Electronic Colloquium on Computational Complexity, Report No. 126 (2015)

Memory, Communication, and Statistical Queries

Jacob Steinhardt* Gregory Valiant† Stefan Wager‡

Stanford University

Abstract

If a concept class can be represented with a certain amount of memory, can it be efficiently learned with the same amount of memory? What concepts can be efficiently learned by algorithms that extract only a few bits of information from each example? We introduce a formal framework for studying these questions, and investigate the relationship between the fundamental resources of memory or communication and the sample complexity of the learning task. We relate our memory-bounded and communication-bounded learning models to the well-studied statistical query model. This connection can be leveraged to obtain both upper and lower bounds: we show strong lower bounds on learning parity functions with bounded communication, as well as the first upper bounds on solving generic sparse linear regression problems with limited memory.

1 Introduction

The increasing scale of problems we want to solve has led to new computing architectures, ranging from GPUs to distributed networks of computers, for which performance is not constrained by the maximum possible number of operations per second. Instead, the performance of such systems often depends on resources such as the memory per processor or machine, or the amount of communication required by the algorithm. This trajectory of modern computing suggests that we revisit the theory of the learnable with an eye not only towards the usual resources of time (number of operations required) and data (number of examples required), but with a consideration of both memory and communication.

From a very different perspective, it seems both practically and philosophically important to understand what factors drive the difficulty of a learning problem. Intuitively, we might assume that easy problems can still be learned in the presence of various constraints, whereas hard problems have fragile solutions that cannot withstand them; under this view, robustness to resource constraints can provide us with a richer understanding of the hardness of learning than traditional PAC theory. One natural question in this line is to explore which concept classes can be efficiently learned by an algorithm that does not require significantly more memory than is required to store the true concept—essentially by an algorithm that can learn the concept class "in memory" without requiring additional side computations or "scratch paper." Similarly, many seemingly easy learning tasks have the property that even if relatively little information from each example is used, learning is still possible and rapid; it is natural to ask for a characterization of the class of problems that can be learned in this way. While these constraints on memory or communication may seem

*jsteinhardt@cs.stanford.edu

†valiant@stanford.edu

‡swager@stanford.edu

ISSN 1433-8092

stringent, it is tempting to argue that nearly all of the processes by which humans learn—in addition to requiring relatively few examples and time—have little memory overhead and allow for a considerable compression of the information given in each example (for instance, millions of pixels may simply be remembered as "a persimmon").[1]

1.1 Our Contributions

Motivated by the dual goals of attending to the memory and communication considerations of modern systems, and developing a more nuanced understanding of the difficulty of learning problems, we introduce and study three complexity classes for investigating what concepts can be learned efficiently given memory or communication constraints (see Section 2 for formal definitions):

- The class MEM(b) of concepts that are learnable from a polynomial number of examples, by an algorithm that uses at most b bits of memory.
- The class COM(b) of concepts that are learnable from a polynomial number of examples, by an algorithm that can extract at most b bits of information from each example. Equivalently, this class can be viewed as those concepts that are learnable via an arbitrary multiround multiparty communication protocol in which there are a polynomial number of parties, each party is given a labeled example, and each party can broadcast at most b bits over the course of the algorithm.
- The class sCOM(b) of concepts that are learnable in a streaming setting by a protocol in which each example must be compressed to b bits before the next example is given. Note that the compression function can be adaptive, and can depend arbitrarily on the earlier compressed examples. The class sCOM(b) can also be viewed as the restriction of COM(b) to the setting in which each party is allowed to broadcast a single b-bit message.

We now provide an informal summary of our main theorems. Our main results connect the classes COM and MEM with the class of concepts efficiently learnable via statistical query algorithms (SQ); see Section 2 for formal definitions. These connections are then leveraged to yield both upper, and lower bounds for several learning problems of interest.

Our first theorem shows that classes of function that can be learned via statistical queries can be learned with relatively little memory. We stress that our proof of this result is of an existential nature, and the reduction may not be computationally efficient.

Theorem. *If a class of functions $\mathcal{F}$ over length n examples is learnable with* $\mathrm{poly}(n)$ *statistical queries of tolerance* $1/\mathrm{poly}(n)$*, then it is learnable with at most* $\mathrm{poly}(n, \log|\mathcal{F}|)$ *samples and* $b = \mathcal{O}\left(\log|\mathcal{F}| \log(n)\right)$ *bits of memory.*

Our next theorem shows a tight correspondence between those classes of function that are learnable via statistical queries, and those that are learnable using a single bit of communication per example:

Theorem. *A concept class is learnable from a polynomial number of statistical queries (of a polynomial tolerance) if, and only if, it is learnable with a polynomial number of examples in the communication setting in which $\mathcal{O}(\log n)$ bits of information are extracted from each example. That is, for any constant $C > 0$,*

$$\mathrm{COM}(1) = \mathrm{COM}(C \log n) = \mathrm{sCOM}(1) = \mathrm{sCOM}(C \log n) = \mathrm{SQ}\,.$$

[1] Beyond the memory and communication bounds studied in this paper, such constraints could also involve, e.g., limits on the class of algorithms that can be used or robustness requirements with respect to faulty hardware.

One direction of the above theorem—showing that SQ $\subset$ COM(1)—is trivial. The other direction is given as a consequence of the following theorem:

Theorem. *If a concept class can be (ϵ, δ) learned with m examples and b bits of communication per example, then it can be $(\epsilon, 2\delta)$ learned by a statistical query algorithm that asks $2bm$ queries of tolerance $\tau = \frac{1}{2^{b+1}m}$.*

Communication Lower bounds

The preceding theorem serves as a powerful tool allowing one to immediately translate exponential lower bounds for SQ into exponential lower bounds for bounded communication protocols. We illustrate this via lower bounds for the well studied problem of learning a parity (PARITY). Recall that the problem of learning a parity over n-bit examples is the problem of recovering an arbitrary set $S \subseteq \{1, \ldots, n\}$ given access to uniformly random examples $x \in \{0,1\}^n$ together with their label $\ell(x) = \sum_{i \in S} x_i \mod 2$. The above theorem immediately implies that, as a direct corollary to the classical exponential SQ lower bounds for learning PARITY of Blum et al. [1994], any communication protocol for learning PARITY in which at most $n/4$ bits are communicated per example requires $2^{\Omega(n)}$ examples.

The following more striking corollary of the above theorem is also established by first showing the associated SQ lower bound.

Corollary. *Given a multiround multiplayer protocol for learning PARITY over length n examples in which each party receives $n/4$ uniformly random labeled examples and each party can communicate at most $b = n/16$ bits, it is necessary to have $2^{\Omega(n)}$ parties in order to recover the correct parity with probability at least $2/3$.*

This result is surprisingly strong: if each party is allowed to communicate arbitrarily, then with decent probability, 4 parties would suffice to learn the parity. Meanwhile, if each party is allowed to communicate $b = n$ bits, then $\mathcal{O}(n)$ parties would suffice to learn the parity; and if each party receives n examples rather than $n/4$ examples, then even if the communication per party is limited to $b = 1$ bit, $\mathcal{O}(n)$ parties would also be capable of learning the parity.

While our theorem allows exponential SQ lower bounds to be mapped to exponential lower bounds in the bounded communication setting, the constants in the exponent are not in general tight. Intriguingly, we show that an alternate approach based on Assouad's method [Assouad, 1983] seems capable of giving nearly tight results in the streaming setting sCOM. As one example of this approach we show the following polynomially tight bounds on learning PARITY with noise in the streaming bounded communication setting:

Theorem. *Given an instance of PARITY over n bit examples for which the label of each example is corrupted independently with constant probability $\eta < 1/2$, any streaming algorithm that (even adaptively) compresses each example down to b bits necessarily requires $\Omega(2^{n-b})$ examples to learn the parity. This is tight to a linear factor, as $\mathcal{O}(n2^{n-b})$ examples suffice to learn a parity in this setting.*

Upper Bounds

We also consider implications of our reduction from SQ to MEM for sparse high-dimensional learning. In several common statistical learning problems, we seek to learn an n-dimensional parameter vector that only has $k \ll n$ non-zero entries. A natural question to ask is whether we need $\mathcal{O}(n)$ memory to solve such problems, or if maintaining memory that scales with k is enough. Our

framework enables us to move towards an answer to this question: We provide a sample-efficient statistical query algorithm for sparse online linear regression, and then use the connection between SQ and MEM to provide nearly memory-optimal algorithms requiring only $k \cdot \text{polylog}(n)$ memory and $n \cdot \text{poly}(k)$ samples.[2] Of course, because our reduction from SQ to MEM is implicit, and not necessarily computationally efficient, this should be viewed as an existential result rather than a practical result.

Open Problems

This work raises some intriguing open questions. Perhaps the most pressing (and, in some sense, embarrassing) open problem is to prove an assumption-free separation between MEM and PAC. We strongly believe that the problem of learning a parity should provide such a separation. Standard Gaussian elimination can solve parity with $\mathcal{O}(n)$ samples and n^2 memory; however, we conjecture that the required sample complexity grows exponentially if the available memory decreases by a constant factor. While we were able to make some progress towards this result—for example proving that the statement holds for any algorithm whose memory states correspond to subspaces—a complete proof eludes us.

Conjecture 1.1. *Any algorithm for learning parities over n-bit examples requires either at least $2^{n/4}$ labeled examples (in expectation), or at least $n^2/4$ bits of memory.*

In a different direction, it would be useful to give a computationally efficient reduction from SQ to MEM, as opposed to our implicit reduction. One related and perhaps more modest and practically relevant goal would be to provide a practical memory-bounded algorithm for sparse regression.

1.2 Related Work

Building off a number of early efforts to develop classification algorithms, as well as algorithms for learning patterns, grammars, etc., the introduction of the PAC learning framework [Valiant, 1984] offered a formal model in which to investigate a theory of learnability via the rich and rigorous tools of probability, computational complexity, and information theory. Similarly, while there were a number of efforts to develop noise-tolerant learning algorithms, the introduction of the *statistical query* model of Kearns [1998] allowed for the development of a more general theory of how learning algorithms may be robust to noise, and the development of a common set of abstract tools for the design of such algorithms. Informally, a statistical query learning algorithm is one that never sees any actual examples, and, instead, interacts with the data via an adaptive sequence of queries that request noisy estimates of various statistics of the example/label pairs. Any such algorithm is inherently robust to noise in the label. Surprisingly, most common learning algorithms can be adapted to be statistical query algorithms. Since the initial definition of the statistical query framework, there have been successful efforts to characterize which concepts can be learned within this framework [e.g., Blum et al., 1994], including recent results showing inherent connections between statistical query learnability and differential privacy [Gupta et al., 2011, Balcan and Feldman, 2013] and evolvability [Feldman, 2009].

Perhaps the line of investigation most closely related to ours is the recent work of Shamir [2014]. That work considers a class of information theoretic constraints that can be applied to learning

[2] As discussed in Section 4.1, this is essentially the best result we could hope for; Steinhardt and Duchi [2015] show that solving sparse regression with $\text{poly}(k, \log(n))$ memory and $\text{poly}(k, \log(n))$ samples is impossible, even with uncorrelated features.

processes, which implies results for online learning given partial information, and learning with memory and communication restrictions. The main difference is that both their definitions and techniques are strictly in terms of information theoretic quantities. Thus, due to inherent limitations of the information-theoretic approach explained in Section 3.3, their results are necessarily limited to linear trade-offs between the information and the number of samples required, as opposed to the distinctions between polynomial versus super-polynomial that we seek.

Several authors have also considered communication and/or privacy restrictions in the distributed data setting [Balcan et al., 2012, Duchi et al., 2013, Zhang et al., 2013, Garg et al., 2014, M. Braverman, 2015]. As in Shamir [2014], this work is largely restricted to classical information theoretic arguments, and hence again obtains only linear trade-offs between the information exchanged per sample/round and the sample complexity/number of rounds.

An early work of Ben-David and Dichterman [1998] also considers several learning models in the streaming setting in which limited information may be extracted from each example. Among other models, they consider one equivalent to our sCOM streaming model with bounded communication, and establish the polynomial equivalence of sCOM($\log n$) and SQ. They exploit this characterization to provide lower bounds for learning PARITY, though our bounds are stronger in two ways. First, their reduction only shows exponential sample lower bounds for PARITY when $\log n$ bits per example are communicated (in contrast to our result which holds even if $n/4$ bits are communicated per example); second, their results only apply to bounded communication protocols in the streaming setting (sCOM) rather than the more general multi-round communication setting (COM).

There is a huge literature on communication complexity (see Kushilevitz and Nisan [1997], Lee and Shraibman [2009] for surveys), and the related concept of information complexity from the theoretical computer science community [Chakrabarti et al., 2001, Bar-Yossef et al., 2004, Braverman and Rao, 2011]. Such work focuses on understanding the length or entropy/information content of messages that must be communicated between distributed parties in order to compute a desired function, such as the set intersection, of their individual inputs. Recently, it was shown that there exist problems that exhibit an exponential gap between the required message length and the required entropy of the communicated messages [Ganor et al., 2014]. This and other curious connections and discrepancies between the communication and information complexities of problems led us to consider the COM and MEM classes defined in terms of both actual bits as well as information theoretic bits of entropy.

Finally, there is a large body of work on memory bounded computation from the theory community [e.g., Reingold, 2008], and work on the applied side on reducing the memory needs [e.g., Langford et al., 2009, Xiao, 2010, Agarwal et al., 2012, Mitliagkas et al., 2013, Arora et al., 2013, Steinhardt et al., 2014] and/or communication needs [e.g., Niu et al., 2011] for various specific learning tasks, though this literature is beyond the scope of this paper. The topic of finite-memory learning has also received attention from the statistics community, with a focus on hypothesis testing [e.g., Cover, 1969, Hellman and Cover, 1970].

2 Definitions

Before making precise definitions, we will state our notational conventions. We use the symbol m for the number of samples/queries, n to denote the length of each sample (for instance, if samples lie in $\{0,1\}^n$), δ to denote the probability of failure, and ε to denote the accuracy of a solution. For communication and memory constraints, b denotes the number of allowed bits; for statistical queries, τ denotes the tolerance of each query.

Learnability. We consider *learning problems* $(\mathcal{X}, \mathcal{F}, \mathcal{D})$, where $\mathcal{F}$ is a *concept class* of functions $f : \mathcal{X} \to \{-1, +1\}$, and $\mathcal{D}$ is a distribution on $\mathcal{X}$. Our goal is to learn $f \in \mathcal{F}$ given some indirect access to it (for instance, samples of the form $(x, f(x))$ with $x \sim \mathcal{D}$). We say that an algorithm (ε, δ)-*learns* $\mathcal{F}$ if, for each $f \in \mathcal{F}$, with probability $1 - \delta$ the algorithm returns an $\hat{f}$ with $\mathbb{P}_{x\sim\mathcal{D}}[f(x) = \hat{f}(x)] \geq 1 - \varepsilon$. We are interested in what problems can be (ε, δ)-learned given various constraints on the algorithm, as described below. Note that any ε-accurate $\hat{f}$ can with high probability be verified to be accurate with only $\widetilde{\mathcal{O}}(1/\varepsilon^2)$ samples; by standard amplification arguments, it therefore suffices to consider $(\varepsilon, 1/3)$-learnability.

Statistical queries. The class SQ is defined as the class of problems that can be $(\varepsilon, 0)$-learned using $\mathrm{poly}(n, \varepsilon^{-1})$ statistical queries of tolerance $1/\mathrm{poly}(n, \varepsilon^{-1})$. By standard derandomization arguments, this is equivalent to the class of problems that can be $(\varepsilon, 1/3)$-learned in the same way (using a number of queries that is larger by at most an extra logarithmic factor in the size of an ϵ-covering of the concept class). For our purposes, a *statistical query of tolerance* τ takes as input a function $\psi : \mathcal{X} \times \{-1, +1\} \to [-1, +1]$, and outputs $\mu = \mathrm{SQ}(\psi, \tau)$ satisfying $|\mu - \mathbb{E}_{x\sim\mathcal{D}}[\psi(x, f(x))]| < \tau$.

Communication bounded learning: COM. We define $\mathrm{COM}(b)$ to be the class of problems that are learnable with b bits of communication per example using a number of examples that is polynomial in the length n of each example. Formally, a learning problem $(\mathcal{X}, \mathcal{F}, \mathcal{D})$ is in $\mathrm{COM}(b)$ if for all $\epsilon, \delta > 0$, the class can be (ε, δ)-learned with $m = \mathrm{poly}(n, \varepsilon^{-1}, \log(\delta^{-1}))$ labeled examples via a 2-phase algorithm of the following form: the first phase is the communication phase, where at the t-th step of the algorithm, a "player" $i \in \{1, \ldots, m\}$ is chosen according to a (possibly randomized function) of the transcript of communications that occurred in the first $j - 1$ steps of the algorithm. The chosen player, i, broadcasts (i, c) where c is a single bit $c = g_{i,j}(x_i, \ell(x_i))$ which is computed via an arbitrary (possibly randomized) function $g_{i,j}$ of the i-th labeled example, where the function $g_{i,j}$ may be adaptively chosen dependent on the entire transcript of communications in the first $j-1$ rounds. We require that each player $i \in \{1, \ldots, m\}$ broadcasts at most b times during the communication phase of the algorithm. After the communication phase of the algorithm, the second phase of the algorithm outputs a hypothesis $\hat{f}$ that is a (possibly randomized) function of the entire communication transcript.

Streaming communication bounded learning: sCOM. We define $\mathrm{sCOM}(b)$ to be the class of problems that are learnable with b bits of communication per example in a streaming model using a number of examples that is polynomial in the length n of each example. Formally, a learning problem $(\mathcal{X}, \mathcal{F}, \mathcal{D})$ is in $\mathrm{sCOM}(b)$ if for all $\epsilon, \delta > 0$, the class can be (ε, δ)-learned with $m = \mathrm{poly}(n, \varepsilon^{-1}, \log(\delta^{-1}))$ labeled examples via a 2-phase algorithm of the following form: the first phase is the communication phase, where at the i-th step of the algorithm, the i-th example $x_i, \ell(x_i)$ is compressed to $z_i = g_i(x_i, \ell(x_i))$ where $z_i \in \{0, 1\}^b$ and $g_i : \mathcal{X} \times \{-1, 1\} \to \{0, 1\}^b$ is a (possibly randomized) function dependent on $z_1, \ldots, z_{i-1}$. After the communication/compression phase of the algorithm, the second phase of the algorithm outputs a hypothesis $\hat{f}$ that is a (possibly randomized) function of $z_1, \ldots, z_m$.

Memory bounded learning: MEM. We define $\mathrm{MEM}(b)$ to be the class of problems that are learnable with b bits of memory using a number of examples that is polynomial in the length of each example. Formally, a learning problem $(\mathcal{X}, \mathcal{F}, \mathcal{D})$ is in $\mathrm{MEM}(b)$ if for all $\epsilon, \delta > 0$, the class

can be (ε, δ)-learned with $m = \text{poly}(n, \varepsilon^{-1}, \log(\delta^{-1}))$ examples via an algorithm that streams over examples while maintaining only b bits of state in memory.

3 Characterizing COM(b)

In this section, we will show that the complexity classes COM and SQ are closely related. Our main high-level result is that a problem is learnable from statistical queries if and only if it is learnable with logarithmic (equivalently, 1-bit) communication.

Theorem 3.1. *For any constant $C > 0$,*

$$\text{COM}(1) = \text{COM}(C \log n) = \text{sCOM}(1) = \text{sCOM}(C \log n) = \text{SQ}.$$

One direction of the above theorem is easy and unsurprising: showing that SQ $\subset$ sCOM(1) $\subset$ COM(1). This follows from the fact that a statistical query $\text{SQ}(\psi, \tau)$ can be simulated by evaluating ψ on approximately $1/\tau^2$ samples and taking the average. Moreover, this still works if we first randomly round $\psi(x, f(x))$ to either $+1$ or -1, which allows us to send the sample with a single bit of communication in the streaming setting. We make this explicit in Lemma 3.2 whose straightforward proof is deferred to Appendix A.

Lemma 3.2. *If a concept class $\mathcal{F}$ is (ε, δ)-learnable with m statistical queries of tolerance τ, then it is $(\varepsilon, 2\delta)$-learnable with $2m \log(2m/\delta)/\tau^2$ samples and 1 bit of communication per sample.*

For the other direction of Theorem 3.1, we leverage the following theorem that provides a general prescription for converting bounded communication algorithms into statistical query algorithms:

Theorem 3.3. *If a concept class can be (ϵ, δ)-learned with m examples and b bits of communication per example, then the concept class can also be $(\epsilon, 2\delta)$-learned by a statistical query algorithm that asks $2bm$ statistical queries of tolerance $\tau = \delta/\left(2^{b+1}m\right)$.*

Note that as long as $b = \mathcal{O}(\log n)$ and $m = \text{poly}(n)$, the above theorem yields a polynomial number of statistical queries of a polynomial tolerance τ, establishing that $\text{COM}(C \log n) \subset SQ$. Beyond showing this polynomial equivalence, the above theorem allows one to translate exponential hardness for SQ learning into exponential lower bounds for learning with bounded communication. To give a concrete example, consider the following bound of Blum et al. [1994] on the learnability of parity functions in the SQ model:[3]

> Let $\mathcal{P}_{n,r}$ be the class of n-bit parity functions that depend on at most r coordinates: that is, $\mathcal{X} = \{0,1\}^n$, and $f_v(x) = (-1)^{\sum_{i=1}^n v_i x_i}$ for some $v \in \{0,1\}^n$ with $\|v\|_0 \leq r$. Also let $s = |\mathcal{P}_{n,r}|$. Then, for $s \geq 16$, no algorithm can $(1/3, 1/3)$-learn $\mathcal{P}_{n,r}$ with $s^{1/3}/2$ statistical queries of tolerance $s^{-1/3}$.

Via Theorem 3.3, this result directly implies the following bound for parity with bounded communication; in particular $\text{PARITY}(n) \notin \text{COM}(\alpha n)$ for any $\alpha < 1/3$.

Corollary 3.4. *Given only b bits of communication per sample, no algorithm can $(1/3, 1/6)$-learn the class $\mathcal{P}_{n,r}$ with less than $|\mathcal{P}_{n,r}|^{1/3}/\left(2^{b+4}\right)$ examples. In particular, no algorithm can $(1/3, 1/6)$-learn $\mathcal{P}_{n,n}$ with less than $2^{n/3-b-4}$ examples, and hence, for example, any algorithm communicating at most $n/4$ bits per example requires an exponential number of examples.*

[3] The theorem as stated in Blum et al. [1994] is for $(1/3, 0)$-learning, but the same proof establishes the impossibility of $(1/3, 1/3)$-learning as well.

As a second corollary to Theorem 3.3, we can prove similar bounds even in the setting in which each "example" consists of a list of $\Theta(n)$ examples drawn uniformly at random from PARITY(n). The proof of the following corollary is given in Appendix A.1. As discussed in the introduction, this corollary may be a bit surprising, since if each party could either see n examples instead of $n/4$ examples, or communicate n bits instead of $n/16$ bits, then the exponential blow-up would disappear.

Corollary 3.5. *Suppose we have a multiround multiparty protocol for learning a parity in which each party is given $\leq n/4$ uniformly distributed length-n labeled examples, and each party can communicate at most $b \leq n/16$ bits. Then, $\Omega(2^{n/16}/n)$ parties are needed in order to learn* PARITY(n) *with probability* 0.9.

3.1 Proof of Theorem 3.3

The high level proof approach is to argue that we can simulate all of the communications just by using statistical queries. We will proceed via induction, and argue that the total variation distance between the distribution of communications in the communication model, and the distribution of *simulated* communications based on statistical queries, is small—bounded by α—and hence if the communication algorithm succeeds with probability $1-\delta$, the statistical query algorithm will be successful with probability at least $1-\delta-\alpha$. The following protocol describes how each step of this simulation proceeds.

Algorithm 1. Simulating Communication Protocol via Statistical Queries

Given the description of a communication protocol in which each of m players receives an example/label pair $(x, \ell(x)))$ drawn from the distribution over examples, the following algorithm describes how to simulate the protocol using statistical queries.

Consider an intermediate step of the communication protocol in which the i-th player is supposed to communicate a bit. Assume that the i-th player has already communicated $j-1$ bits, $c_1, \ldots, c_{j-1}$ in earlier steps of the protocol. Let $f_{i,j}$ denote the (possibly randomized) function that maps $(x, \ell(x))$ to $\{0,1\}$, representing the function that player i uses to compute the j-th bit c_j to communicate. Note that $f_{i,j}$ might have been chosen dependent on the entire transcript of communications up to this point, and let $f_{i,1}, \ldots, f_{i,j-1}$ denote the analogous functions that were used to compute the first $j-1$ bits that player i communicated (which were each dependent on the transcripts of communication up until the corresponding bits were communicated).

For $k \leq j-1$, let $E_k = \bigwedge_{h=1}^{k}[f_{i,h}(x, \ell(x)) = c_h]$ and $p_k = \Pr_x[E_k]$, and assume that we have estimates of $p_1, \ldots, p_{j-1}$, which we denote by $q_1, \ldots, q_{j-1}$, satisfying $|p_k - q_k| \leq 2\tau$.

1. *We ask two statistical queries of tolerance τ:*

$$ {}_0q_j := SQ\Big(\mathbb{I}[E_{j-1} \wedge f_{i,j}(x, \ell(x)) = 0], \tau\Big),$$
$$ {}_1q_j := SQ\Big(\mathbb{I}[E_{j-1} \wedge f_{i,j}(x, \ell(x)) = 1], \tau\Big),$$

   where $\mathbb{I}$ is the indicator function.

2. *Define $t = \frac{{}_0q_j + (q_{j-1} - {}_1q_j)}{2}$ and $s = \max(0, \min(t, q_{j-1}))$.*
3. *Set $c_j = 0$ with probability s/q_{j-1}, and $c_j = 1$ otherwise.*
4. *If $c_j = 0$, then set $q_j = s$, otherwise set $q_j = q_{j-1} - s$.*

We note that in the above reduction, in Step 1 we use two statistical queries—one to estimate ${}_0q_j \approx \Pr[E_{j-1} \wedge f_{i,j}(x, \ell(x)) = 0]$ and one to estimate ${}_1q_j \approx \Pr[E_{j-1} \wedge f_{i,j}(x, \ell(x)) = 1]$. We could have gotten away with a single query, leveraging the fact that ${}_0q_j + {}_1q_j \approx q_{j-1}$, though the errors in the approximation would increase as more bits are communicated, yielding that $|p_k - q_k| \leq k\tau$, rather than the invariant that $|p_k - q_k| \leq 2\tau$. Hence we could have reduced the total number of statistical queries by a factor of 2, at the expense of needing to decrease the tolerance by a factor of b—the total number of bits communicated per player.

The following lemma justifies the calculations in Step 2 of the above algorithm, showing that the above process preserves the invariant that $|p_k - q_k| \leq 2\tau$.

Lemma 3.6. *Assume ${}_0p_j, {}_1p_j, p_{j-1} \in [0,1]$ satisfy ${}_1p_j + {}_0p_j = p_{j-1}$, and let ${}_0q_j, {}_1q_j$, and q_{j-1} satisfy $|p_{j-1} - q_{j-1}| \leq 2\tau$, $|{}_0p_j - {}_0q_j| \leq \tau$, and $|{}_1p_j - {}_1q_j| \leq \tau$. Then if we define*

$$t = \frac{{}_0q_j + (q_{j-1} - {}_1q_j)}{2} \text{ and } s = \max\left(0, \min\left(s, q_{j-1}\right)\right),$$

the following two inequalities hold:

$$|{}_0p_j - s| \leq 2\tau \text{ and } |{}_1p_j - (q_{j-1} - s)| \leq 2\tau.$$

Proof. First note that since $0 \leq {}_0p_j, {}_1p_j \leq p_{j-1}$, and $|p_{j-1} - q_{j-1}| \leq 2\tau$, it suffices to show that $|{}_0p_j - t| \leq 2\tau$ and $|{}_1p_j - (q_{j-1} - t)| \leq 2\tau$, as the restriction of t to lie in the range $[0, q_{j-1}]$ can never cause these equations to go from being true to being false. For the first inequality, note that $q_{j-1} - {}_1q_j \in [{}_1p_j - 3\tau, {}_0p_j - 3\tau]$, and hence ${}_0q_j + (q_{j-1} - {}_1q_j) \in [2{}_0p_j - 4\tau, 2{}_0p_j + 4\tau]$, from which the first inequality follows. For the second inequality, first note that $({}_0q_j - {}_1q_j) - ({}_0p_j - {}_1p_j)| \leq 2\tau$, and hence $({}_0q_j - {}_1q_j) - (p_{j-1} - 2{}_1p_j)| \leq 2\tau$. Plugging this in, we have the following:

$$\begin{aligned}
\left|{}_1p_j - \left(q_{j-1} - \frac{{}_0q_j + (q_{j-1} - {}_1q_j)}{2}\right)\right| &= \left|{}_1p_j + \frac{-q_{j-1} + ({}_0q_j - {}_1q_j)}{2}\right| \\
&\leq \left|{}_1p_j + \frac{-q_{j-1} + p_{j-1} - 2{}_1p_j}{2}\right| + \tau, \\
&\leq \left|\frac{p_{j-1} - q_{j-1}}{2}\right| + \tau \\
&\leq 2\tau.
\end{aligned}$$

□

We are now equipped to prove the validity of the simulation algorithm, establishing Theorem 3.3.

Proof of Theorem 3.3. Let c_t denote the bit of communication communicated in round t, and let i_t and j_t denote the corresponding party and index of the bit. For shorthand, define the tuple $z_t = (i_t, j_t, c_t)$, and let $m' = bm$ denote the total number of rounds of communication. Then, we can bound the total variational distance as

$$\begin{aligned}
\|p - q\|_{TV} &= \frac{1}{2} \sum_{z_{1:m'}} |p(z_{1:m'}) - q(z_{1:m'})| \\
&\leq \frac{1}{2} \sum_{t=1}^{m'} \mathbb{E}_{z_{1:t-1}} \left[|p(i_t, j_t \mid z_{1:t-1}) - q(i_t, j_t \mid z_{1:t-1})|\right] + \mathbb{E}_{z_{1:t-1}, i_t, j_t} \left[|p(c_t \mid i_t, j_t, c_{1:t-1}) - q(c_t \mid i_t, j_1, c_{1:t-1})|\right] \\
&= \frac{1}{2} \sum_{t=1}^{m'} \mathbb{E}_{z_{1:t-1}, i_t, j_t} \left[|p(c_t \mid c_{1:t-1}, i_t, j_t) - q(c_t \mid c_{1:t-1}, i_t, j_t)|\right],
\end{aligned}$$

where the final equality is because i_t is selected using the same protocol for both p and q, and j_t is a deterministic function of $(z_{1:t-1}, i_t)$.

Consequently, for a given party i, the contribution from the simulation of bit c_j to the total variation distance is bounded by $\frac{1}{2}\mathbb{E}\left[\left|\frac{p_j}{p_{j-1}} - \frac{q_j}{q_{j-1}}\right|\right]$. We now analyze this quantity.

First observe that

$$\mathbb{E}\left[\frac{1}{q_j} \mid q_{j-1}\right] = u\frac{1}{uq_{j-1}} + (1-u)\frac{1}{(1-u)q_{j-1}} = \frac{2}{q_{j-1}},$$

where $u = s/q_{j-1}$ is the probability of the coin used to decide c_j as specified in Step 3 of the algorithm. Hence $\frac{1}{2q_1}, \frac{1}{2^2 q_2}, \ldots, \frac{1}{2^j q_j}$ is a martingale, from which it follows that $\mathbb{E}[\frac{1}{q_j}] = 2^j$, where the expectation is taken over the randomness of the entire simulation.

From Lemma 3.6, $|q_k - p_k| \leq 2\tau$, hence we have the following inequality:

$$\begin{aligned}\left|\frac{p_j}{p_{j-1}} - \frac{q_j}{q_{j-1}}\right| &= \frac{|p_j q_{j-1} - q_j p_{j-1}|}{p_{j-1}q_{j-1}} \\ &\leq \frac{p_j|q_{j-1} - p_{j-1}| + p_{j-1}|p_j - q_j|}{p_{j-1}q_{j-1}} \\ &\leq \frac{2\tau(p_j + p_{j-1})}{p_{j-1}q_{j-1}} \leq \frac{4\tau p_{j-1}}{p_{j-1}q_{j-1}} \leq \frac{4\tau}{q_{j-1}}.\end{aligned}$$

Consequently, $\frac{1}{2}\mathbb{E}\left[\left|\frac{p_j}{p_{j-1}} - \frac{q_j}{q_{j-1}}\right|\right] \leq 2^j\tau$, and so the overall contribution of the ith party to the total variational distance is at most $\sum_{j=1}^{b} 2^j\tau < 2^{b+1}\tau$. Summing over all parties yields a bound of $\|p-q\|_{TV} < m2^{b+1}\tau$. Hence, as long as $\tau \leq \frac{\delta}{m2^{b+1}}$, the total variational distance is bounded by δ, which means that the probability of failure can increase by at most δ, and so is bounded by 2δ, as desired.

□

3.2 Direct Bounds on Communication in the Streaming Setting

In the previous section, we showed that learnability with bounded communication is closely tied to statistical query learnability, and used this to prove lower bounds on the amount of communication needed to learn PARITY in polynomially many samples. In this section, we complement this result with a more direct communication lower bound based on Assouad's method [Assouad, 1983] that applies to the streaming model of bounded communication. As illustrated in the case of learning parity with noise, this approach can yield essentially sharp lower bounds on the hardness of communication-constrained learning in the streaming settings. It remains an intriguing open question to extend these techniques from the streaming setting sCOM to the more general bounded communication setting COM.

Suppose that we are trying to learn some unknown $x \in \mathcal{X}$ by observing a stream of i.i.d. samples from a distribution $p_x(y)$. Intuitively, given a sample y, a communication-bounded algorithm would like to approximate the likelihood function $l_y(x) = p_x(y)$ with as few bits as possible. If there is a cluster $S \subseteq \mathcal{Y}$ across which the values of l_y are highly correlated, then we can save communication by transmitting the cluster rather than the particular value of y. If no such clusters exist, communication-bounded learning should be difficult. Theorem 3.7 below formalizes this, by measuring the covariance M between the probabilities of different points in $\mathcal{Y}$.

Theorem 3.7. *Consider any problem where $x \in \mathcal{X}$ is drawn according to a measure $\mu(x)$, and $y_1, \ldots, y_m \in \mathcal{Y}$ are drawn i.i.d. (conditioned on x) with probability $p_x(y)$. Let $p_0(y)$ be any distribution on $\mathcal{Y}$ and define the matrix $M \in \mathbb{R}^{\mathcal{Y}\times\mathcal{Y}}$ by*

$$M_{y,y'} = \sum_{x\in\mathcal{X}} \mu(x)\left(p_x(y) - p_0(y)\right)\left(p_x(y') - p_0(y')\right); \tag{1}$$

moreover, write $\lambda = |\mathcal{Y}|\cdot\lambda_{\max}(M)$ for the maximum eigenvalue of $|\mathcal{Y}|\cdot M$ and assume that $\mu(x) \leq 1/2$ for all x. Then any procedure using b bits of communication requires at least $\sqrt{2}^{\log_2(1/\lambda)-b-8}$ samples y_i to identify x with probability greater than $3/4$.

If we take $p_0(y) = p_\mu(y) := \sum_x p_x(y)\mu(x)$, then we can interpret $M_{y,\,y'}$ as the covariance between $p_x(y)$ and $p_x(y')$, where y, y' are treated as fixed and x is random. Note that $\log_2(1/\lambda)$ identifies a threshold in communication below which sample complexity increases exponentially. Assuming that $p_x(y) \approx 1/|\mathcal{Y}|$, we will have $\mathrm{tr}(M) \approx |\mathcal{Y}|^{-1}$, and hence $\lambda \in [|\mathcal{Y}|^{-1}, 1]$. Therefore, $\log_2(1/\lambda)$ will typically be between 0 and $\log_2 |\mathcal{Y}|$.

The two key steps to proving our result are given as lemmas below. The first is the standard lemma on which Assouad's method is based, while the second is a new bound that lets us leverage the bounded communication assumption. The idea behind the latter result is that, letting z be a message in one of the rounds, the KL divergence between $p_0(z)$ and $p_x(z)$ can be bounded by the χ^2-divergence $\sum_z |p_0(z) - p_x(z)|^2/p_x(z)$ (Tsybakov [2009], Lemma 2.7). Since $z \in \{0,1\}^b$, the denominator $1/p_x(z)$ is with high probability not much larger than 2^b, which yields a bound that leverages the communication constraint b. Proofs are provided in Appendix B.

Lemma 3.8 (Assouad [1983]; Arias-Castro et al. [2013])**.** *Let $Z^{(1:m)}$ be the messages sent by a communication-bounded streaming algorithm, and suppose that*

$$\sum_{i=1}^{m} \sum_{x \in \mathcal{X}} \mu(x)\, \mathbb{E}_{z^{(1:i-1)} \sim p_0} \left[\mathrm{KL}\left(p_0(Z^{(i)} \mid z^{(1:i-1)}) \parallel p_x(Z^{(i)} \mid z^{(1:i-1)}) \right) \right] \le K. \tag{2}$$

Then, the probability of recovering x is at most $\mu_{\max} + \sqrt{K/2}$, where $\mu_{\max} = \max_x \mu(x)$.

Lemma 3.9. *Let the message $Z \in \{0,1\}^b$ depend on Y and $\hat{Z}$ and suppose that $p_x(y) \ge \epsilon/|\mathcal{Y}|$ for all x, y. Let M be as defined in (1). Then, for any $\hat{z}$, we have*

$$\sum_{x \in \mathcal{X}} \mu(x)\, \mathrm{KL}\left(p_0(Z \mid \hat{z}) \parallel p_x(Z \mid \hat{z}) \right) \le 2^b \lambda/\epsilon, \tag{3}$$

where $\lambda = |\mathcal{Y}| \cdot \lambda_{\max}(M)$ is the measure of problem complexity in the statement of Theorem 3.7.

Given these two lemmas, the proof of Theorem 3.7 proceeds as follows. For any given distribution $p_x(y)$, we modify both it and $p_0(y)$ by averaging in a multiple of $1/(8m)$ of the uniform distribution on $\mathcal{Y}$; this guarantees that $p_x(y) \ge 1/(8m|\mathcal{Y}|)$ uniformly, and with probability $7/8$ this difference will not manifest even once over m samples. Also, M becomes scaled down by a factor of $(1 - 1/(8m))^2$, so λ can only decrease due to this modification. We can then apply Lemma 3.9 to conclude that

$$\sum_{i=1}^{m} \sum_{x \in \mathcal{X}} \mu(x) \mathbb{E}_{z^{(1:i-1)} \sim p_0} \left[\mathrm{KL}\left(p_0(Z^{(i)} \mid z^{(1:i-1)}) \parallel p_x(Z^{(i)} \mid z^{(1:i-1)}) \right) \right] \le m^2\, 2^{b+3} \lambda, \tag{4}$$

so that the probability of recovering x is at most $1/2 + 1/8 + \sqrt{m^2\, 2^{b+2} \lambda}$. As long as $m^2\, 2^{b+2} \lambda \le 1/64$, the probability of success is at most $3/4$; this corresponds to $m \le 1/\sqrt{\lambda 2^{b+8}}$, as claimed.

Applying the bound. We can use the lower bound in Theorem 3.7 to provide a short proof of the communication lower bounds for PARITY in the streaming setting. For convenience, we consider the equivalent version of PARITY where instead of getting samples labeled by the parity, we see samples drawn uniformly at random from the positive class. In the language of Theorem 3.7, we have $\mathcal{X} = \mathcal{Y} = \{0,1\}^n$ and $p_x(y) = \left(1 + (-1)^{\langle x,y \rangle}\right)/2^n$. If we let $p_0(y) = 2^{-n}$, then we have

$$M_{y,y'} = \frac{1}{4^n} \sum_{x \in \mathcal{X}} \mu(x)(-1)^{\langle x,y \rangle}(-1)^{\langle x,y' \rangle} = \frac{1}{4^n} \sum_{x \in \mathcal{X}} \mu(x)(-1)^{\langle x,y+y' \rangle}. \tag{5}$$

From this, we see that $M_{y,y'}$ is a circulant matrix over $\{0,1\}^n$, and hence is diagonalized by the Fourier transform. More concretely, if we let $\chi_s(y) = (-1)^{\langle s,y \rangle}$, then one can easily check (Lemma B.1) that χ_s is an eigenvector of M with eigenvalue $\mu(s)/2^n$. Since $|\mathcal{Y}| = 2^n$, we have $\lambda = \max_s \mu(s) = \mu_{\max}$. Hence we need at least $\sqrt{2}^{\log_2(1/\mu_{\max}) - b - 8}$ samples to learn PARITY, for any distribution $\mu(x)$ with $\mu(x) \le \mu_{\max}$.

As an immediate consequence, if $\mu(x)$ is uniform over $\{0,1\}^n$, then we need $2^{(n-b)/2-4}$ samples to solve PARITY on n bits. This result is substantially stronger than Corollary 3.4. In particular, it implies that $\text{PARITY}(n) \notin \text{sCOM}(\alpha n)$ for any $\alpha < 1$ (instead of $\alpha < 1/3$). Note that we can trivially verify that $\text{PARITY}(n) \in \text{sCOM}(n)$; thus, Corollary 3.10 is sharp to within $o(n)$ for the amount of communication required to efficiently learn a parity.

Corollary 3.10. *Any algorithm that learns* PARITY *on n bits with probability of success at least $3/4$, using only b bits of communication per example in the streaming setting, must use at least $2^{(n-b)/2-5}$ examples.*

In the case of learning a noisy parity, where samples from the negative class occur with some probability $0 < \eta < 0.5$, we obtain even tighter bounds. This is because we no longer need to add in artificial noise in the proof of Theorem 3.7 and instead can directly bound the left-hand side of (4) by $m\,2^b\,\lambda\eta = m\,2^{b-n}\eta$, thus yielding a lower bound of $\Omega\left(2^{n-b}\right)$.

Holding η fixed, a naïve algorithm that guesses the first $n-b$ bits of the answer can learn noisy parity with $\mathcal{O}\left(n2^{n-b}\right)$ samples (since it takes $\mathcal{O}(b)$ samples to compute the answer conditioned on a given guess, followed by $\mathcal{O}(n-b)$ samples to verify the guess with sufficiently high probability). We can therefore characterize the number of samples required to learn a noisy parity to within a factor n. Moreover, this result implies that the noisy parity problem is not in the class $\text{sCOM}(n - \omega(\log(n)))$.

Corollary 3.11. *For a fixed noise level, learning a noisy parity on n bits with b bits of communication and success probability $\frac{7}{8}$ in the streaming setting requires at least $\Omega\left(2^{n-b}\right)$ samples and at most $\mathcal{O}\left(n\,2^{n-b}\right)$ samples.*

3.3 The Limits of Information Theory

In the above sections, we established exponential thresholds for communication-bounded procedures, showing that below a certain threshold the sample complexity doubles for every 1-bit reduction in the allowed communication. This lower bound is very different in kind than much of the prior work on resource-constrained learning [e.g., Bar-Yossef et al., 2004, Zhang et al., 2013, Garg et al., 2014, Shamir, 2014], which reduce resource constraints to information constraints in order to build on well-developed mathematical tools from information theory. Instead of finding exponential thresholds, prior work has only established *linear thresholds*, where the sample complexity doubles every time the allowed communication is halved. We will show that, in a strong sense, exponential thresholds can never be obtained from a reduction to information-theoretic constraints: the linear thresholds exhibited thus far are the best possible with a purely information-theoretic approach.

The idea is the following: given any algorithm that communicates using b bits of entropy per example, we can run the same algorithm while only sending a message once every k steps, e.g., by only looking at the examples whose index is $i \bmod k$, with $i \sim \text{Uniform}(\{0,\dots,k-1\})$. This requires k times as many examples total, but the amount of communication per example is only about b/k in expectation. Therefore, we can always increase the number of samples by a factor of k while decreasing the entropy by the same factor. We prove the following more precise result in Appendix C, as well as an analogous result for memory-constrained problems.

Proposition 3.12. *Any communication-constrained problem that can be learned with m samples and $H(z_i) \leq b$ can be learned with mk samples and $H(z_i) \leq b/k + \log_2(k \cdot e)/k$.*

In addition to justifying the statistical query techniques presented above, Proposition 3.12 also validates our definitions: if we had instead, for instance, defined the class $\text{COM}_H(b)$ of problems learnable with polynomially many examples and b bits of entropy communicated per example, then we would already have $\text{COM}_H(1) \supseteq \text{PAC}$. The class COM_H is therefore completely trivial, while the previous sections show that COM has rich structure.

4 Reducing SQ to MEM

Many classical algorithms use $\text{poly}(n)$ memory to solve n-dimensional problems; e.g., Gaussian elimination uses on the order of n^2 memory to solve an n-dimensional linear system; this is infeasible in modern applications, where n could easily be 10^9 or larger. In such cases, streaming algorithms such as stochastic gradient

have generated enormous practical benefit by solving regression problems using nearly-linear memory in exchange for modest increases in sample complexity.

The success of such algorithms leads us to ask the more general question: which problems can be solved in such a memory-efficient way? Theorem 4.1 below provides a positive result for problems that are learnable from statistical queries: all such problems may also be learned with nearly-linear memory memory, using only a polynomial number of samples. We note, however, that our reduction procedure is not necessarily computationally efficient. Proofs for this section may be found in Appendix D.

Theorem 4.1. *Suppose that the distribution $\mathcal{D}$ over $\mathcal{X}$ is known. If a class $\mathcal{F}$ is $(\varepsilon, 0)$-learnable with m_0 statistical queries of tolerance τ, then it is (ε, δ)-learnable with at most*

$$m = \mathcal{O}\left(\frac{m_0 \log|\mathcal{F}|}{\tau^2}\left(\log\log|\mathcal{F}| + \log(m_0) + \log(1/\delta)\right)\right) \text{ samples and} \tag{6}$$

$$b = \mathcal{O}\left(\log|\mathcal{F}|\left(\log(m_0) + \log\log(1/\tau)\right) + \log(1/\tau)\right) \text{ bits of memory.} \tag{7}$$

Let REP be the class of *efficiently representable* problems (i.e., with $\log|\mathcal{F}| = \widetilde{\mathcal{O}}(n)$). Note that $\mathrm{MEM}(\widetilde{\mathcal{O}}(n)) \subseteq \mathrm{REP}$. Theorem 4.1 implies a partial converse:

Corollary 4.2. *If a class $\mathcal{F}$ is learnable with $\mathrm{poly}(n)$ statistical queries of tolerance $1/\mathrm{poly}(n)$, then it is learnable with at most $\mathrm{poly}(n, \log|\mathcal{F}|)$ samples and $b = \mathcal{O}(\log|\mathcal{F}|\log(n))$ bits of memory. In particular, $\mathrm{SQ} \cap \mathrm{REP} \subseteq \mathrm{MEM}(\widetilde{\mathcal{O}}(n))$.*

We remark here that the naïve approach of simply remembering the results of the statistical queries will not work, since the number m of statistical queries could be much larger than $\log|\mathcal{F}|$. Instead, we show that it is always possible to identify a subset of $\mathcal{O}(\log|\mathcal{F}|)$ "important" queries, such that remembering the results of only these queries suffices to recover the answer.

More specifically, the proof of Theorem 4.1 relies on a reduction of statistical queries to a canonical form. First, we replace statistical queries with *statistical threshold queries*, which return a binary yes-no answer about whether a statistical query lies above or below a given threshold. We also apply a normalization procedure to ensure that at least one of the two answers will narrow down the space of concepts by a factor of 3/4 or better. Then, we simply record the first point at which we receive an "important" answer that narrows the space by at least this factor: by construction, all of the previous answers are then uniquely determined, so that this is enough to recover the full sequence of queries and responses up to that point. Iterating this process $\mathcal{O}(\log|\mathcal{F}|)$ times leaves us with a unique concept.

To make the canonicalization process more precise, recall that a statistical query oracle SQ takes a statistic $\psi : \mathcal{X} \times \{-1, +1\} \to [-1, +1]$ and a tolerance τ, and outputs a value $\mu = \mathrm{SQ}(\psi, \tau)$ satisfying $|\mu - \mathbb{E}_{x\sim\mathcal{D}}[\psi(x, f(x))]| < \tau$. We define a related *statistical threshold query* oracle, which takes a triple (ψ, μ, τ) and outputs a response $r = \mathrm{STQ}(\psi, \mu, \tau) \in \{0, 1\}$, such that $r = 1$ if $\mathbb{E}_{x\sim\mathcal{D}}[\psi(x, f(x))] \geq \mu + \tau$, $r = 0$ if $\mathbb{E}_{x\sim\mathcal{D}}[\psi(x, f(x))] \leq \mu - \tau$, and r may be arbitrary otherwise. Then, for any statistical threshold query $q = (\psi, \mu, \tau)$, define $\mathcal{F}^1(q)$ and $\mathcal{F}^0(q)$ to be the subsets of $\mathcal{F}$ consistent with a response of 1 and 0 to the query, respectively:

$$\mathcal{F}^1(q) \stackrel{\text{def}}{=} \{f \in \mathcal{F} : \mathbb{E}_{x\sim\mathcal{D}}[\psi(x, f(x))] > \mu - \tau\}, \tag{8}$$

$$\mathcal{F}^0(q) \stackrel{\text{def}}{=} \{f \in \mathcal{F} : \mathbb{E}_{x\sim\mathcal{D}}[\psi(x, f(x))] < \mu + \tau\}. \tag{9}$$

Note that $\mathcal{F}^1(q) \cup \mathcal{F}^0(q) = \mathcal{F}$. We say that a statistical threshold query $q = (\psi, \mu, \tau)$ is *valid* if

$$\min\left\{\left|\mathcal{F}^1(q)\right|, \left|\mathcal{F}^0(q)\right|\right\} \leq \frac{3}{4}|\mathcal{F}|. \tag{10}$$

As shown in the following result, any statistical query can be replaced with a small number of valid statistical threshold queries.

Lemma 4.3. *Any statistical query with tolerance $\tau \leq 2$ can be implemented with $\left\lceil \log\left(\frac{2}{\tau}\right)\right\rceil$ statistical threshold queries with tolerance $\tau/2$. Moreover, any invalid statistical threshold query with tolerance $\tau/2$ can be simulated with a valid statistical threshold query with tolerance $\tau/4$.*

Now, for a query q, call a response r *good* if $|\mathcal{F}^r(q)| \leq \frac{3}{4}|\mathcal{F}|$, and *bad* otherwise. If we get a good response to q, then we can simply remember this response and recursively solve the problem for the class $\mathcal{F}^r(q)$, which is at least 25% smaller than before. If we have only received bad responses so far, then the sequence of responses and queries is uniquely determined (since there is at most one bad response to each query and the algorithm is deterministic) and we can remember this with 1 bit of memory. This yields:

Lemma 4.4. *If a problem is $(\epsilon, 0)$-learnable with m statistical threshold queries with tolerance $\tau/2$, then it can be $(\epsilon, 0)$-learned with $\mathcal{O}(m \log |\mathcal{F}|)$ statistical threshold queries with tolerance at least $\tau/4$ and $\mathcal{O}(\log |\mathcal{F}| \log(m))$ bits of memory.*

Theorem 4.1 follows by using the fact that all of the statistical threshold queries in Lemma 4.3 can be obtained from a single statistical query of tolerance $\tau/4$, and that m such statistical queries can be simulated with probability $1 - \delta$ using $\mathcal{O}\left(m \log(m/\delta)/\tau^2\right)$ samples.

Separating SQ from MEM($\widetilde{\mathcal{O}}(n)$). Learning from statistical queries is in fact strictly harder than learning with memory constraints. To show this, let $\mathcal{F}$ be the class of parity functions that depend only on the first $\sqrt{n}$ bits. This class belongs to MEM($\widetilde{\mathcal{O}}(n)$), since we can recover the true concept by performing Gaussian elimination over $\mathbb{F}_2$ on the first $\mathcal{O}(\sqrt{n})$ samples, which requires $\mathcal{O}((\sqrt{n})^2) = \mathcal{O}(n)$ bits of memory. On the other hand, this class is not in SQ [Blum et al., 1994]. Therefore, we have MEM($\widetilde{\mathcal{O}}(n)$) $\supsetneq$ SQ $\cap$ REP.

4.1 Application: Sparse Linear Regression

Sparse linear regression is one of the most widely studied problems in machine learning (see, e.g., Hastie et al. [2015]). A simple form of the linear regression problem is as follows: we observe independent and identically distributed pairs $(x, y) \in [-1, 1]^n \times [-R, R]$ where the x are drawn from a known distribution $x \sim \mathcal{D}$, $y = w^* \cdot x + v$, and v is drawn from a known noise distribution $\mathcal{D}_v$ that is uncorrelated with x. We then seek to recover w^* via the relation

$$w^* = \arg\min_w \left\{ \mathbb{E}\left[\|y - w \cdot x\|_2^2 \right] : \|w\|_1 \leq R \right\}. \tag{11}$$

We say that this problem is a *k-sparse* regression problem if, moreover, $\|w^*\|_0 \leq k$ for some $k \ll n$; in this case, we also set $R = k$. It is well known that the regression problem (11) can be efficiently solved with $\mathcal{O}(k \log n)$ samples using, for instance, the exponentiated gradient (EG) algorithm of Kivinen and Warmuth [1997]. However, the EG algorithm uses $\Omega(n)$ bits of memory, whereas the answer w^* can be stored with only $\mathcal{O}(k \log n)$ bits of memory. This discrepancy leads us to a natural question: is it possible to efficiently solve k-sparse linear regression as defined above using poly(n) samples and only poly(k, log(n)) memory?

Using the relationship between SQ and MEM, we answer this question in the affirmative—at least from the point of view of sample complexity. To our knowledge, this is the first result on memory-efficient linear regression that does not require structural assumptions on the covariance matrix. The result below extends directly to other sparse convex optimization problems such as logistic regression. All proofs in this section may be found in Appendix E.

Theorem 4.5. *Suppose that the linear regression problem (11) has k-sparse true parameters w^*, known covariate distribution $\mathcal{D}$, and known noise distribution $\mathcal{D}_v$. Then, for any $\varepsilon, \delta > 0$, there exists an algorithm that (ε, δ)-solves (11); that is, with probability at least $1 - \delta$ the algorithm returns a predictor $\hat{w}$ satisfying*

$$\mathbb{E}\left[\|y - \hat{w} \cdot x\|_2^2 \right] \leq \mathbb{E}\left[\|y - w^* \cdot x\|_2^2 \right] + \varepsilon, \text{ while requiring only} \tag{12}$$

$$\mathcal{O}\left(k \log^2\left(\frac{n}{\varepsilon}\right) \right) \text{ bits of memory and } \widetilde{\mathcal{O}}\left(\frac{nk^8 \log \delta^{-1}}{\varepsilon^4} \right) \text{ samples.} \tag{13}$$

As a baseline to Theorem 4.5, we note that a simple "guess-and-check" algorithm that can store s coordinates at a time would need at least $\binom{n}{s}$ samples to solve sparse linear regression. The only other result we are aware of that enables memory-efficient sparse linear regression is the algorithm of Steinhardt and Duchi [2015], which runs with a polynomial number of samples but only works in the special case where the

features x are known to have identity covariance. Thus, it appears that our abstract reduction from SQ to MEM has enabled us to state results that may have been difficult to establish without this machinery.

To prove Theorem 4.5, it suffices to provide an efficient SQ learning algorithm for linear regression and then apply Theorem 4.1. To this end, we note the following standard result on performing gradient descent with small errors on the gradients:

Lemma 4.6. *Let f be a convex function. Suppose that for any point w with $\|w\|_1 \leq R$ ($R \geq 1$), one can obtain an approximate gradient z satisfying $\|z\|_\infty \leq B$ and $\|z - \nabla f(w)\|_\infty \leq \tau B$. Then, after m approximate gradient computations, exponentiated gradient obtains an estimate $\hat{w}$ satisfying*

$$f(\hat{w}) - \min_{\|w\|_1 \leq R} f(w) \leq \mathcal{O}\left(\sqrt{\frac{RB^2 \log(n)}{m}} + \tau RB\right). \tag{14}$$

The proof of Theorem 4.5 now follows by taking $f(w) = \mathbb{E}_{x \sim \mathcal{D}}[(w \cdot x - y)^2]$; the gradient $\nabla f(w)$ is equal to $\mathbb{E}_{x \sim \mathcal{D}}[2x(w \cdot x - y)]$, which has coordinates bounded by $B = 4R$. Then, $\nabla f(w)/B$ is a mean of random variables over $[-1, 1]^n$ and so can be estimated with n statistical queries; in particular, queries with tolerance τ lead to an accuracy of τB for the estimate z of $\nabla f(w)$. Thus, by Lemma 4.6, we obtain a vector $\hat{w}$ satisfying (12) in

$$M = \mathcal{O}\left(n\varepsilon^{-2} \log(n) R^3\right) \text{ statistical queries of tolerance } \tau = \mathcal{O}\left(\varepsilon / R^2\right), \tag{15}$$

whether or not w^* is k-sparse. Now, if in addition w^* is known to be k-sparse, the answer w^* can be represented (to accuracy ε) with $\log|\mathcal{F}| = \mathcal{O}(k \log(n/\varepsilon))$ bits of memory. We immediately conclude from Theorem 4.1 that (11) can be (ε, δ)-solved with[4]

$$b = \mathcal{O}\left(k \log^2\left(\frac{n}{\varepsilon}\right) + \log\left(\frac{R^2}{\varepsilon}\right)\right) \text{ bits of memory and } M = \widetilde{\mathcal{O}}\left(\frac{nR^7 k}{\varepsilon^4} \log\left(\frac{1}{\delta}\right)\right) \text{ samples.} \tag{16}$$

The statement of Theorem 4.5 follows by setting $R = k$. As a follow-up to Theorem 4.5 we might be tempted ask whether it is possible to do better yet: can we efficiently solve k-sparse linear regression using $\text{poly}(k, \log(n))$ samples and $\text{poly}(k, \log(n))$ memory? The answer to this question, however, is in general no: as shown by Steinhardt and Duchi [2015], n-dimensional regression with b bits of memory requires at least $\Omega(nk/(b\varepsilon))$ samples to attain an error rate ε for k-sparse regression.

If, however, we also know that x is r-sparse for some $r \ll n$, we can eliminate the linear dependence on n completely by using the count-min sketch algorithm of Cormode and Muthukrishnan [2005] to reduce the number of statistical queries required for sparse regression in (15). To our knowledge, this is the first bound for sparse linear regression that requires neither memory nor sample resources that scale linearly in n.

Theorem 4.7. *Under the conditions of Theorem 4.5, suppose in addition that $\|x\|_1 \leq r$ for all x. Then, there exists an algorithm that (ε, δ)-solves (11) with*

$$\mathcal{O}\left(k \log^2(n/\varepsilon)\right) \text{ bits of memory and } \widetilde{\mathcal{O}}\left(r^3 k^{10} \log^2(1/\delta)/\varepsilon^5\right) \text{ samples.} \tag{17}$$

References

A. Agarwal, S. Negahban, and M.J. Wainwright. Stochastic optimization and sparse statistical recovery: Optimal algorithms for high dimensions. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1538–1546, 2012.

E. Arias-Castro, E.J. Candes, and M.A. Davenport. On the fundamental limits of adaptive sensing. *IEEE Transactions on Information Theory*, 59(1):472–481, 2013.

R. Arora, A. Cotter, and N. Srebro. Stochastic optimization of PCA with capped MSG. In *Advances in Neural Information Processing Systems (NIPS)*, 2013.

[4] Theorem 4.1 does not directly apply, since the notion of error ε is different here than before. However, the proof of Theorem 4.1 does not rely on any important properties of the error metric and so still applies in this case.

P. Assouad. Deux remarques sur l'estimation. *Comptes rendus des séances de l'Académie des sciences. Série 1, Mathématique*, 296(23):1021–1024, 1983.

M.F. Balcan and V. Feldman. Statistical active learning algorithms for noise tolerance and differential privacy. In *Advances in Neural Information Processing Systems (NIPS)*, 2013.

M.F. Balcan, A. Blum, S. Fine, and Y. Mansour. Distributed learning, communication complexity and privacy. In *Conference on Learning Theory (COLT)*, 2012.

Z. Bar-Yossef, T.S. Jayram, R. Kumar, and D. Sivakumar. An information statistics approach to data stream and communication complexity. *Journal of Computer and System Science*, 68:702–732, 2004.

S. Ben-David and E. Dichterman. Learning with restricted focus of attention. *Journal of Computer and System Sciences*, 56:277–298, 1998.

A. Blum, M. Furst, J. Jackson, M. Kearns, Y. Mansour, and S. Rudich. Weakly learning DNF and characterizing statistical query learning using Fourier analysis. In *Proceedings of the ACM Symposium on Theory of Computing (STOC)*, pages 253–262. ACM, 1994.

M. Braverman and A. Rao. Information equals amortized communication. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, 2011.

A. Chakrabarti, Y. Shi, A. Wirth, and A. Yao. Informational complexity and the direct sum problem for simultaneous message complexity. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, 2001.

G. Cormode and S Muthukrishnan. An improved data stream summary: The count-min sketch and its applications. *Journal of Algorithms*, 55(1):58–75, 2005.

T.M. Cover. Hypothesis testing with finite statistics. *The Annals of Mathematical Statistics*, pages 828–835, 1969.

J. Duchi, M. Jordan, and M. Wainwright. Local privacy and statistical minimax rates. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, 2013.

V. Feldman. A complete characterization of statistical query learning with applications to evolvability. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, 2009.

A. Ganor, G. Kol, and R. Raz. Exponential separation of information and communication. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, 2014.

A. Garg, T. Ma, and H. Nguyen. On communication cost of distributed statistical estimation and dimensionality. In *Advances in Neural Information Processing Systems (NIPS)*, 2014.

A. Gilbert and P. Indyk. Sparse recovery using sparse matrices. *Proceedings of the IEEE*, 98(6):937–947, 2010.

A. Gupta, M. Hardt, A. Roth, and J. Ullman. Privately releasing conjunctions and the statistical query barrier. In *Proceedings of the ACM Symposium on Theory of Computing (STOC)*, 2011.

T. Hastie, R. Tibshirani, and M. Wainwright. *Statistical Learning with Sparsity: The Lasso and Generalizations*. CRC Press, 2015.

M.E. Hellman and T.M. Cover. Learning with finite memory. *The Annals of Mathematical Statistics*, pages 765–782, 1970.

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

J. Kivinen and M.K. Warmuth. Exponentiated gradient versus gradient descent for linear predictors. *Information and Computation*, 132(1):1–63, 1997.

E. Kushilevitz and N. Nisan. *Communication Complexity*. Cambridge University Press, New York, NY, USA, 1997.

J. Langford, L. Li, and T. Zhang. Sparse online learning via truncated gradient. *Journal of Machine Learning Research*, 10:777–801, 2009.

T. Lee and A. Shraibman. Lower bounds in communication complexity. *Foundations and Trends in Theoretical Computer Science*, 3(4):263398, 2009.

T. Ma H.L. Nguyen D.P. Woodruff M. Braverman, A. Garg. Communication lower bounds for statistical estimation problems via a distributed data processing inequality. *arXiv preprint arXiv:1506.07216*, 2015.

I. Mitliagkas, C. Caramanis, and P. Jain. Memory limited, streaming PCA. In *Advances in Neural Information Processing Systems (NIPS)*, 2013.

F. Niu, B. Recht, C. Re, and S. Wright. Hogwild: A lock-free approach to parallelizing stochastic gradient descent. In *Advances in Neural Information Processing Systems (NIPS)*, 2011.

O. Reingold. Undirected connectivity in log-space. *Journal of the ACM*, 55(4), 2008.

S. Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 4(2):107–194, 2011.

O. Shamir. Fundamental limits of online and distributed algorithms for statistical learning and estimation. In *Advances in Neural Information Processing Systems (NIPS)*, 2014.

J. Steinhardt and J. Duchi. Minimax rates for memory-bounded sparse linear regression. In *Conference on Learning Theory*, 2015.

J. Steinhardt, S. Wager, and P. Liang. The statistics of streaming sparse regression. *arXiv preprint arXiv:1412.4182*, 2014.

A.B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer, 2009.

L.G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

L. Xiao. Dual averaging methods for regularized stochastic learning and online optimization. *Journal of Machine Learning Research*, 11:2543–2596, 2010.

Y. Zhang, J. Duchi, M. Jordan, and M. Wainwright. Information-theoretic lower bounds for distributed statistical estimation with communication constraints. In *Advances in Neural Information Processing Systems (NIPS)*, 2013.

A The Equivalence of SQ and COM

Proof of Lemma 3.2

We will simply simulate whatever statistical queries would have been made, using only a small number of samples and 1 bit of communication per sample.

More precisely, we will show that we can with probability $1 - \frac{\delta}{m}$ simulate a single statistical query of tolerance τ, using $\frac{2\log(2\delta/m)}{\tau^2}$ samples and 1 bit of communication per sample. By the union bound, we can then simulate m statistical queries with probability $1 - \delta$.

To simulate a single statistical query, suppose we have k samples $(x_1, f(x_1)), \ldots, (x_k, f(x_k))$. For each sample i, evaluate $\psi_i = \psi(x_i, f(x_i))$, and round it to $+1$ with probability $\frac{1+\psi_i}{2}$ and to -1 with probability

$\frac{1-\psi_i}{2}$. If we let z_i be the result of this rounding process, then clearly $\mathbb{E}[z_i] = \mathbb{E}_{x\sim\mathcal{D}}[\psi(x, f(x))]$. Moreover, if we let $\overline{z} = \frac{1}{k}(z_1 + \cdots + z_k)$, then by the Chernoff bound, we have

$$\mathbb{P}[|\overline{z} - \mathbb{E}[\psi(x, f(x))]| \geq \tau] \leq 2\exp(-k\tau^2/2). \tag{18}$$

For $k \geq \frac{2\log(2m/\delta)}{\tau^2}$, this probability is less than $\frac{\delta}{m}$, as desired. □

A.1 Lower bounds on PARITY

In this section we prove Corollary 3.5, which shows that even if we have a multiparty communication protocol in which each party is given a set of $n/4$ examples drawn from PARITY(n), then if each party communicates at most $b \leq n/16$ bits, an exponential number of parties is required to learn the parity. The proof follows from first proving a statistical query lower bound in the setting in which one can ask arbitrary queries of lists of $n/4$ examples (Proposition A.1), and then applying Theorem 3.3 to transform the statistical query lower bound into a bounded communication lower bound.

We will prove the lower bound in the following setting: a vector $s \in \{0,1\}^n$ is chosen uniformly at random from the set of nonzero vectors. Each s defines a rank $n-1$ subspace S of $\{0,1\}^n$ defined by $S = \{x : \sum_{i:s(i)=1} x_i = 0 \mod 2\}$, and "examples" consist of uniformly drawn elements from this PARITY subspace, $x \leftarrow Unif[S]$. The goal is to recover the set s given access to a set of such examples. This is equivalent to the standard problem of learning PARITY, except that instead of being given labelled examples $(x, \ell(x))$, we are considering the label as just another bit of the example. (This can also be regarded as the problem of learning PARITY given random positively labelled examples—the two learning tasks are equivalent, up to a factor of 2 in number of examples.)

For our proof, we will consider the setting where we are given *batches* of n/k examples drawn from a PARITY subspace $S \subset \{0,1\}^n$. Throughout, we denote a "batch" of examples from subspace S as $b = (x_1, \ldots, x_{n/k})$, where $x_i \in S$, and we denote the process of drawing this batch uniformly at random from S via "$b \leftarrow S$". Additionally, let "$S \leftarrow Unif$" denote the process of choosing such a PARITY subspace uniformly at random from such subspaces (namely, by choosing the set of parity indices $s \in \{0,1\}^n$, s.t. $s \neq \vec{0}$ uniformly at random).

Proposition A.1. *Consider statistical query algorithms for PARITY that are allowed to ask queries of a batch of $n/4$ examples—hence queries that propose functions $f : \{0,1\}^{n^2/4} \to [-1,1]$. Any such statistical query algorithm that learns PARITY with probability at least 0.6 after adaptively proposing functions $f_1, f_2, \ldots, f_N$ with $f_i : \{0,1\}^{n^2/4} \to [-1,1]$ and receiving $r_1, r_2, \ldots, r_N$ with r_i adversarially chosen subject to*

$$|r_i - \mathbb{E}_{b\leftarrow S}[f_i(b)]| \leq \tau,$$

where S is the true parity subspace, either requires $\tau \leq 2^{-n/8}$ or $N = \Omega(\frac{2^{n/8}}{n})$.

The following lemma, which is a simple application of Chebyshev's inequality to the random variable representing $E_{b\leftarrow S}[f(b)]$ over the choice of random subspace S, is the core of the proof of Proposition A.1. For a rank $n-1$ subspace $S \subset \{0,1\}^n$ and function mapping a list of n/k examples to the range $[-1,1]$, $f : \{0,1\}^{n^2/k} \to [-1,1]$ define $X_{S,f} = \mathbb{E}_{b\leftarrow S}[f(b)]$.

Lemma A.2. *Let $f : \{0,1\}^{n^2/k} \to [-1,1]$ be a function satisfying $\mathbb{E}_{S\leftarrow Unif}[X_{S,f}] = 0$, where $S \leftarrow Unif$ denotes the process of selecting a rank $n-1$ subspace of $\{0,1\}^n$ uniformly at random from the set of all $2^n - 1$ such subspaces; the following holds:*

$$\Pr_{S\leftarrow Unif}[|X_{S,f}| > \alpha] \leq \frac{3n}{\alpha^2 2^{n-2n/k}}.$$

Hence for $k = 4$, for sufficiently large n,

$$\Pr[|X_{S,f}| > 2^{-n/8}] \leq \frac{3n}{2^{n/4}}.$$

Before proving the lemma, we show why it implies Proposition A.1:

Proof of Proposition A.1. Fix any $\delta \in [0,1]$, $\tau = 2^{-n/8}$ and $N \leq \delta \frac{2^{n/4}}{3n}$, and consider the learning setting in which a parity S is chosen uniformly at random, and a statistical query algorithm begins by asking statistical query $SQ(f_1, \tau)$. By Lemma A.2, with probability at least $1 - \frac{3n}{2^{n/4}}$, the response $r_1 = \mathbb{E}_{S' \leftarrow Unif}[X_{S',f}]$ will satisfy $|r_i - \mathbb{E}_{b \leftarrow S}[f_1(b)]| \leq \tau$, and hence r_1, which is a deterministic function of f_1 will be an appropriate response to the statistical query. Inductively, defining f_i to be the i-th function proposed by the statistical query algorithm given the sequence of responses$r_1, \ldots, r_{i-1}$ as defined above, the probability that the responses $r_1, \ldots, r_N$ all satisfy $|r_i - \mathbb{E}_{b \leftarrow S}[f_i(b)]| \leq \tau|$ is at least $1 - \delta$. Since r_i are deterministic functions of f_i, and do not depend on the true parity S, for any two parities $S_1 \neq S_2$, the distribution of outcomes of the algorithm given parity S_1 will have total variation distance at most δ from the analogous distribution corresponding to the true parity being S_2, and hence no such algorithm can correctly output the parity set with probability greater than $1/2 + \delta/2$. □

We now prove Lemma A.2.

Proof of Lemma A.2. The proof will be a simple application of Chebyshev's inequality. We begin by bounding $\mathrm{Var}[X_{S,f}]$. In the following calculations, $\#S = 2^n - 1$ denotes the number of rank $n-1$ subspaces of $\{0,1\}^n$.

$$\begin{aligned}
\mathbb{E}[X_{S,f}^2] &= \mathbb{E}_S\left[\left(\frac{1}{2^{(n-1)(n/k)}}\sum_{b\in S} f(b)\right)^2\right] \\
&= \frac{1}{2^{2n^2/k-2n/k}}\frac{1}{(\#S)}\sum_S\sum_{b\in S} f(b)\sum_{b'\in S} f(b') \\
&= \frac{1}{(\#S)2^{2n^2/k-2n/k}}\sum_{b\in\{0,1\}^{n^2/k}} f(b)\sum_{b'\in\{0,1\}^{n^2/k}} f(b')\cdot \Pr[b,b'\in S \leftarrow Unif]\cdot(\#S), \qquad (19)
\end{aligned}$$

where $\Pr[b, b' \in S \leftarrow Unif]$ in the above equation denotes the probability that all the examples contained in the batches b and b' lie within a subspace S that is drawn uniformly at random from the set of rank $n-1$ subspaces. To bound this probability, let $r = rank(b \cup b')$ denote the rank of the space spanned by the list of $2n/k$ vectors represented by the two lists b and b'. Let $x_1, \ldots, x_r$ denote a basis for the space spanned by $b \cup b'$, and note that the probability that $x_1 \in S$ for a randomly chosen S is exactly $\frac{2^{n-1}-1}{2^n-1}$ (where the minus 1's take into account the fact that $x_i \neq \vec{0}$), and

$$\Pr[x_i \in S | x_1, \ldots, x_{i-1} \in S] = \frac{2^{n-i}-1}{2^{n-i+1}-1}\Pr[x_{i-1} \in S | x_1, \ldots, x_{i-2} \in S].$$

Hence

$$\Pr[b, b' \in S \leftarrow Unif] = \frac{(2^{n-1}-1)(2^{n-2}-1)\cdots(2^{n-r}-1)}{(2^n-1)(2^{n-1}-1)\cdots(2^{n-r+1}-1)} \in [\frac{1}{2^r} - \frac{r}{2^n}, \frac{1}{2^r}].$$

Combining this with Equation 19 yields that

$$\begin{aligned}
\mathbb{E}[X_{S,f}^2] &\leq \frac{2^{2n^2/k}}{2^{2n^2/k-2n/k}}\frac{n}{2^n} + \left|\frac{1}{2^{2n^2/k-2n/k}}\sum_b f(b)\sum_{b'} f(b')\frac{1}{2^{rank(b\cup b')}}\right| \\
&= \frac{n}{2^{n-2n/k}} + \left|\frac{1}{2^{2n^2/k-2n/k}}\sum_b f(b)\sum_{b'} f(b')\frac{1}{2^{rank(b\cup b')}}\right|. \qquad (20)
\end{aligned}$$

We will now bound this second term. Note that by our assumption that $\mathbb{E}_{S\leftarrow Unif}[X_{S,f}] = 0$, we have that

$$0 = \sum_b f(b)\Pr[b \in S \leftarrow Unif] = \sum_b f(b)\frac{(2^{n-1}-1)(2^{n-2}-1)\cdots(2^{n-rank(b)}-1)}{(2^n-1)(2^{n-1}-1)\cdots(2^{n-rank(b)+1}-1)},$$

and hence it follows that $|\sum_b f(b)\frac{1}{2^{rank(b)}}| \leq 2^{n^2/k}\frac{n}{2^n}$, and

$$|\sum_b f(b)\sum_{b'} f(b')\frac{1}{2^{rank(b)+rank(b')}}| = |\sum_b f(b)\frac{1}{2^{rank(b)}}\sum_{b'} f(b')\frac{1}{2^{rank(b')}}| \leq 2^{2n^2/k}\frac{n}{2^n}. \qquad (21)$$

To relate the above bounds to Equation 20, we now argue that

$$\left|\left(\sum_b f(b)\sum_{b'} f(b')\frac{1}{2^{rank(b)+rank(b')}}\right) - \left(\sum_b f(b)\sum_{b'} f(b')\frac{1}{2^{rank(b\cup b')}}\right)\right| \leq (2n/k)2^{2n^2/k-n+2n/k}. \quad (22)$$

To see this, note that for any pair b, b' such that $rank(b\cup b') = 2n/k$, we have $rank(b\cup b') = rank(b)+rank(b')$, and hence the above difference is trivially bounded in magnitude by

$$|\{b, b' | rank(b \cup b') < 2n/k\}| \leq 2^{2n^2/k}\frac{2n/k}{2^{n-2n/k}},$$

where the first term is the total number of pairs b, b', and the second term is a crude upper bound on the probability that a list of $2n/k$ randomly chosen vectors has rank less than $2n/k$.

We now combine Equations 20, 21, and 22:

$$\begin{aligned}
\mathbb{E}[X_{S,f}^2] &\leq \frac{n}{2^{n-2n/k}} + \left|\frac{1}{2^{2n^2/k-2n/k}}\sum_b f(b)\sum_{b'} f(b')\frac{1}{2^{rank(b\cup b')}}\right| \\
&\leq \frac{n}{2^{n-2n/k}} + \frac{2^{2n^2/k}(n/2^n)}{2^{2n^2/k-2n/k}} + \left|\frac{1}{2^{2n^2/k-2n/k}}\sum_b f(b)\frac{1}{2^{rank(b)}}\sum_{b'} f(b')\frac{1}{2^{rank(b')}}\right| \\
&\leq \frac{n}{2^{n-2n/k}} + \frac{2^{2n^2/k}(n/2^n)}{2^{2n^2/k-2n/k}} + \frac{2^{2n^2/k}(n/2^n)}{2^{2n^2/k-2n/k}} \\
&\leq \frac{3n}{2^{n-2n/k}}.
\end{aligned}$$

□

B Direct Communication Bounds with Assouad's Method

Proof of Lemma 3.8

Note that the left-hand-side of (2) is just the expansion of $\sum_x \mu(x)\,\mathrm{KL}\left(p_0(Z^{(1:m)}) \parallel p_x(Z^{(1:m)})\right)$ under the chain rule. Let $\hat{X}$ be any estimator of x. Then the probability of recovering x is $\sum_x \mu(x)p_x(\hat{X} = x)$, which we can bound as follows:

$$\begin{aligned}
\sum_x \mu(x)p_x(\hat{X} = x) &\leq \sum_x \mu(x)\left(p_0(\hat{X} = x) + |p_x(\hat{X} = x) - p_0(\hat{X} = x)|\right) \\
&\leq \mu_{\max}\sum_x p_0(\hat{X} = x) + \sum_x \mu(x)\|p_x - p_0\|_{TV} \\
&\leq \mu_{\max} + \sum_x \mu(x)\sqrt{\mathrm{KL}\left(p_0(Z^{(1:m)}) \parallel p_x(Z^{(1:m)})\right)/2} \qquad \text{(Pinsker's inequality)} \\
&\leq \mu_{\max} + \sqrt{\sum_x \mu(x)\,\mathrm{KL}\left(p_0(Z^{(1:m)}) \parallel p_x(Z^{(1:m)})\right)/2} \qquad \text{(Jensen's inequality)} \\
&\leq \mu_{\max} + \sqrt{K/2},
\end{aligned}$$

as claimed. □

Proof of Lemma 3.9

Recall that Z is a potentially random function of Y and $\hat{Z}$; writing $\pi(z \mid y, \hat{z})$ for its conditional distribution, we have that

$$p_0(z \mid \hat{z}) = \sum_y \pi(z \mid y, \hat{z})\, p_0(y), \quad p_x(z \mid \hat{z}) = \sum_y \pi(z \mid y, \hat{z})\, p_x(y).$$

We then have:

$$\begin{aligned}
\mathbb{E}_{x\sim\mu}[|p_x(z|\hat{z}) - p_0(z|\hat{z})|^2] &= \mathbb{E}_{x\sim\mu}\left[\sum_{y,y'}(p_x(y) - p_0(y))(p_x(y') - p_0(y'))\,\pi(z|y,\hat{z})\,\pi(z|y',\hat{z})\right] \\
&= \sum_{y,y'}\mathbb{E}_{x\sim\mu}\left[(p_x(y) - p_0(y))\,(p_x(y') - p_0(y'))\right]\,\pi(z|y,\hat{z})\,\pi(z|y',\hat{z}) \\
&= \sum_{y,y'} M_{y,y'}\,\pi(z|y,\hat{z})\pi(z|y',\hat{z}) \\
&\le \lambda_{\max}(M)\,\|\pi(z|\cdot,\hat{z})\|_2^2 \\
&\le \lambda_{\max}(M)\,\|\pi(z|\cdot,\hat{z})\|_1, \qquad (23)
\end{aligned}$$

where the last step follows since $\pi(z|y,\hat{z}) \le 1$. Note also that, since $p_x(y) \ge \epsilon/|\mathcal{Y}|$ for all x and y by assumption,

$$p_x(z|\hat{z}) = \sum_{y\in\mathcal{Y}} p_x(y)\,\pi(z|y,\hat{z}) \ge \frac{\epsilon}{|\mathcal{Y}|}\|\pi(z|\cdot,\hat{z})\|_1. \qquad (24)$$

We are now ready to bound the KL divergence:

$$\begin{aligned}
\mathrm{KL}\left(p_0(Z|\hat{z}) \parallel p_x(Z|\hat{z})\right) &\overset{(i)}{\le} \mathrm{D}_{\chi^2}\left(p_0(Z|\hat{z}) \parallel p_x(Z|\hat{z})\right) \\
&= \sum_z \frac{|p_x(z|\hat{z}) - p_0(z|\hat{z})|^2}{p_x(z|\hat{z})} \\
&\le \frac{|\mathcal{Y}|}{\epsilon}\sum_z \frac{|p_x(z|\hat{z}) - p_0(z|\hat{z})|^2}{\|\pi(z|\cdot,\hat{z})\|_1},
\end{aligned}$$

where (i) is Lemma 2.7 of Tsybakov [2009]. Hence,

$$\begin{aligned}
\sum_{x\in\mathcal{X}}\mu(x)\,\mathrm{KL}\left(p_0(Z|\hat{z}) \parallel p_x(Z|\hat{z})\right) &\le \frac{|\mathcal{Y}|}{\epsilon}\sum_z \frac{1}{\|\pi(z|\cdot,\hat{z})\|_1}\sum_x \mu(x)\,|p_x(z|\hat{z}) - p_0(z|\hat{z})|^2 \\
&= \frac{|\mathcal{Y}|}{\epsilon}\sum_z \frac{\mathbb{E}_{x\sim\mu}[|p_x(z|\hat{z}) - p_0(z|\hat{z})|^2]}{\|\pi(z|\cdot,\hat{z})\|_1} \\
&\overset{(ii)}{\le} \frac{|\mathcal{Y}|}{\epsilon}\sum_z \lambda_{\max}(M) \\
&= 2^b\lambda/\epsilon,
\end{aligned}$$

where (ii) is just the inequality (23). We thus have the claimed result. □

Lemma B.1. *For M as given in (5), the eigenvectors of M are $\chi_s(y) = (-1)^{\langle s,y\rangle}$, with corresponding eigenvalues $\frac{\mu(s)}{2^n}$.*

Proof. The verification is straightforward:

$$\begin{aligned}
[M\chi_s](y) &= \sum_{y'} M_{y,y'}\chi_s(y') \\
&= \frac{1}{4^n}\sum_{x\in\mathcal{X}}\mu(x)\sum_{y'}(-1)^{\langle x,y+y'\rangle}(-1)^{\langle s,y'\rangle} \\
&= \frac{1}{4^n}\sum_{x\in\mathcal{X}}\mu(x)(-1)^{\langle x,y\rangle}\sum_{y'}(-1)^{\langle x+s,y'\rangle} \\
&= \frac{1}{4^n}\sum_{x\in\mathcal{X}}\mu(x)(-1)^{\langle x,y\rangle}2^n\mathbb{I}[x=s] \\
&= \frac{1}{2^n}\mu(s)(-1)^{\langle s,y\rangle} \\
&= \frac{\mu(s)}{2^n}\chi_s(y).
\end{aligned}$$

□

C Polynomial Insensitivity of the Information Theoretic Approach

Proof of Proposition 3.12

Suppose that we have an algorithm that takes m steps. We will perform step $i \in \{1,\dots,m\}$ on the $(i-1)k+r$-th sample, where $r \sim \text{Uniform}(\{1,\dots,k\})$. For all other samples we transmit a blank message.

Let us now compute the entropy of each message; each message is either blank (with probability $1-1/k$) or else is a message whose entropy is b. Letting $h_2(p)$ be the entropy of a coin flip with probability p, we can then bound the total entropy of each message by

$$\begin{aligned}
\frac{b}{k} + h_2(1/k) &= \frac{b}{k} + \frac{1}{k}\log_2(k) + \frac{k-1}{k}\log_2\left(\frac{k}{k-1}\right) \\
&= \frac{b}{k} + \frac{\log_2(k)}{k} + \frac{1}{k}\log_2\left(\left(1+\frac{1}{k-1}\right)^{k-1}\right) \\
&\le \frac{b+\log_2(k)+\log_2(e)}{k},
\end{aligned}$$

as was to be shown. □

We can also prove a similar result for memory-constrained procedures:

Proposition C.1. *Any memory-constrained problem that can be learned to accuracy ε with m samples and $H(z_i) \le b$ can be learned to accuracy 2ε with mk samples and $H(z_i) \le \frac{b}{k} + \frac{\log_2(k\cdot e)}{k} + \log_2(mk^2|\mathcal{F}|)$ for any integer k. In particular, it can be learned with mb samples and $H(z_i) \le 1 + \log_2(emb^3|\mathcal{F}|)$.*

Proof. We split our memory into three chunks; at a high level, the first chunk will be used to run an instance of the original algorithm, the second chunk will be used to store the answer, and the third chunk will store a small amount of auxiliary data.

Our procedure first samples a random number $j \in \{1,\dots,k\}$. Then, it runs the original algorithm on the samples with index $i \in \{(j-1)m+1,\dots,jm\}$ (using the third chunk to track j and i). Afterwards, it writes the recovered answer to the second chunk, and zeroes out the first chunk. One the index i reaches km, it returns the answer in the second chunk (which is guaranteed to have been written to exactly once).

First, to see that this algorithm works, we need to make sure that the answer can be represented with $\log_2|\mathcal{F}|$ bits. This is straightforward if the answer $\hat{f}$ lies in $\mathcal{F}$, but this need not be the case. On the other hand, we know that $\mathbb{P}_{x\sim\mathcal{D}}[f(x)=\hat{f}(x)] \ge 1-\varepsilon$. Therefore, take any $f' \in \mathcal{F}$ satisfying $\mathbb{P}_{x\sim\mathcal{D}}[f'(x)=\hat{f}(x)] \ge$

$1-\varepsilon$, and use this as the answer[5]; this f' will satisfy $\mathbb{P}_{x\sim\mathcal{D}}[f(x) = f'(x)] \geq 1-2\varepsilon$, which is all that we require.

Now, let us measure the entropy of z_i under this procedure. As before, the first chunk is zeroes with probability $1-\frac{1}{k}$, and is otherwise a random variable with entropy at most b. Hence, as in Proposition 3.12, the entropy of this chunk is at most $\frac{\log_2(k\cdot e)}{k} + \frac{b}{k}$. Furthermore, the second chunk takes on at most $|\mathcal{F}|$ values and the third chunk at most $k \cdot km$ values (for the random draw j together with the counter i), and so they together add at most $\log_2(k^2 m|\mathcal{F}|)$ bits to the entropy. The claimed result follows. □

D Reductions with Valid Queries

Proof of Lemma 4.3

We first show that any statistical query with tolerance $\tau \leq 2$ can be implemented with $\lceil \log\left(\frac{2}{\tau}\right) \rceil$ statistical threshold queries with tolerance $\frac{\tau}{2}$. To do so, we use the following binary search algorithm to implement $\mathrm{SQ}(\psi, \tau)$:

$L \leftarrow -1 - \tau/2$
$R \leftarrow 1 + \tau/2$
while $R - L > 2\tau$ **do**
  $M \leftarrow \frac{L+R}{2}$
  $b \leftarrow \mathrm{STQ}(\psi, M, \tau/2)$
  if $b = 1$ **then**
    $L \leftarrow M - \tau/2$
  else
    $R \leftarrow M + \tau/2$
  end if
end while
return $\frac{L+R}{2}$

Consider the value of $R - L$; it is initially $2 + \tau$, and on each loop iteration updates as $(R-L)_{\text{new}} = \frac{1}{2}(R-L)_{\text{old}} + \frac{\tau}{2}$. After m loop iterations (and hence m calls to $\mathrm{STQ}(\psi, \cdot, \tau/2)$) we thus have $R - L = (2+\tau)\cdot 2^{-m} + \tau(1-2^{-m}) = \tau + 2^{1-m}$. For $m = \lceil \log_2\left(\frac{2}{\tau}\right) \rceil$, we then have $R - L \leq 2\tau$, at which point the loop exits. But we also then know that $\mathbb{E}[\psi(x, f(x))] \in (L, R)$, whence $\frac{R+L}{2}$ is a valid output of $\mathrm{SQ}(\psi, \tau)$.

Next, we show that any statistical threshold query with tolerance τ can be simulated with a valid statistical threshold query with tolerance at least $\tau/2$. Consider any statistical threshold query $q = (\psi, \mu, \tau)$. If q is already valid, then we are done. Otherwise, $\min(|\mathcal{F}^0(q)|, |\mathcal{F}^1(q)|) > \frac{3}{4}|\mathcal{F}|$, which means that $|\mathcal{F}^0(q) \cap \mathcal{F}^1(q)| > \frac{1}{2}|\mathcal{F}|$. This means that more than half of all f satisfy $\mathbb{E}[\psi(x, f(x))] \in (\mu-\tau, \mu+\tau)$. Therefore, more than one-fourth of all f satisfy either $\mathbb{E}[\psi(x, f(x))] \in (\mu-\tau, \mu]$ or $\mathbb{E}[\psi(x, f(x))] \in [\mu, \mu+\tau)$. We will assume that it is the former (the other case is symmetric). In this case, we can make a query $q' = (\psi, \mu+\tau/2, \tau/2)$. First, note that q' is valid, since

$$|\mathcal{F}^1(q')| = |\{f : \mathbb{E}[\psi(x, f(x))] > \mu\}| \leq |\{f : \mathbb{E}[\psi(x, f(x))] \notin (\mu-\tau, \mu]| \leq \frac{3}{4}|\mathcal{F}|.$$

In addition, given $r' = \mathrm{STQ}(q')$, we can also use r' as the answer to $\mathrm{STQ}(q)$: if $r' = 1$, then $\mathbb{E}[\psi(x, f(x))] > \mu > \mu - \tau$, and if $r' = 0$, then $\mathbb{E}[\psi(x, f(x))] < (\mu + \tau/2) + \tau/2 = \mu + \tau$. So, the valid query q' can be used to simulate q, which completes the proof. □

Proof of Lemma 4.4

We formalize our procedure with the following pseudocode.

procedure `BoundedMemoryLearn`($\mathcal{F}$, $\mathcal{D}$):
if $|\mathcal{F}| = 1$ **then**
  return the unique element of $\mathcal{F}$

[5] Finding such an f' may in general require superpolynomial computation.

end if
for $i = 1$ **to** m **do**
  $q \leftarrow$ `queryIfAllBad`$(i, \mathcal{F}, \mathcal{D})$ {gets query i assuming all previous responses are bad}
  if not `valid`$(q, \mathcal{F}, \mathcal{D})$ **then**
    $q \leftarrow$ `makeValid`$(q, \mathcal{F}, \mathcal{D})$
  end if
  $r \leftarrow \mathrm{STQ}(q)$
  if `good`$(r, \mathcal{F}, \mathcal{D})$ **then**
    $firstGoodIndex \leftarrow i$
    $goodResponse \leftarrow r$
    return `BoundedMemoryLearn`$(\mathcal{F}^r(q), \mathcal{D})$ {q is the first query to get a good response}
  end if
end for
return `answerIfAllBad`$(\mathcal{F}, \mathcal{D})$

Note that when we recurse to subsets of $\mathcal{F}$, we can use the same statistical query algorithm as before (since if it learns $\mathcal{F}$ it will certainly learn any subset of $\mathcal{F}$), though the return values of `valid` and `good` may change. By construction, the above procedure can only recurse $\mathcal{O}(\log|\mathcal{F}|)$ times, and only requires m queries and $\mathcal{O}(\log(m))$ bits at each level of recursion (to keep track of i, $firstGoodIndex$, and $goodResponse$).[6] This establishes the result. □

E Sparse Regression with Memory Constraints

Proof of Lemma 4.6

We make use of the following version of exponentiated gradient, which restricts the domain of optimization to the l^1-ball of radius R; the vector γ_j measures the error in the approximation z_j:

$$z_j = \nabla f(w_j) + \gamma_j, \tag{25}$$

$$w_{j+1} = \arg\min_{w} \left\{ \eta^{-1} \sum_{i=1}^{n} w^{(i)} \log\left(w^{(i)}\right) + \left\langle w, \sum_{j'=1}^{j} z_{j'} \right\rangle : \|w\|_1 \le R,\, w \ge 0 \right\}. \tag{26}$$

Note that this algorithm restricts to the positive orthant $w \ge 0$; however, we can remove this assumption (while making our constants worse by a factor of 2) by splitting each coordinate of w into a positive and negative part [Kivinen and Warmuth, 1997].

The approximate gradient z_j can be interpreted as the exact gradient of the modified function $\tilde{f}(w) = f(w) + \gamma_j \cdot w$. Thus, by standard online convex optimization results [e.g., Shalev-Shwartz, 2011], for any $\eta > 0$ and $n \ge 3$ we have

$$\begin{aligned}
&\sum_{j=1}^{m} \left(f(w_j) - f(w^*) + \gamma_j \cdot (w_j - w^*)\right) \\
&\quad \le \eta^{-1}\left(R\log(n/R) + \sum_{i=1}^{n} w^{*,(i)} \log\left(w^{*,(i)}\right)\right) + \eta \sum_{j=1}^{m} \|z_j\|_\infty^2 \\
&\quad \le \eta^{-1}\left(R\log(n) - R\log(R) + \|w^*\|_1 \log\|w^*\|_1\right) + \eta m B^2 \\
&\quad \le \eta^{-1} R \log(n) + \eta m B^2,
\end{aligned}$$

where the last inequality uses the fact that $x\log(x) \le y\log(y)$ whenever $x \le y$ and $y \ge 1$. Optimizing our choice of η, we then get

$$\sum_{j=1}^{m} \left(f(w_j) - f(w^*) + \gamma_j \cdot (w_j - w^*)\right) \le 2\sqrt{mR\log(n)}B.$$

[6] Note that, even though `answerIfAllBad`$(\mathcal{F}, \mathcal{D})$ need not lie in $\mathcal{F}$, it is a fixed quantity depending only on $\mathcal{F}$ and $\mathcal{D}$, and so requires no additional bits to represent.

Moreover, since $\|w_j - w^*\|_1 \le 2R$, we know that $\gamma_j \cdot (w_j - w^*) \ge -2\tau RB$, and so

$$\sum_{j=1}^{m} \left(f\left(w_j\right) - f\left(w^*\right)\right) \le 2\sqrt{mR\log\left(n\right)}B + 2\tau RBm.$$

Finally, the desired conclusion follows by convexity of $f(\cdot)$.

Proof of Theorem 4.7

Our proof relies on the min-count sketch construction [Cormode and Muthukrishnan, 2005]. For our purposes, the main implication of the construction is as follows: there is a distribution of matrices $A \in \{0,1\}^{\omega d \times n}$ such that, for any fixed vector $u \in \mathbb{R}^n$, we can recover $\hat{u}$ from Au such that (see, e.g., Section 2 of Gilbert and Indyk [2010])

$$\|\hat{u} - u\|_\infty \le \frac{2}{\omega}\|u\|_1, \text{ with probability at least } 1 - n\,2^{-d}.$$

Moreover, if we only get to observe Au up to tolerance τ in each coordinate, resulting in a recovered vector $\hat{u}_\tau$, then the result correspondingly weakens to

$$\|\hat{u}_\tau - u\|_\infty \le \frac{2}{\omega}\|u\|_1 + \tau, \text{ with probability at least } 1 - n\,2^{-d}.$$

In our situation, we want to estimate $z^* = \nabla f(w)$ as defined in Lemma 4.6 with as few statistical queries as possible. To do this, we proceed as follows (letting $u = z^*$):

- Draw a matrix A from the count-sketch distribution with parameters d and ω,
- Estimate Az^* to tolerance $\tau\|Az^*\|_\infty$ using ωd statistical queries of tolerance τ, and finally
- Generate a recovered vector z using the count-min sketch algorithm.

Note that $\|Az^*\|_\infty \le \|z^*\|_1$ (since all entries of A are in $\{0,1\}$) and hence the error on Az^* is at most $\tau\|z^*\|_1 \le \tau\|x\|_1|y - w\cdot x| \le 2\tau rR$. We of course also have $\|u\|_1 = \|z^*\|_1 \le 2rR$. Toegether, these imply that $\|z - z^*\|_\infty \le 2\left(\frac{2}{\omega} + \tau\right)rR$. Thus, setting

$$\omega = \left\lceil \frac{1}{\tau} \right\rceil \text{ and } d = \lceil \log\left(n\right) + \log\left(\delta^{-1}\right) + \log\left(m\right)\rceil, \text{ we conclude that } \|z_m - z_m^*\|_\infty \le 6\tau rR, \tag{27}$$

with probability at least $1 - \delta/m$; by the union bound, (27) holds with probability at least $1 - \delta$ for all m samples simultaneously. The upshot is that we can now get the gradients z from $\omega d = \mathcal{O}\left(\log(n/\delta)/\tau\right)$ statistical queries instead of n queries. In summary, we can then obtain error ε using

$$\mathcal{O}\left(R^3\log(n/\delta)/(\tau\varepsilon^2)\right) \text{ queries of tolerance } \tau = \mathcal{O}\left(\varepsilon/rR^2\right). \tag{28}$$

Substituting in for R and τ, this leads to $rk^5\log(n/\delta)/\varepsilon^3$ queries of tolerance $\varepsilon/(rk^2)$. To apply Theorem 4.1, we then need to derandomize, which we can do by standard amplification techniques, at the cost of increasing the number of queries by a further factor of $k\log(n/\varepsilon)$. Finally, applying Theorem 4.1, the number of samples needed is (up to polylogarithmic factors in k, n, and ε) the number of statistical queries divided by the square of the tolerance, times $\log(1/\delta)$. This yields a final number of samples equal to

$$\widetilde{\mathcal{O}}\left(r^3k^{10}\log^2(1/\delta)/\varepsilon^5\right), \tag{29}$$

as claimed.

ECCC ISSN 1433-8092
http://eccc.hpi-web.de