Sublinear-Time Adaptive Data Analysis

Benjamin Fish and **Lev Reyzin**
Department of Mathematics, Statistics,
& Computer Science
University of Illinois at Chicago
{bfish3,lreyzin}@uic.edu

Benjamin I. P. Rubinstein
School of Computing & Information Systems
University of Melbourne, Australia
benjamin.rubinstein@unimelb.edu.au

Abstract

The central aim of most fields of data analysis and experimental scientific investigation is to draw valid conclusions from a given data set. But when past inferences guide future inquiries into the same dataset, reaching valid inferences becomes significantly more difficult. In addition to avoiding the overfitting that can result from adaptive analysis, a data analyst often wants to use as little time and data as possible. A recent line of work in the theory community has established mechanisms that provide low generalization error on adaptive queries, yet there remain large gaps between established theoretical results and how data analysis is done in practice. Many practitioners, for instance, successfully employ bootstrapping and related sampling approaches in order to maintain validity and speed up analysis, but prior to this work, no theoretical analysis existed to justify employing such techniques in this adaptive setting.

In this paper, we show how these techniques can be used to provably guarantee validity while speeding up analysis. Through this investigation, we initiate the study of sub-linear time mechanisms to answer adaptive queries into datasets. Perhaps surprisingly, we describe mechanisms that provide an exponential speed-up per query over previous mechanisms, without needing to increase the total amount of data needed for low generalization error. We also provide a method for achieving statistically-meaningful responses even when the mechanism is only allowed to see a constant number of samples from the data per query.

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 pre-determined sequence of analyses [19]. In practice, most data analysis is adaptive: 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' [12]. While basing each analysis on new data drawn from the same distribution might appear an appealing solution, repeated data collection and analysis time can be prohibitively costly.

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. However, the results in this burgeoning field of adaptive data analysis have ignored bootstrapping and related sampling techniques, even though it has enjoyed widespread and successful use in practice in a variety of settings [18, 26], including in adaptive settings [13]. 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 exponentially faster than previous approaches, and we open up a host of interesting open problems at the intersection of sublinear-time algorithm design and this important new field. For example, sub-linear time algorithms are a necessary component to establish nontrivial results in property testing. We also enable the introduction of anytime algorithms in adaptive data analysis, by defining mechanisms that provide guarantees on accuracy when the time allotted is restricted.

As in previous literature, a mechanism $\mathcal{M}$ is given an i.i.d. sample S of size n from an unknown distribution D over a finite space X, and is given 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.

In previous work, the mechanisms execute in $\Omega\left(n\right)$ time per query. In this work, we introduce mechanisms that make an exponential improvement on this bound. Remarkably, we show that these results come at almost no tradeoff—we can obtain these exponential improvements in running time and yet use essentially the same sample sizes.

1.1 Motivation and results

Our results are summarized in Table 1. Our first result, in Section 3, is a method to answer low-sensitivity queries (defined in Section 2) that still has $\tilde{O}(\sqrt{k}/\alpha^2)$ sample complexity (as in previous work) but takes exponentially less time per query than previous approaches (Theorem 10). More-

Table 1: Time per query

Query Type	Previous Work	This Work
Low-sensitivity queries with $\tilde{O}\left(\frac{\sqrt{k}}{\alpha^2}\right)$ sample complexity	$\tilde{O}\left(\frac{\sqrt{k}}{\alpha^2}\right)$ [1]	$\tilde{O}\left(\frac{\log^2(k)}{\alpha^2}\right)$
Sampling counting queries with $\tilde{O}\left(\frac{\sqrt{k}}{\alpha^2}\right)$ sample complexity	$\tilde{O}\left(\frac{\sqrt{k}}{\alpha^2}\right)$	$\tilde{O}\left(\log\left(\frac{k}{\alpha}\right)\right)$

Summary of our results. k is the number of queries and 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.

over, our mechanism to answer a query is simple: given a database S, we first sample ℓ points i.i.d. from S, compute the empirical mean of q on that subsample, and then add Laplacian noise, which guarantees a *differentially-private* mechanism. The intuition behind this approach is that sampling offers two benefits: it can decrease computation time while simultaneously boosting privacy. Privacy yields a strong notion of stability, which in turn allows us to reduce the computation time without sacrificing accuracy. In particular, this mechanism takes only $\tilde{O}(\log^2(k)/\alpha^2)$ time per query and a sample size of $\ell = \tilde{O}(\log(k)/\alpha^2)$, all while matching the established sample complexity bound $\tilde{O}(\sqrt{k}/\alpha^2)$. Even in the non-adaptive case, it must take $\Omega(\log(k)/\alpha^2)$ samples to answer k such queries [1]. This means our results are tight in ℓ and come close to matching the best known lower-bound for the time complexity of answering such queries adaptively, which is simply $\Omega(\log(k)/\alpha^2)$. We show that this holds both when using uniform sampling with replacement and sampling without replacement.

While both sampling methods require examining $\ell = \tilde{O}(\log(k)/\alpha^2)$ samples per query, an analyst may wish to control the number of samples used. For example, the analyst might want the answer to a counting query using a very small number of sample points from the database, even just a single sample point. The above methods cannot handle this case gracefully because when ℓ is sufficiently small, the guarantees on accuracy (using Definition 2 below) become trivial—we get only that $\alpha = O(1)$, which any mechanism will satisfy. Instead, we want the mechanism to have to return a statistically-meaningful reply even if $\ell = 1$. Indeed, the empirical answer to such a query is $\{0, 1\}$-valued, while a response using Laplacian noise will not be.

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 [27] of the statistical query (SQ) setting for learning [2, 16], or the 1-STAT oracle for optimization [11]. Thus we introduce *sampling counting queries*, which imitate the process of an analyst requesting the value of a query on a single random sample. This allows for greater control over how long each query takes, in addition to greater control over the outputs. Namely, we require that for a query of the form $q : X \to \{0, 1\}$, the mechanism must output a $\{0, 1\}$-valued answer that is accurate in expectation. We show how to answer queries of this form by sampling a single point x from S and then applying a simple differentially-private algorithm to $q(x)$ that has not been used in adaptive data analysis prior to this work (Theorem 14). Finally, in Section 5, we compare sampling counting queries to counting queries.

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 [1, 6, 7, 21, 22]. Most applicable to our work is that of Bassily et al. [1] who consider, among others, *low-sensitivity queries*, which are merely any function of X^n whose output does not change much when the input is perturbed (for a more precise definition, see below). 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 order k/α^2 instead—exponentially worse.

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. [1], building on the work of Dwork et al. [7], provide a mechanism with sample complexity $n = \tilde{O}(\sqrt{k}/\alpha^2)$ to answer k low-sensitivity queries. Furthermore, for efficient mechanisms, this bound is tight in k [23]. This literature shows that the key to finding such mechanisms with this quadratic improvement over the naive 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 [1, 6, 7, 22], which offers a strong notion of stability.

Since this work uses differentially-private mechanisms after sampling, we are acutely interested in the impact on privacy when sampling. In both theory and practice, sampling in settings where privacy matters has long been deemed useful, in a variety of areas [14, 15, 17].

In our setting, we need an efficient uniform sampling method that not only maintains privacy, but actually boosts it. In particular, for an ϵ-private mechanism on a database of size n, we want to show that if you sample ℓ points uniformly and efficiently from those n points, and then apply the same mechanism, the result is $O\left(\frac{\ell}{n}\epsilon\right)$-private.

Fortunately, folklore has it that sampling boosts privacy–implicitly in Kasiviswanathan et al. [15], and certainly explicitly in the work of Lin et al. [20], who show that sampling without replacement boosts privacy to the degree we require for a particular setting. We note that their proof method easily generalizes to arbitrary domains and ϵ-private mechanisms. In addition, Bun et al. [4] show that sampling with replacement also boosts privacy.

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 Low-sensitivity queries

In this work, the first type of query we consider is a *low-sensitivity query*, which is specified by a function $q : X^n \to [0,1]$ with the property that for all samples $S, S' \in X^n$ where S and S' differ by at most one element, we have $|q(S) - q(S)'| \leq 1/n$, where we define $q(D) = \mathbb{E}_{S \sim D^n}[q(S)]$. We can now define the accuracy of $\mathcal{M}$.

Definition 1. *A mechanism $\mathcal{M}$ is said to be (α, β)*-accurate *over a sample S on queries $q_1, \ldots, q_k$ if for its responses $a_1, \ldots, a_k$ we have*

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

The key requirement is stronger. Namely, we seek accuracy over the unknown distribution.

Definition 2. *A mechanism $\mathcal{M}$ is (α, β)*-accurate *on distribution D, and on queries $q_1, \ldots, q_k$, if for its responses $a_1, \ldots, a_k$ we have*

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

In this work, we not only want (α, β)-accuracy but we also want to consider the time per query $\mathcal{M}$ takes. In this work, we assume we will have oracle access to q, which will compute $q(x)$ for a sample point x in unit time (and also $q(S)$ in at most $|S|$ time). This is not a strong assumption: If 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).

2.2 Counting queries and sampling counting queries

In this work we also consider *counting queries*, which ask the question "What proportion of the data satisfies property q?" Counting queries are a simple and important restriction of low-sensitivity queries [3, 5, 22]. More formally, a counting query is specified by a function $q : X \to \{0,1\}$, where $q(S) = \frac{1}{|S|}\sum_{x \in S} q(x)$ and $q(D) = \mathbb{E}_{x \sim D}[q(x)]$. As in the low-sensitivity setting, an answer to a counting query must be close to $q(D)$ (Definition 2).

This means, however, that answers 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 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 x from D and evaluate $q(x)$. The average value the mechanism returns (over the coins of the mechanism) should be close to the expected value of q. More precisely, we want the following:

Definition 3. *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 analagously. 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.

2.3 Differential privacy

Differential privacy, first introduced by Dwork et al. [8], provides a strong notion of stability.

Definition 4 (Differential privacy)**.** *Let $\mathcal{M} : X^n \to Z$ a randomized algorithm. We call $\mathcal{M}$ (ϵ, δ)*-differentially private *if for every two samples $S, S' \in X^n$, and every $z \subset Z$,*

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

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.

Proposition 5 (Adaptive composition [9, 10])**.** *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 6 (Post-processing [9])**.** *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.*

In this paper, we use two well-established differentially-private mechanisms: the Laplace and exponential mechanisms. See Dwork & Roth [9] for more on differential privacy and these mechanisms.

2.4 The transfer theorem

A key method of Bassily et al. [1] 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 7 (Bassily et al. [1])**.** *Let $\mathcal{M}$ be a mechanism that on input sample $S \sim D^n$ answers k adaptively chosen low-sensitivity 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 does 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. [1].

In order to prove our own transfer theorem for SCQ's, we will use some of the tools they developed. 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 8 (Bassily et al. [1])**.** *Let $\mathcal{W} : (X^n)^T \to Q \times [T]$ be (ϵ, δ)-private where Q is the class of counting 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 9 (Bassily et al. [1])**.** *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*

$$\mathbb{E}[f(X)] \geq \max_{r \in \mathcal{R}} f(r) - \frac{1}{\eta} \log |\mathcal{R}|.$$

3 Fast mechanisms using sampling

In this section, we provide simple and fast mechanisms for answering low-sensitivity queries. Our mechanism $\mathcal{M}$ for answering low-sensitivity queries is very simple: 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 Lap$(b)$ refers to the zero-mean Laplacian distribution with scale parameter b.

We may now state our main theorem for mechanism $\mathcal{M}$, using suitable values for ϵ and ℓ.

Theorem 10. *When $\epsilon = O(1/\alpha)$ and $\ell \geq \frac{2\log(4k/\beta)}{\alpha^2}$ for k low-sensitivity queries,*

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

$$n = \Omega\left(\frac{\sqrt{k}\log k \cdot \log^{3/2}(\frac{1}{\alpha\beta})}{\alpha^2}\right).$$

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. It is also the case that sampling without replacement may take $O(\log n)$ time per sample, for a total of $O(\ell \log n)$ time over ℓ samples, in several settings. Again, this is sufficient, but may come at the cost of space complexity, e.g. by keeping track of which elements have not been chosen so far [25]. Alternatively, there are methods that enjoy optimal space complexity at the cost of worst-case running times, as in rejection sampling [24].

To prove part 2), we must establish that sampling boosts privacy. If sampling before a a 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. Fortunately, sampling can boost privacy:

Proposition 11 (Adapted from Lin et al. [20])**.** *Given a mechanism $\mathcal{P} : X^\ell \to Y$, $\mathcal{M}$ will be the mechanism that does 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* $\log(1 + \frac{\ell}{n}(e^\epsilon - 1)) = O\left(\frac{\ell}{n} \cdot \epsilon\right)$ *private for $\ell \geq 1$.*

Sampling with replacement also boosts privacy:

Proposition 12 (Bun et al. [4])**.** *Given a mechanism $\mathcal{P} : X^\ell \to Y$, $\mathcal{M}$ will be the mechanism that does the following: Sample uniformly at random* with *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 $\frac{6\epsilon\ell}{n}$-private for $\ell \geq 1$.*

We may now return to the main theorem:

Proof of Theorem 10. 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 for any $\epsilon'' > 0$:

$$\mathbb{P}[|a_q - q(S_\ell)| \geq \alpha/2] \leq e^{-\frac{\alpha\epsilon''\ell}{2}}.$$

We can bound this above by $\frac{\beta}{2k}$ provided $\epsilon'' \geq \frac{\log(2k/\beta)}{\ell\alpha}$; and this follows from a Chernoff bound

$$\mathbb{P}[|q(S_\ell) - q(S)| \geq \alpha/2] \leq e^{-\frac{\alpha^2\ell}{2}}.$$

Once again we can bound this above by $\frac{\beta}{2k}$ so long as $\ell \geq \frac{2\log(4k/\beta)}{\alpha^2}$.

Thus we have, 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 over all k queries. From Proposition 11, we also have $\left(\frac{\ell}{n}\epsilon''\right)$-privacy, where $\frac{\ell}{n}\epsilon'' = \frac{\log(2k/\beta)}{n\alpha}$. Equivalently, we have ϵ'-privacy when $n \geq \frac{\log(2k/\beta)}{\epsilon'\alpha}$. With adaptive composition (Proposition 5), we can answer k queries with (ϵ, δ)-privacy when $\epsilon' = \frac{\epsilon}{2\sqrt{2k\log(1/\delta)}}$, resulting in (α, β)-accuracy and (ϵ, δ)-privacy on S so long as $n > \frac{2\sqrt{2k\log(1/\delta)}\log(2k/\beta)}{\alpha\epsilon}$. The proof is concluded by applying Theorem 7. □

4 Sampling counting queries

We now turn to sampling counting queries. Unlike in the previous section, we cannot leverage an existing transfer theorem, so instead we establish a new one.

Theorem 13. *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 14. *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 sampling counting queries, where*

$$n \geq \Omega\left(\max\left(\frac{\sqrt{k\log(\frac{1}{\alpha\beta})}}{\alpha^2}, \frac{\log(k/\beta)}{\alpha^2}\right)\right).$$

We prove our transfer theorem using the following monitoring algorithm, 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.

Definition 15 (Monitor with exponential mechanism)**.** *Define a monitoring algorithm $\mathcal{W}_D$ as the following: Given input $\mathbf{S} = \{S_1, \ldots, S_T\}$, for each of $t \in [T]$, simulate $\mathcal{M}(S_t)$ and $\mathcal{A}$ interacting, and let $q_{t,1}, \ldots, q_{t,k}$ be the queries of $\mathcal{A}$.*

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 $u(\mathbf{S}, r_{t,i}) = |q_{t,i}(S_t) - q_{t,i}(D)|$. Release $r \in \mathcal{R}$ with probability proportional to $\exp\left(\frac{\epsilon\cdot n\cdot u(\mathbf{S},r)}{2}\right)$.

Lemma 16. *If $\mathcal{M}$ is (ϵ, δ)-private for k queries, then $\mathcal{W}_D$ is $(2\epsilon, \delta)$-private.*

Proof. A single pertubation 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. Using the standard composition theorem finishes the proof. □

We can now bound the probability that the query that the monitor outputs on the sample are far away from the distribution on both sides, if $\mathcal{M}$ is not accurate, by using both Lemmas 8 and 9.

Proof of Theorem 13. 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, 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

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

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 9 implies that under the exponential mechanism, we have

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

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

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

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 16, $\mathcal{W}_D$ is $(2\epsilon, \delta)$-private, and then in turn Lemma 8 implies that

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

We will now ensure (2) $\geq \alpha/8$ and (3) $\leq \alpha/8$, 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 $n \geq \frac{1024\log(k/\beta)}{\alpha^2}$. □

With a transfer theorem in hand, we now introduce a private mechanism that is accurate on a sample for answering sampling counting queries.

Lemma 17 (SCQ mechanism)**.** *For $\epsilon \leq 1$, There is an (ϵ, δ)-private mechanism to release k SSQs 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 a $(\alpha, 0)$-accurate SCQ for $n > \frac{1}{\alpha\epsilon}$ and then use Proposition 5. The mechanism is simple: sample x i.i.d. from S. Then release $q(x)$ with probability $1-\alpha$ and $1-q(x)$ with probability α. Let $i = \sum_{x\in S} q(x)$. 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 $\frac{i}{n} - \alpha \leq \mathbb{E}_{\mathcal{M}}[\mathcal{M}(q)] \leq \frac{i}{n} + \alpha$, implying that $\mathcal{M}$ is $(\alpha, 0)$-accurate on S.

Now let S' differ from S on one element x, where $q(x) = 0$ but for $x' \in S'$, $q(x') = 1$. Consider

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

for $i = 0$ to $i = n-1$. The other cases are similar. Note this is at least 1 since $1 - 2\alpha \geq 0$. Thus it suffices to show when this is upper-bounded by e^ϵ. By computing the partial derivative with respect to i, it is easy to see that it suffices to consider the cases when $i = 0$ or $i = n-1$. When $i = 0$,

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

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

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

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

We now use this mechanism to answer sampling counting queries.

Proof of Theorem 14. We use the mechanism of Lemma 17. This gives an (ϵ, δ)-private mechanism that is $(\alpha/2, 0)$-accurate so long as $n \geq \frac{4\sqrt{2k\log(1/\delta)}}{\alpha\epsilon}$.

Setting ϵ and δ as required by Theorem 13 implies that we need $n \geq \Omega\left(\sqrt{k\log(\frac{1}{\alpha\beta})}/\alpha^2\right)$.

Note to use Theorem 13 we also need $n \geq \Omega\left(\log(k/\beta)/\alpha^2\right)$. The sample complexity bound follows. This mechanism samples a single random point, which takes $O(\log(n))$ time, completing the proof. □

5 Comparing counting and sampling counting queries

How do our mechanisms for counting queries and sampling counting queries compare to each other? 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 an extra $O(1/\alpha)$ factor (although it does enjoy a slightly better dependence on k). This represents a $O(1/\alpha)$ overhead in order to ensure that the mechanism returns meaningful results for all sample sizes ℓ.

Proposition 18. *Using ℓ SCQ's to estimate each counting query is an (α, β)-accurate mechanism for k counting queries if $\ell \geq \frac{2\log(4k/\beta)}{\alpha^2}$ 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 Chernoff 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 14, 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

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

Plugging in ℓ into the above expression for n completes 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 naive 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 Future work

In this paper, we have introduced new faster mechanisms that take advantage of sampling's simultaneous ability to boost privacy while decreasing running time. In what other adaptive settings can sampling help as much as it does in this work? Sub-linear time algorithms are frequently required for a variety of problems, such as property testing or large-data environments. How can fast algorithms for adaptive analysis be developed in these types of settings?

7 Acknowledgements

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

References

[1] R. Bassily, K. Nissim, A. D. Smith, T. Steinke, U. Stemmer, and J. 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.

[2] A. Blum, M. L. Furst, J. C. Jackson, M. J. Kearns, Y. Mansour, and S. 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.

[3] A. Blum, K. Ligett, and A. 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.

[4] M. Bun, K. Nissim, U. Stemmer, and S. 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.

[5] M. Bun, J. Ullman, and S. 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.

[6] C. Dwork, V. Feldman, M. Hardt, T. Pitassi, O. Reingold, and A. 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, 2015.

[7] C. Dwork, V. Feldman, M. Hardt, T. Pitassi, O. Reingold, and A. L. 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, 2015.

[8] C. Dwork, F. McSherry, K. Nissim, and A. 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.

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

[10] C. Dwork, G. N. Rothblum, and S. 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.

[11] V. Feldman, E. Grigorescu, L. Reyzin, S. S. Vempala, and Y. Xiao. Statistical algorithms and a lower bound for detecting planted cliques. *Journal of the ACM*, 64(2):8:1–8:37, 2017.

[12] A. Gelman and E. 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.

[13] N. Golbandi, Y. Koren, and R. 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.

[14] Z. Jorgensen, T. Yu, and G. 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.

[15] S. P. Kasiviswanathan, H. K. Lee, K. Nissim, S. Raskhodnikova, and A. 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.

[16] M. J. Kearns. Efficient noise-tolerant learning from statistical queries. *J. ACM*, 45(6):983–1006, 1998.

[17] G. Kellaris and S. Papadopoulos. Practical differential privacy via grouping and smoothing. *PVLDB*, 6(5):301–312, 2013.

[18] A. Kleiner, A. Talwalkar, P. Sarkar, and M. 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.

[19] J. D. Lee, D. L. Sun, Y. Sun, and J. E. Taylor. Exact post-selection inference, with application to the lasso. *Ann. Statist.*, 44(3):907–927, 06 2016.

[20] B.-R. Lin, Y. Wang, and S. Rane. On the benefits of sampling in privacy preserving statistical analysis on distributed databases. *arXiv preprint arXiv:1304.4613*, 2013.

[21] R. M. Rogers, A. Roth, A. D. Smith, and O. 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.

[22] T. Steinke and J. Ullman. Between pure and approximate differential privacy. In *Theory and Practice of Differential Privacy (TPDP 2015), London, UK*, 2015.

[23] T. Steinke and J. 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, 2015.

[24] J. S. Vitter. Faster methods for random sampling. *Communications of the ACM*, 27(7):703–718, 1984.

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

[26] H. Xiao, J. Gao, Q. Li, F. Ma, L. Su, Y. Feng, and A. 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.

[27] K. 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.