)$ time per query (and a constant number of samples per query).

These queries allow responses to use fewer than the number of sampled points ℓ required for answering statistical queries while also returning an integer-valued response. And if we do want to answer a statistical query, they also allow an anytime algorithm as we can trade off ℓ with the accuracy by averaging over the responses to repeatedly asking the same SCQ. In this way, SCQ's are 'honest,' because repeatedly asking the same SCQ always yields an integer fraction instead of a real number.

2.3. Differential Privacy and the Transfer Theorem

Differential privacy, first introduced by Dwork et al. (2006), provides a strong notion of stability.

Definition 3 (Differential privacy) *Let $\mathcal{M}$ be a randomized algorithm with domain X^n and image Z. We call $\mathcal{M}$* (ϵ, δ)-differentially private *if for every two samples $S, S' \in X^n$ differing on one instance, and every measurable $z \subset Z$,*

$$\mathbb{P}[\mathcal{M}(S) \in z] \leq e^{\epsilon} \cdot \mathbb{P}[\mathcal{M}(S') \in z] + \delta.$$

If $\mathcal{M}$ is $(\epsilon, 0)$-private, we may simply call it ϵ-private.

Differential privacy comes with several guarantees useful for developing new mechanisms. In this paper, we use two well-established differentially-private mechanisms: the Laplace and exponential mechanisms. See (Dwork and Roth, 2014) for more on these mechanisms and properties of differential privacy, including adaptive composition and post-processing, which are also given in Appendix E for convenience.

A key method of Bassily et al. (2016) for answering queries adaptively is a 'transfer theorem,' which states that if a mechanism is both accurate on a sample and differentially private, then it will be accurate on the sample's generating distribution.

Theorem 4 (Bassily et al., 2016) *Let $\mathcal{M}$ be a mechanism that on input sample $S \sim D^n$ answers k adaptively chosen statistical queries, is $(\frac{\alpha}{64}, \frac{\alpha\beta}{32})$-private for some $\alpha, \beta > 0$ and $(\frac{\alpha}{8}, \frac{\alpha\beta}{16})$-accurate on S. Then $\mathcal{M}$ is (α, β)-accurate on D.*

Their 'monitoring algorithm' proof technique involves a thought experiment in which an algorithm, called the monitor, assesses how accurately an input mechanism replies to an adversary, and remembers the query it performs the worst on. It repeats this process some T times, and outputs the query that the mechanism does the worst on over all T rounds. Since the mechanism is private, so too is the monitor; and since privacy implies stability, this will ensure that the accuracy of the worst query is not too bad. For more details see Bassily et al. (2016).

3. Answering Statistical Queries

In this section, we provide simple and fast mechanisms for answering statistical queries. We then show that this mechanism is as fast as possible up to poly-log factors when the mechanism gets to ask for evaluations of queries at given sample points. Our mechanism $\mathcal{M}$ for answering statistical

queries is as follows: Given a data set S of size n and query q, sample some ℓ points uniformly at random from S (with or without replacement), and call this new set S_ℓ. Then the mechanism returns $q(S_\ell) + \text{Lap}\left(\frac{1}{\ell\epsilon'}\right)$, where $\text{Lap}(b)$ refers to the zero-mean Laplacian distribution with scale parameter b, and ϵ' is a carefully chosen privacy setting.

Algorithm 1 Fast mechanism for statistical queries

Parameters: Sub-sample size ℓ, target privacy parameters (ϵ, δ), number of queries k

Input: Sample S, query q

$S_\ell := \{s_1, \ldots, s_\ell\}$, where $s_i \sim S$ uniformly at random (with or without replacement).

$\epsilon' := \frac{\epsilon n}{4\ell\sqrt{2k\log(1/\delta)}}$

return $q(S_\ell) + \text{Lap}\left(\frac{1}{\ell\epsilon'}\right)$.

We may now state our result for mechanism $\mathcal{M}$ (Algorithm 1), using suitable values for ϵ, δ, and ℓ.

Theorem 5 *For any $\alpha, \beta > 0$ and $k \geq 1$, when we run $\mathcal{M}$ (Algorithm 1) on k statistical queries with parameters $\ell = \frac{2\log(4k/\beta)}{\alpha^2}$, $\epsilon = \alpha/64$, and $\delta = \alpha\beta/32$, we have*

1. *$\mathcal{M}$ takes $\tilde{O}\left(\frac{\log(k)\log(k/\beta)}{\alpha^2}\right)$ time per query.*
2. *$\mathcal{M}$ is (α, β)-accurate on the distribution so long as $n = \Omega\left(\frac{\sqrt{k}\log k \cdot \log^{3/2}\left(\frac{1}{\alpha\beta}\right)}{\alpha^2}\right)$.*

Sampling with replacement takes $O(\log n)$ time per sample, for a total of $O(\ell \log n)$ time over ℓ samples. This suffices to prove part 1. for the values of ℓ and n given. Sampling without replacement may also take $O(\log n)$ time per sample.[2]

To prove part 2, we need to take advantage of the fact that sampling amplifies privacy. If sampling before an ϵ-private mechanism were to only deliver $O(\epsilon)$ instead of $O(\frac{\ell}{n}\epsilon)$ privacy then we would need $\ell > \frac{2\sqrt{2k\log(1/\delta)}\log(2k/\beta)}{\alpha\epsilon}$, which would be undesirable: ℓ then becomes the size of the entire database and sampling yields no time savings over computing $q(S)$ exactly. With these savings in our privacy budget, we can decrease the amount of noise we add to the outputs, compensating for the accuracy loss we incur by sampling.

Proposition 6 (Lin et al., 2013; Balle et al., 2018) *Given mechanism $\mathcal{P} : X^\ell \to Y$, let $\mathcal{M}$ do the following: Sample uniformly at random* without *replacement ℓ points from an input sample $S \in X^n$ of size n, and call this set S_ℓ. Output $\mathcal{P}(S_\ell)$. Then if $\mathcal{P}$ is ϵ-private, then $\mathcal{M}$ is $\log(1 + \frac{\ell}{n}(e^\epsilon - 1))$-private for $\ell \geq 1$.*

Sampling with replacement also amplifies privacy:

2. This may come at the cost of space complexity, e.g. by keeping track of which elements have not been chosen so far (Wong and Easton, 1980). Alternatively, there are methods that enjoy optimal space complexity at the cost of worst-case running times, as in rejection sampling (Vitter, 1984).

Proposition 7 (Bun et al., 2015; Balle et al., 2018) *Given mechanism $\mathcal{P} : X^\ell \to Y$, let $\mathcal{M}$ do the following: Sample uniformly at random* with *replacement ℓ points from an input sample $S \in X^n$, and call this set S_ℓ. Output $\mathcal{P}(S_\ell)$. Then if $\mathcal{P}$ is ϵ-private, then $\mathcal{M}$ is $\log(1+(1-(1-\frac{1}{n})^\ell)(e^\epsilon-1))$-private for $\ell \geq 1$.*

Note we have that whenever $\epsilon \leq 1$, both $\log(1 + \frac{\ell}{n}(e^\epsilon - 1)) \leq 2\frac{\ell}{n}\epsilon$ and $\log(1 + (1 - (1 - \frac{1}{n})^\ell)(e^\epsilon - 1)) \leq 2\frac{\ell}{n}\epsilon$, so privacy amplification is linear in the sub-sample size ℓ. This linear amplification allows us to set ϵ' as proportional to $\frac{\epsilon n}{\ell\sqrt{k\log(1/\delta)}}$ instead of $\frac{\epsilon}{\sqrt{k\log(1/\delta)}}$ which would be required without any privacy amplification. For the proof of part 2, see Appendix A.

We also have a version of this theorem that demonstrates that this mechanism will still be accurate in expectation at any point along the execution, even if in the first t rounds it (with small probability) failed to be accurate. This requires a slight variant of Theorem 4, provided in Appendix B.2.

Theorem 8 *For any $\alpha \geq \alpha_0 = \tilde{O}\left(\frac{k^{1/4}}{\sqrt{n}} + \frac{1}{\sqrt{\ell}}\right)$, when we run $\mathcal{M}$ (Algorithm 1) with parameters $\ell \geq 1$, $\epsilon = \alpha/8$, and $\delta = \alpha/4$, with respect to any possible simulation between $\mathcal{A}$ and $\mathcal{M}$ up to the first $t-1$ rounds, and denoting the expectation while conditioning on any such possibility $E_{t-1}[\cdot]$, for any $i \geq t$,*

$$\mathbb{E}_{t-1,S,\mathcal{A},\mathcal{M}}[|a_i - q_i(D)|] \leq \alpha.$$

The proof is similar to the proof of Theorem 5, but using Proposition 20 and the fact that

$$\mathbb{E}_{t-1,S,\mathcal{A},\mathcal{M}}[|a_i - q_i(S)|] \leq \mathbb{E}_{t-1,S,\mathcal{A},\mathcal{M}}[|a_i - q_i(S_\ell)|] + \mathbb{E}_{t-1,S,\mathcal{A},\mathcal{M}}[|q_i(S_\ell) - q_i(S)|] \lesssim \frac{1}{\epsilon'\ell} + \frac{1}{\sqrt{\ell}}.$$

The computational complexity of the mechanism in Theorem 5 is tight up to poly-log factors, even in the non-adaptive case when all queries must be made before seeing any replies from the mechanism. We show this by considering random queries, which for the purposes of this construction, the learner can access by asking for evaluations at given points. The query values will simulate flipping a coin with given bias from one of two biases randomly selected. Then it takes computing each query on $\Omega(1/\alpha^2)$ sample points to distinguish between a fair coin and a weighted coin, resulting in a total computational complexity of at least $\Omega(k/\alpha^2)$ points. The proof may be found in Appendix C.

Proposition 9 *Suppose for any sequence $q_1, \ldots, q_k$ of k statistical queries chosen non-adaptively, there is a mechanism $\mathcal{M}$ that is $(\alpha, 1/5)$-accurate on the uniform distribution over a universe X with $|X| \geq \frac{2\log(10)}{\alpha^2}$. Then $\mathcal{M}$ must evaluate the queries on at least $\Omega(k/\alpha^2)$ points.*

4. Answering Sampling Counting Queries

We now turn to sampling counting queries. Because of the different notion of accuracy for these queries, we establish a new transfer theorem.

Theorem 10 *Let $\mathcal{M}$ be a mechanism that on input sample $S \sim D^n$ answers k adaptively chosen sampling counting queries, is $(\frac{\alpha}{64}, \frac{\alpha\beta}{16})$-private for some $\alpha, \beta > 0$ and $(\alpha/2, 0)$-accurate on S. Suppose further that $n \geq \frac{1024\log(k/\beta)}{\alpha^2}$. Then $\mathcal{M}$ is (α, β)-accurate on D.*

This allows us to answer sampling counting queries:

Theorem 11 *For any $\alpha, \beta > 0$ and $k \geq 1$, there is a mechanism $\mathcal{M}$ that satisfies the following:*

1. *$\mathcal{M}$ takes $\tilde{O}\left(\log\left(\frac{k\log(\frac{1}{\beta})}{\alpha}\right)\right)$ time per query.*
2. *$\mathcal{M}$ is (α, β)-accurate on k SCQ's, where $n \geq \Omega\left(\max\left(\sqrt{k\log(\frac{1}{\alpha\beta})}/\alpha^2, \log(k/\beta)/\alpha^2\right)\right)$.*

This results in spending $\tilde{O}\left(k\log\left(\frac{k\log(\frac{1}{\beta})}{\alpha}\right)\right)$ time over the course of k queries, which must be tight up to log factors, as the mechanism of course must spend at least unit time per query.

We prove our transfer theorem using the monitoring algorithm $\mathcal{W}_D$ (Algorithm 2), which takes as input T sample sets, and outputs a query with probability proportional to how far away the query is on the sample as opposed to the distribution.

Algorithm 2 Monitor with exponential mechanism $\mathcal{W}_D$

Parameters: Mechanisms $\mathcal{M}$ and $\mathcal{A}$, distribution D
Input: Set of samples $\mathbf{S} = \{S_1, \ldots, S_T\}$
for t in $[T]$ **do**
Simulate $\mathcal{M}(S_t)$ and $\mathcal{A}$ interacting.
Let $q_{t,1}, \ldots, q_{t,k}$ be the queries of $\mathcal{A}$.
end for
Let $\mathcal{R} := \{(q_{t,i}, t)\}_{t\in[T], i\in[k]}$.
Abusing notation, for each t and $i \in [k]$, consider the corresponding element $r_{t,i}$ of $\mathcal{R}$ and define the utility of $r_{t,i}$ as $u(\mathbf{S}, r_{t,i}) = |q_{t,i}(S_t) - q_{t,i}(D)|$.
return $r \in \mathcal{R}$ with probability proportional to $\exp\left(\frac{\epsilon \cdot n \cdot u(\mathbf{S}, r)}{2}\right)$.

$\mathcal{W}_D$ must be private if $\mathcal{M}$ is: $\mathcal{R}$ represents post-processing from the differentially private $\mathcal{M}$, and outputting an element from $\mathcal{R}$ is achieved with the exponential mechanism. We can then bound the probability that $q(S)$ is far from $q(D)$ for q the query that the monitor outputs, by using the fact that private algorithms like the monitor are also stable. This yields the transfer theorem given in Theorem 11. The full proof is provided in Appendix B.1.

With a transfer theorem in hand, we now introduce a private mechanism for answering SCQ's.

Algorithm 3 SCQ mechanism

Parameters: Target accuracy α
Input: Sample S, query q
Sample $s \sim S$ uniformly at random.
return $q(s)$ with probability $1 - \alpha$ and $1 - q(s)$ with probability α.

Lemma 12 (SCQ mechanism) *For $\epsilon \leq 1$, There is an (ϵ, δ)-private mechanism to release k SCQ's that is $(\alpha, 0)$-accurate, for $\alpha \leq 1/2$, with respect to a fixed sample S of size n so long as $n > \frac{2\sqrt{2k\log(1/\delta)}}{\alpha\epsilon}$.*

Proof We design a mechanism $\mathcal{M}$ to release an $(\alpha, 0)$-accurate SCQ for $n > \frac{1}{\alpha\epsilon}$ and then use adaptive composition. The mechanism (Algorithm 3) is simple: sample s i.i.d. from S. Then

release $q(s)$ with probability $1-\alpha$ and $1-q(s)$ with probability α. Let $i = \sum_{s\in S} q(s)$. Then $\mathbb{E}_{\mathcal{M}}[\mathcal{M}(q)] = \frac{(1-\alpha)i+\alpha(n-i)}{n} = \frac{i}{n} + \alpha\left(\frac{n-2i}{n}\right)$, so $\frac{i}{n} - \alpha \le \mathbb{E}_{\mathcal{M}}[\mathcal{M}(q)] \le \frac{i}{n} + \alpha$, implying that $\mathcal{M}$ is $(\alpha, 0)$-accurate on S.

Now let S' differ from S on one element s, where $q(s) = 0$ but for $s' \in S'$, $q(s') = 1$. The other cases are very similar. Consider

$$\frac{\mathbb{P}[\mathcal{M}(S') = 1]}{\mathbb{P}[\mathcal{M}(S) = 1]} = \frac{(1-\alpha)\frac{i+1}{n} + \alpha\left(\frac{n-(i+1)}{n}\right)}{(1-\alpha)\frac{i}{n} + \alpha\left(\frac{n-i}{n}\right)} = 1 + \frac{1-2\alpha}{(1-2\alpha)i + \alpha n}.$$

Note this is at least 1 since $1-2\alpha \ge 0$. By computing the partial derivative with respect to i, it is easy to see that this is maximized when $i = 0$ or $i = n-1$. When $i = 0$,

$$\log\left(\frac{\mathbb{P}[\mathcal{M}(S') = 1]}{\mathbb{P}[\mathcal{M}(S) = 1]}\right) \le \frac{1-2\alpha}{\alpha n} \le \frac{1}{\alpha n} \le \epsilon$$

when $n \ge \frac{1}{\epsilon\alpha}$. When $i = n-1$,

$$\log\left(\frac{\mathbb{P}[\mathcal{M}(S') = 1]}{\mathbb{P}[\mathcal{M}(S) = 1]}\right) \le \frac{1-2\alpha}{n(1-\alpha) - (1-2\alpha)} \le \epsilon$$

when $n \ge \frac{(1-2\alpha)(\epsilon+1)}{(1-\alpha)\epsilon}$ but because $\frac{1-2\alpha}{1-\alpha} \le 1$, it suffices to set $n \ge 1 + \frac{1}{\epsilon}$. The proof is completed by noting that $\frac{1}{\epsilon\alpha} \ge 1 + \frac{1}{\epsilon}$ because $\epsilon \le 1$. ■

We now use this mechanism to answer sampling counting queries.

Proof of Theorem 11 We use Algorithm 3 for each query. This gives an (ϵ, δ)-private mechanism that is $(\alpha/2, 0)$-accurate so long as $n \ge \frac{4\sqrt{2k\log(1/\delta)}}{\alpha\epsilon}$. Setting ϵ and δ as required by Theorem 10 implies that we need $n \ge \Omega\left(\sqrt{k\log(\frac{1}{\alpha\beta})}/\alpha^2\right)$. Note to use Theorem 10 we also need $n \ge \Omega\left(\log(k/\beta)/\alpha^2\right)$. The sample complexity bound follows. This mechanism samples a single random point, taking $O(\log(n))$ time, completing the proof. ■

5. Comparing Counting and Sampling Counting Queries

How do our mechanisms for counting queries and sampling counting queries compare? Can we use a mechanism for SCQ's to simulate a mechanism for counting queries, or vice-versa? We now show that the natural approach to simulate a counting query with SCQ's results in similar running times but an extra $O(1/\alpha)$ factor in its sample size (although it does enjoy a slightly better dependence on k). This represents a $O(1/\alpha)$ overhead to enforce 'honesty' for counting queries as well, since the returned value is now an actual count: it is always an integer fraction of ℓ, instead of an arbitrary real number due to added noise.

Proposition 13 *Using ℓ SCQ's to estimate each counting query is an (α, β)-accurate mechanism for k counting queries if $\ell \ge \frac{2\log(4k/\beta)}{\alpha^2}$ and $n = \Omega\left(\frac{\sqrt{k\log k}\log^{3/2}(\frac{1}{\alpha\beta})}{\alpha^3}\right)$.*

Proof The mechanism, for each query q, will query the SCQ mechanism $\mathcal{M}$ described in Section 4 ℓ times with the query q, and return the average, call this a_q. Note that $\mathbb{E}[a_q] = \mathbb{E}[\mathcal{M}(q)]$. Since each SCQ is independent of each other, a Hoeffding bound gives $\mathbb{P}[|a_q - \mathbb{E}[a_q]| \geq \alpha/2] \leq 2e^{-\ell\alpha^2/2} \leq \beta/2k$ when $\ell \geq \frac{2\log(4k/\beta)}{\alpha^2}$. Using Theorem 11, as long as $n = \Omega\left(\frac{\sqrt{k\ell}\log(\frac{1}{\alpha\beta})}{\alpha^2}\right)$, we have that $\mathbb{P}[\max_q |\mathbb{E}[\mathcal{M}(q)] - q(D)| \geq \alpha/2] \leq \beta/2$, over all $k\ell$ queries. Then the union bound implies that

$$\begin{aligned}\mathbb{P}[\max_q |a_q - q(D)| \geq \alpha] &\leq \mathbb{P}[\max_q |a_q - \mathbb{E}[\mathcal{M}(q)]| + |\mathbb{E}[\mathcal{M}(q)] - q(D)| \geq \alpha] \\ &\leq \beta/2 + \beta/2 \leq \beta,\end{aligned}$$

completing the proof. ■

Meanwhile, it is possible to use a mechanism for counting queries to attempt to answer SCQ's, but it has higher sample complexity than the mechanism for SCQ's proposed above. Indeed, there is the naïve approach that ignores time constraints by first computing $q(S)$ exactly, adding noise to obtain a value $\tilde{a}_q$, and then returning 1 with probability $\tilde{a}_q$ and 0 otherwise. For this mechanism we obtain an (ϵ, δ)-private mechanism to release k SCQ's that is (α, β)-accurate with respect to a fixed sample S of size n so long as $n > \frac{2\sqrt{2k\log(1/\delta)}\log(1/\beta)}{\alpha\epsilon}$, which is strictly worse than the mechanism for SCQ's we actually use. This motivates our approach to SCQ's.

6. An Application to Convex Optimization

We now show how to use our fast mechanism for statistical queries to get improved responses to adaptive convex optimization queries. To minimize a loss function $\mathcal{L}$, we will perform gradient descent but we calculate each coordinate $j \in [d]$ of each gradient using Algorithm 1 via the statistical query $q_{t-1,j}(S) = \nabla\mathcal{L}(S, x_{t-1})^{(j)}$, as described in Algorithm 4. The mechanism, recall, draws a random subsample S_ℓ and adds independent noise which we'll call b, so that it returns $\tilde{\nabla}\mathcal{L}(S, x_{t-1})^{(j)} := \nabla\mathcal{L}(S_\ell, x_{t-1})^{(j)} + b_{j,t-1}$. We may abbreviate $\tilde{\nabla}\mathcal{L}(S, x_t)$ as $\tilde{\nabla}\mathcal{L}(x_t)$, or $\tilde{\nabla}_t$. We then repeat Algorithm 4 k times, once for each convex optimization query.

To do this, we need to assume the restriction of the gradient to each coordinate $\nabla\mathcal{L}(S, x)^{(j)}$ is a statistical query. If this is the case, we call such a gradient *statistical*. This is not a strong assumption: it is the case when for example the loss is of the form $\mathcal{L}(S, x) = \frac{1}{|S|}\sum_{s\in S}\ell(s, x)$ for $\ell : X \times \Theta \to \mathbb{R}$ and $\nabla\ell \in [0, 1]$.[3]

We first show that the *expected excess loss* $\mathbb{E}_{S,\mathcal{M},\mathcal{A}}[\mathcal{L}(S, x) - \min_{x\in\Theta}\mathcal{L}(S, x)]$ for x the output of Algorithm 4 is small for convex functions.

Theorem 14 *For each $i \in [k]$, let $\mathcal{L}_i$ be differentiable and convex, let $\nabla\mathcal{L}_i$ be statistical, for any $x \in \Theta$, $\mathbb{E}_{S,S'\sim S}[\|\nabla\mathcal{L}_i(S', x)\|^2] \leq G^2$, and finally, for any $x, y \in \Theta$, $\|x - y\|^2 \leq D^2$. Then there is a mechanism that answers k adaptive optimization queries $\mathcal{L}_i$ each with expected excess loss α if $n = \tilde{O}\left(\frac{d^{3/2}\sqrt{k}}{\alpha^5}\right)$ in a total of $\tilde{O}\left(\frac{dk}{\alpha^2}\right)$ calls to Algorithm 1 using parameter $\ell = \tilde{O}\left(\frac{d}{\alpha^4}\right)$ and $\tilde{O}\left(\frac{1}{\alpha^2}\right)$ iterations of gradient descent per query.*

3. This last requirement may be weakened so that we just require $\nabla\ell$ to be bounded (which happens when X and Θ are compact, for example). The stronger requirement for being in $[0, 1]$ is because, for convenience, we also required this of statistical queries themselves.

Algorithm 4 Gradient descent with an adaptive mechanism for gradients

Parameters: Loss function $\mathcal{L}$, Mechanism $\mathcal{M}$, learning rate η
Input: number of rounds T, initial point x_0
  for t in $[T]$ **do**
    for j in $[d]$ **do**
      $q_{t-1,j}(S) := \nabla\mathcal{L}(S, x_{t-1})^{(j)}$
      Receive response $a_j := \mathcal{M}(q_{t-1,j}, S)$
    end for
    $\tilde{\nabla}\mathcal{L}(S, x_{t-1}) := (a_1, \ldots, a_d)$
    $x_t := x_{t-1} - \eta\tilde{\nabla}\mathcal{L}(S, x_{t-1})$
  end for
  return $\frac{1}{T}\sum_t x_t$.

Since $\mathcal{L}_i$ is convex, we have $\sum_{t=1}^{T}\mathbb{E}[\mathcal{L}_i(x_t) - \mathcal{L}_i(x^*)] \leq \sum_{t=1}^{T}\mathbb{E}[\langle\nabla_t, x_t - x^*\rangle]$, where $x^* = \arg\min_{x\in\Theta}\mathcal{L}_i(x)$. Then we can bound each term on the right-hand side using the fact that $\mathbb{E}_{t-1}[\nabla_t^{(j)}] \leq \mathbb{E}_{t-1}[\tilde{\nabla}_t^{(j)}] + \tilde{O}\left(\frac{R^{1/4}}{\sqrt{n}} + \frac{1}{\sqrt{\ell}}\right)$, i.e. Theorem 8. Proofs may be found in Appendix D.

We can boost this to a high-probability result by running the gradient-descent algorithm $O(\log(k/\beta))$ times and use the exponential mechanism to pick the best run among them, similarly to previous work that does this kind of boosting (Bassily et al., 2014).

Corollary 15 *For each $i \in [k]$, let $\mathcal{L}_i$ be differentiable and convex, let $\nabla\mathcal{L}_i$ be statistical, for any $x \in \Theta$, $\mathbb{E}_{S,S'\sim S}[\|\nabla\mathcal{L}_i(S', x)\|^2] \leq G^2$, and finally, for any $x, y \in \Theta$, $\|x - y\|^2 \leq D^2$. Then there is an (α, β)-accurate mechanism that answers k adaptive optimization queries $\mathcal{L}_i$ when $n = \tilde{O}\left(\frac{d^{3/2}\sqrt{k}\log(k/\beta)}{\alpha^5}\right)$ in a total of $\tilde{O}\left(\frac{dk\log(k/\beta)}{\alpha^2}\right)$ calls to Algorithm 1 using parameter $\ell = \tilde{O}\left(\frac{d}{\alpha^4}\right)$ and $\tilde{O}\left(\frac{\log(k/\beta)}{\alpha^2}\right)$ iterations of gradient descent per query.*

We also show an equivalent result holds when the loss function is not only convex but strongly convex (again the proof is in Appendix D).

Corollary 16 *For each $i \in [k]$, let $\mathcal{L}_i$ be differentiable and H-strongly convex, let $\nabla\mathcal{L}_i$ be statistical, and for any $x \in \Theta$, $\mathbb{E}_{S,S'\sim S}[\|\nabla\mathcal{L}_i(S', x)\|^2] \leq G^2$. Then there is an (α, β)-accurate mechanism for k adaptive optimization queries $\mathcal{L}_i$ when $n = \tilde{O}\left(\frac{d^{3/2}\sqrt{k}\log(k/\beta)}{\alpha^{5/2}}\right)$ in a total of $\tilde{O}\left(\frac{dk\log(k/\beta)}{\alpha}\right)$ calls to Algorithm 1 using parameter $\ell = \tilde{O}\left(\frac{d}{\alpha^2}\right)$ and $\tilde{O}\left(\frac{\log(k/\beta)}{\alpha}\right)$ iterations of gradient descent per query.*

Acknowledgments

Benjamin Fish was supported in part by the NSF EAPSI fellowship and NSF grant IIS-1526379. Lev Reyzin was supported in part by NSF grants IIS-1526379 and CCF-1848966. Benjamin Rubinstein acknowledges support of the Australian Research Council (DP150103710).

References

Borja Balle, Gilles Barthe, and Marco Gaboardi. Privacy amplification by subsampling: Tight analyses via couplings and divergences. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, pages 6280–6290, 2018.

Ziv Bar-Yossef. *The complexity of massive data set computations*. PhD thesis, University of California, Berkeley, 2002.

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.

Raef Bassily, Kobbi Nissim, Adam D. Smith, Thomas Steinke, Uri Stemmer, and Jonathan Ullman. Algorithmic stability for adaptive data analysis. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 1046–1059, 2016.

Avrim Blum, Merrick L. Furst, Jeffrey C. Jackson, Michael J. Kearns, Yishay Mansour, and Steven Rudich. Weakly learning DNF and characterizing statistical query learning using Fourier analysis. In *Proceedings of the Twenty-Sixth Annual ACM Symposium on Theory of Computing, 23-25 May 1994, Montréal, Québec, Canada*, pages 253–262, 1994.

Avrim Blum, Katrina Ligett, and Aaron Roth. A learning theory approach to non-interactive database privacy. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, British Columbia, Canada, May 17-20, 2008*, pages 609–618, 2008.

Mark Bun, Jonathan Ullman, and Salil P. Vadhan. Fingerprinting codes and the price of approximate differential privacy. In *Proceedings of the 46th Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 1–10, 2014.

Mark Bun, Kobbi Nissim, Uri Stemmer, and Salil P. Vadhan. Differentially private release and learning of threshold functions. In *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, pages 634–649, 2015.

Olivier Devolder, François Glineur, and Yurii Nesterov. First-order methods with inexact oracle: the strongly convex case. CORE Discussion Papers 2013016, Université catholique de Louvain, Center for Operations Research and Econometrics (CORE), 2013.

Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014. doi: 10.1561/0400000042. URL https://doi.org/10.1561/0400000042.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam D. Smith. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006*, pages 265–284, 2006.

Cynthia Dwork, Guy N. Rothblum, and Salil P. Vadhan. Boosting and differential privacy. In *51th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2010, October 23-26, 2010, Las Vegas, Nevada, USA*, pages 51–60, 2010.

Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Roth. Generalization in adaptive data analysis and holdout reuse. In *Advances in Neural Information Processing Systems 28, December 7-12, 2015, Montreal, Quebec, Canada*, pages 2350–2358, 2015a.

Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Leon Roth. Preserving statistical validity in adaptive data analysis. In *Proceedings of the Forty-Seventh Annual ACM Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 117–126, 2015b.

Vitaly Feldman, Elena Grigorescu, Lev Reyzin, Santosh Srinivas Vempala, and Ying Xiao. Statistical algorithms and a lower bound for detecting planted cliques. *Journal of the ACM*, 64(2):8:1–8:37, 2017a.

Vitaly Feldman, Cristobal Guzman, and Santosh Vempala. Statistical query algorithms for mean vector estimation and stochastic convex optimization. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, pages 1265–1277, 2017b.

Andrew Gelman and Eric Loken. The statistical crisis in science data-dependent analysis—a "garden of forking paths"—explains why many statistically significant comparisons don't hold up. *American Scientist*, 102(6):460–465, 2014.

Nadav Golbandi, Yehuda Koren, and Ronny Lempel. Adaptive bootstrapping of recommender systems using decision trees. In *Proceedings of the Forth International Conference on Web Search and Web Data Mining, WSDM 2011, Hong Kong, China, February 9-12, 2011*, pages 595–604, 2011.

Zach Jorgensen, Ting Yu, and Graham Cormode. Conservative or liberal? Personalized differential privacy. In *31st IEEE International Conference on Data Engineering, ICDE 2015, Seoul, South Korea, April 13-17, 2015*, pages 1023–1034, 2015.

Shiva Prasad Kasiviswanathan, Homin K. Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam D. Smith. What can we learn privately? In *49th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2008, October 25-28, 2008, Philadelphia, PA, USA*, pages 531–540, 2008.

Michael J. Kearns. Efficient noise-tolerant learning from statistical queries. *Journal of the ACM*, 45 (6):983–1006, 1998.

Georgios Kellaris and Stavros Papadopoulos. Practical differential privacy via grouping and smoothing. *Proceedings of the VLDB Endowment*, 6(5):301–312, 2013.

Ariel Kleiner, Ameet Talwalkar, Purnamrita Sarkar, and Michael I. Jordan. The big data bootstrap. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK, June 26 - July 1, 2012*, 2012.

Jason D. Lee, Dennis L. Sun, Yuekai Sun, and Jonathan E. Taylor. Exact post-selection inference, with application to the lasso. *Annals of Statistics*, 44(3):907–927, 06 2016. doi: 10.1214/15-AOS1371. URL http://dx.doi.org/10.1214/15-AOS1371.

Bing-Rong Lin, Ye Wang, and Shantanu Rane. On the benefits of sampling in privacy preserving statistical analysis on distributed databases. *arXiv preprint arXiv:1304.4613*, 2013.

Kobbi Nissim, Adam D. Smith, Thomas Steinke, Uri Stemmer, and Jonathan Ullman. The limits of post-selection generalization. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, pages 6402–6411, 2018.

Ryan M. Rogers, Aaron Roth, Adam D. Smith, and Om Thakkar. Max-information, differential privacy, and post-selection hypothesis testing. In *57th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA*, pages 487–494, 2016.

Ohad Shamir and Tong Zhang. Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, pages 71–79, 2013.

Thomas Steinke and Jon Ullman. Between pure and approximate differential privacy. In *Theory and Practice of Differential Privacy (TPDP 2015), London, UK*, 2015a. URL http://tpdp.computing.dundee.ac.uk/abstracts/TPDP_2015_3.pdf.

Thomas Steinke and Jonathan Ullman. Interactive fingerprinting codes and the hardness of preventing false discovery. In *Proceedings of the 28th Conference on Learning Theory, COLT 2015, Paris, France, July 3-6, 2015*, pages 1588–1628, 2015b.

Jeffrey Scott Vitter. Faster methods for random sampling. *Communications of the ACM*, 27(7): 703–718, 1984.

C. K. Wong and Malcolm C. Easton. An efficient method for weighted sampling without replacement. *SIAM Journal of Computing*, 9(1):111–113, 1980.

Houping Xiao, Jing Gao, Qi Li, Fenglong Ma, Lu Su, Yunlong Feng, and Aidong Zhang. Towards confidence in the truth: A bootstrapping based truth discovery approach. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pages 1935–1944, 2016.

Ke Yang. On learning correlated Boolean functions using statistical queries. In *Proceedings of the 12th International Conference on Algorithmic Learning Theory ALT 2001, Washington, DC, USA, November 25-28, 2001*, pages 59–76, 2001.

Appendix A. Answering Statistical Queries

In this section, we point out how Algorithm 1 may be used with appropriate sub-sample size and privacy parameters to answer statistical queries both quickly and accurately:

Proof of Theorem 5, part 2 Since the Laplace mechanism receives a sample S_ℓ of size ℓ, output a_q can be bounded with the standard accuracy result for the Laplace mechanism ensuring ϵ'-privacy: $\mathbb{P}[|a_q - q(S_\ell)| \geq \alpha/2] \leq e^{-\frac{\alpha\epsilon'\ell}{2}}$. We can bound this above by $\frac{\beta}{2k}$ provided $\epsilon' \geq \frac{\log(2k/\beta)}{\ell\alpha}$. Recalling that $\epsilon' = \frac{\epsilon n}{4\ell\sqrt{2k\log(1/\delta)}}$, this occurs when

$$n \geq \frac{4\sqrt{2k\log(1/\delta)}\log(2k/\beta)}{\alpha\epsilon}.$$

From the Hoeffding bound, we also get that $\mathbb{P}[|q(S_\ell) - q(S)| \geq \alpha/2] \leq 2e^{-\frac{\alpha^2\ell}{2}}$. Once again we can bound this above by $\frac{\beta}{2k}$ so long as $\ell \geq \frac{2\log(4k/\beta)}{\alpha^2}$.

Thus for all q, $\mathbb{P}[|a_q - q(S)| \geq \alpha] \leq \mathbb{P}[|a_q - q(S_\ell)| \geq \alpha/2] + \mathbb{P}[|q(S_\ell) + q(S)| \geq \alpha/2] \leq \beta/k$. The union bound immediately yields (α, β)-accuracy on the sample over all k queries. From either Proposition 6 or 7, we also have that on a single query this mechanism is $\left(2\frac{\ell}{n}\epsilon'\right)$-private, where $2\frac{\ell}{n}\epsilon' = \frac{\epsilon}{2\sqrt{2k\log(1/\delta)}}$. Thus by the adaptive composition lemma (see Appendix E), the mechanism over the course of k queries is ϵ-private. The proof is concluded by applying Theorem 4. ■

Appendix B. Transfer Theorems

In this section, we prove our required transfer theorems, which state that if a mechanism is accurate on the sample and private, it will also be accurate on the distribution.

B.1. Transfer Theorem for Sampling Counting Queries

We return to the proof of Theorem 10, our transfer theorem for sampling counting queries.

First, we show the monitor is private.

Lemma 17 *If $\mathcal{M}$ is (ϵ, δ)-private for k queries, then $\mathcal{W}_D$ is $(2\epsilon, \delta)$-private.*

Proof A single perturbation to $\mathbf{S}$ can only change one S_t, for some t. Then since $\mathcal{M}$ on S_t is (ϵ, δ)-private, $\mathcal{M}$ remains (ϵ, δ)-private over the course of the T simulations. Since $\mathcal{A}$ uses only the outputs of $\mathcal{M}$, $\mathcal{A}$ is just post-processing $\mathcal{M}$, and therefore it is (ϵ, δ)-private as well: releasing all of $\mathcal{R}$ remains (ϵ, δ)-private.

Since the sensitivity of u is $\Delta = 1/n$, the monitor is just using the exponential mechanism to release some $r \in \mathcal{R}$, which is ϵ-private. The standard composition theorem completes the proof. ■

We will also need some of the tools used by Bassily et al. (2016). First, for a monitoring algorithm $\mathcal{W}$, the expected value of the outputted query on the sample will be close to its expected value over the distribution—formalizing a connection between privacy and stability.

Lemma 18 (Bassily et al., 2016) *Let $\mathcal{W} : (X^n)^T \to Q \times [T]$ be (ϵ, δ)-private where Q is the class of statistical queries. Let $S_i \sim D^n$ for each of $i \in [T]$ and $\mathbf{S} = \{S_1, \ldots, S_T\}$. Then*

$$|\mathbb{E}_{\mathbf{S},\mathcal{W}}[q(D)|(q,t) = \mathcal{W}(\mathbf{S})] - \mathbb{E}_{\mathbf{S},\mathcal{W}}[q(S_t)|(q,t) = \mathcal{W}(\mathbf{S})]| \leq e^\epsilon - 1 + T\delta.$$

We will also use a convenient form of accuracy bound for the exponential mechanism.

Lemma 19 (Bassily et al., 2016) *Let $\mathcal{R}$ be a finite set, $f : \mathcal{R} \to \mathbb{R}$ a function, and $\eta > 0$. Define a random variable X on $\mathcal{R}$ by $\mathbb{P}[X = r] = e^{\eta f(r)}/C$, where $C = \sum_{r\in\mathcal{R}} e^{\eta f(r)}$. Then $\mathbb{E}[f(X)] \geq \max_{r\in\mathcal{R}} f(r) - \frac{1}{\eta}\log|\mathcal{R}|$.*

Now we can provide the proof of the transfer theorem:

Proof of Theorem 10 Consider the results for simulating T times the interaction between $\mathcal{M}$ and $\mathcal{A}$. Suppose for the sake of contradiction that $\mathcal{M}$ is not (α, β)-accurate on D. Then for every i in $[k]$ and t in T, since $|\mathbb{E}_{\mathcal{M}}[\mathcal{M}(q_{t,i})] - q(S_t)| \leq \alpha/2$, we have

$$\mathbb{P}_{S_t,\mathcal{M},\mathcal{A}}\left[\max_i |q_{t,i}(S_t) - q_{t,i}(D)| > \alpha/2\right] > \beta.$$

Call some q and t that achieves the maximum $|q(S_t) - q(D)|$ over the T independent rounds of $\mathcal{M}$ and $\mathcal{A}$ interacting, as $\mathcal{W}_D$ does (Algorithm 2), by q_w and t_w. Since each round t is independent, the probability that $|q_w(S_{t_w}) - q_w(D)| \leq \alpha/2$ is then no more than $(1-\beta)^T$. Then using Markov's inequality immediately grants us that

$$\mathbb{E}_{\mathbf{S},\mathcal{W}_D}\left[|q_w(S_{t_w}) - q_w(D)|\right] > \frac{\alpha}{2}(1 - (1-\beta)^T). \tag{1}$$

Let $\Gamma = \mathbb{E}_{\mathbf{S},\mathcal{W}_D}\left[|q^*(S_{t^*}) - q^*(D)| : (q^*, t^*) = \mathcal{W}_D(\mathbf{S})\right]$.

Setting $f(r) = u(\mathbf{S}, r)$, Lemma 19 implies that under the exponential mechanism, we have

$$\mathbb{E}[|q^*(S_{t^*}) - q^*(D)| : (q^*, t^*) = \mathcal{W}_D(\mathbf{S})] \geq |q_w(S_{t_w}) - q_w(D)| - \frac{2}{\epsilon n}\log(kT).$$

Taking the expected value of both sides with respect to $\mathbf{S}$ and the randomness of the rest of $\mathcal{W}_D$, we obtain

$$\Gamma \geq \mathbb{E}_{\mathbf{S},\mathcal{W}_D}[|q_w(S_{t_w}) - q_w(D)|] - \frac{2}{\epsilon n}\log(kT) > \frac{\alpha}{2}(1 - (1-\beta)^T) - \frac{2}{\epsilon n}\log(kT), \tag{2}$$

which follows from employing Equation (1). On the other hand, suppose that $\mathcal{M}$ is (ϵ, δ)-private for some $\epsilon, \delta > 0$. Then by Lemma 17, $\mathcal{W}_D$ is $(2\epsilon, \delta)$-private, and then in turn Lemma 18 implies that

$$\Gamma \leq e^{2\epsilon} - 1 + T\delta. \tag{3}$$

We will now ensure $\Gamma \geq \alpha/8$, via (2), and $\Gamma \leq \alpha/8$, via (3), yielding a contradiction. Set $T = \lfloor \frac{1}{\beta} \rfloor$ and $\delta = \frac{\alpha\beta}{16}$. Then

$$e^{2\epsilon} - 1 + T\delta \leq e^{2\epsilon} - 1 + \alpha/16 \leq \alpha/8$$

when $e^{2\epsilon} - 1 \leq \alpha/16$, which in turn is satisfied when $\epsilon \leq \alpha/64$, since $0 \leq \alpha \leq 1$.

On the other side, $1 - (1-\beta)^{\lfloor \frac{1}{\beta} \rfloor} \geq 1/2$. Then it suffices to set ϵ such that $\frac{2}{\epsilon n}\log(kT) \leq \alpha/8$. Thus we need ϵ such that

$$\frac{16\log(k/\beta)}{\alpha n} \;\leq\; \epsilon \;\leq\; \alpha/64.$$

Such an ϵ exists, since we explicitly required $n \geq \frac{1024\log(k/\beta)}{\alpha^2}$. ■

B.2. Transfer Theorem for Statistical Queries in Expectation

We also need a transfer theorem for Theorem 8.

Proposition 20 *Consider any possibility for the simulation between $\mathcal{A}$ and $\mathcal{M}$ up to the first $t-1$ rounds. Denoting the expectation while conditioning on any such possibility $E_{t-1}[\cdot]$, we have for any round $i \geq t$, if $\mathcal{M}$ is $(\alpha/8, \alpha/4)$-private for $\alpha \leq 1$, and $E_{t-1,S,\mathcal{M},\mathcal{A}}[|q_i(S) - a_i|] \leq \alpha/2$, then*

$$E_{t-1,S,\mathcal{M},\mathcal{A}}[|q_i(D) - a_i|] \leq \alpha.$$

Algorithm 5 Monitor $\mathcal{W}$

Parameters: Mechanisms $\mathcal{M}$ and $\mathcal{A}$, index i, and initial sequence of queries $q_1, \ldots, q_{t-1}$ and responses $a_1, \ldots, a_{t-1}$

Input: Sample S

Set the internal states of $\mathcal{M}(S)$ and $\mathcal{A}$ to be what they would be if the resulting simulation had produced $q_1, \ldots, q_{t-1}$ and $a_1, \ldots, a_{t-1}$.

Now simulate $\mathcal{M}(S)$ and $\mathcal{A}$ interacting starting in those states for $i - t + 1$ rounds. Let $q_t, \ldots, q_i$ be the resulting queries.

return q_i.

Proof Suppose by way of contradiction that $\mathbb{E}_{t-1,S,\mathcal{M},\mathcal{A}}[|q_i(D) - a_i|] > \alpha$. Note the monitor $\mathcal{W}$, given in Algorithm 5, simply outputs q_i, conditioned on $q_1, \ldots, q_{t-1}$ and $a_1, \ldots, a_{t-1}$ being the initial sequence of queries and responses, so

$$\begin{aligned}|\mathbb{E}_{S,\mathcal{W}}[q(D) - q(S) | q = \mathcal{W}(S)]| =& |\mathbb{E}_{t-1,S,\mathcal{M},\mathcal{A}}[q_i(S) - q_i(D)]| \\ \geq& |\mathbb{E}_{t-1,S,\mathcal{M},\mathcal{A}}[q_i(D) - a_i]| - |\mathbb{E}_{t-1,S,\mathcal{M},\mathcal{A}}[q_i(S) - a_i]| \\ >& \alpha - \alpha/2 = \alpha/2.\end{aligned}$$

Since the monitor $\mathcal{W}$ only outputs q_i, which is post-processing from a private mechanism $\mathcal{M}$, $\mathcal{W}$ remains $(\alpha/8, \alpha/4)$-private. Therefore by Lemma 18, $|\mathbb{E}_{S,\mathcal{W}}[q(D) - q(S) | q = \mathcal{W}(S)]| \leq e^\epsilon - 1 + \delta \leq \alpha/2$ with the above values of ϵ and δ for $\alpha \leq 1$. ■

Appendix C. Lower Bound

Proposition 9 *Suppose for any sequence $q_1, \ldots, q_k$ of k statistical queries chosen non-adaptively, there is a mechanism $\mathcal{M}$ that is $(\alpha, 1/5)$-accurate on the uniform distribution over a universe X with $|X| \geq \frac{2\log(10)}{\alpha^2}$. Then $\mathcal{M}$ must evaluate the queries on at least $\Omega(k/\alpha^2)$ points.*

Proof Consider a distribution Q over statistical queries defined by the following process: For each $i \in [k]$, let $p_i = 1/2$ independently with probability $1/2$ and $p_i = 1/2 + 4\alpha$ with probability $1/2$. Then set $q_i(x) = 1$ with probability p_i independently for each $x \in X$. Now suppose $\mathcal{M}$ is $(\alpha, 1/5)$-accurate on the uniform distribution U. Since $\mathcal{M}$ is $(\alpha, 1/5)$-accurate for any set of k statistical queries, in particular it remains that accurate for a random set of k queries drawn from Q for any i:

$$P_{Q,\mathcal{M}}[|q_i(U) - a_i| > \alpha] \leq 1/5.$$

From the Hoeffding bound and our assumption on the size of $|X|$, we also have

$$P_Q[|q_i(U) - p_i| > \alpha/2] \leq 2e^{-|X|\alpha^2/2} \leq 1/5.$$

Thus with probability at least $3/5$, $|a_i - p_i| \leq 3\alpha/2$. For the values $q_i(s_1), \ldots, q_i(s_m)$ that $\mathcal{M}$ computed, define a mechanism $A(q_i(s_1), \ldots, q_i(s_m)) = 1/2$ if $a_i \leq 1/2 + 2\alpha$ and otherwise $A(q_i(s_1), \ldots, q_i(s_m)) = 1/2 + 4\alpha$. Recall $q_i(s_1), \ldots, q_i(s_m)$ are i.i.d. draws from a coin with bias either $1/2$ or $1/2 + 4\alpha$. Thus with probability at least $3/5$, A distinguishes between the two coins. This is well known to require $m \geq \Omega(1/\alpha^2)$ (e.g. see Bar-Yossef (2002)), which in turn implies that $\mathcal{M}$ computed the value of queries at least $\Omega(k/\alpha^2)$ times. ■

Appendix D. Convex Optimization

We now return to the omitted proofs in Section 6. Bounding regret here is similar to typical analyses, but is complicated by one major difference: A typical assumption in stochastic gradient descent is that the oracle returning the oracle for the gradient is unbiased, so that $\mathbb{E}[\tilde{\nabla}\mathcal{L}] = \nabla\mathcal{L}$ (e.g. Shamir and Zhang, 2013), whereas here $\mathbb{E}[\tilde{\nabla}\mathcal{L}]$ is only guaranteed to be close to the true gradient $\mathcal{L}$. We take advantage of (strong) convexity to show that for sufficiently large sample size, gradient descent still converges sufficiently quickly.

Theorem 21 *For each $i \in [k]$, let $\mathcal{L}_i$ be differentiable, H-strongly convex, let $\nabla\mathcal{L}_i$ be statistical, and for any $x \in \Theta$, $\mathbb{E}_{S,S' \sim S}[\|\nabla\mathcal{L}_i(S', x)\|^2] \leq G^2$. Then there is a mechanism that answers k adaptive optimization queries $\mathcal{L}_i$ each with expected excess risk α if $n = \tilde{O}\left(\frac{d^{3/2}\sqrt{k}}{\alpha^{5/2}}\right)$ in a total of $\tilde{O}\left(\frac{dk}{\alpha}\right)$ calls to Algorithm 1 using parameter $\ell = \tilde{O}\left(\frac{d}{\alpha^2}\right)$ and $\tilde{O}\left(\frac{1}{\alpha}\right)$ iterations of gradient descent per query.*

Proof We use Algorithm 4 to answer k optimization queries, which in turn uses our statistical query oracle (Algorithm 1) to get each component of $\nabla\mathcal{L}_i$, for a total of $R := k \cdot T \cdot d$ rounds, where T is the number of iterations per optimization. For each optimization query, we now bound regret. As is standard, we pick $x^* = \arg\min_{x \in \Theta} \mathcal{L}_i(x)$ to plug in to the definition of strong convexity to get, rearranging,

$$\mathbb{E}[\mathcal{L}_i(x_t) - \mathcal{L}_i(x^*)] \leq \mathbb{E}[\langle \nabla_t, x_t - x^* \rangle] - \frac{H}{2}\mathbb{E}[\|x_t - x^*\|^2].$$

Again following the standard analysis,

$$\begin{aligned}\|x_{t+1} - x^*\|^2 =& \|\Pi(x_t - \eta_t\tilde{\nabla}_t) - x^*\| \leq \|x_t - \eta_t\tilde{\nabla}_t - x^*\|^2 \\ \leq& \|x_t - x^*\|^2 + \eta_t^2\|\tilde{\nabla}_t\|^2 - 2\eta_t\langle\tilde{\nabla}_t, x_t - x^*\rangle.\end{aligned}$$

In other words,

$$\langle\tilde{\nabla}_t, x_t - x^*\rangle \leq \frac{\|x_t - x^*\|^2 - \|x_{t+1} - x^*\|^2}{2\eta_t} + \frac{\eta_t}{2}\|\tilde{\nabla}_t\|^2.$$

Moreover, we can upper-bound $\mathbb{E}[\|\tilde{\nabla}_t\|^2]$ since $\tilde{\nabla}_t = \nabla\mathcal{L}_i(S_\ell, x_t) + b_t$, where b_t is the noise vector.

$$\begin{aligned}\mathbb{E}[\|\tilde{\nabla}_t\|^2] =& \mathbb{E}[\|\nabla\mathcal{L}_i(S_\ell, x_t)\|^2] + \mathbb{E}[\|b_t\|^2] + 2\mathbb{E}[\langle\nabla\mathcal{L}_i(S_\ell, x_t), b_t\rangle] \\ \leq& G^2 + 2d\sigma^2 = G^2 + \frac{cdR\log(1/\alpha')}{n^2\alpha'^2},\end{aligned}$$

where σ^2 is the variance of the noise, α' is the accuracy of Algorithm 1, and c is a sufficiently large constant. Note $\mathbb{E}[\langle\nabla\mathcal{L}_i(S_\ell, x_t), b_t\rangle] = 0$ because b_t is independent of both S_ℓ and x_t.

Now, using the bounds on our oracle, we upper-bound $\langle\nabla_t, x_t - x^*\rangle$ using $\langle\tilde{\nabla}_t, x_t - x^*\rangle$.

Using $\mathbb{E}_{t-1}[\cdot]$ to denote the expectation conditioned on all of the previous $t-1$ iterations, the promise of our mechanism (Theorem 8) is that we can guarantee that for each coordinate j, $\mathbb{E}_{t-1}[\nabla_t^{(j)}] \leq \mathbb{E}_{t-1}[\tilde{\nabla}_t^{(j)}] + \alpha'$, where

$$\alpha' = \tilde{O}\left(\frac{R^{1/4}}{\sqrt{n}} + \frac{1}{\sqrt{\ell}}\right).$$

Then

$$\begin{aligned}\mathbb{E}[\langle\nabla_t, x_t - x^*\rangle] =& \sum_i \mathbb{E}[\mathbb{E}_{t-1}[\nabla_t^{(j)}(x_t - x^*)^{(j)}]] \\ \leq& \sum_i \mathbb{E}[\mathbb{E}_{t-1}[(\tilde{\nabla}_t^{(j)} + \alpha')(x_t - x^*)^{(j)}]] \\ =& \mathbb{E}[\langle\tilde{\nabla}_t, x_t - x^*\rangle] + \alpha'\mathbb{E}\left[\sum_i (x_t - x^*)^{(j)}\right] \\ \leq& \mathbb{E}[\langle\tilde{\nabla}_t, x_t - x^*\rangle] + \alpha'\mathbb{E}[\|x_t - x^*\|_1] \\ \leq& \mathbb{E}[\langle\tilde{\nabla}_t, x_t - x^*\rangle] + \alpha'\sqrt{d}\,\mathbb{E}[\|x_t - x^*\|_2].\end{aligned}$$

The first equality conditions on the first $t-1$ rounds and then expands the inner product. The first inequality follows because once we condition on the first $t-1$ rounds, ∇_t and x_t are independent, so we can use the mechanism's guarantee. $\tilde{\nabla}_t$ and x_t are also independent when conditioned on the first $t-1$ rounds, from which the second equality follows. The last inequality follows from Cauchy-Schwartz.

Note further that $\mathbb{E}[\|x_t - x^*\|_2] \leq 1 + \mathbb{E}[\|x_t - x^*\|_2^2]$, simply because either $\|x_t - x^*\|_2 \leq 1$ or $\|x_t - x^*\|_2 < \|x_t - x^*\|_2^2$. Thus

$$\mathbb{E}[\langle\nabla_t, x_t - x^*\rangle] \leq \mathbb{E}[\langle\tilde{\nabla}_t, x_t - x^*\rangle] + \alpha'\sqrt{d} + \alpha'\sqrt{d}\mathbb{E}[\|x_t - x^*\|^2].$$

Thus we have

$$\begin{aligned}&\sum_{t=1}^T \mathbb{E}[\mathcal{L}_i(x_t) - \mathcal{L}_i(x^*)] \\ \leq& \sum_{t=1}^T \left(\frac{(1+\alpha'\sqrt{d})E[\|x_t - x^*\|^2] - \mathbb{E}[\|x_{t+1} - x^*\|^2]}{2\eta_t} - \right. \\ &\left. \frac{H}{2}\mathbb{E}[\|x_t - x^*\|^2] + \frac{\eta_t}{2}\left(G^2 + \frac{cdR\log(1/\alpha')}{n^2\alpha'^2}\right) + \alpha'\sqrt{d}\right)\end{aligned}$$

$$\leq \frac{1}{2}\sum_{t=1}^{T}\mathbb{E}[\|x_t - x^*\|^2]\left(\frac{1+\alpha'\sqrt{d}}{\eta_t} - \frac{1}{\eta_{t-1}} - H\right) + \left(\frac{G^2}{2} + \frac{cdR\log(1/\alpha')}{2n^2\alpha'^2}\right)\left(\sum_{t=1}^{T}\eta_t\right) + \alpha'\sqrt{d}T.$$

Now if we set $\eta_t = \frac{2}{Ht}$, then $\frac{1+\alpha'\sqrt{d}}{\eta_t} - \frac{1}{\eta_{t-1}} - H \leq 0$ when $\alpha'\sqrt{d} \leq 1/t$.
Then setting $\alpha'\sqrt{d} \leq \frac{1}{T}$, the average loss is

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\mathcal{L}_i(x_t) - \mathcal{L}_i(x^*)] \leq& \frac{2}{HT}\left(\frac{G^2}{2} + \frac{cdR\log(1/\alpha')}{2n^2\alpha'^2}\right)\sum_{t=1}^{T}1/t + \alpha'\sqrt{d} \\ \leq& \frac{G^2}{H}\cdot\frac{1+\log(T)}{T} + \frac{cdR\log(1/\alpha')}{Hn^2\alpha'^2}\cdot\frac{1+\log(T)}{T} + \alpha'\sqrt{d}.\end{aligned}$$

Thus to show that the average loss is no more than α it suffices to show that $\frac{G^2}{H}\cdot\frac{1+\log(T)}{T} \leq \alpha/3$, $\alpha'\sqrt{d} \leq \alpha/3$, $\frac{cdR\log(1/\alpha')}{Hn^2\alpha'^2}\cdot\frac{1+\log(T)}{T} \leq \alpha/3$, and $\alpha'\sqrt{d} \leq 1/T$. For the first, it suffices to set $T = \tilde{O}\left(\frac{G^2}{\alpha H}\right)$. Then, as long as α is sufficiently small[4], it suffices so that $n = \tilde{O}\left(\frac{G^5}{H^{5/2}}\cdot\frac{d^{3/2}\sqrt{k}}{\alpha^{5/2}}\right)$ and $\ell = \tilde{O}\left(\frac{G^4}{H^2}\cdot\frac{d}{\alpha^2}\right)$. Finally, the number of times we need to compute a gradient over k rounds is $R = k\cdot T\cdot d = \tilde{O}\left(\frac{G^2kd}{H\alpha}\right)$. ■

Corollary 16 then follows by boosting this to a high-probability result via running the gradient-descent algorithm $\log(k/\beta)$ times and choosing the best run among them using the exponential mechanism.

We now turn to the proof of Theorem 14, which is restated here:

Theorem 14 *For each $i \in [k]$, let $\mathcal{L}_i$ be differentiable and convex, let $\nabla\mathcal{L}_i$ be statistical, for any $x \in \Theta$, $\mathbb{E}_{S,S'\sim S}[\|\nabla\mathcal{L}_i(S', x)\|^2] \leq G^2$, and finally, for any $x, y \in \Theta$, $\|x - y\|^2 \leq D^2$. Then there is a mechanism that answers k adaptive optimization queries $\mathcal{L}_i$ each with expected excess loss α if $n = \tilde{O}\left(\frac{d^{3/2}\sqrt{k}}{\alpha^5}\right)$ in a total of $\tilde{O}\left(\frac{dk}{\alpha^2}\right)$ calls to Algorithm 1 using parameter $\ell = \tilde{O}\left(\frac{d}{\alpha^4}\right)$ and $\tilde{O}\left(\frac{1}{\alpha^2}\right)$ iterations of gradient descent per query.*

Proof The proof is very similar to that of the proof of Theorem 21, using the same algorithm, except now we only have

$$\mathbb{E}[\mathcal{L}(x_t) - \mathcal{L}(x^*)] \leq \mathbb{E}[\langle\nabla_t, x_t - x^*\rangle].$$

But as before, we have

$$\mathbb{E}[\langle\nabla_t, x_t - x^*\rangle] \leq \mathbb{E}[\langle\tilde{\nabla}_t, x_t - x^*\rangle] + \alpha'\sqrt{d} + \alpha'\sqrt{d}\,\mathbb{E}[\|x_t - x^*\|^2],$$

$$\langle\tilde{\nabla}_t, x_t - x^*\rangle \leq \frac{\|x_t - x^*\|^2 - \|x_{t+1} - x^*\|^2}{2\eta_t} + \frac{\eta_t}{2}\|\tilde{\nabla}_t\|^2,$$

4. That is, α is sufficiently small as a function of d, G, and H. Or, we can instead assume G and H are absolute constants. Otherwise, the dependence of n on G and H is messier and we omit these calculations for the sake of brevity.

and

$$\mathbb{E}[\|\tilde{\nabla}_t\|^2] \leq G^2 + \frac{cdR\log(1/\alpha')}{n^2\alpha'^2},$$

for sufficiently large constant c. Then

$$\begin{aligned}
&\sum_{t=1}^{T}\mathbb{E}[\mathcal{L}(x_t)-\mathcal{L}(x^*)]\\
\leq&\sum_{t=1}^{T}\left(\frac{(1+\alpha'\sqrt{d})\mathbb{E}[\|x_t-x^*\|^2]-\mathbb{E}[\|x_{t+1}-x^*\|^2]}{2\eta_t}+\frac{\eta_t}{2}\left(G^2+\frac{cdR\log(1/\alpha')}{n^2\alpha'^2}\right)+\alpha'\sqrt{d}\right)\\
\leq&\frac{1}{2}\sum_{t=1}^{T}\mathbb{E}[\|x_t-x^*\|^2]\left(\frac{1+\alpha'\sqrt{d}}{\eta_t}-\frac{1}{\eta_{t-1}}\right)+\left(\frac{G^2}{2}+\frac{cdR\log(1/\alpha')}{2n^2\alpha'^2}\right)\left(\sum_{t=1}^{T}\eta_t\right)+\alpha'\sqrt{d}\cdot T\\
\leq&\frac{D^2}{2\eta_T}+\frac{D^2\alpha'\sqrt{d}}{2}\sum_{t=1}^{T}\frac{1}{\eta_t}+\left(\frac{G^2}{2}+\frac{cdR\log(1/\alpha')}{2n^2\alpha'^2}\right)\left(\sum_{t=1}^{T}\eta_t\right)+\alpha'\sqrt{d}\cdot T,
\end{aligned}$$

where the last inequality comes from upper-bounding $\|x_t-x^*\|^2$ by the diameter, and collapsing the telescoping series. Set $\eta_t=\frac{D}{G\sqrt{t}}$. This gives the average loss as

$$\begin{aligned}
\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\mathcal{L}(x_t)-\mathcal{L}(x^*)]\leq&\frac{DG}{2\sqrt{T}}+O\left(\frac{DG\alpha'\sqrt{dT}}{2}+\frac{DG}{2\sqrt{T}}+\frac{DdR\log(1/\alpha')\sqrt{T}}{Gn^2\alpha'^2}\right)+\alpha'\sqrt{d}\\
=&O\left(\frac{DG}{\sqrt{T}}+DG\alpha'\sqrt{dT}+\frac{DdR\log(1/\alpha')\sqrt{T}}{Gn^2\alpha'^2}+\alpha'\sqrt{d}\right).
\end{aligned}$$

It suffices to show that each of these four terms are upper-bounded by $\alpha/4$, in which case, for sufficiently small α,[5] we require $T\geq O(\frac{D^2G^2}{\alpha^2})$, $n\geq\tilde{O}\left(\frac{D^5G^5d^{3/2}\sqrt{k}}{\alpha^5}\right)$, and $\ell\geq\tilde{O}\left(\frac{D^4G^4d}{\alpha^4}\right)$. Thus the number of times we need to compute a gradient over k rounds is $R=k\cdot T\cdot d=\tilde{O}\left(\frac{D^2G^2kd}{\alpha^2}\right)$. ■

Appendix E. Differential Privacy Review

Differential privacy has several nice guarantees, among which is that it composes adaptively.

Lemma 22 (Adaptive composition; Dwork and Roth, 2014; Dwork et al., 2010) *Given parameters $0<\epsilon<1$ and $\delta>0$, to ensure $(\epsilon, k\delta'+\delta)$-privacy over k adaptive mechanisms, it suffices that each mechanism is (ϵ',δ')-private, where $\epsilon'=\frac{\epsilon}{2\sqrt{2k\log(1/\delta)}}$.*

We also have a post-processing guarantee:

Lemma 23 (Post-processing; Dwork and Roth, 2014) *Let $\mathcal{M}:X^n\to Z$ be an (ϵ,δ)-private mechanism and $f:Z\to Z'$ a (possibly randomized) algorithm. Then $f\circ\mathcal{M}$ is (ϵ,δ)-private.*

5. As in the proof of Theorem 21, this assumption is only required to write n as a function of D and G without having to resort to a much messier formula. Another alternative is to assume that D and G are absolute constants.