Reproducibility in Learning

Russell Impagliazzo*
University of California-San Diego
russell@eng.ucsd.edu

Rex Lei*
University of California-San Diego
rlei@eng.ucsd.edu

Jessica Sorrell*
University of California-San Diego
jlsorrel@eng.ucsd.edu

1 Introduction

Reproducibility is vital to ensuring scientific conclusions are reliable, and researchers have an obligation to ensure that their results are replicable. However, many scientific fields are suffering from a "reproducibility crisis," a term coined circa 2010 to refer to the failure of results from a variety of scientific disciplines to replicate [Ioa05, OKS+15]. A 2012 Nature article by Begley and Ellis reported that the biotechnology company Amgen was only able to replicate 6 out of 53 landmark studies in haematology and oncology [BE12]. In a 2016 Nature article, Baker published a survey of 1500 researchers, reporting that 70% of scientists had tried and failed to replicate the findings of another researcher, and that 52% believed there is a significant crisis in reproducibility [Bak16].

Within the subfields of machine learning and data science, there are similar concerns about the reliability of published findings. The performance of models produced by machine learning algorithms may be affected by the values of random seeds or hyperparameters chosen during training, and performance may be brittle to deviations from the values disseminated in published results [HIB+17, IHGP17, LKM+18]. To begin addressing concerns about reproducibility, several prominent machine learning conferences have begun hosting reproducibility workshops and holding reproducibility challenges, to promote best practices and encourage researchers to share the code used to generate their results [PVLS+20].

In this work, we aim to initiate the study of reproducibility as a property of algorithms themselves, rather than the process by which their results are collected and reported. We define the following notion of reproducibility, which informally says that a randomized algorithm is reproducible if two distinct runs of the algorithm on two samples drawn from the same distribution, with internal randomness fixed between both runs, produces the same output with high probability.

Definition 1.1 (Reproducibility)**.** Let D be an arbitrary distribution over $\mathcal{X}$. Let $\mathcal{A}$ be a probabilistic algorithm with sample access to D. Then $\mathcal{A}$ is *ρ-reproducible* if $\mathbf{Pr}_{S_1,S_2,r_A}\left[\mathcal{A}(S_1;r_A)=\mathcal{A}(S_2;r_A)\right]\geq 1-\rho$, where S_1 and S_2 are two samples drawn i.i.d. from D, and r_A is the internal randomness used by $\mathcal{A}$.

More generally, we can define reproducibility with respect to access to an arbitrary probabilistic oracle.

Definition 1.2 (Reproducibility)**.** Let $\mathcal{A}^{\mathcal{O}}$ be a probabilistic algorithm with access to probabilistic oracle $\mathcal{O}$. $\mathcal{A}$ is *ρ-reproducible* if $\mathbf{Pr}_{r^1_{\mathcal{O}},r^2_{\mathcal{O}},r_A}\left[\mathcal{A}^{\mathcal{O}(;r^1_{\mathcal{O}})}(;r_A)=A^{\mathcal{O}(;r^2_{\mathcal{O}})}(;r_A)\right]\geq 1-\rho$, where $r^1_{\mathcal{O}}$ (and $r^2_{\mathcal{O}}$) is the randomness used by $\mathcal{O}$ in an entire execution of $\mathcal{A}$, and r_A is the internal randomness used by $\mathcal{A}$.

This definition is inspired by the literature on pseudodeterministic algorithms (see "Related Work"). Reproducibility is a strong stability property that, applied to machine learning algorithms, implies the algorithm is in fact learning something about the underlying distribution from which its sample is drawn, rather than overfitting to its training data.

*Supported by the Simons Foundation and NSF grant CCF-1909634.

Our Results. We first demonstrate the usefulness of reproducibility by giving a reproducible algorithm `rHeavyHitters` for identifying approximate v-heavy-hitters of a distribution, i.e. the elements in the support of the distribution with probability mass at least v.

We compare reproducibile algorithms to statistical query algorithms, showing how to simulate statistical queries reproducibly, and prove the following Theorem.

Theorem 1.3 (Simulating SQ algorithms reproducibly)**.** *Let A be an SQ algorithm that makes $q \in \text{poly}(1/\tau, 1/\delta)$ queries to an SQ oracle with tolerance τ and failure rate δ. Then there exists a reproducible SQ algorithm A' that makes q queries to an SQ oracle with tolerance $\tau/(\text{poly}(q)+1)$ and error rate $1/\text{poly}(q)$ that is $\text{poly}(\tau, \delta)$-reproducible.*

We show that our reproducible heavy-hitter algorithm also gives a separation between reproducible and statistical query algorithms. The algorithm `rHeavyHitters` has sample complexity independent of the size of the domain $\mathcal{X}$, but we show an ensemble of distributions such that any statistical query algorithm must make a number of queries to its oracle that depends on $\mathcal{X}$.

Claim 1.4 (Learning Heavy-hitters using Statistical Queries)**.** *Any statistical query algorithm for the v-heavy-hitters problem requires $\Omega(\log_{1/\tau} |\mathcal{X}|)$ calls to the SQ oracle.*

We compare reproducibility to differential privacy. Differential privacy is an important notion of algorithmic stability that, informally, asks for bounds on the distance between the two distributions induced by an algorithm, when run on datasets that differ in a single element. Crucially, it is asks for the guarantees in the *worst case over datasets*. Reproducibility asks for equality between outputs of an algorithm *with high probability*, for a fixed random string, and so it is natural to ask whether a reproducible algorithm can be generically transformed into an (approximately) differentially private one. We answer this question by showing that ρ-reproducibility implies $(0, 4\rho)$-differential privacy. Finally, we show that answering statistical queries reproducibly allows for adaptive data reuse without significantly compromising the validity of the analysis.

Related Work. Our definition of reproducibility (Definition 1.1) is inspired by the literature on pseudodeterministic algorithms, particularly the work of Grossman and Liu [GL19] and Goldreich [Gol19]. We adapt their notion of reproducibility from the setting of pseudodeterministic algorithms, where they are primarily concerned with reproducing the output of an algorithm given the same input and different randomness. Our notion is more suitable for the setting of machine learning, where we wish to reproduce the output of an algorithm given different inputs (samples) so long as they are drawn from the same distribution, but for a fixed random string.

Other notions of stability that are similar to reproducibility have also been studied; some have been shown to have connections to privacy and generalization (see Bassily, Nissim, Smith, Steinke, Stemmer, and Ullman [BNS+16]). In the context of clustering algorithms, a notion of instability defined as the expected distance between two clusterings of two datasets has been used to design and analyze convergence of clustering algorithms such as K-means (see [vL10] for a survey overview). Definitions of algorithmic and distributional stability such as ϵ-UCO stable (ϵ-uniform change-one) and ϵ-TV (variation distance) stable provide generalization bounds, and these distributional stability notions compose in the adaptive data analysis model.[1] The work of [TS13] utilizes a stability notion called "subsampling stability", getting a differentially private algorithm for computing subsampling-stable functions. Their notion defines stability as a property of (deterministic) functions when subsampling from a dataset, while our notion of reproducibility is a property of (randomized) algorithms.

Among these notions of stability, many are defined so that stability holds as long as outputs are close (i.e. outputs need not be identical). We reiterate that our focus is on algorithms that return the *exact* same output for different samples, not just outputs that are similar.

Open Questions. In this work, we aim to initiate the study of algorithmic reproducibility in learning. To this end, we propose a few general directions of study we consider interesting. We believe designing

[1] For example, see the lecture notes in `https://adaptivedataanalysis.files.wordpress.com/2017/10/lect07-10-draft-v1.pdf`

reproducible approaches to hypothesis testing would be particularly valuable, not only to the machine learning community, but to the broader scientific community as well. If such techniques can be developed and adopted, we believe they could inform a new standard of experimental study design. Even looking backwards, it would be interesting to see if reproducible approaches to data analysis could be used to "audit" published results, in the cases where researchers have made their data public.

To understand the usefulness and limitations of reproducibility, we would naturally like to have upper and lower bounds on sample complexity for standard problems in machine learning theory. In addition, we hope to identify a wider array of techniques for designing reproducible algorithms, beyond the randomized rounding technique we use throughout this work. We would also like to continue the study of the relationship between reproducibility and other desirable properties of machine learning algorithms, such as robustness and low generalization error. Lastly, we believe it would be interesting to study stronger and weaker variants of our definition, and investigate separations or equivalences between them.

2 Heavy-hitters

Here we present our reproducible approximate heavy-hitters algorithm, and give claims stating its sample complexity and reproducibility.

Definition 2.1 (Heavy-Hitter)**.** Let D be a distribution over $\mathcal{X}$. Then we say $x \in \mathcal{X}$ is a *v-heavy-hitter* of D if $\mathbf{Pr}_{x' \sim D}[x' = x] \geq v$.

Let D be a distribution over $\mathcal{X}$. The following algorithm reproducibly returns a set of v'-heavy-hitters of D, where v' is a random value in $[v - \epsilon, v + \epsilon]$. Picking v' randomly allows the algorithm to, with high probability, avoid a situation where the cutoff for being a heavy-hitter (i.e. v') is close to the probability density of any $x \in \text{supp}(D)$.

Algorithm 1 `rHeavyHitters`(S, v, ϵ)

$\mathcal{X}_{\text{set}} \leftarrow Q_1 \stackrel{\text{def}}{=} 6/(\rho(v-\epsilon)^2)$ examples from S // Step 1: Find candidate heavy-hitters
for all $x \in \mathcal{X}_{\text{set}}$ **do** // Step 2: Estimate probabilities
  $S_1 \leftarrow Q_2 \stackrel{\text{def}}{=} \frac{\log(12Q_1/\rho) \cdot Q_1^2}{(\rho\epsilon)^2}$ fresh examples from S
  Estimate $p_x \stackrel{\text{def}}{=} \mathbf{Pr}_{x' \sim D}[x' = x]$ using S_1
$v' \leftarrow_r [v - \epsilon, v + \epsilon]$ uniformly at random // Step 3: Remove non-v'-heavy-hitters
Remove from $\mathcal{X}_{\text{set}}$ all x for which $p_x < v'$.
return $\mathcal{X}_{\text{set}}$

Essentially, `rHeavyHitters` returns exactly the list of v'-heavy-hitters so long as the following holds. In Step 1 of Algorithm 1, all $(v-\epsilon)$-heavy-hitters of D are included in $\mathcal{X}_{\text{set}}$. In Step 2, the probabilities for all $x \in \mathcal{X}_{\text{set}}$ are correctly estimated to within error $\rho\epsilon/(3Q_1)$. In Step 3, the randomly sampled v' does not fall within an interval of width $\rho\epsilon/(3Q_1)$ centered on the true probability of a $(v-\epsilon)$-heavy-hitter of D. We show that these 3 conditions will hold with probability at least $1 - \rho/2$, and so will hold for two executions with probability $1 - \rho$.

Claim 2.2. `rHeavyHitters`$^{\text{EX}}(v)$ *is ρ-reproducible, and has sample complexity* $|S| \in O\left(\frac{\log(1/(\rho^2\epsilon^2(v-\epsilon)^2))}{\rho^4\epsilon^2(v-\epsilon^4)}\right)$.

Proof. We say Step 1 of Algorithm 1 succeeds if all $(v-\epsilon)$-heavy-hitters of D are included in $\mathcal{X}_{\text{set}}$ after Step 1. Step 2 succeeds if the probabilities for all $x \in \mathcal{X}_{\text{set}}$ are correctly estimated to within error $\rho\epsilon/(3Q_1)$. Step 3 succeeds if the returned $\mathcal{X}_{\text{set}}$ is exactly the set of v'-heavy-hitters of D.

In Step 1, an individual $(v-\epsilon)$-heavy-hitter is not included with probabilility at most $(1 - v + \epsilon)^{Q_1}$; union bounding over all $1/(v-\epsilon)$ possible $(v-\epsilon)$-heavy-hitters, Step 1 succeeds with probability at least $1 - \frac{(1-v+\epsilon)^{Q_1}}{v-\epsilon} \geq 1 - \rho/6$.

By a Chernoff bound, each p_x is estimated to within error $\rho\epsilon/(3Q_1)$ with all but probability $\rho/(6Q_1)$ in Step 2. Union bounding over all Q_1 possible $x \in \mathcal{X}_{\text{set}}$, Step 2 succeeds except with probability $\rho/6$.

Conditioned on the previous steps succeeding, Step 3 succeeds if the randomly chosen v' is not within $\rho\epsilon/(3Q_1)$ of the true probability of any $x \in \mathcal{X}_{\text{set}}$ under distribution D. A v' chosen randomly from the interval $[v-\epsilon, v+\epsilon]$ lands in any given subinterval of width $\rho\epsilon/(3Q_1)$ with probability $\rho/(6Q_1)$, and so by a union bound, Step 3 succeeds with probability at least $1-\rho/6$.

Therefore, Algorithm 1 outputs exactly the set of v'-heavy-hitters of D with probability at least $1-\rho/2$. If we consider two executions of Algorithm 1, both using the same shared randomness for chooosing v', output the set of v'-heavy-hitters of D with probability at least $1-\rho$, and so `rHeavyHitters` is ρ-reproducible.

□

3 Statistical Queries and Reproducibility

We show how to use randomized rounding to reproducibly simulate any SQ oracle, and therefore SQ algorithm. The statistical query model introduced by [Kea98] is a restriction of the PAC-learning model introduced by [Val84]. Rather than giving direct access to samples from distribution D, a statistical query oracle takes queries that are functions $\phi : \mathcal{X} \to [0,1]$, and returns the expectation of that function on D up to some specified tolerance τ. We consider the statistical query oracle in the context of unsupervised learning (e.g. see [Fel16]).

Definition 3.1 (Statistical Query Oracle)**.** Let $\tau \in [0,1]$ be the tolerance parameter, and function $\phi : \mathcal{X} \to [0,1]$. Statistical query oracle $\texttt{STAT}(D,\tau)$, on query ϕ, outputs a value v such that $|v - \mathbb{E}_{x\sim D}\phi(x)| \le \tau$ with probability at least $1-\delta$.

Algorithm 2 $\texttt{rSTAT}^{\texttt{STAT}(D,\tau))}(D, 11\tau)(\phi)$ // a reproducible SQ oracle
ϕ: a query $X \times \{\pm 1\} \to [0,1]$

$\alpha \leftarrow 20\tau$
$\alpha_{\text{off}} \leftarrow_r [0,\alpha]$
Split $[0,1]$ in regions: $R = \{[0,\alpha_{\text{off}}], [\alpha_{\text{off}}, \alpha_{\text{off}}+\alpha], \ldots, [\alpha_{\text{off}}+i\alpha, \alpha_{\text{off}}+(i+1)\alpha], \ldots, [\alpha_{\text{off}}+k\alpha, 1]\}$
$v \leftarrow \texttt{STAT}(D,\tau)(\phi)$
Let r_v denote the region in R to which v belongs (break ties arbitrarily)
return the midpoint of region r_v

Claim 3.2 ($\texttt{rSTAT}$ is an SQ oracle)**.** *If $\texttt{STAT}(D,\tau)$ is an SQ oracle for D_f with tolerance τ and failure rate δ, then $\texttt{rSTAT}$ is an SQ oracle for D_f with tolerance 11τ and failure rate δ.*

Proof. With probability at least $1-\delta$, $\texttt{STAT}(D,\tau)(\phi)$ returns a value v within τ of $\mathbb{E}_{(x,y)\sim D}\,\phi(x)$. Outputting the midpoint of region r_v can further offset this result by at most $\alpha/2 = 10\tau$. □

Claim 3.3 ($\texttt{rSTAT}$ is reproducible)**.** $\texttt{rSTAT}$ *is $(2\delta + 1/10)$-reproducible.*

Proof. The probability that either call to $\texttt{STAT}(D,\tau)(\phi)$ fails is at most 2δ. Assuming the two calls to $\texttt{STAT}(D,\tau)$ succed, the values v_1, v_2 returned by $\texttt{STAT}(D,\tau)(\phi)$ differ by at most 2τ. $\texttt{rSTAT}$ outputs different values for the two runs iff v_1 and v_2 are in different regions of R. Since these regions are chosen by a random offset α_{off}, the probability that v_1 and v_2 land in different regions is at most $2\tau/20\tau = 1/10$. □

Corollary 3.4. *The construction in Algorithm 2 converts an SQ oracle $\texttt{STAT}(D,\tau)$ into a reproducible SQ oracle $\texttt{rSTAT}$ with tolerance $\tau + \alpha/2$, failure rate δ, and reproducibility $\rho \le 2\tau/\alpha + 2\delta$.*

Theorem 3.5 (Simulating SQ algorithms reproducibly)**.** *Let A be an SQ algorithm that makes $q \in \text{poly}(1/\tau, 1/\delta)$ queries to an SQ oracle with tolerance τ and failure rate δ. Then there exists a reproducible SQ algorithm A' that makes q queries to an SQ oracle with tolerance $\tau/(\text{poly}(q)+1)$ and error rate $1/\text{poly}(q)$ that is $\text{poly}(\tau,\delta)$-reproducible.*

Proof. Let $n = 1/(\tau\delta)$ and $\tau' \stackrel{\text{def}}{=} \tau/(nq+1)$. Apply the construction in Algorithm 2 to an SQ oracle $\texttt{STAT}(D, \tau')$ with error rate $\delta' \stackrel{\text{def}}{=} 1/(nq)$. Using $\alpha = 2\tau'/\delta'$ yields a reproducible SQ oracle $\texttt{rSTAT}$ with tolerance τ, error δ', and reproducibility $3\delta'$. (Note: by definition, $\delta' < \delta$.) Let A' be the algorithm that runs A using $\texttt{rSTAT}$. A' reproduces if each call to $\texttt{rSTAT}$ yields the same output for both executions. By a union bound over the q calls to $\texttt{rSTAT}$, A' is $6/n$-reproducible. □

Learning Heavy-hitters using Statistical Queries. Next, we show that any statistical query algorithm for the v-heavy-hitters problem requires $\Omega(\log|\mathcal{X}|/\log(1/\tau))$ calls to the SQ oracle. Since Algorithm 1 has a sample complexity independent of the domain size, this implies a separation between reproducible problems and problems solvable with SQ queries.

Claim 3.6 (Learning Heavy-hitters using Statistical Queries)**.** *Let $\{D_x\}_{x\in\mathcal{X}}$ be an ensemble of distributions over $\mathcal{X}$, where D_x is supported entirely on a single $x \in \mathcal{X}$. Then any statistical query algorithm for the v-heavy-hitters problem on this ensemble requires $\Omega(\log|\mathcal{X}|/\log(1/\tau))$ calls to the SQ oracle.*

Proof. Consider the ensemble $\{D_x\}_{x\in\mathcal{X}}$ on $\mathcal{X}$, where distribution D_x is supported entirely on a single $x \in \mathcal{X}$. An adversarial SQ oracle has tolerance τ to permute the result of a statistical query ϕ. So, for any ϕ, there must be some distribution D_x for which the following holds: at least a τ-fraction of the distributions $D_{x'}$ in the ensemble satisfy $|\phi(x') - \phi(x)| \le \tau$. Thus, any correct SQ algorithm can rule out at most a $(1-\tau)$-fraction of the distributions in the ensemble with one query. If $\mathcal{X}$ is finite, then an SQ algorithm needs at least $\log_{1/\tau}(|\mathcal{X}|)$ queries. □

4 Privacy

We show that reproducibility implies approximate differential privacy.

Definition 4.1 ((ϵ,δ)-Differential Privacy [DMNS06])**.** A randomized algorithm $\mathcal{A} : \mathcal{X}^n \to \mathcal{Y}$ is (ϵ,δ)-differentially private if for all datasets $S, S' \in \mathcal{X}^n$ differing in at most a single element, and for all measurable $T \subseteq \mathcal{Y}$, we have that $\mathbf{Pr}[\mathcal{A}(S) \in T] \le e^{\epsilon}\mathbf{Pr}[\mathcal{A}(S') \in T] + \delta$.

Intuitively, we can construct a differentially private algorithm $\mathcal{A}'$ as follows. The algorithm $\mathcal{A}'$ will draw a subsample of size n from its own sample, and return the output of the reproducible algorithm $\mathcal{A}$ on this subsample. Reproducibility implies the output of $\mathcal{A}$ cannot be too different depending on the presence or absence of a single data point in its input subsample, and therefore the same is true of the output of $\mathcal{A}'$.

Theorem 4.2 (Reproducibility ⇒ Privacy)**.** *If a randomized algorithm $\mathcal{A} : \mathcal{X}^n \to \mathcal{Y}$ is ρ-reproducible, then there exists an algorithm $\mathcal{A}' : \mathcal{X}^m \to \mathcal{Y}$ that is $(0, \frac{2\rho m}{m-n})$-differentially private.*

Proof. The algorithm $\mathcal{A}'$ proceeds as follows. On input $S \in \mathcal{X}^m$, $\mathcal{A}'$ draws a subsample $U \in \mathcal{X}^n$ and outputs $h \leftarrow \mathcal{A}(U)$.

To show that $\mathcal{A}'$ is differentially private, we first consider the behavior of $\mathcal{A}$ on two independently drawn subsamples from S, and on two independently drawn subsamples from S'. Let x and x' denote the elements on which neighboring sets S and S' differ, assuming without loss of generality that they are unique in their respective multisets so that $x' \in S'$, $x' \notin S$ and $x \in S$, $x \notin S'$. We observe that for a subsample U of size n, we have $\mathbf{Pr}_{U\sim S}[x \notin U] = (1-1/m)^n \ge 1 - n/m$. Then the ρ-reproducibility of $\mathcal{A}$ give us

$$
\begin{aligned}
1-\rho &\le \mathop{\mathbf{Pr}}_{\substack{U_0,U_1\sim S\\ r}}[\mathcal{A}(U_0;r) = \mathcal{A}(U_1;r)] \\
&= \mathop{\mathbf{Pr}}_{\substack{U_0,U_1\sim S\\ r}}[\mathcal{A}(U_0;r) = \mathcal{A}(U_1;r) \mid x \notin U_1] \cdot \mathbf{Pr}[x \notin U_1] + \mathop{\mathbf{Pr}}_{\substack{U_0,U_1\sim S\\ r}}[\mathcal{A}(U_0;r) = \mathcal{A}(U_1;r) \mid x \in U_1] \cdot \mathbf{Pr}[x \in U_1].
\end{aligned}
$$

Because $\mathbf{Pr}[x \notin U_1] \ge 1 - n/m$, we then have

$$\begin{aligned}\Pr_{\substack{U_0,U_1\sim S\\ r}}[\mathcal{A}(U_0;r)=\mathcal{A}(U_1;r)\mid x\notin U_1] &\geq \frac{1-\rho-\Pr_{\substack{U_0,U_1\sim S\\ r}}[\mathcal{A}(U_0;r)=\mathcal{A}(U_1;r)\mid x\in U_1]\cdot\Pr[x\in U_1]}{\Pr[x\notin U_1]}\\ &\geq \frac{1-\rho-\Pr[x\in U_1]}{\Pr[x\notin U_1]}\\ &= \frac{\Pr[x\notin U_1]-\rho}{\Pr[x\notin U_1]}\\ &\geq 1-\frac{\rho m}{m-n}.\end{aligned}$$

Analogously, for subsamples from S' we have $\Pr_{\substack{U_0,U_1\sim S'\\ r}}[\mathcal{A}(U_0;r)=\mathcal{A}(U_1;r)\mid x'\notin U_1]\geq 1-\frac{\rho m}{m-n}$.

Note that each subsample U_1 from S' such that $x'\notin U_1$ has equal probability of being subsampled from S, and so conditioning on $x\notin U_1$, we can draw U_1 from S in the probability above, rather than from S', which gives

$$\Pr_{\substack{U\sim S\\ U'\sim S'\\ r}}[\mathcal{A}(U;r)=\mathcal{A}(U';r)]\geq 1-\frac{2\rho m}{m-n}.$$

Since we have just shown that $\mathcal{A}(U;r)=\mathcal{A}(U';r)$ except with probability $\frac{2\rho m}{m-n}$, it follows that

$$\Pr_{U\underset{\mathcal{A}}{\sim}S}[\mathcal{A}(U)\in T]\leq \Pr_{U'\underset{\mathcal{A}}{\sim}S'}[\mathcal{A}(U')\in T]+\frac{2\rho m}{m-n},$$

and so we can bound the privacy loss of $\mathcal{A}'$ by

$$\Pr_{\mathcal{A}'}[\mathcal{A}'(S)\in T]=\Pr_{U\underset{\mathcal{A}}{\sim}S}[\mathcal{A}(U)\in T]\leq \Pr_{U'\underset{\mathcal{A}}{\sim}S'}[\mathcal{A}(U')\in T]+\frac{2\rho m}{m-n}=\Pr_{\mathcal{A}'}[\mathcal{A}'(S')\in T]+\frac{2\rho m}{m-n}.$$

So as long as $m>2n$, $\mathcal{A}'$ is $(0,4\rho)$-differentially private. □

5 Reproducibility Implies Adaptivity

We consider adaptive data analysis as discussed in [DFH+15b] and [DFH+15a]. The proof of Claim 5.2 follows from a hybrid argument. First, we define a slightly stronger notion of reproducibility:

Definition 5.1 (Reproducibility w.r.t. Inputs)**.** Let $\mathcal{X}$ be a set of strings. Let $\mathcal{A}^{\mathcal{O}}(x)$ be a probabilistic algorithm with access to probabilistic oracle $\mathcal{O}$ and input string $x\in\mathcal{X}$. $\mathcal{A}$ is *ρ-reproducible with respect to $\mathcal{X}$* if $\forall x\in\mathcal{X}, \Pr_{r^1_{\mathcal{O}},r^2_{\mathcal{O}},r_A}\left[\mathcal{A}^{\mathcal{O}(;r^1_{\mathcal{O}})}(x;r_A)=A^{\mathcal{O}(;r^2_{\mathcal{O}})}(x;r_A)\right]\geq 1-\rho$, where $r^1_{\mathcal{O}}$ (and $r^2_{\mathcal{O}}$) is the randomness used by $\mathcal{O}$ in an entire execution of $\mathcal{A}$, and r_A is the internal randomness used by $\mathcal{A}$.

Claim 5.2 (Reproducibility $\Longrightarrow$ Data Reusability)**.** *Let D be a distribution over domain $\mathcal{X}$. Let $\mathcal{M}$ be a mechanism that answers queries of the form $q:\mathcal{X}\to\{0,1\}$ by drawing a sample S of n i.i.d. examples from D and returning answer a. Let $\mathcal{A}$ denote an algorithm making m* adaptive *queries, chosen from a set of queries Q, so that the choice of q_i may depend on q_j, a_j for all $j<i$. Denote by $[\mathcal{A},\mathcal{M}]$ the distribution over transcripts $\{q_1,a_1,\ldots q_m,a_m\}$ of queries and answers induced by $\mathcal{A}$ making queries of $\mathcal{M}$. Let $\mathcal{M}'$ be a mechanism that behaves identically to $\mathcal{M}$, except it draws a single sample S' of n i.i.d. examples from D and answers all queries with S'.*

If $\mathcal{M}$ is ρ-reproducible with respect to Q, then $SD_\Delta([\mathcal{A},\mathcal{M}],[\mathcal{A},\mathcal{M}'])\leq(m-1)\rho$.

Proof. For $i\in[m]$, let $[\mathcal{A},\mathcal{M}_i]$ denote the distribution on transcripts output by algorithm $\mathcal{A}$'s interaction with $\mathcal{M}_i$, where $\mathcal{M}_i$ is the analogous mechanism that draws new samples $S_1,\ldots,S_i$ for the first i queries, and reuses sample S_i for the remaining $m-i$ queries. Note that $\mathcal{M}'=\mathcal{M}_1$ and $\mathcal{M}=\mathcal{M}_m$.

For $i \in [m-1]$, consider distributions $[\mathcal{A}, \mathcal{M}_i]$ and $[\mathcal{A}, \mathcal{M}_{i+1}]$. We will bound the statistical distance by a coupling argument. Let $S_1, \ldots, S_{i+1}$ denote random variables describing the samples used, and let r denote the randomness used over the entire procedure. $[\mathcal{A}, \mathcal{M}_i]$ can be described as running the entire procedure (with randomness R) on $S_1, \ldots, S_{i-1}, S_{i+1}, S_{i+1}, \ldots, S_{i+1}$, and $[\mathcal{A}, \mathcal{M}_{i+1}]$ can be described as running the entire procedure (with randomness R) on $S_1, \ldots, S_{i-1}, S_i, S_{i+1}, S_{i+1}, \ldots, S_{i+1}$.

These distributions are identical for the first $i-1$ queries and answers, so the i'th query q_i is identical, conditioned on using the same randomness. Both S_i and S_{i+1} are chosen by i.i.d. sampling from D, so reproducibility implies that, $\mathbf{Pr}_{S_i, S_{i+1}, r}\left[\mathcal{A}(q_i, S_{i+1}; r) = A(q_i, S_i; r)\right] \geq 1-\rho$. Conditioned on both transcripts including the same $(i+1)$'th answer a_{i+1} (and continuing to couple S_{i+1} and r for both runs), the remaining queries and answers $q_{i+1}, a_{i+1}, \ldots, q_m, a_m$ is identical. Therefore, $SD_\Delta([\mathcal{A}, \mathcal{M}_i], [\mathcal{A}, \mathcal{M}_{i+1}]) \leq \rho$ for all $i \in [m-1]$. Unraveling, $SD_\Delta([\mathcal{A}, \mathcal{M}], [\mathcal{A}, \mathcal{M}']) \leq (m-1)\rho$. □

Remark 5.3. *This connection may be helpful for showing that reproducibility cannot be achieved efficiently in contexts where data reuse is not efficiently achievable.*

Acknowledgements. The authors would like to thank Cynthia Dwork, Toni Pitassi, Rahul Santhanam, and Ryan Williams for interesting discussions.

References

[Bak16] Monya Baker. 1,500 scientists lift the lid on reproducibility. *Nature News*, 533(7604):452, May 2016.

[BE12] C. Glenn Begley and Lee M Ellis. Raise standards for preclinical cancer research. *Nature (London)*, 483(7391):531–533, 2012.

[BNS+16] Raef Bassily, Kobbi Nissim, Adam Smith, Thomas Steinke, Uri Stemmer, and Jonathan Ullman. Algorithmic stability for adaptive data analysis. In *Proceedings of the Forty-Eighth Annual ACM Symposium on Theory of Computing*, STOC 2016. Association for Computing Machinery, 2016.

[DFH+15a] Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Roth. Generalization in adaptive data analysis and holdout reuse. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'15, page 23502358, Cambridge, MA, USA, 2015. MIT Press.

[DFH+15b] Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Leon Roth. Preserving statistical validity in adaptive data analysis. In *Proceedings of the Forty-Seventh Annual ACM Symposium on Theory of Computing*, STOC '15, page 117126, New York, NY, USA, 2015. Association for Computing Machinery.

[DMNS06] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In Shai Halevi and Tal Rabin, editors, *Theory of Cryptography*, pages 265–284, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[Fel16] V. Feldman. A general characterization of the statistical query complexity. *CoRR*, abs/1608.02198, 2016.

[GL19] Ofer Grossman and Yang P. Liu. Reproducibility and pseudo-determinism in log-space. In Timothy M. Chan, editor, *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, San Diego, California, USA, January 6-9, 2019*, pages 606–620. SIAM, 2019.

[Gol19] Oded Goldreich. Multi-pseudodeterministic algorithms. *Electron. Colloquium Comput. Complex.*, 26:12, 2019.

[HIB+17] Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. Deep reinforcement learning that matters, 2017. cite arxiv:1709.06560Comment: Accepted to the Thirthy-Second AAAI Conference On Artificial Intelligence (AAAI), 2018.

[IHGP17] Riashat Islam, Peter Henderson, Maziar Gomrokchi, and Doina Precup. Reproducibility of benchmarked deep reinforcement learning tasks for continuous control. *CoRR*, abs/1708.04133, 2017.

[Ioa05] John P. A. Ioannidis. Why most published research findings are false. *PLOS Medicine*, 2, 08 2005.

[Kea98] M. J. Kearns. Efficient noise-tolerant learning from statistical queries. *Journal of the ACM*, 45(6):983–1006, 1998.

[LKM+18] Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, and Olivier Bousquet. Are gans created equal? a large-scale study. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[OKS+15] Open Science Collaboration, Robert Wilhelm Krause, Sabine Scholz, Hedderik van Rijn, and Eric-Jan Wagenmakers. Estimating the reproducibility of psychological science. *Science*, 349(6251), August 2015. Copyright © 2015, American Association for the Advancement of Science.

[PVLS+20] Joelle Pineau, Philippe Vincent-Lamarre, Koustuv Sinha, Vincent Larivière, Alina Beygelzimer, Florence d'Alché Buc, Emily Fox, and Hugo Larochelle. Improving reproducibility in machine learning research (a report from the neurips 2019 reproducibility program). *arXiv preprint arXiv:2003.12206*, 2020.

[TS13] Abhradeep Guha Thakurta and Adam Smith. Differentially private feature selection via stability arguments, and the robustness of the lasso. In Shai Shalev-Shwartz and Ingo Steinwart, editors, *Proceedings of the 26th Annual Conference on Learning Theory*, volume 30 of *Proceedings of Machine Learning Research*, pages 819–850, Princeton, NJ, USA, 12–14 Jun 2013. PMLR.

[Val84] L. G. Valiant. A theory of the learnable. In *Proc. 16th Annual ACM Symposium on Theory of Computing (STOC)*, pages 436–445. ACM Press, 1984.

[vL10] Ulrike von Luxburg. Clustering stability: An overview. *Foundations and TrendsÂ® in Machine Learning*, 2(3):235–274, 2010.