Natural Analysts in Adaptive Data Analysis

Tijana Zrnic [1] **Moritz Hardt** [1]

Abstract

Adaptive data analysis is frequently criticized for its pessimistic generalization guarantees. The source of these pessimistic bounds is a model that permits arbitrary, possibly adversarial analysts that optimally use information to bias results. While being a central issue in the field, still lacking are notions of natural analysts that allow for more optimistic bounds faithful to the reality that typical analysts aren't adversarial. In this work, we propose notions of natural analysts that smoothly interpolate between the optimal non-adaptive bounds and the best-known adaptive generalization bounds. To accomplish this, we model the analyst's knowledge as evolving according to the rules of an unknown dynamical system that takes in revealed information and outputs new statistical queries to the data. This allows us to restrict the analyst through different natural control-theoretic notions. One such notion corresponds to a *recency bias*, formalizing an inability to arbitrarily use distant information. Another complementary notion formalizes an *anchoring bias*, a tendency to weight initial information more strongly. Both notions come with quantitative parameters that smoothly interpolate between the non-adaptive case and the fully adaptive case, allowing for a rich spectrum of intermediate analysts that are neither non-adaptive nor adversarial. Natural not only from a cognitive perspective, we show that our notions also capture standard optimization methods, like gradient descent in various settings. This gives a new interpretation to the fact that gradient descent tends to overfit much less than its adaptive nature might suggest.

[1]Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, Berkeley, USA. Correspondence to: Tijana Zrnic <tijana@eecs.berkeley.edu>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

1. Introduction

Modern data analysis is usually adaptive in the sense that past analyses shape future analyses. This practice offers power and flexibility to data science at the cost of a greater potential for spurious results. The issue is now well recognized in multiple communities. The problem of *inference after selection* is an active research area in statistics, while computer science has developed an area known as *adaptive data analysis*.

The statistical community has focused on analyzing concrete two-step procedures, such as, variable selection followed by a significance test on the chosen variables (Fithian et al., 2014; Belloni et al., 2014). This approach leads to precise insight into some concrete procedures, but it does not capture the workflow of typical analysts that proceed in more than two steps.

Computer scientists took an alternative route by focusing on a powerful *statistical query model* that in principle captures all sorts of different analyses involving many adaptive steps. In this model, an analyst interacts with a data set through a primitive called *statistical queries*. In each round, the analyst can evaluate one statistical query on the data. Future statistical queries may depend arbitrarily on the revealed transcript of past queries and query results. This level of generality comes at the cost of diminished generalization ability.

To review what's known, the generalization error on t non-adaptively chosen statistical queries on a data set of size n is on the order of $O(\sqrt{\log(t)/n})$, as follows from Hoeffding's bound. In the fully adaptive model, Hoeffding's bound would only give a rate of $\tilde{O}(\sqrt{t/n})$. This disappointing bound coincides with the naive strategy of splitting the data set into t chunks, each of size n/t and using one chunk for each query. Noise addition techniques combined with the mature technical repertoire of differential privacy yield a better bound of $\tilde{O}(t^{1/4}/\sqrt{n})$ (Bassily et al., 2016). However, this bound still features a polynomial dependence on the number of queries t that has resisted improvement for years. Negative results suggest that it might in fact be computationally hard to improve on this bound (Hardt & Ullman, 2014; Ullman et al., 2018).

For years, the knee jerk reaction to such pessimistic bounds

has been to point out that natural analysts aren't adversarial. However, it has proved challenging to formalize what makes natural analysts more benign than the worst-case bounds suggest. Indeed, to date there is still no comprehensive proposal for a class of analysts that allows for interesting intermediate points between the fully adaptive and non-adaptive case.

1.1. Our Contributions

In this work, we tackle the central conceptual challenge of formalizing classes of natural analysts using ideas from dynamical systems theory. Specifically, we model the analyst's knowledge as evolving according to the rules of an unknown dynamical system in discrete time. The system takes in query results a_t at each step and maintains a hidden state h_t at time t. Based on its hidden state, the system chooses a new query $q_t = f_t(h_t)$ as a function of the hidden state (that may vary with time) and updates its hidden state $h_{t+1} = \psi_t(h_t, a_t)$ according to a state-transition map ψ_t that is allowed to vary with time. It is clear that we can recover the non-adaptive case by forcing the hidden state to be constant at all steps, whereas the fully adaptive case corresponds to an unrestricted hidden state and state transition rule.

What is interesting is that this dynamical perspective allows us to restrict the analyst in natural ways, which we show lead to interesting trade-offs. These restrictions simultaneously correspond to natural control-theoretic notions, subsume common optimization procedures, and can be seen as formalizing well-known cognitive biases. We focus on two complementary notions of natural analysts that we call *progressive* and *conservative*.

Progressive analysts. Progressive analysts, intuitively speaking, have a recency bias and weight recent information more strongly than information received far into the past. We can think of a discount factor $\lambda \in (0, 1)$ by which the analyst downweights past observations. Formally, we call an analyst λ-*progressive* if the state transition map is contractive[1]: $\|\psi_t(h, a) - \psi_t(h', a)\| \leq \lambda \|h - h'\|$.

To gain intuition, in the case of a linear state-transition map $h_{t+1} = Ah_t + Ba_t$, this requirement corresponds to the condition $\|A\|_{\text{op}} \leq \lambda$, where $\|\cdot\|_{\text{op}}$ denotes the operator norm. In control-theoretic terms, this requirement expresses that the system is stable. Trajectories cannot blow up under repeated application of the state transition map. We show that this control-theoretic stability has a strong regularizing effect.

Theorem 1 (Informal result for progressive analysts)**.** *There is a computationally efficient algorithm to answer t statistical queries chosen adaptively by a λ-progressive analyst so that the error on each query is at most $\tilde{O}\left(\sqrt{K(\lambda) d_q \log(t)/n}\right)$, where $K(\lambda) = O\left(\frac{\log(1/(1-\lambda))}{\log(1/\lambda)}\right)$, d_q is the dimension of the queries, and n is the size of the data set.*

Since Theorem 1 allows queries of arbitrary dimension, d_q can also be thought of as the number of parallel statistical queries in one round, making the total number of one-dimensional queries after t rounds equal to td_q. With this in mind, we can see that for $\lambda = 1 - 1/t$, the bound reduces to the adaptive Hoeffding bound $\tilde{O}(\sqrt{td_q/n})$ (by a first-order Taylor approximation). For any constant λ bounded away from 1, we recover the non-adaptive bound. The proof of this result combines a simple compression argument with recent ideas in the context of recurrent neural networks (Miller & Hardt, 2019).

We could hope that as λ approaches 1 we not only recover the Hoeffding bound but rather the best known adaptive bounds that follow from differential privacy techniques. While this turns out to be difficult for progressive analysts for reasons we elaborate on later, we can indeed achieve this better trade-off for our second notion.

Conservative analysts. Conservative analysts favor initial information over new information in their decision making. Intuitively, this can be seen as a kind of anchoring bias. One of the ways we can express this is by requiring that the state-transition map gets increasingly Lipschitz in its second argument over time:

$$\|\psi_t(h, a) - \psi_t(h, a')\| \leq \eta \|\psi_{t-1}(h, a) - \psi_{t-1}(h, a')\|,$$

for some $\eta \in (0, 1)$. We call analysts satisfying this requirement η^t-*conservative*, leading to the following result.

Theorem 2 (Informal result for conservative analysts, special case)**.** *There is a computationally efficient algorithm to answer t statistical queries chosen adaptively by a η^t-conservative analyst so that the error on each query is at most $\tilde{O}\left((K(\eta^t) d_q \log(t))^{1/4}/\sqrt{n}\right)$, where $K(\eta^t) = O\left(\frac{1}{\log(1/\eta)}\right)$, d_q is the dimension of the queries, and n is the size of the data set.*

Contrary to progressive analysts, if $\eta = 1 - 1/t$, the bound reduces to a multi-dimensional generalization of the hard-to-improve generalization bound $\tilde{O}((td_q)^{1/4}/\sqrt{n})$.

1.2. Proof Technique Overview

The main technical tool used in our generalization proofs is an algorithmic abstraction called the *truncated analyst*. For both progressive and conservative analysts, we design their respective truncated counterpart, which acts according to the

[1] From here forward, we will assume $\|\cdot\|$ denotes some ℓ_p-norm, $p \geq 1$.

same dynamics ψ_t. By construction, the truncated analyst has a time-independent number of rounds of adaptivity. We will also refer to the true analyst as the *full analyst*, to contrast it with the corresponding truncated abstraction.

We first derive a natural conclusion stating that truncated analysts have time-independent generalization properties. Then, we show that, for a large enough level of truncation (which is still time-independent), the truncated analyst closely approximates the full one. This observation will enable us to claim that the full analyst, which is either progressive or conservative, inherits the generalization properties of its corresponding truncated version. One of the conclusions we will derive from here is the following: setting the parameters of progressiveness or conservatism to be constant with respect to the number of interactions yields a generalization error that scales only logarithmically with the number of queries.

2. Analysts as Dynamical Systems

2.1. Problem Setting

Let $\mathcal{S} := \{X_1, \ldots, X_n\} \in \mathcal{D}^n$ be a data set of n i.i.d. samples from a distribution $\mathcal{P}$ supported on $\mathcal{D}$. On one side, there is a data analyst, who initially has no information about the drawn samples in $\mathcal{S}$. On the other side there is a statistical mechanism with access to $\mathcal{S}$, however with no knowledge of its true underlying distribution. At each time step $t \in \mathbb{N}$, the analyst and statistical mechanism have an interaction: the analyst asks a statistical query $q_t \in \mathcal{Q}$, and the statistical mechanism responds with an answer $a_t \in \mathcal{A}$. In the adaptive data analysis literature, statistical queries are typically defined as one-dimensional bounded functions, however in this work we generalize this definition to allow bounded functions in higher dimensions. The motivation for this is that many common procedures query a vector of values; for example, gradient descent queries a gradient of the loss at the current point. Formally, we define statistical queries as functions of the form $q_t : \mathcal{D} \to [0,1]^{d_q}$. In this generalized setting, a single query q_t is equivalent to a set of d_q one-dimensional queries. It is only natural to assume that the dimension of answers matches that of the posed queries, and hence we take $\mathcal{A} \subseteq \mathbb{R}^{d_q}$.

Before deciding on q_t, the analyst takes into account the previous interactions with the statistical mechanism, typically called the *transcript*. In classical work on adaptive data analysis, the transcript at time t consists of all query-answer pairs thus far, $(q_1, a_1, \ldots, q_{t-1}, a_{t-1})$. Recall that, in this work, the analyst only has access to the transcript through its hidden state, or history, $h_t \in \mathcal{H} \subseteq \mathbb{R}^d$, acting according to the recursion:

$$h_t = \psi_t(h_{t-1}, a_{t-1}), \tag{1}$$

where we initialize $h_0 = 0$. The variable h_t serves as a possibly lossy encoding of the knowledge the analyst has gathered about data $\mathcal{S}$ up to time t. Based on this encoding, the analyst picks the next query $q_t \in \mathcal{Q}$:

$$q_t = f_t(h_t), \tag{2}$$

where $f_t : \mathcal{H} \to \mathcal{Q}$ is an arbitrary measurable function.

The goal of designing a statistical mechanism is to have the analyst learn about the *distribution* $\mathcal{P}$, and not just the samples in $\mathcal{S}$. Mathematically, we want the *generalization error*

$$\max_{1 \le i \le t} \|a_i - \mathbb{E}_{X \sim \mathcal{P}}[q_i(X)]\|_\infty$$

to be small with high probability, for any given number of rounds t. The difficulty is this task lies in the fact that the statistical mechanism does not have access to $\mathcal{P}$. It might seem intuitive to set $a_t = q_t(\mathcal{S}) := \frac{1}{n}\sum_{i=1}^n q_i(X_i)$. However, in general, this standard choice quickly leads to overfitting (see the paper (Blum & Hardt, 2015) for an example attack).

A better solution stems from a connection with privacy-preserving data analysis. In particular, it has been shown that good *sample accuracy* combined with *differential privacy* ensures small generalization error (Dwork et al., 2015b; Bassily et al., 2016; Dwork et al., 2015a).

We say that a possibly randomized function $\mathcal{F} : \mathcal{D}^n \to \mathcal{Y} \subseteq \mathbb{R}^d$ is (α, β)-differentially private for some $\alpha, \beta \ge 0$, if for all data sets $S, S' \in \mathcal{D}^n$, such that S and S' differ in at most one entry, it holds that:

$$\mathbb{P}(\mathcal{F}(S) \in \mathcal{O}) \le e^\alpha \mathbb{P}(\mathcal{F}(S') \in \mathcal{O}) + \beta,$$

for any event $\mathcal{O}$. We will extensively rely on some of the well-known properties of differential privacy that we collect in the supplementary materials.

A possibly randomized function $\mathcal{M} : \mathcal{D}^n \times \mathcal{Q} \to \mathcal{Y}$, where $\mathcal{Y} \subseteq \mathbb{R}^d$, is (ϵ, δ)-sample accurate if for every data set $\mathcal{S} = \{X_1, \ldots, X_n\} \in \mathcal{D}^n$ and every query $q \in \mathcal{Q}$, where $q : \mathcal{D} \to \mathcal{Y}$, it holds that:

$$\mathbb{P}(\|\frac{1}{n}\sum_{i=1}^n q(X_i) - \mathcal{M}(\mathcal{S}, q)\|_\infty \ge \epsilon) \le \delta.$$

Applying these definitions to the problem of adaptive data analysis, we simultaneously want $\max_{1 \le i \le t} \|a_i - q_i(\mathcal{S})\|_\infty$ to be small, and a_i to be constructed in a differentially private manner, thus obscuring the exact value of $q_i(\mathcal{S})$. We will show that, with an appropriate choice of a differentially private mechanism, these two conditions will result in favorable generalization properties in our setting.

Our analysis will primarily make use of Gaussian noise addition, as it achieves the hard-to-improve rate of $\tilde{O}(t^{1/4}/\sqrt{n})$,

in the one-dimensional statistical query model. We will use ξ_t to denote a generic Gaussian noise vector; with this, the classical Gaussian mechanism is given by $a_t = q_t(\mathcal{S}) + \xi_t$, where ξ_t is zero-mean noise of d_q independent Gaussians with appropriately chosen variance.

The main idea for preventing adversarial behavior of analysts will be some form of contraction characterizing the state-transition map sequence $\{\psi_t\}$. This approach requires a way to translate closeness in norm into a form of information-theoretic closeness. In general, however, if two different analysts have histories h_t^1 and h_t^2, such that $\|h_t^1 - h_t^2\| \leq \epsilon$ for some very small $\epsilon > 0$, it is impossible to say whether their knowledge of $\mathcal{S}$ is indeed "ϵ-close". For this reason, we introduce the assumption that $\mathcal{H}$ is a discrete grid in $\mathbb{R}^d$ with coordinate-wise resolution $\Delta > 0$, where Δ is sufficiently small. Mathematically, if $h = (h_1, \ldots, h_d) \in \mathcal{H}$, then $h_i = k_i\Delta$, for some $k_i \in \mathbb{Z}$. This way, if two histories are close enough in norm, they have to be semantically identical. This condition is satisfied by a great majority of real-world data analysts. First, all "transcripts" generated by numerical algorithms are memorized in computers using finite-bit precision. Second, human analysts typically use only the first few digits after the decimal of any performed numerical evaluation. It is also worth pointing out that all generalization results obtained for the set $\mathcal{H}$ also hold for all uniformly discrete sets which have a packing radius at least Δ.

3. Progressive Analysts: Motivation and Generalization

The first class of analysts is oblivious in that its knowledge of past events diminishes over time. We will aptly refer to such data analysts as *progressive*.

Definition 1 (Progressive analyst)**.** An adaptive analyst is *λ-progressive* if the maps $\{\psi_t\}$ are λ-contractive in their first argument; for every $h, h' \in \mathcal{H}$ and $a \in \mathcal{A}$, ψ_t satisfies:

$$\|\psi_t(h, a) - \psi_t(h', a)\| \leq \lambda\|h - h'\|,$$

for some $\lambda \in [0, 1]$. Additionally, we require $\psi_t(h, \cdot)$ to be L-Lipschitz for any fixed $h \in \mathcal{H}$; that is, for all $a, a' \in \mathcal{A}$, ad some $L \geq 0$:

$$\|\psi_t(h, a) - \psi_t(h, a')\| \leq L\|a - a'\|.$$

Without loss of generality, we also assume that the maps $\{\psi_t\}$ are normalized to satisfy $\psi_t(0, 0) = 0$. This does not limit their expressiveness.

We now motivate the definition of progressive analysts via three examples, before proving our main generalization bound for this class of analysts.

It is well-known that humans exhibit numerous cognitive biases while performing analytical tasks. One well-known bias is the recency bias (Cheadle et al., 2014). This bias is defined as a tendency to focus more on recent evidence than the history. We can think of recency bias as a motivating analogy for our definition of progressive. In our definition, the parameter λ determines how fast prior information are forgotten. The case $\lambda = 0$ corresponds to full recency bias and virtually no adaptivity in query formulation, while $\lambda = 1$ implies no recency bias and arbitrarily adaptive queries.

As another, contrasting example, iterative algorithms which interact with a fixed data set can also be thought of as adaptive analysts. Suppose that $\mathcal{S}$ contains simulation samples of an agent interacting with a stochastic environment, which returns noisy rewards from an unknown distribution and has known random transitions between a possibly large number of states. This problem can be modeled as a classical Markov decision process (Bertsekas, 2005). Suppose that the analyst wishes to define a set of d states, possibly by grouping the existing elementary states, such that the value function, which is the expected reward-to-go under the optimal policy, satisfies some criterion: for example, one objective could be maximizing the value function in one of the states of the model. First, the analyst initializes the set of states to some arbitrary set of fixed size d. Then, they recurse their hidden state, whose coordinates $i \leq d$ are updated as:

$$h_{t,i} = \sup_a (r_t(i, a) + \gamma \sum_{j=1}^{d} \mathbb{P}(i, a, j) h_{t-1,j}), \qquad (3)$$

where the supremum is taken over the possible actions, $\gamma \in (0, 1)$ is a discount factor, $\mathbb{P}(i, a, j)$ is the probability of landing in state j after taking action a in state i, and $r_t(i, a)$ is the estimated average reward of taking action a in state i. Equation (3) is called the Bellman equation, and the algorithm given by repeated iterations of this equation is called value iteration (Bellman, 1957), as it is used to find the value function. For example, if every sample $X_k \in \mathcal{S}$ is vector containing the initial state, action, reward, and subsequent state, $(s_{1,k}, a_k, r_k, s_{2,k})$, then the estimated reward is given by $r_t(i, a) = \sum_{k=1}^{n} r_k \mathbf{1}\{s_{1,k} = i, a_k = a\} / \sum_{k=1}^{n} \mathbf{1}\{s_{1,k} = i, a_k = a\}$. The analyst's queries are therefore asking for the reward estimates across all states and all actions. After running the Bellman update for a certain number of rounds, the analyst can now adaptively change the set of states, using the previously learned value of h_t for initialization. Since the Bellman equation contracts h_t by factor γ in ℓ_∞-norm, such an analyst would be γ-progressive. The Bellman equation is at the core of numerous dynamic programs, thus making many algorithmic solvers of such problems progressive analysts.

Stable recurrent neural networks are another algorithmic example of progressive analysts. Recurrent neural networks

are given by the update:

$$h_t = \rho(Wh_{t-1} + Ua_{t-1}),$$

where $U \in \mathbb{R}^{d_q \times d}$, $W \in \mathbb{R}^{d \times d}$, and ρ is a point-wise non-linearity. The variable a_t is the empirical answer to an arbitrary query based on h_t. In this case, the analyst is λ-progressive if $\|W\|_{\text{op}} \leq 1/L_\rho$, where L_ρ is the Lipschitz constant of the map ρ. For a detailed treatment of this case, see the paper (Miller & Hardt, 2019). The work also shows how other stateful models, such as LSTMs, can be made stable and how stable models perform well in practice.

Now we argue that the parametrization of progressive analysts allows interpolation between that of non-adaptive analysts and fully adaptive analysts. Then, we move on to proving the generalization error in regimes between these two extremes.

First, consider $L = 0$. In this case, h_t has no sensitivity to the answers of the statistical mechanism, so queries are trivially non-adaptive.

On the other end, $\lambda = 1$ allows full adaptivity, for any $L > 0$. To see this, imagine that h_t is an infinite-dimensional vector[2], where each coordinate is initially 0, and coordinate-wise, h_t can take values in $L\mathcal{A}$. At time t, simply set the coordinates $(t-1)d_q + 1$ through td_q of h_t to La_{t-1}. Since all queries are computed via a deterministic function of the current history, which is composed by stacking the answers, the vector of all answers encodes the whole transcript in a lossless fashion. Consequently, this analyst is fully adaptive. One can easily verify that the described transition maps satisfy the conditions of Definition 1 with $\lambda = 1$.

Since these two extreme cases reduce to generalization rates which are known from prior work, in the rest of this section we focus on the parameter set $\lambda \in [0, 1)$, $L > 0$.

3.1. Truncated Analyst

Now we introduce a useful counterpart of a λ-progressive analyst, who only has access to the last k answers of the full analyst, for some constant k. This truncated analyst will be the main abstraction used in the proofs of this section.

Define the truncated analyst corresponding to a full progressive analyst as:

$$h_t^k = \psi_t(h_{t-1}^k, a_{t-1}), \;\; h_{t-j}^k = 0, \;\; \forall j \geq k,$$
$$q_t^k = f_t(h_t^k),$$

for fixed $k \in \mathbb{N}$. The truncated analyst updates their history according to the same map sequence as the full analyst, and receives exactly k answers *of the full analyst*.

[2]This can be formalized in the framework of separable Hilbert spaces, however this example is only intended to be illustrative.

First we show that, as aligned with intuition, each query of the truncated analyst has a time-independent generalization error.

Lemma 1. *Let h_t^k be the history of a truncated analyst, and let the range of answers be of size A^{d_q}, where A is polynomial in n. Then, at time t, the query q_t^k asked by the truncated analysts satisfies the following:*

$$\mathbb{P}(\|q_t^k(\mathcal{S}) - \mathbb{E}_{X\sim\mathcal{P}}[q_t^k(X)]\|_\infty > \epsilon) \leq 2d_q \exp(kd_q \log A - 2n\epsilon^2).$$

Now we show that contractiveness implied by the progressivenes condition forces the full analyst to be close in norm to its corresponding truncated version.

Lemma 2. *Let $a_t \in [0, 1]^{d_q}$ for all $t \in \mathbb{N}$. For any $k \in \mathbb{N}$, the progressive analyst and the corresponding truncated analyst satisfy $\|h_t^k - h_t\| \leq \frac{\lambda^k L C_{\mathbf{1}}}{1-\lambda}$, where $C_{\mathbf{1}} := \|(1, \ldots, 1)\|$ is the norm of the d_q-dimensional all-ones vector.*

3.2. Generalization via Compression

For a large enough level of truncation k, which depends on the radius Δ of the set of all possible histories $\mathcal{H}$, the truncated analyst and the full analyst are identical. This level of truncation is time-independent, and hence, by Lemma 1, progressive analysts also have a time-independent scaling of the generalization error.

Theorem 1. *Answering t queries chosen adaptively by a λ-progressive analyst by rounding the empirical answer to $O(1/n)$ precision achieves overall generalization error at most $\tilde{O}(\sqrt{K(\lambda) d_q \log(t)/n})$, where $K(\lambda) = \frac{\log\left(\frac{LC_{\mathbf{1}}}{(1-\lambda)\lambda\Delta}\right)}{\log(1/\lambda)}$.*

In other words, having $n = \tilde{O}(K(\lambda) d_q \log(t)/\epsilon^2)$ samples suffices to guarantee ϵ-generalization error with high probability.

Let $\lambda = 1 - \frac{1}{t}$. Then, by the first-order Taylor approximation, $\log(1/\lambda) \approx \frac{1}{t}$, and hence the generalization error of Theorem 1 grows as $\tilde{O}(\sqrt{td_q/n})$. The same scaling of generalization error is achieved by fully adaptive analysts in the case of d_q-dimensional queries, when there is no use of privacy mechanisms. As argued earlier, $\lambda = 1$ corresponds to full adaptivity, so it comes as no surprise that the same rate is achieved.

Note also that the generalization error is completely independent of the dimension of the *history* d. This justifies our "infinite-dimensional" example earlier in this section.

3.3. Limitations

Differential privacy. It is natural to wonder why we never used differential privacy to prevent progressive analysts from overfitting. In the proof of Theorem 1, we allow the

statistical mechanism to return unobscured empirical answers (up to a small rounding error), although we initially argued that differentially private perturbations provide a quadratic improvement.

The main reason is that the truncated analyst *does not* in general have a time-independent composition of differential privacy, in spite of the fact that the number of observed answers is time-independent. This follows from the observation that it receives answers of the full analyst, whose uncertainty grows with time. On a high level, changing one element in the data set $\mathcal{S}$ allows minor changes in the history in each step, even if differential privacy is used. After $t-k$ steps, for some $k \in \mathbb{N}$, these changes might pile up to lead to a completely different query than the one that resulted from the original data set $\mathcal{S}$. The initial input from $\mathcal{S}$ of the truncated analyst is the answer to this query, which is highly unstable for large enough t. Therefore, claiming time-independent generalization, if possible, would require a novel framework for designing mechanisms for adaptive data analysis, one that does not rely on differential privacy.

Naive definition. Stepping away from the dynamical systems perspective for a moment, one might argue that a simple way to smoothly interpolate between no adaptivity and full adaptivity through recency bias is to truncate the analyst's view of the transcript. More formally, define $q_t^K = g_t^K(q_{t-1}^K, a_{t-1}^K, \ldots, q_{t-K}^K, a_{t-K}^K)$, for some fixed $K \in \mathbb{N}$ and functions $\{g_t^K\}$. The input to g_t^K consists of the last K query-response pairs. This seems to be in contrast with the usual adaptive query construction $q_t = g_t(q_{t-1}, a_{t-1}, \ldots, q_1, a_1)$, for some $\{g_t\}$; here, the argument of g_t is *all* query-response pairs so far. However, we claim that this intuitive construction does not necessarily rule out full adaptivity.

Claim 1. *Suppose that an adaptive data analyst has a truncated view of the transcript with truncation depth K. In full generality, this analyst generalizes no better than an analyst with a full view of the transcript, regardless of the mechanism for constructing responses and value of K.*

4. Conservative Analysts, Type A: Motivation and Generalization

The second main class of natural analysts operates in a manner opposite to progressive analysts; namely, these discount new evidence increasingly with time, making their knowledge saturate. We will call such analysts *conservative*.

We consider two possible causes for saturation. Either the maps $\{\psi_t\}$ become less sensitive to new evidence, or the queries $\{q_t\}$ are chosen in such a way that the values $\{q_t(\mathcal{S})\}$ saturate. This distinction leads to two notions of conservative analysts. Type A conservatives and type B conservatives. We will see that each correspond to natural algorithms.

Below we define type A conservative analysts, while we leave the definition of type B for the following section.

Definition 2 (Conservative analyst, type A)**.** An adaptive analyst is type A η_t-conservative if the maps $\{\psi_t(h, \cdot)\}$ are η_t-Lipschitz, where $\lim_{t\to\infty} \eta_t = 0$. Mathematically, this corresponds to:

$$\|\psi_t(h, a) - \psi_t(h, a')\| \leq \eta_t \|a - a'\|,$$

for every $h \in \mathcal{H}$ and $a, a' \in \mathcal{A}$.

The construction of conservative analysts is primarily motivated by gradient descent in various settings in which it experiences saturation. As in the case of progressive analysts, however, there is also a connection between human data analysts and our definition of conservative analysts.

A common cognitive bias that humans experience in analytical tasks is called the anchoring bias (Campbell & Sharpe, 2009; Cen et al., 2013). It is characterized by relying heavily on initial evidence, and becoming decreasingly sensitive to new evidence, as mathematically formulated in Definition 2. The sequence $\{\eta_t\}$ in the definition of conservative analysts can be thought of as the strength of one's anchoring phenomenon. In particular, $\eta_t = 0$ for all $t \in \mathbb{N}$ implies complete anchoring and no adaptivity in formulating queries, while a slow decrease in η_t represents analysts with a mild anchoring effect.

From the algorithmic perspective, examples of conservative analysts include optimization algorithms with decaying step size. Consider the problem of empirical risk minimization using gradient descent. In particular, let the loss be:

$$L(h) = \frac{1}{n} \sum_{i=1}^{n} \ell(h; X_i),$$

where $h \in \mathbb{R}^d$ is a vector of weights for the given optimization model, and $\ell(h; X_i)$ is the loss incurred by this model on sample $X_i \in \mathcal{S}$. The well-known gradient descent update is the following:

$$h_{t+1} = \psi_{t+1}(h_t, \nabla_h L(h_t)) = h_t - \eta_t \nabla_h L(h_t),$$

where $\nabla_h L(h_t)$ is the gradient of the loss on data $\mathcal{S}$ at point h_t, and η_t is a time-dependent, decreasing step. Notice that this gradient decomposes as $\nabla_h L(h_t) = \frac{1}{n}\sum_{i=1}^{n} \nabla_h \ell(h_t; X_i)$. Therefore, gradient descent for empirical risk minimization is an η_t-conservative analyst, whose queries are equal to the gradient of the loss incurred at each point of $\mathcal{S}$ at the current weight iterate.

The rate of step size decay determines the rate of saturation of the analyst, allowing the class of conservative analysts to cover a wide spectrum of gradient-based optimization

algorithms. Notable examples of step size decays include $\eta_t = O(1/t^\alpha)$, where $\alpha \in (0.5, 1]$ (Robbins & Monro, 1951), or the more recent schemes which cut the learning rate by a constant factor in every so-called epoch, which implies an essentially exponential decay (Hazan & Kale, 2014; Ge et al., 2018).

4.1. Truncated Analyst

As its name suggests, the adaptiveness of a conservative analyst essentially saturates after some number of rounds of interaction with the data set. Again, we prove this via truncation of the full analyst. Let the truncated analyst corresponding to the full conservative analyst be the following:

$$\begin{aligned} h_t^k &= \psi_t(h_{t-1}^k, a_{t-1}^k), \\ \text{where } \ a_t^k &= a_t, \forall t \le k \text{ and } a_t^k = 0, \forall t > k, \\ q_t^k &= f_t(h_t^k). \end{aligned}$$

In words, the truncated analyst only sees the first k true answers and deterministically sets the second input to 0 for the remaining $t - k$ rounds.

Lemma 3. *Assume that the answers of the statistical mechanism are bounded to $[0,1]^{d_q}$, and let $C_{\mathbf{1}} := \|(1, \dots, 1)\|$ denote the norm of the d_q-dimensional all-ones vector. Then, for $K(\eta_t) := \min\{t : \eta_t < \frac{\Delta}{C_{\mathbf{1}}}\}$, the history of the full analyst matches the history of the truncated analyst with truncation depth $K(\eta_t)$, $h_t = h_t^{K(\eta_t)}$.*

Since the approximating truncated analyst in this setting sees the first k answers, instead of the last k, the privacy parameters of its history degrade gracefully with k, given that the statistical mechanism is differentially private. The Gaussian mechanism is, however, not bounded, as required by Lemma 3. For this reason, we introduce a slight modification of this mechanism, where the answers are computed as $a_t = [q_t(\mathcal{S}) + \xi_t]_{[0,1]^{d_q}}$, where $[\cdot]_{[0,1]^{d_q}}$ is truncation to the box $[0,1]^{d_q}$:

$$([x]_{[0,1]^{d_q}})_i = \begin{cases} 0, \text{ if } x_i \le 0, \\ x_i, \text{ if } x_i \le 0, \\ 1, \text{ if } x_i \ge 1, \end{cases}$$

where subscript i denotes the i-th coordinate. As before, ξ_t is d_q-dimensional Gaussian noise. By post-processing of differential privacy, this truncated mechanism preserves the parameters of differential privacy of the Gaussian mechanism, determined by the variance of ξ_t.

The next lemma formalizes the gradual degradation of differential privacy of the truncated analyst's history.

Lemma 4. *Let h_t^k be the history of the truncated analyst at time $t \in \mathbb{N}$, and let the statistical mechanism be (α, β)-differentially private. Then, h_t^k is $(\sqrt{2k\log(1/\beta')}\alpha + 2k\alpha^2, k\beta + \beta')$-differentially private.*

4.2. Generalization via Differential Privacy

Since we proved that type A conservative analysts have the same history as their corresponding truncated analyst, for a large enough level of truncation, and that truncated analysts have a time-independent composition of differential privacy, we can conclude a time-independent composition of privacy for the full analyst as well.

Proposition 1. *Let h_t be the hidden state of an oblivious analyst at time t. Let the statistical mechanism answering queries be (α, β)-differentially private. Then, for* arbitrarily large *t, h_t is $(\sqrt{2K(\eta_t)\log(1/\beta')}\alpha + 2K(\eta_t)\alpha^2, K(\eta_t)\beta + \beta')$-differentially private, where $K(\eta_t) := \min\{t : \eta_t \le \frac{\Delta}{C_{\mathbf{1}}}\}$.*

To prove the generalization error of conservative analysts, we turn to the main transfer theorem of Bassily et al. (2016), which is the main technical tool used to establish the celebrated rate of $\tilde{O}((td_q)^{1/4}/\sqrt{n})$. This transfer theorem will allow us to compute the generalization error by balancing out the sample accuracy and differential privacy parameters of the Gaussian mechanism.

Theorem 2. *There is a computationally efficient mechanism to answer t queries chosen adaptively by a type A η_t-conservative analyst so that the overall generalization error is at most $\tilde{O}((K(\eta_t)d_q\log(t))^{1/4}/\sqrt{n})$, where $K(\eta_t) := \min\{t : \eta_t \le \frac{\Delta}{C_{\mathbf{1}}}\}$ and $C_{\mathbf{1}} = \|(1, \dots, 1)\|$ is the norm of the d_q-dimensional all-ones vector.*

Said in terms of sample complexity, it suffices to have $n = \tilde{O}(\sqrt{K(\eta_t)d_q\log(t)}/\epsilon^2)$ samples for ϵ-generalization error. Notice that, just like for progressive analysts, there is no direct dependence on the dimension of the history.

As an example, taking $\eta_t = \eta^t$, where $\eta = 1 - 1/t$, results in error rate $\tilde{O}((td_q)^{1/4}/\sqrt{n})$ by the first-order Taylor approximation. As expected, this is the tight rate for fully adaptive queries under differential privacy.

5. Conservative Analysts, Type B: Motivation and Generalization

In this section we define and analyze linear analysts whose histories saturate despite non-decreasing sensitivity to revealed information about $\mathcal{S}$.

Definition 3 (Conservative analyst, type B)**.** An adaptive analyst is type B λ-conservative if, first, it contracts when given empirical answers:

$$\|\psi_t(h_{t-1}, q_{t-1}(S)) - \psi_t(h'_{t-1}, q'_{t-1}(S))\| \le \lambda\|h_{t-1} - h'_{t-1}\|,$$

for some $\lambda \in [0,1]$, where $q_t = f_t(h_t)$ and $q'_t = f_t(h'_t)$. Second, we require the analyst to be linear:

$$h_t = \psi_t(h_{t-1}, a_{t-1}) = A_t h_{t-1} + B_t a_{t-1},$$

for some sequences $\{A_t\}, \{B_t\}$, where $A_i \in \mathbb{R}^{d\times d}$, with $\|A_i\|_{\text{op}} \leq 1$, and $B_i \in \mathbb{R}^{d\times d_q}$ for all $i \in \mathbb{N}$.

The motivation for type B conservative analysts comes from the observation that gradient-based methods sometimes saturate even if there is no step decay.

Consider again the problem of empirical risk minimization using gradient descent. In this setting, let the gradient descent update have a constant step size $\eta > 0$:

$$h_{t+1} = \psi_{t+1}(h_t, \nabla_h L(h_t)) = h_t - \eta \nabla_h L(h_t),$$

where $\nabla_h L(h_t) = \frac{1}{n}\sum_{i=1}^n \ell(h_t; X_i)$ is again the gradient of the loss on data $\mathcal{S}$. If the loss is β-smooth and μ-strongly convex, and the step size is $\eta \leq \frac{2}{\beta+\mu}$, then the gradient descent update is type B λ-conservative, where $\lambda = 1 - \frac{\eta\beta\mu}{\beta+\mu}$ (Hardt et al., 2016). If the objective is not strongly convex, however is still smooth, gradient descent is non-expansive, meaning it has contraction parameter $\lambda = 1$. In that case, one can induce contractiveness in many ways; one is to add an ℓ_2-regularizer to the objective, that is transform the loss into $L^{\text{reg}}(h) = L(h) + \frac{\mu}{2}\|h\|^2$, for some $\mu > \beta$.

5.1. Truncated Analyst

As in the previous section, due to saturation of conservative analysts, we will define a truncated analyst that has access to k responses of the statistical mechanism. In this case, however, the interaction happens in the *last* k rounds.

Suppose that the statistical mechanism is the usual Gaussian mechanism. Worth mentioning is that this time we deploy no truncation. Denote by ξ_t the noise variable added to the empirical answer at time t. For fixed k, define the truncated analyst corresponding to a type B conservative analyst as:

$$\begin{aligned} h_t^k &= \psi_t(h_{t-1}^k, a_{t-1}^k), \;\; h_{t-j}^k = 0, \forall j \geq k, \\ \text{where } a_t^k &= q_t^k(\mathcal{S}) + \xi_t, \\ q_t^k &= f_t(h_t^k), \end{aligned}$$

In this setting, we assume that $\mathcal{H}$ is a norm-ball with radius D, where D is large enough with respect to $\sum_{i=1}^t \|B_i\|_{\text{op}} C_{\mathbf{1}}$, so that there is no need for projecting the norm of the current history iterates to D. Since this "escaping" event happens with negligible probability, in all subsequent arguments for simplicity we treat it as being of measure zero. First we establish closeness between the full analyst and the truncated version.

Lemma 5. *Suppose that the statistical mechanism is the Gaussian mechanism. For any $k \in \mathbb{N}$, the truncated analyst with truncation depth k and the full analyst satisfy $\|h_t - h_t^k\| \leq \lambda^k D$.*

Additionally, we show that the truncated analyst has a composition of differential privacy which only depends on the truncation depth.

Lemma 6. *Let h_t^k be the history of a truncated analyst corresponding to a type B conservative analyst, and let the statistical mechanism be (α, β)-differentially private. Then, for all $t \in \mathbb{N}$ and $\beta' > 0$, h_t^k is $(\sqrt{2k\log(1/\beta')}\alpha + 2k\alpha^2, k\beta + \beta')$-differentially private.*

5.2. Generalization via Differential Privacy

Lemma 5 allows us to find the effective memory of a conservative analyst, resulting in the following time-independent composition of differential privacy parameters.

Proposition 2. *Let h_t be the history of a type B conservative analyst at time t. Let the statistical mechanism answering queries be the Gaussian mechanism, such that the answers are (α, β)-differentially private. Then, for arbitrarily large t, h_t is $(\sqrt{2K(\lambda)\log(1/\beta')}\alpha + 2K(\lambda)\alpha^2, K(\lambda)\beta + \beta')$-differentially private, where $K(\lambda) := \frac{\log(D/\Delta\lambda)}{\log(1/\lambda)}$.*

The main transfer theorem of Bassily et al. (2016) will now show that the generalization error of type B conservative analysts is essentially the same as for type A conservative analysts, justifying their unification into one broader class.

Theorem 3. *There is a computationally efficient mechanism to answer t queries chosen adaptively by a type B λ-conservative analyst so that the overall generalization error is at most $\tilde{O}((K(\lambda)d_q\log(t))^{1/4}/\sqrt{n})$, where $K(\lambda) := \frac{\log(D/\Delta\lambda)}{\log(1/\lambda)}$.*

In other words, $n = \tilde{O}(\sqrt{K(\lambda)d_q\log(t)}/\epsilon^2)$ samples suffice for ϵ-generalization error. Moreover, under a few additional commonly satisfied assumptions, this sample complexity holds for much more general sets $\mathcal{H}$, which need not be discrete. This essentially follows by linearity of maps ψ_t. We prove this claim, as well as negative results for progressive and type A conservative analysts under continuous sets $\mathcal{H}$, in the supplementary material.

6. Summary

We introduce progressive and conservative analysts by modeling the evolution of their knowledge using different control-theoretic constraints. In addition to serving as mathematical analogies of human cognitive biases, these categories also capture various iterative algorithms, like value iteration or gradient descent. The natural analysts we define achieve generalization error essentially independent of the number of queries, in stark contrast with arbitrary adversarial analysts whose error scales polynomially. In doing so, we combine control-theoretic notions of stability with the algorithmic stability notions underpinning adaptive generalization bounds. The connection between control-theoretic and algorithmic stability for the sake of proving stronger generalization bounds is worth studying further.

Acknowledgements

The authors thank John Miller for his careful reading of a draft of this paper and constructive feedback provided.

References

Bassily, R., Nissim, K., Smith, A., Steinke, T., Stemmer, U., and Ullman, J. Algorithmic stability for adaptive data analysis. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pp. 1046–1059. ACM, 2016.

Bellman, R. E. Dynamic programming. *Princeton University Press*, 1957.

Belloni, A., Chernozhukov, V., and Hansen, C. Inference on treatment effects after selection among high-dimensional controls. *The Review of Economic Studies*, 81(2):608–650, 2014.

Bertsekas, D. P. *Dynamic programming and optimal control*, volume 1. Athena scientific Belmont, MA, 2005.

Blum, A. and Hardt, M. The ladder: A reliable leaderboard for machine learning competitions. In *International Conference on Machine Learning*, pp. 1006–1014, 2015.

Campbell, S. D. and Sharpe, S. A. Anchoring bias in consensus forecasts and its effect on market prices. *Journal of Financial and Quantitative Analysis*, 44(2):369–390, 2009.

Cen, L., Hilary, G., and Wei, K. J. The role of anchoring bias in the equity market: Evidence from analysts' earnings forecasts and stock returns. *Journal of Financial and Quantitative Analysis*, 48(1):47–76, 2013.

Cheadle, S., Wyart, V., Tsetsos, K., Myers, N., De Gardelle, V., Castañón, S. H., and Summerfield, C. Adaptive gain control during human perceptual choice. *Neuron*, 81(6): 1429–1441, 2014.

Dwork, C., Feldman, V., Hardt, M., Pitassi, T., Reingold, O., and Roth, A. Generalization in adaptive data analysis and holdout reuse. In *Advances in Neural Information Processing Systems*, pp. 2350–2358, 2015a.

Dwork, C., Feldman, V., Hardt, M., Pitassi, T., Reingold, O., and Roth, A. L. Preserving statistical validity in adaptive data analysis. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pp. 117–126. ACM, 2015b.

Fithian, W., Sun, D., and Taylor, J. Optimal inference after model selection. *arXiv preprint arXiv:1410.2597*, 2014.

Ge, R., Kakade, S. M., Kidambi, R., and Netrapalli, P. Rethinking learning rate schedules for stochastic optimization. 2018.

Hardt, M. and Ullman, J. Preventing false discovery in interactive data analysis is hard. In *Foundations of Computer Science (FOCS), 2014 IEEE 55th Annual Symposium on*, pp. 454–463. IEEE, 2014.

Hardt, M., Recht, B., and Singer, Y. Train faster, generalize better: Stability of stochastic gradient descent. In *International Conference on Machine Learning*, pp. 1225–1234, 2016.

Hazan, E. and Kale, S. Beyond the regret minimization barrier: optimal algorithms for stochastic strongly-convex optimization. *The Journal of Machine Learning Research*, 15(1):2489–2512, 2014.

Miller, J. and Hardt, M. Stable recurrent models. In *International Conference on Learning Representations (to appear)*, 2019.

Robbins, H. and Monro, S. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3): 400–407, 1951.

Ullman, J., Smith, A., Nissim, K., Stemmer, U., and Steinke, T. The limits of post-selection generalization. In *Advances in Neural Information Processing Systems*, pp. 6402–6411, 2018.