Making Progress Based on False Discoveries

Roi Livni

Department of Electrical Engineering, Tel Aviv University, Israel

Abstract

We consider Stochastic Convex Optimization as a case-study for Adaptive Data Analysis. A basic question is how many samples are needed in order to compute ε-accurate estimates of $O(1/\varepsilon^2)$ gradients queried by gradient descent. We provide two intermediate answers to this question.

First, we show that for a general analyst (not necessarily gradient descent) $\Omega(1/\varepsilon^3)$ samples are required, which is more than the number of sample required to simply optimize the population loss. Our construction builds upon a new lower bound (that may be of interest of its own right) for an analyst that may ask several non adaptive questions in a batch of fixed and known T rounds of adaptivity and requires a fraction of true discoveries. We show that for such an analyst $\Omega(\sqrt{T}/\varepsilon^2)$ samples are necessary.

Second, we show that, under certain assumptions on the oracle, in an interaction with gradient descent $\tilde{\Omega}(1/\varepsilon^{2.5})$ samples are necessary. Which is again suboptimal in terms of optimization. Our assumptions are that the oracle has only *first order access* and is *post-hoc generalizing*. First order access means that it can only compute the gradients of the sampled function at points queried by the algorithm. Our assumption of *post-hoc generalization* follows from existing lower bounds for statistical queries. More generally then, we provide a generic reduction from the standard setting of statistical queries to the problem of estimating gradients queried by gradient descent.

Overall these results are in contrast with classical bounds that show that with $O(1/\varepsilon^2)$ samples one can optimize the population risk to accuracy of $O(\varepsilon)$ but, as it turns out, with spurious gradients.

2012 ACM Subject Classification Theory of computation → Machine learning theory

Keywords and phrases Adaptive Data Analysis, Stochastic Convex Optimization, Learning Theory

Digital Object Identifier 10.4230/LIPIcs.ITCS.2024.76

Related Version *Full Version*: https://arxiv.org/abs/2204.08809

Funding This work has been supported by an ISF grant no 2188/20 and the research was funded in part by the ERC grant (FOG- 101116258 FoG).

1 Introduction

Adaptive data analysis is a recent mathematical framework [13] that aims to deal with modern issues of false discoveries [16, 17, 23]. These issues, potentially, occur when datasets are being reused in order to find statistically significant discoveries which, without proper care, may lead to overfitting due to adaptivity of the analysis to the data. Motivated by these issues, the framework of adaptive data analysis investigates a mathematical model that considers an *analyst* that interacts with a fixed dataset through acces to an *oracle* or *mechanism*. In this framework, the analyst is assumed to be malicious and its objective is to find *false discoveries* i.e. queries for which the oracle fails to answer correctly. The oracle on the other hand, tries to maintain the validity of the answers. Through this formalism, investigators were able to provide new algorithms and methods that allow more principled use of the data [3, 12, 5, 11], as well as complementary work that provides limitations to what can be done with limited resources [25, 15, 24].

A very appealing, and studied [14, 26], example for adaptive data analysis emerges from the setup of *Stochastic Convex Optimization* (SCO). In SCO, we want to construct an algorithm that is given access to finite (convex) noisy samples of a convex function and its objective is to minimize the function. A standard approach to the problem is via an iterative process where the algorithm initializes at some point w_1 and then incrementally updates w_t using estimates of the gradient. For example, (full-batch) Gradient Descent performs an update step of the form:

$$w_{t+1} = \Pi\left(w_t - \frac{\eta}{m}\sum_{i=1}^{m} \nabla f_i(w_t)\right), \tag{1}$$

where f_i are the i.i.d noisy instances of the function to be minimized and Π is a projection operator. This approach to stochastic convex optimization falls nicely into the framework of adaptive data analysis as follows: At each iteration the optimizer (which takes the role of an analyst) asks the gradient for the population risk at w_t, receives an estimation that is computed from a dataset and updates its state according to the answer. In Equation (1), the estimates are provided by the naive empirical mean.

Motivated by this example of adaptivity, in this work we revisit the framework of adaptive data analysis, and consider it from the lens of optimization. One motivation is to allow us to study the problem of adaptivity in a setting where the analyst has a well defined objective which is not necessarily to generate, maliciously, false discoveries. This affects much of the analysis, as the algorithms are fixed, as well as the number of needed queries and tolerance to noise.

Perhaps the first result that comes to mind in this context is that of SGD: In SGD the algorithm uses highly noisy estimates of the gradient. Instead of taking the empirical mean, we just sample one point (without replacement) as an estimate. On the one hand, it avoids the problem of adaptivity by using few examples per iteration. On the other hand, it does not even try to provide correct gradients. From the optimization point of view this algorithm achieves the optimal statistical rates [21].

But there are algorithms that don't necessarily avoid adaptiveness and reuse the data to estimate the gradient. For example, full-batch GD as depicted in Equation (1). How do they perform, and how does the problem of adaptivity affect them? Perhaps as expected, adaptivity does come with a certain cost. A recent construction by [2] shows that GD, with standard choice of hyperparameters (i.e. learning rate and no. of iterations) can minimize the empirical risk, and at the same time overfit and fail to generalize. A close examination of the construction shows that, already in the second step, the gradient starts to be biased and does not represent the true gradient of the loss function. In a subsequent work, [1], it was shown that this problem is inherent in the estimator. Namely, no method can accept the empirical gradients as estimates and optimize the problem with optimal rates. This is perhaps a good example to the shortcoming of naive reuse of data.

There is a fix though. It leads to suboptimal rates in terms of computation but maintain optimal sample complexity rate. Specifically, [3] showed that a smaller step size, and more iterations (quadratically more, infact) would make the algorithm stable, and $m = O(1/\varepsilon^2)$ samples suffice for this iterative process to work. But, from the adaptive data analysis lens, this solution sort of shifts the responsibility from the estimation mechanism to the analyst. In particular, the gradients are still provided as is, it is the analyst that is being stabilized. Moreover, the analyst *increased* the number of queries. Namely, not only it did not improve the estimates, it actually enhanced its interaction with this inexact mechanism. Importantly, one can show that the solution does not make the gradients any more accurate (see Theorem 4). Only the final output is being salvaged. The natural question, then, is whether we can think of another fix within the framework of adaptive data analysis. Thus, let us consider Gradient Descent as any algorithm that makes adaptive steps as in Equation (1) but with any estimate of the gradient, not necessarily empirical mean. Then the question we would like to answer is:

What is the sample complexity of providing $O(1/\varepsilon^2)$, ε-accurate gradients of a 1-Lipschitz convex function to Gradient Descent with learning rate $\eta = O(\varepsilon)$?

We require $O(1/\varepsilon^2)$ gradients and $O(\varepsilon)$ learning rate as these are known to be necessary for optimization [22, 20] (and it is also easy to see that it is sufficient). We focus here on dimension-independent bounds as these are the optimal achievable rates. It is easy to see that $\tilde{O}(1/\varepsilon^4)$ is a naive, dimension independent bound that one could achieve (where the oracle uses $O(1/\varepsilon^2)$-fresh new samples at each iteration, hence by standard dimension independent concentration bounds [6]). Standard techniques of adaptive data analysis can also be used to achieve rates of $\tilde{O}(\sqrt{d}/\varepsilon^3)$ [4] but this is both dimension dependent and remains suboptimal for optimization purposes. The question above remains open, but we provide two intermediate answers, which we next describe:

Our first result is for general analysts and not for GD: We show that if an analyst is allowed to query gradients of a convex function then $\Omega(1/\varepsilon^3)$ samples are needed in order to provide $O(1/\varepsilon^2)$ ε–accurate answers. Our result here builds upon existing techniques and attacks [8, 10, 24], and we obtain a new lower bound (which may be of interest of its own right) to an analyst, in the standard statistical query setting, that queries many non-adaptive queries in sequential bulks of adaptive rounds (where the rounds of adaptivity are known, distinctively from [11]) and needs to obtain a fraction of true discoveries. We show that for such an analyst there exists a lower bound of $\Omega(\sqrt{T}/\varepsilon^2)$ samples, where T is the number of rounds of adaptivity and ε is the accuracy. We then show a generic reduction to the setting of convex optimization. Though the analyst is not GD, this result does demonstrate that one cannot design a complete mechanism for any analyst with optimal rates. It does leave open, though, the possibility of designing incomplete oracles that interact with specific types of analysts (or algorithms) such as GD.

The second result is for GD. We provide a bound of $\tilde{\Omega}(1/\varepsilon^{2.5})$, but under further assumptions: First, we assume the oracle is *post-hoc generalizing* [9, 25]. Roughly, post-hoc generalization means that the algorithm does not query points where the empirical loss and true loss differ. While this assumption may seem restrictive, we point out that we inherit it from existing known bounds in the standard statistical query setting. Specifically, we provide a generic reduction from statistical queries to the framework of GD. Then, we apply the lower bound of [25] that assumes post-hoc generalization. But the reduction is generic, and guarantees, given a lower bound for statistical queries of the form $f(T, \varepsilon)$ where T is the number of queries and ε is the desired accuracy, a lower bound of the form $f(O(1/\varepsilon), O(\varepsilon))$ in the setting of convex optimization (under a further first-order access assumption which we discuss next).

The second assumption we make is what we term *first-order access*. Here we assume that the oracle must compute the estimate only from the gradients at $\{w_1, \ldots w_t\}$ and not, say, by using the global structure of the function (we mention that our result can easily be extended to allow any local, but at a small neighbourhood, information of the function). Note that, since the function must be fixed throughout the optimization process, and since the optimization algorithm is fixed, allowing the oracle global access to the function restricts us from using any type of randomness other than the randomness of the distribution. Hence, while slightly more delicate then the first assumption, here too we require this assumption since in the standard statistical query setting lower bounds are provided with respect to random analysts. Our reduction, then, can turn a more general oracle (without first-order access) into a procedure that can answer statistical queries against a deterministic analyst (in the sense that the distribution may be random, but the analyst's strategy is fixed and known). This seems like an interesting question for future study.

It is interesting, then, to compare these results to recent adaptations of the standard model that restrict the analyst [26]. This is largely motivated by the reasoning that analysts are not necessarily adversarial. Our result, though, may hint (if one considers GD as a non malicious algorithm in this context) that the problem may be in the distribution of the data and not necessarily in the analyst. Namely, a general reduction from statistical queries to the framework of GD along our lines, will show that any lower bound can be described as constructing a malicious distribution which leads to overfitting together with a non-malicious analyst.

2 Background

2.1 Adaptive Data Analysis

We begin by revisiting the standard statistical queries setting of adaptive data analysis introduced by [13]. In this model, we consider a subset $\mathcal{Q}$ of statistical queries over a domain $\mathcal{X}$. A statistical query is defined to be any function $q : \mathcal{X} \to [-1, 1]$. We consider a sequential interaction between an *analyst* A and a statistical queries *oracle* $\mathcal{O}$ (or simply oracle) that continues for T iterations and is depicted as follows:

At the beginning of the interaction the analyst A chooses a distribution D over $\mathcal{X}$. The Oracle $\mathcal{O}$ is provided with a finite collection of samples $S = \{x_1, \ldots, x_m\}$ drawn i.i.d from the distribution D. Then the interaction continues for T sequential rounds: At round $t \geq 1$, A provides a statistical query $q_t \in \mathcal{Q}$, and the oracle $\mathcal{O}$ returns an answer $a_t \in [-1, 1]$. The answer a_t may depend on the dataset S as well as on previous answers and queries $\{q_1, \ldots, q_t\}$. The query q_t may depend on previous answers $\{a_1 \ldots, a_{t-1}\}$, as well as on the distribution D (which is thought of as known to the analyst). We denote by $q_t(D)$ and $q_t(S)$ the following quantities:

$$q_t(D) := \mathop{\mathbb{E}}_{x \sim D}[q_t(x)], \quad q_t(S) := \frac{1}{m}\sum_{i=1}^{m} q_t(x_i).$$

The goal of the oracle is to preserve accuracy, as next defined. And, here, we mostly care about the minimal size m that is required by $\mathcal{O}$ in order to succeed.

▶ **Definition 1.** *An oracle $\mathcal{O}$ is $(\varepsilon, \gamma, \delta)$-accurate for T adaptively chosen queries given m samples in $\mathcal{X}$ if for every analyst A and distribution D, with probability at least $(1 - \delta)$ for $(1 - \gamma)T$ fraction of the queries output by A:*

$$|a_t - q_t(D)| \leq \varepsilon.$$

We will write, for brevity, (ε, δ)-accurate instead of $(\varepsilon, 0, \delta)$-accurate. An additional requirement of *post-hoc generalization* [9], is also sometimes imposed:

▶ **Definition 2.** *An oracle $\mathcal{O}$ is (ε, δ)-post hoc generalizing for T adaptive queries with m samples if: given m samples, for every analyst A, with probability at least $(1 - \delta)$: for all $t \in [T]$*

$$|q_t(S) - q_t(D)| < \varepsilon.$$

The following result bounds the sample complexity of a post-hoc generalizing oracle:

▶ **Theorem 3** (Cor 3.2 [25]). *There exists an analyst A, such that for every Oracle $\mathcal{O}$, if $\mathcal{O}$ is $(\varepsilon, 0.005)$ post hoc-generalizing and $(\varepsilon, 0.005)$-accurate, given m samples, for T adaptively chosen queries by A, then*

$$m = \Omega(\sqrt{T}/\varepsilon^2). \tag{2}$$

2.2 Stochastic Convex Optimization

We next review the setting of stochastic convex optimization. In this model we consider a function $f(w,x):\mathbb{R}^d\times\mathcal{X}\to\mathbb{R}$, which is *convex* and $O(1)$-Lipschitz in the parameter w, for every $x\in\mathcal{X}$. We also consider a distribution D over $\mathcal{X}$ and we denote by F the *population risk*:

$$F(w)=\mathop{\mathbb{E}}_{x\sim D}[f(w,x)].$$

The objective of an optimization algorithm A (or analyst) is to calculate $w^\star$ such that:

$$F(w^\star)\le\min_{\|w\|\le 1}F(w)+\varepsilon.$$

In order to achieve this goal, we also assume an interaction with what we'll call here *exact-first-order* oracle, $\mathcal{O}_{f,\nabla f}(w,x)$, for the function f that, given w and x returns

$$\mathcal{O}_{f,\nabla f}(w,x)=(\mathcal{O}_f(w,x),\mathcal{O}_{\nabla f}(w,x)):=(f(w,x),\nabla f(w,x)). \tag{3}$$

Gradient Descent over the empirical risk

A very popular first-order approach to solve the above optimization problem is by performing Gradient-Descent over the *empirical loss* . Here we perform a very simple update rule: At first, the algorithm initializes at $w_1=0$. Then, at each iteration t the algorithm updates

$$w_{t+1}=\Pi\left[w_t-\frac{\eta}{m}\sum_{i=1}^m\nabla f(w_t,x_i)\right], \tag{4}$$

where Π is a projection on the unit ball and ∇f is provided by access to an exact-first-order oracle. The output of the procedure is then:

$$w_S=\frac{1}{T}\sum_{i=1}^T w_i.$$

This procedure can be considered as an algorithm that minimizes the empirical loss, where given a sample S, we define the empirical loss to be

$$F_S(w)=\frac{1}{m}\sum_{i=1}^m f(w,x_i).$$

It is well known (see for example, [7]) that ,given the above procedure:

$$F_S(w_S)\le\min_{\|w\|\le 1}F_S(w)+O\left(\eta+\frac{1}{\eta T}\right).$$

In particular, a choice of $\eta=O(1/\sqrt{T})$ leads to an error of $O(1/\sqrt{T})$. But the output of the procedure can also be related to the population risk through the following upper bound:

▶ **Theorem** ([3]). *Let D be an unknown distribution, over $\mathcal{X}$ and suppose that $f(w,x)$ is $O(1)$ Lipschitz and convex with respect to $w\in\mathbb{R}^d$. Let $S=\{x_1,\dots,x_m\}$ be a sample drawn i.i.d from distribution D, and consider the update rule in Equation (4). Then for $w_S=\frac{1}{T}\sum_{t=1}^T w_t$*

$$\mathop{\mathbb{E}}_{S\sim D^m}[F(w_S)]\le\min_{\|w^\star\|\le 1}F(w^\star)+O\left(\eta\sqrt{T}+\frac{1}{\eta T}+\frac{\eta T}{m}\right). \tag{5}$$

A choice of $T = O(m^2)$, $\eta = 1/m^{3/2}$ leads to an error of $O(1/\sqrt{m})$ which is statistically optimal. As discussed, [2] provided a matching lower bound for the number of iteration required to achieve $O(1/\sqrt{m})$ error.

One could also ask whether the empirical estimates of the gradients also generalize. Namely, is the empirical mean of the gradients close to their true expectations throughout the procedure? A close examination of the construction used by [2] shows that, without special care, the empirical estimate of the gradient fails to provide accurate gradients, even if we choose the learning rate and number of iteration to minimize Equation (5). We provide a proof sketch in the full version [19].

▶ **Theorem 4.** *Given a sample $x_1, \ldots, x_m$ of i.i.d samples, suppose we run GD over the empirical risk, as depicted in Equation* (4)*. There exists a distribution over $\mathcal{X}$ and an $O(1)$ convex Lipschitz function. Such that if S is a sample drawn i.i.d from the distribution D of size m and w_t is defined as in Equation* (4) *then for $t = 2$, with probability $1/2$ over w_2:*

$$\left\| \mathop{\mathbb{E}}_{S\sim D^m}\left[\frac{1}{m}\sum_{i=1}^{m} \nabla f(w_2, x_i)|w_2\right] - \nabla F(w_2)]\right\| = \Omega(1).$$

As discussed, there is in fact a simpler example to the fact that optimizing the objective doesn't require accurate gradients which is SGD. It is known that when $T = m$ and $\eta = 1/\sqrt{T}$, the analyst optimizes the objective to the same accuracy of $O(1/\sqrt{m})$. Remarkably, this requires even less iterations and the gradient doesn't even presume to be accurate. Nevertheless, the analysis here rely on the fact that the gradient is an *unbiased* estimate, where adaptivity is avoided since we use a fresh example at every round.

It is also worth mentioning that recently [18] showed that, in adaptive data analysis terminology, SGD is an example to a non post-hoc generalizing algorithm in the following sense: It can be shown that the output parameter w_S provided by SGD may minimize the population loss, but there is a constant gap between the empirical and population loss at w_S.

3 Problem setup

We next describe our setting. We consider the problem of adaptiveness within the context of stochastic convex optimization. We consider an interaction between an analyst A and a first-order optimization oracle, $\mathcal{O}_F$. At the beginning of the interaction the analyst chooses a function f and a distribution D. Then a sample $S = \{x_1, \ldots, x_m\}$ is drawn and provided to the oracle. The interaction then proceeds for T rounds, where at round $t \in [T]$ the analyst queries for a point w_t, and the oracle returns $\mathcal{O}_F(w_t) \in \mathbb{R}^d$. The query points $w_1, \ldots, w_T$ may depend on the distribution D and the oracle answer $\mathcal{O}_F(w_t)$ may depend on the sample S, the function f, as well as on the sequence of previously seen $w_1, \ldots w_{t-1}$.

Gradient Descent

Within our framework we describe GD as the following procedure: For every $\eta > 0$ we let GD with learning rate η be defined by the following update at each iteration $t \geq 1$ (setting $w_1 = 0$):

$$w_t = \Pi\left(w_t - \eta \nabla \mathcal{O}_F(w_t)\right), \tag{6}$$

where Π is the projection operator over the ℓ_2-unit ball. Notice that, if $\mathcal{O}_F$ has an access to an exact first order oracle for f, $\mathcal{O}_{\nabla f}$, and returns the empirical mean at each iteration then we obtain GD over the empirical risk as described in Equation (1). Before we continue, we notice that good generalization of $\mathcal{O}_F$ is sufficient for optimization. Indeed, the following result is an easy adaptation of the classical optimization bound for GD:

▶ **Theorem 5.** *Let D be an unknown distribution over $\mathcal{X}$ and suppose that $f(w,x)$ is $O(1)$-Lipschitz and convex with respect to $w \in \mathbb{R}^d$. Let $S = \{x_1, \ldots, x_m\}$ be a sample drawn i.i.d from distribution D, and consider the update rule in Equation* (6)*. Assume that for every iteration $\mathcal{O}_F$ satisfies*

$$|\mathcal{O}_F(w_t) - \nabla F(w_t)| \le \varepsilon,$$

Then for $w_S = \frac{1}{T}\sum_{t=1}^T w_t$

$$E_{S\sim D^m}[F(w_S)] \le \min_{\|w^\star\|\le 1} F(w^\star) + O\left(\eta + \frac{1}{\eta T} + \varepsilon\right).$$

The above rate is optimal, which leads to the natural question, what is the sample needed by an oracle that returns $O(1/\varepsilon^2)$ ε-accurate gradients for GD with learning rate $O(\varepsilon)$. Such an oracle improves over, the naive, empirical mean estimate which induces GD over the empirical risk which requires $\tilde{\Theta}(1/\varepsilon^4)$ iterations to achieve error of $O(\varepsilon)$. The performance of such an oracle should also be compared with SGD that can achieve a comparable bound on the number of iterations and requires the optimal sample size of $m = O(1/\varepsilon^2)$. Next, we provide natural extentions to the definition of adaptive oracles to the setting of stochastic optimization.

▶ **Definition 6.** *A first order oracle $\mathcal{O}_F$ is $(\varepsilon, \gamma, \delta)$-accurate against algorithm A for T iterations, given m samples, if $\mathcal{O}_F$ is provided with m samples and with probability at least $(1-\delta)$ for $(1-\gamma)T$ fractions of the $t \in [T]$:*

$$\|\mathcal{O}_F(w_t) - \nabla F(w_t)\| \le \varepsilon.$$

If $\mathcal{O}$ is $(\varepsilon, \gamma, \delta)$-accurate against any algorithm A we say it is $(\varepsilon, \gamma, \delta)$-accurate.

We will write in short (ε, δ)-accurate for $(\varepsilon, 0, \delta)$-accurate.

▶ **Definition 7.** *A first-order oracle $\mathcal{O}_F$ is (ε, δ)-post hoc generalizing against algorithm A for T iterations, given m samples if with probability at least $(1-\delta)$: for every $t \in [T]$*

$$\|\nabla F(w_t) - \frac{1}{m}\sum_{i=1}^m \nabla f(w_t, x_i)\| \le \varepsilon.$$

If $\mathcal{O}$ is (ε, δ)-post hoc generalizing against any algorithm A we simply say it is (ε, δ)-post hoc generalizing.

First order local access

We next introduce the following assumption on the oracle:

▶ **Definition 8.** *A first order first-order-access (FOA)-oracle $\mathcal{O}_F$ is a procedure that, given access to an exact-first-order oracle $\mathcal{O}_{f,\nabla f}$ to the function f, and access to a sample S of size m, returns for every point w_t a gradient estimate $\mathcal{O}_F(w_t)$ that may depend only on*

$$\{(f(w_{t'}, x_i), \nabla f(w_{t'}, x_i\}\}_{\{(x_i, w_{t'}): x_i \in S, t' \le t\}}.$$

Equivalently, we may think of an FOA oracle as a procedure that does not have access to f, instead, at each iteration t receives as input the parameter w_t as well as a function

$$\bar{\rho}_t : \mathcal{X} \to \mathbb{R} \times \mathbb{R}^d,$$

such that $\bar{\rho}_t(x) = (f(w_t, x), \nabla f(w_t, x))$ for every x. The output of the FOA at round t may depend on $\bar{\rho}_1, \ldots, \bar{\rho}_t$

The assumption of a FOA-oracle is very natural in the context of Stochastic Convex Optimization, and in general, we do not assume access to a global structure of a convex function. The above assumption indeed captures oracles that have only such local access.

4 Main Results

We are now ready to state our main results. Our first result state that, for a general analyst, the oracle cannot generalize for $T = O(1/\varepsilon^2)$ estimated gradients, unless it is provided with $m = \Omega(1/\varepsilon^3)$ examples. The proof is provided in Section 5.2.1

▶ **Theorem 9.** *There exists constants $\gamma, \delta > 0$ and a randomized analyst A the chooses a determined 1-Lipschitz function f, defined over sufficiently large d, such if $\mathcal{O}_F$ is a first-order oracle that is $(\varepsilon, \gamma, \delta)$-accurate against A for T iterations, then $m = \Omega\left(\frac{\sqrt{T}}{\varepsilon^2}\right)$. In particular, any oracle $\mathcal{O}_F$ that is $(\varepsilon, \gamma, \delta)$-accurate for $T = O(1/\varepsilon^2)$ iterations must observe $m = \Omega\left(1/\varepsilon^3\right)$ examples.*

Making no assumption on the analyst may seem non-realistic, especially to assume it is malicious and attempts to achieve false gradients. Nevertheless there is value in producing oracles that are foolproof. The above theorem shows that such security guarantees are impossible with the standard sample complexity.

The next natural thing that we might want to consider is an oracle that is principled under certain assumptions on the optimization algorithm. We might even hope to design an incomplete oracle that can interact with specific optimization algorithms and halt when certain assumptions are broken. The next result demonstrate that limitations from general statistical queries can be translated into limitations for (FOA) oracles that interact with gradient descent. The proof is provided in the full version [19].

▶ **Theorem 10.** *For sufficiently large d, suppose that there exists a FOA oracle, $\mathcal{O}_F$, that is (ε, δ)-accurate that receives m samples and answers T adaptive queries against Gradient Descent with learning rate $\eta = O(\varepsilon)$. Then there exists a $(O(\varepsilon), O(\delta))$-accurate statistical queries oracle, $\mathcal{O}$, that receives m samples and answers $\tilde{\Omega}\left(\min\{T, 1/\eta\}\right)$ adaptive queries.*

Moreover, if $\mathcal{O}_F$ is (ε, δ)-post-hoc generalizing then $\mathcal{O}$ is $(O(\varepsilon), O(\delta))$-post-hoc generalizing.

Together with Theorem 3 we obtain the following corollary

▶ **Corollary 11.** *For sufficiently large d, let $\mathcal{O}_F$ be an (ε, δ)-accurate and post-hoc generalizing FOA that receives m samples and answers $T > \Omega(1/\varepsilon)$ adaptive queries against Gradient Descent with learning rate $\eta = O(\varepsilon)$. Then $m = \tilde{\Omega}\left(1/\varepsilon^{2.5}\right)$.*

We stress again that, in contrast with these results, an optimization algorithm can correctly minimize the true loss using no more than $\tilde{O}(1/\varepsilon^2)$ iterations and $\tilde{O}(1/\varepsilon^2)$ examples [18].

5 Lower bounds against malicious analysts

In this section we set out to prove Theorem 9 and provide a lower bound to general oracles against *adversarial* analysts. In section Section 5.2 we show how to turn a generic lower bound for statistical queries to a lower bound in convex optimization. However, to the best of the author's knoweledge, there is no (unconditional) known lower bound that shows that $m = \Omega(\sqrt{T}/\varepsilon^2)$ examples are necessary to answer T queries. So we actually rely on a slightly stronger reduction than from the standard setting of statistical queries. We rely, then, on the fact that querying a single gradient carries more information than a single statistical query, in fact d more. However, these may not be chosen adaptively, and the errors are spreaded.

The setting from which we provide the reduction is as follows: We consider an analyst that at each iteration t can ask k non-adaptive queries. This is reminiscent to a similar problem that was studied by [13], but there it is unknown what are the rounds of adaptivity.

Here we consider a significantly simpler problem where the rounds of adaptivity are known in advance and we show, using ideas from [8] (that constructs a similar lower bound but in the setting of privacy), that for certain $k = \Omega(1/\varepsilon^2)$,$\Omega(\sqrt{T}/\varepsilon^2)$ samples are needed to ensure a large enough fraction of the answers are correct. Then, as discussed, we provide a generic reduction to convex optimization. We now turn to describe the setting of adaptive non-adaptive queries and state our main lower bound for this setting.

5.1 Adaptive-non-Adaptive queries

In this section we take a little detour from our basic setting and return to the setting of statistical queries.

5.1.1 Setup

We will consider now a natural generalization of the standard setting of adaptive data analysis. Here, we allow the analyst to query at each round k queries simultaneously. In this setting, as before, we have a family of queries $\mathcal{Q}$ as well as an analyst A and oracle $\mathcal{O}$ which interact for T rounds. Distinictively from before, at round t we assume A asks k-statistical queries $\mathbf{q_t} = \{q_{t,1}, \dots, q_{t,k}\} \subseteq \mathcal{Q}^k$, and $\mathcal{O}$ returns an answer vector $\mathbf{a}_t = (a_{t,1}, \dots, a_{t,k})$. The answer vector $\mathbf{a}_t$ may depend on the sample S and on previously published queries $\mathbf{q}_1, \dots, \mathbf{q}_t$. Similarly the query vector may depend on previous answer vectors $\mathbf{a}_1, \dots, \mathbf{a}_{t-1}$ and the distribution D.

▶ **Definition 12.** *Similar to Definition 1, we say that $\mathcal{O}$ is $(\varepsilon, \gamma_T, \gamma_k, \delta)$-accurate for T adaptively chosen queries, given m samples, if the oracle samples at most m samples and with probability $1-\delta$ we have for $(1-\gamma_T)$ fraction of the rounds, for $(1-\gamma_k)$ fraction of the queries:*

$$|\mathcal{O}(q_{t,i}) - q_{t,i}(D)| \leq \varepsilon.$$

We next set out to prove the following lower bound:

▶ **Theorem 13.** *For $\mathcal{X} = \{0,1\}$ For $k = \Omega(1/\varepsilon^{2.01})$, there exists a finite family of queries $\mathcal{Q}$ over the domain $\mathcal{X} = \{0,1\}^k$, constants γ_T, γ_k, δ, such that no oracle $\mathcal{O}$ is $(\varepsilon, \gamma_T, \gamma_k, \delta)$ accurate for k-non adaptive T adaptively chosen queries given m samples unless $m = \Omega\left(\sqrt{T}/\varepsilon^2\right)$*

Before we begin with the proof, we provide several preliminary results that we build upon.

5.1.2 Overview Technical Preliminaries

The proof of Theorem 13 relies on a technical idea that appears in [8]. [8] starts by considering two constructions in the context of privacy. The first, demonstrates a sample complexity lower bound of $\Omega(\sqrt{T})$ for T private queries, and a second construction, a reconstruction attack, that allows a certain reconstruction of the data unless the sample size is order of $\Omega(1/\varepsilon^2)$ for ε-accurate answers. Then, they provide a new construction that consolidates these two bounds into one construction that operates on a certain product space of the two domains. Here we do something similar only we replace the privacy attack with an adaptive data analysis attacks that operates on i.i.d samples (which is not necessary when privacy is considered). The consolidation is a little bit different as we must consider a dataset that is generated by sampling i.i.d examples (as opposed to worst-case dataset in the case of privacy).

In more detail, the proof of Theorem 13 relies on two types of attacks that were introduced by [24, 10]. Our first type of attack is a reconstruction attack, and we follow the definition of [8]:

▶ **Definition 14** (Reconstruction Attack). *For a dataset $\mathcal{S} = \{x_1, \ldots, x_m\}$, we will say that $\mathcal{S}$ enables an ε'-reconstruction attack from (ε, γ)-accurate answers to the family of statistical queries $\mathcal{Q}$ if: There exists a function*

$$\mathcal{B} : \mathbb{R}^{|Q|} \to [0,1]^m,$$

such that for every vector $v \in [0,1]^m$ and every answer sequence $a = (a_q)_{q\in\mathcal{Q}} \in [0,1]^{\mathcal{Q}}$: If for at least $1-\gamma$ fraction of the queries $q \in \mathcal{Q}$ holds:

$$\left| a_q - \frac{1}{m}\sum_{i=1}^{m} q(x_i)v(i) \right| < \varepsilon,$$

then for $\mathbf{b} = \mathcal{B}(a)$:

$$\frac{1}{m}\sum_{i=1}^{m} |b(i) - v(i)| < \varepsilon'.$$

The following result is due to [10], we state it as in [8] for the special case of considering 1-way marginals[1] :

▶ **Theorem 15.** *Let $k \geq 1/\varepsilon^{2.01}$, and assume ε is sufficiently small. There exists a constant γ_0 (independent of ε and k) such that for every ε', there exists a dataset $\mathcal{S} = (\{0,1\}^k)^m$ with $m = \Omega_{\varepsilon'}(1/\varepsilon^2)$ such that $\mathcal{S}$ enables an ε'-reconstruction attack from (ε, γ_0)-accurate answers to a family of queries $\mathcal{Q}$ of size k.*

The second attack that we rely on provides an information theoretic lower bound of $\Omega(\sqrt{T})$ to answer adaptive statistical queries:

▶ **Theorem 16** (Thm 3.10 [24]). *For all $\gamma < 1/2$, there is a function $T(m,\gamma) \in O\left(m^2/(1/2-\gamma)^4\right)$, such that there is no oracle $\mathcal{O}$ that is $(0.99, \gamma, 1/2)$-accurate for $T(m,\gamma)$ adaptively chosen queries, given m samples in $\{0,1\}^d$, where $d \geq T(m,\gamma)$.*

We will require a dual restatement of Theorem 16, which essentially follows the same proof together with standard minmax theorem:

▶ **Theorem 17.** *There exists a randomized analyst A such that for any oracle $\mathcal{O}$ that interacts with A for $T(m,\gamma)$ rounds having m samples, then with probability at least $1/2$ for at least γT of the rounds:*

$$|a_t - q_t(D)| > 0.01.$$

Sketch. The proof is essentialy the proof of Theorem 16 as depicted by [24]. We only need to argue that in the construction of [24] the advarsarial analysts that are being constructed are from a finite set and then use standard minmax duality. To see that the analysts in the original proof are supported on a finite set, first observe that the analyst chooses (randomly)

[1] Note that in [8] k denotes the k-way marginal query class which we fix to be the 1-way marginal, and the k in our statement is denoted by d in [8]

a uniform distribution over a sequence of pairs $(1, v_1), (2, v_2), \ldots, (N, v_N)$ where $N = T(m, \gamma)$ and each v_i depicts a secret key (where for the information-theoretic lower bound we choose a one-time pad encryption and then $v_i \in \{\pm 1\}^N$). Hence the set of feasible distributions is of size 2^N, and $N = O(T(m, \gamma))$. Next, we note that at each iteration, the analyst rounds the answer for $\mathcal{O}$, a_t and chooses as a query, q_t which is parameterized by a vector in $\{-1, 0, 1\}^N$. Hence, the query at round t depends on t vectors in $\{-1, 0, 1\}^N$ and $\{\mathrm{sgn}(a_1), \ldots, \mathrm{sgn}(a_t)\}$ hence overall there is a finite set of states to which the analyst can transition at each iteration, so overall there is only a finite amount of analysts on which the distribution is supported. ◀

5.2 Proof of Theorem 13

Let $k \geq 1/\varepsilon^{2.01}$, and set $\mathcal{Q}_\varepsilon$ be a set of at most k queries over a dataset $\mathcal{S}_\varepsilon$ and γ_0 a parameter that enables an ε'-reconstruction attack from (ε, γ_0)-accurate answers to $\mathcal{Q}_\varepsilon$ as guaranteed in Theorem 15. Without loss of generality we assume $\gamma_0 < 1/2$, and ε' is chosen such that:

$$\varepsilon' < \frac{0.01}{3 \cdot 2^6}.$$

Now we let $d = |S_\varepsilon| = O(1/\varepsilon^2)$. Without loss of generality and for simplicity of notations we assume $S_\varepsilon = [d] = \{1, \ldots, d\}$.

Suppose $\mathcal{Q}$ is a family of queries, and assume we have d analysts, in the standard model (i.e. each analyst asks a single question), $(A_1, \ldots, A_d)$. We define an analyst that asks k queries $\mathcal{A}(A_1, \ldots, A_d)$ as follows:

First, when the analysts choose distributions $D_1, \ldots, D_d$ over $\mathcal{X}$, the analyst $\mathcal{A}$ defines a distribution D over $\mathcal{S}_\varepsilon \times \mathcal{X}$ that chooses first randomly and uniformly $i \in [d]$ and returns (i, x) where $x \sim D_i$. The oracle, in turn, observes i.i.d samples from the given product distribution.

The interplay with the oracle proceeds as follows: At each iteration t, we assume by induction that each analyst, $A_1, \ldots, A_d$, provides a query $q_{t,1}, \ldots, q_{t,d}$. The analyst $\mathcal{A}(A_1, \ldots, A_d)$ constructs for each query $q \in \mathcal{Q}_\varepsilon$ the query

$$q'_t((i, x)) = q(i) q_{t,i}(x),$$

and asks these k non-adaptive queries.

Then, given the answer vector $\{\mathbf{a}_{q'_t}\}$, we provide analyst A_i with the answer $a_{t,i}$, where

$$\mathbf{a}_t = \mathcal{B}(\mathbf{a}_{q'_t}),$$

and $\mathcal{B}$ defines the reconstruction attack in Theorem 15. The analysts then provide the queries $q_{t+1,1}, \ldots, q_{t+1,d}$ and the analyst $\mathcal{A}(A_1, \ldots, A_d)$ continues to the next round until round T. Our analyst then depends on the d analysts, We choose them to be d i.i.d copies of the analyst in Theorem 17 and we let $\bar{A}$ be the analyst induced by such $A_1, \ldots, A_d$.

Notice that when we fix $A_1, \ldots, A_{i-1}, A_{i+1}, \ldots, A_d$, that are provided to $\mathcal{A}(A_1, \ldots, A_d)$ we induce an oracle, that we denote by $\mathcal{O}_i$ that interacts with analyst A_i. In more detail, we consider a randomized oracle $\mathcal{O}_i$ that operates as follows:

At the beginning of the interaction, before the first round, $\mathcal{O}_i$ draw a uniform sample $\{s_1, \ldots, s_m\}$. For each sample $s_j \neq i$ the oracle also draws a sample $(x \sim D_i)$. Then, given m_i samples $\{x_1, \ldots, x_{m_i}\}$ from D_i where m_i is the number of times i was drawn, the oracle adds to the sample the sample points $\{(i, x_j)\}_{j=1}^{m_i}$. Notice that this sample is drawn exactly according to the process depicted above where (i, x) is drawn such that i is uniform and $x \sim D_i$. The interaction with A_i along the rounds is continued where at each round $\bar{A}$ transmit the question, $\mathcal{O}$ answers, and $\bar{A}$ transmit the answer back, as described above.

In this interaction the number of samples is random, but notice that if $T \geq T(m_i, 1/4)$, where

$$T(m, 1/4) = O(m^2),$$

is defined in Theorem 17, then with probability at least 1/2 for $T/4$ of the rounds, by Theorem 17

$$\|a_{t,i} - q_{t,i}(D_i)\| > 0.01.$$

This also entails that for every i:

$$\mathop{\mathbb{E}}_{A_i,Q_i}\left(\frac{1}{T}\sum_{t=1}^{T}|a_{t,i} - q_{t,i}(D_i)|\right) > \frac{0.01}{2^3}\mathbb{P}(T \geq T(m_i, 1/4)). \tag{7}$$

Now assume $\mathcal{O}$ is $(\varepsilon, \varepsilon', \gamma_0, \varepsilon')$-accurate for $T = T(4m/3d, 1/4)$ rounds, and consider $\mathcal{A}$ as defined above. Then with probability $1 - \varepsilon'$: for $(1 - \varepsilon')$-fractions of the rounds, for $(1 - \gamma_0)$-fraction of the queries q'_t:

$$\left|a_{q'_t} - \frac{1}{k}\sum_{i=1}^{k} q(i) q_{t,i}(D_i)\right| = \left|a_{q'_t} - q'_t(D)\right| < \varepsilon.$$

Which entails by reconstruction attack, for the same fraction of rounds:

$$\frac{1}{k}\sum_{i=1}^{k}|a_{t,i} - q_{t,i}(D_i)| \leq \varepsilon'.$$

Taken together we have

$$\mathbb{E}\left[\frac{1}{Tk}\sum_{i=1}^{k}|a_{t,i} - q_{t,i}(D_i)|\right] \leq 3\varepsilon' \leq \frac{0.01}{2^6}. \tag{8}$$

On the other hand, notice that for any analyst, with probability 1/4, $\mathcal{O}_i$ is provided with less than $\frac{4m}{3d}$ samples from the distribution D_i

So by choice $T \geq T(4m/3d, 1/4) = O(m^2\varepsilon^4)$, and by Equation (7), we have:

$$\mathop{\mathbb{E}}_{A,Q}\left(\frac{1}{Tk}\sum_{i=1}^{T}\sum_{i=1}^{k}|a_{t,i} - q_{t,i}|\right) = \frac{1}{k}\sum_{i=1}^{k}\mathop{\mathbb{E}}_{A_i,Q_i}\left(\frac{1}{T}\sum_{t=1}^{T}|a_{t,i} - q_{t,i}|\right) > \frac{0.01}{2^5}.$$

contradicting Equation (8).

5.2.1 Proof of Theorem 9

We now proceed with the formal proof of Theorem 9. Given a family of queries $|\mathcal{Q}| \leq d$ index the coordinates of $\mathbb{R}^d$ by the elements of $\mathcal{Q}$. Namely, we think of $\mathbb{R}^d$ as $\mathbb{R}^{\mathcal{Q}}$ where each vector $w \in \mathbb{R}^{Q}$ is thought of as a function $w : \mathcal{Q} \to \mathbb{R}$.

We define a convex, over the parameter w, function in $\mathbb{R}^{\mathcal{Q}}$:

$$f(w, x) = \sum_{q \in \mathcal{Q}} \frac{q(x) + 1}{4} w^2(q). \tag{9}$$

Note that, since $|q(x)| \leq 1$, the above function is always convex and 1-Lipschitz for any choice of queries and x. Let $\mathcal{O}_F$ be a first order $(\varepsilon, \gamma_T, \delta)$-accurate oracle, and let us consider the setting of an analyst that requires k non adaptive queries for T rounds. Let A be an analyst that asks k non adaptive, T adaptive queries, and consider an oracle $\mathcal{O}$ that performs as follows: Given queries $q_{t,1}, \ldots, q_{t,k}$, the oracle $\mathcal{O}$ transmit to the oracle $\mathcal{O}_F$ the point:

$$w_t = \frac{1}{\sqrt{k}} \sum_{i=1}^{k} e_{q_{t,i}},$$

where e_q is the vector in $\mathbb{R}^Q$ that is $e_q(q) = 1$, and $e_q(q') = 0$ when $q' \neq q$. In turn, the oracle receives the answer vector $\mathcal{O}_F(w_t) = g_t$ and returns the answers $a_{t,i} = 2\sqrt{k} g_t(q_{t,i}) - 1$

Now suppose that with probability $(1 - \delta)$ for γ_T fraction of the rounds:

$$\|g_t - \nabla F(w_t)\| \leq \varepsilon,$$

then:

$$\frac{1}{k} \sum \left(\frac{a_{t,i} - q_{t,i}(D)}{2} \right)^2 = \sum \left(\frac{a_{t,i} + 1}{2\sqrt{k}} - \frac{q_{t,i}(D) + 1}{2\sqrt{k}} \right)^2 \leq \|g_t - \nabla F(w_t)\|^2 \leq \varepsilon^2.$$

Then by Markov's inequality for any γ_k, for γ_k fraction of the queries we have:

$$|a_{t,i} - q_{t,i}(\mathcal{D})| \leq \frac{2\varepsilon}{\sqrt{\gamma_k}}.$$

Which means $\mathcal{O}$ is an $(\frac{2\varepsilon}{\sqrt{\gamma_k}}, \gamma_T, \gamma_k, \delta)$-accurate oracle that answers k non adaptive T adaptive queries. By Theorem 13, for small enough ε, with correct choice of γ_k, γ_T and δ:

$$m = \Omega\left(\sqrt{T}/\varepsilon^2\right).$$

6 Gradient Descent

In this section we set out to prove Theorem 10. In contrast to previous result, here we fix the analyst and assume that it performs predefined update steps. This puts several complications into the lower bound as we cannot actively make it "adversarial", at least not in the standard way. Nevertheless our construction builds on a similar idea as the proof before. The idea here is to think of the function as a "state" machine, where each coordinate represents a query that may be asked. The analyst, given answers to the queries, moves to the next query. The complication though, is that here the dynamic is predefined and we need to design our function carefully so that GD will induce the correct transition between states.

The idea is captured in what is our main technical workhorse which is the notion of a *GD wrapper*, which we build in the full version [19]. GD wrappers will be used to provide a reduction from a special class of analysts termed *Boolean analysts*, which are depicted in Section 6.1. Then we use a simple reduction from general analysts (see Lemma 18) to obtain a reduction from general analysts to our setting.

We begin with a brief overview of the construction. After that, in Section 6.2 we depict the technical notion of GD wrapper. We then explain, in the full version [19], how to deduce Theorem 10 from the existence of a GD wrapper. In thethe full version [19], we provide a construction of a GD wrapper, which concludes the proof.

6.1 Brief overview

As discussed, the heart of our construction is the notion of a *GD wrapper*. The idea is quite straightforward. Given an analyst A we want to construct a convex function f_A that is convex and such that the trajectory of the function provides the answers to our question. We've done something similar in the construction of Equation (9). There too, we constructed a convex function that the gradient of w_t at a certain coordinate provides an answer to a given query. The issue though is that there we could ask to query any coordinate we wanted. Here we need to make sure that the dynamic of GD moves us from query q_t to query q_{t+1}.

Thus, the first requirement that we want from our function f_A is that by looking at the outputs: $\mathcal{O}(w_1), \ldots, \mathcal{O}(w_{t-1})$ we can identify the answer to query q_t. For one non-adaptive query this is straight forward. Indeed, consider the linear function

$$f_q(w, x) = q(x) \cdot w(1).$$

Then, the gradient $\nabla F_q(w) = \mathbb{E}_{x \sim D}[q(x)]e_1$. So we can identify the answer to the first query.

Simulating an adaptive analysis with 2 questions

As a next step, let us construct a GD wrapper that works as follows: At the first step the wrapper will provide answer to query q_1, and if $q_1(x) > b_1$ for some threshold b_1, then the wrapper transitions to a state that identifies the answer to a query q^+ and else moves to a state that identifies q^- (to simplify, we will assume that the answer is never $q = b_1$). This is still far from a general strategy of an analyst, but at the end of this overview we will discuss how we can reduce the general problem to a problem of a similar form.

Also, for the exposition we don't want to consider the oracle's strategy, hence assume that at each iteration the oracle returns the *true gradient* and we will show how the trajectory simulates the adaptive query interaction:

$$f_{q,q^+,q^-} = \max\left\{w(1) + \eta - \frac{1-q(x)}{3}w(2) - \frac{1+q(x)}{3}w(3), w(2) + q^+(x)w(4), w(3) + q^-(x)w(5)\right\}.$$

Our function is described as the maximum of three linear functions hence it is convex. Now let us follow the trajectory for the first two steps. At the first step, note that the first term maximizes the term. Recall that the gradient of a function $f = \max\{g_1, \ldots, g_k\}$, is given by $\nabla f = \arg\max \nabla_{i \in [k]} g_i$ hence we have that for $w_1 = 0$, for every x:

$$\nabla f_{q,q^+,q^-}(0, x) = e_1 - \frac{(1-q(x)}{3}e_2 - \frac{1+q(x)}{3}e_3.$$

hence:

$$w_2 = w_1 - \eta\nabla \mathop{\mathbb{E}}_{x \sim D}[f_{q,q^+,q^-}(0, x)] = -\eta e_1 + \frac{(1-q(D))}{3}\eta e_2 + \frac{1+q(D)}{3}\eta e_3.$$

Now, note that the first term is no longer maximized by w_2 for any x, as we moved against the gradient and now it is smaller. On the other hand, if $q(D) < 0$ the second term is maximized, and else the last. Assume the first: then

$$\mathop{\mathbb{E}}_{x \sim D} \nabla f(w_2, x) = e_2 + q^+(D)e_4,$$

Note that the gradient at w_2 tells us excatly whether $q(D) > 0$ or $q(D) < 0$. In particular, if $\mathcal{O}(w_2)_2 > 1/2$, then we know the $q(D) > 0$. Any oracle that returns an approximate answer will also identify the answer. Using a recursive process, along these lines, we then can construct a convex function that moves from one query to another using gradient descent.

In general, the state of an analyst does not depend, necessarily, on some threshold value as in our case above. However, as the next reduction shows, if we are willing to suffer a $\log 1/\varepsilon$ factor increase in the number of queries, we can turn a general analyst to an analyst whose decision indeed depend on some threshold as depicted here:

Boolean Analysts

We will call an analyst *Boolean* if it provides to the oracle a query $q_i : \mathcal{X} \to [-1, 1]$ and its state at time t depends only on $\{\mathrm{sgn}(a_1), \mathrm{sgn}(a_2), \dots, \mathrm{sgn}(a_{t-1})\}$. A more general setup could allow at each iteration a query q_i and a threshold b_i and the state may depend only on $\mathrm{sgn}(a_1 - b_1), \dots, \mathrm{sgn}(a_{t-1} - b_{t-1})$. However, up to rescaling it can be seen that the two types of analysts are equivalent.

For such a boolean analyst, we define an oracle to be (ε, δ)-accurate for T adaptive queries against a Boolean analyst, if given m samples for every Boolean analyst A with probability at least $(1 - \delta)$ for all $t \in [T]$ if $a_t = 1$ then

$$q_t(D) > -\varepsilon,$$

and if $a_t = -1$

$$q_t(D) < \varepsilon.$$

Similarly, an oracle $\mathcal{O}$ is (ε, δ)-post hoc generalizing for T adaptive queries against a Boolean analyst, if given m samples for every Boolean analyst A with probability at least $(1 - \delta)$ for all $t \in [T]$ if $q(D) > 0$ then

$$\frac{1}{m} \sum q_t(x_i) > -\varepsilon,$$

and if $q(D) < 0$

$$\frac{1}{m} \sum q_t(x_i) < \varepsilon.$$

The following statement is easy to see:

▶ **Lemma 18.** *Suppose that there exists an oracle $\mathcal{O}_1$ that is (ε, δ)-post hoc generalizing (ε, δ)-accurate oracle against any Boolean analyst that answers T queries with m samples. Then there exists a $(4\varepsilon, \delta)$-post-hoc generalizing $(4\varepsilon, \delta)$-accurate oracle, $\mathcal{O}_2$, that answers any analyst (not necessarily Boolean) $\Omega\left(\frac{T}{\log 1/\varepsilon}\right)$ queries with m samples.*

General first order local access oracles

We so far assume a truthful Oracle, that provides the true answer. We now need to deal with an oracle whose decision is both based on finite data and is *strategic*, in the sense that it can manipulate the wrapper above with his answers.

Note that in order to construct the wrapper we designed a function whose dependent on *all* possible states of the analyst A. In particular, an all powerful oracle can basically look at the design of the function and get to know the exact strategy. That includes random bits as the function needs to be determined and chosen at the beginning of the game. That is why, as long as existing lower bounds for statistical queries rely on randomized analysts, for this strategy to work, we need to somehow prohibit from the oracle to identify the random bits.

Note that the reverse is also true. Without some restrictions on the oracle, the construction against Gradient Descent becomes a pure strategy (modulus the choice of the distribution of the data).

We thus assume that the oracle has only access through the local gradients at points visited which restricts him from identifying the whole strategy of the analyst. In turn, we need to make sure, in our construction, that such internal bits are indeed not transmitted through the gradients. We therfore add in our notion of GD wrapper a further restriction called *correctness of gradients*: In addition to requiring that answers can be extracted from the gradients, we also require that the gradients of the instance functions can be completely derived from the current state. This ensures us that an FOA Oracle may be operated solely on already published information, hence its decision cannot be dependent on internal bits which are independent of the current state. We now move on to introduce the notion of GD wrapper, which explains exactly these requirements:

6.2 GD wrapper for an anlayst

We next describe the main technical tool for the proof which is a *GD wrapper for an analyst* which can be thought of as an object that allows a statistical query analyst to interact with a FOA Oracle. A GD wrapper (with learning rate η and initialization s), consists of

1. A *wrapper function* which is a function $f(A; w, x)$ that accepts a deterministic Boolean analyst and is convex and $O(1)$-Lipschitz in $w \in \mathbb{R}^d$ for every x.
2. A strictly increasing mapping $\mathcal{T} : [T_1] \to [T_2]$, $\mathcal{T}(1) > 1$. The wrapper is said to answer T_1 queries and to perform T_2 iterations, and $\mathcal{T}$ is called the iteration complexity.
3. A sequence $\vec{\kappa} = \{\kappa_t\}_{t=1}^{T_1}$ of T_1 functions which are termed *answering mechanisms*:

   $$\kappa_t : \left(\mathbb{R}^d\right)^{\mathcal{T}(t)} \to [0, 1],$$

4. A sequence $\vec{\rho} = \{\rho_t\}_{t=1}^{T_2}$ of T_2 functions which are termed *gradient access functions*:

   $$\rho_t := (\rho_t^0, \rho_t^1) : \mathcal{Q} \times \left(\mathbb{R}^d\right)^t \times \mathcal{X} \to \mathbb{R} \times \mathbb{R}^d,$$

   where $\mathcal{T}(t') < t$.

We consider the interaction between a GD wrapper and what we will term here a *pseudo* oracle $\mathcal{O}$, which is a deterministic mapping that outputs at step t $\mathcal{O}(t) \in \mathbb{R}^d$ and receives a vector $\bar{\rho}_t : \mathcal{X} \to \mathbb{R}^d$ (whose chosen in a manner that we will define next). The output of $\mathcal{O}$ at times t, then, may depend on $\bar{\rho}_1, \ldots, \bar{\rho}_t$, but it doesn't necessarily obtain a query.

Given $\mathcal{O}$, we define two sequences iteratively: *the trajectory of the wrapper* $w_1, \ldots, w_{T_2}$, which is defined inductively where $w_1 = s$, and

$$w_t = \Pi\left(w_{t-1} - \eta\mathcal{O}(t-1)\right).$$

Second we define the answering sequence, which is updated whenver $t = \mathcal{T}(t')$ for some t':

$$a_{t'} = \kappa_{t'}(\mathcal{O}(1), \ldots, \mathcal{O}(t)).$$

Next, at each round t, suppose we have the answering sequence $a_1, \ldots, a_{t'}$. Then, the oracle receives, as input, the t-th gradient access function, considered as a function of the argument x, defined as follows:

$$\bar{\rho}_t(x) = \rho_t(q_{t'+1}, \mathcal{O}(1), \ldots, \mathcal{O}(t), x),$$

where $q_{t'+1}$ is the $t'+1$-th query asked by analyst A, given the answering sequence $a_1, \dots, a_{t'}$ (if $t' = 0$, i.e. $\mathcal{T}(1) > t$, then, $q_{t'+1} = q_1$). In turn, $\mathcal{O}$ outputs $\mathcal{O}(t+1) \in \mathbb{R}^d$.

The GD wrapper is said to be $(\varepsilon_1, \varepsilon_2, \delta)$-accurate against $\mathcal{O}$ if for every distribution D, the following occurs w.p. $(1-\delta)$, for *every* $t_0 \le T_1$:

If, for analyst A:

$$\|\mathcal{O}(t) - \mathop{\mathbb{E}}_{x \sim D}[\nabla f(A; w_t, x)]\| \le \varepsilon_1. \tag{10}$$

for every $t < t_0$ Then:

- (Correct gradients at t_0:) For every $x \in \mathcal{X}$:

$$\bar{\rho}_{t_0}(x) = \left(f(A; w_{t_0}, x), \nabla f(A; w_{t_0}, x)\right), \tag{11}$$

- (Accurate answers) If Equation (10), in addition, holds for $t = t_0$, and $\mathcal{T}(i) = t_0$ then $a_i = 1$ implies $q_i(D) > -\varepsilon_2$, and $a_i = -1$ implies $q_i(D) < \varepsilon_2$, where q_i is the i-th query provided by A when provided with answer sequence $a_1, \dots, a_{i-1}$.

If a GD wrapper is $(\varepsilon_1, \varepsilon_2, \delta)$-accurate against any oracle, we simply say it is $(\varepsilon_1, \varepsilon_2, \delta)$-accurate.

It can be seen that a GD wrapper together with an FOA Oracle imply an oracle that answers statistical queries (we provide the proof in the full version [19]).

▶ **Lemma 19.** *Suppose that there exists $(2\varepsilon_1, \varepsilon_2, \delta)$-accurate GD wrapper with learning $\eta > 0$ that answers T_1 queries and perform T_2 iterations. Suppose also, that there exists an oracle that is a (ε_1, δ)-accurate FOA oracle, $\mathcal{O}_F$, that receives m samples and answers T_2 adaptive queries against Gradient Descent with learning rate $\eta > 0$.*

Then there exists an $(\varepsilon_2, 2\delta)$-accurate oracle, $\mathcal{O}$, that receives m samples and answers T_1 adaptive queries against any Boolean analysts. Moreover, if $\mathcal{O}_F$ is (ε_1, δ)-post-hoc generalizing then $\mathcal{O}$ is $(2\varepsilon_2, 2\delta)$-post-hoc generalizing.

In the full version [19] we provide a construction of a GD wrapper. Specifically, we prove the following:

▶ **Lemma 20.** *For sufficiently small $\varepsilon, \eta > 0$, and $\delta > 0$. Assume $\eta < \sqrt{\varepsilon/48}$, and $T < \min\{1/16\eta, 1/24\varepsilon\}$. For sufficiently large d, there exists a $(\varepsilon, O(\varepsilon), \delta)$-accurate GD wrapper with a learning rate $\eta > 0$ and 1-Lipschitz wrapping function that answers T queries and performs $2T$ iterations.*

References

1 Idan Amir, Yair Carmon, Tomer Koren, and Roi Livni. Never go full batch (in stochastic convex optimization). *Advances in Neural Information Processing Systems*, 34, 2021.

2 Idan Amir, Tomer Koren, and Roi Livni. Sgd generalizes better than gd (and regularization doesn't help). In *Conference on Learning Theory*, pages 63–92. PMLR, 2021.

3 Raef Bassily, Vitaly Feldman, Cristóbal Guzmán, and Kunal Talwar. Stability of stochastic gradient descent on nonsmooth convex losses. *Advances in Neural Information Processing Systems*, 33:4381–4391, 2020.

4 Raef Bassily, Kobbi Nissim, Adam Smith, Thomas Steinke, Uri Stemmer, and Jonathan Ullman. Algorithmic stability for adaptive data analysis. *SIAM Journal on Computing*, 50(3):STOC16–377, 2021.

5 Avrim Blum and Moritz Hardt. The ladder: A reliable leaderboard for machine learning competitions. In *International Conference on Machine Learning*, pages 1006–1014. PMLR, 2015.

6 Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration inequalities: A nonasymptotic theory of independence.* Oxford university press, 2013.

7 Sébastien Bubeck et al. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

8 Mark Bun, Jonathan Ullman, and Salil Vadhan. Fingerprinting codes and the price of approximate differential privacy. *SIAM Journal on Computing*, 47(5):1888–1938, 2018.

9 Rachel Cummings, Katrina Ligett, Kobbi Nissim, Aaron Roth, and Zhiwei Steven Wu. Adaptive learning with robust generalization guarantees. In *Conference on Learning Theory*, pages 772–814. PMLR, 2016.

10 Anindya De. Lower bounds in differential privacy. In *Theory of cryptography conference*, pages 321–338. Springer, 2012.

11 Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toni Pitassi, Omer Reingold, and Aaron Roth. Generalization in adaptive data analysis and holdout reuse. *Advances in Neural Information Processing Systems*, 28, 2015.

12 Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Roth. The reusable holdout: Preserving validity in adaptive data analysis. *Science*, 349(6248):636–638, 2015.

13 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*, pages 117–126, 2015.

14 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*, pages 1265–1277. SIAM, 2017.

15 Moritz Hardt and Jonathan Ullman. Preventing false discovery in interactive data analysis is hard. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, pages 454–463. IEEE, 2014.

16 John PA Ioannidis. Contradicted and initially stronger effects in highly cited clinical research. *Jama*, 294(2):218–228, 2005.

17 John PA Ioannidis. Why most published research findings are false. *PLoS medicine*, 2(8):e124, 2005.

18 Tomer Koren, Roi Livni, Yishay Mansour, and Uri Sherman. Benign underfitting of stochastic gradient descent. *arXiv preprint*, 2022. `arXiv:2202.13361`.

19 Roi Livni. Making progress based on false discoveries. *arXiv preprint*, 2022. `arXiv:2204.08809`.

20 Arkadi Nemirovski and Dmitry Yudin. Problem complexity and method efficiency in optimization (as nemirovsky and db yudin). *Wiley, Interscience*, 1985.

21 Arkadij Semenovič Nemirovskij and David Borisovich Yudin. *Problem complexity and method efficiency in optimization.* Wiley-Interscience, 1983.

22 Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2003.

23 Florian Prinz, Thomas Schlange, and Khusru Asadullah. Believe it or not: how much can we rely on published data on potential drug targets? *Nature reviews Drug discovery*, 10(9):712–712, 2011.

24 Thomas Steinke and Jonathan Ullman. Interactive fingerprinting codes and the hardness of preventing false discovery. In *Conference on learning theory*, pages 1588–1628. PMLR, 2015.

25 Jonathan Ullman, Adam Smith, Kobbi Nissim, Uri Stemmer, and Thomas Steinke. The limits of post-selection generalization. *Advances in Neural Information Processing Systems*, 31, 2018.

26 Tijana Zrnic and Moritz Hardt. Natural analysts in adaptive data analysis. In *International Conference on Machine Learning*, pages 7703–7711. PMLR, 2019.