The complexity of approximating the entropy*

Tuğkan Batu† Sanjoy Dasgupta‡ Ravi Kumar§ Ronitt Rubinfeld¶

February 23, 2004

Abstract

We consider the problem of approximating the entropy of a discrete distribution under several oracle access models to the distribution. In the evaluation oracle model, the algorithm has access to the distribution in the form of an array where the i-th location is the probability of the i-th element. In this model, linear time in the size of the domain of the distribution is both necessary and sufficient for approximating the entropy.

We consider a generation oracle model in which the algorithm is given access only to independent samples from the distribution. In this model, we show that a γ-multiplicative approximation to the entropy can be obtained in $O\left(n^{(1+\eta)/\gamma^2}\log n\right)$ time for distributions with entropy $\Omega(\gamma/\eta)$, where n is the size of the domain of the distribution and η is an arbitrarily small positive constant. We show that one cannot get a multiplicative approximation to the entropy in general in this model. Even for the class of distributions to which our upper bound applies, we obtain a lower bound of $\Omega\left(n^{\max(1/(2\gamma^2),2/(5\gamma^2-2))}\right)$.

We next consider a combined oracle model in which the algorithm has access to both the generation and the evaluation oracles of the distribution. In this model, significantly more efficient algorithms can be achieved: a γ-multiplicative approximation to the entropy can be obtained in $O\left(\frac{\gamma^2\log^2 n}{h^2(\gamma-1)^2}\right)$ time for distributions with entropy $\Omega(h)$; for these class of distributions, we show a lower bound of $\Omega\left(\frac{\log n}{h(\gamma^2-1)}\right)$.

Finally, we consider two special families of distributions: those for which the probabilities of the elements decrease monotonically according to a known order, and those that are uniform over a subset of the domain. In each case, we give more efficient algorithms for approximating the entropy.

*A preliminary version of this paper appeared in the 34th ACM Symposium on Theory of Computing, 2002, Montreal, Canada.

†Department of Computer Sciences, University of Texas, Austin, TX 78712. This work was supported by ONR N00014-97-1-0505, MURI. Phone: (512) 471 9790, Fax: (512) 471 8885, Email: `batu@cs.utexas.edu`

‡University of California, San Diego. Phone: (858) 822 5270, Fax: (858) 534 7029, Email: `dasgupta@cs.ucsd.edu`

§IBM Almaden Research Center, San Jose, CA 95120. Phone: (408) 927 1885, Fax: (408) 927 3215, Email: `ravi@almaden.ibm.com`

¶NEC Laboratories America, Princeton, NJ 08540. Phone: (609) 951 2719, Fax: (609) 951 2488, Email: `ronitt@nec-labs.com`

1 Introduction

The Shannon entropy is a measure of the randomness of a distribution, and plays a central role in statistics, information theory, and data compression. Knowing the entropy of a random source can shed light on the compressibility of data produced by such a source. In this paper we consider the complexity of approximating the entropy under various different assumptions on the way the input is presented.

Suppose the algorithm has access to an *evaluation oracle*[1] in which the distribution $\mathbf{p}$ is given as an array where the i-th location contains the probability p_i assigned to the i-th element of the domain. It is clear that an algorithm that reads the whole representation can calculate the exact value of the entropy. However, it is also easy to see that in this model, linear time in the size of the domain is required even to approximate the entropy: Consider two distributions, one with a singleton support set (zero entropy) and the other with a two-element support set (positive entropy). Any algorithm which approximates the entropy to within any multiplicative factor must distinguish these two distributions, however, any randomized algorithm which distinguishes between two such distributions requires linear time in general.

Next suppose the distribution $\mathbf{p} = \langle p_1, \ldots, p_n \rangle$ is given as a *generation oracle*[1] which generates samples according to the distribution. This model has been considered in both the statistics and physics communities (c.f., [6, 10, 8, 9]), though none of the previous works provide a rigorous analysis of computational efficiency and sample complexity in terms of approximation quality. To the best of our knowledge, the only previously known algorithms which do not require superlinear (in the domain size) sample complexity are those presented in [8, 9]. These algorithms use an estimate of the collision probability, $\|\mathbf{p}\|^2$, to give a lower bound estimate of the entropy: using Jensen's inequality, it is shown [9] that

$$\log \|\mathbf{p}\|^2 = \log \sum_i p_i^2 \geq \sum_i p_i \log p_i = -H(\mathbf{p}).$$

In fact, when the infinity norm of $\mathbf{p}$, $\|\mathbf{p}\|_\infty$, is at most $n^{-\alpha}$, (in other words, when the min-entropy of $\mathbf{p}$ is large) the collision probability can be used to give an upper bound on the entropy of the distribution: using the relationship between norms,

$$\log \|\mathbf{p}\|^2 \leq \log(\|\mathbf{p}\|_\infty) \leq \log n^{-\alpha} = -\alpha \log n \leq -\alpha H(\mathbf{p}).$$

It is, however, unclear how to use the collision probability to obtain an arbitrary multiplicative approximation with a better sample complexity than our results (stated below), since approximating the collision probability itself will require $\Omega(\sqrt{n})$ samples. However, the collision probability can be used to understand much more about certain types of distributions; for instance, it exactly determines the entropy in the special case of distributions that are known to be uniform over an unknown subset of arbitrary size (see Section 7).

1.1 Our results

(1) THE GENERATION ORACLE MODEL: When the distribution is given as a generation oracle, we show that the entropy can be approximated well in sublinear time for a large class of distributions. Informally, a γ-multiplicative approximation to the entropy can be obtained in time $O(n^{(1+\eta)/\gamma^2} \log n)$, where n is the size of the domain of the distribution and η is an arbitrarily small

[1]We use the terminology from [7].

positive constant, provided that the distribution has $\Omega(\gamma/\eta)$ entropy. We show that one cannot get a multiplicative approximation to the entropy in general. But, even for the class of distributions to which our upper bound applies, we obtain a lower bound of $\Omega(n^{\max(1/(2\gamma^2), 2/(5\gamma^2-2))})$. Our algorithm is simple—we partition the elements in the domain into big or small based on their probability masses and approximate the entropy of the big and small elements separately.

It is interesting to consider what these bounds imply for the complexity of achieving a 2-approximation for distributions with non-constant entropy: Our upper bound yields an algorithm which runs in $O\left(n^{\frac{1+o(1)}{4}}\right)$ time. Our lower bound demonstrates that any algorithm that 2-approximates the entropy requires $\Omega(n^{1/8})$ time.

(2) The evaluation oracle model: When the distribution is given as an evaluation oracle, we show a lower bound of $\Omega(n2^{-\gamma^2(h+1)})$ to γ-approximate the entropy for the class of distributions with entropy at least h.

(3) The combined oracle model: We then consider a *combined oracle* model, in which the algorithm has access to both the generation and the evaluation oracles of the distribution. We assume that the two oracles are consistent, which is a natural assumption for such a model. In the combined oracle model, we give a γ-approximation algorithm which runs in time $O\left(\frac{\gamma^2 \log^2 n}{h^2(\gamma-1)^2}\right)$ for distributions with entropy $\Omega(h)$; we also show a lower bound of $\Omega\left(\frac{\log n}{h(\gamma^2-1)}\right)$ for this class of distributions. For example, to achieve a constant approximation for distributions with entropy $\Omega(h)$, our upper bound yields an algorithm which runs in time $O((1/h^2)\log^2 n)$; our lower bound is $\Omega((1/h)\log n)$, i.e., quadratically smaller than the upper bound.

(4) Special families of distributions: Finally we consider two families of distributions for which we show more efficient upper bounds. The first family is that of *monotone distributions*, in which the probability of an element decreases monotonically in the label of the element (i.e., $p_i \geq p_{i+1}$). We give an $O(\log n(1+\log^{-1}\gamma))$-time (resp. $O((\log n)^6 \mathrm{poly}(\gamma))$-time) algorithm for γ-approximating the entropy in the evaluation oracle model (resp. generation oracle model). The second family is that of *subset-uniform distributions*, in which the distribution is uniform over some subset of the domain. In this case we give $O(\sqrt{k})$-time algorithms for approximating the entropy, where k is the size of the support set.

Model		Lower bound	Upper bound
Evaluation oracle:	General	$\Omega(n)$	$O(n)$
	When $H(\mathbf{p}) \geq h$	$\Omega(n2^{-\gamma^2(h+1)})$, Thm. 13	?
Generation oracle:	General	∞, Thm. 6	–
	High enough entropy	$\Omega\left(\max(n^{1/(2\gamma^2)}, n^{2/(5\gamma^2-2)})\right)$, $H(\mathbf{p}) > \Omega((\log n)/\gamma^2)$, Thm. 12	$\tilde{O}(n^{1/\gamma^2})$, $H(\mathbf{p}) > \Omega(\gamma)$, Thm. 1
Combined oracle:	General	$\Omega(n^{(1-o(1))/\gamma^2})$, Thm. 17	?
	When $H(\mathbf{p}) \geq h$	$\Omega\left(\frac{\log n}{h(\gamma^2-1)}\right)$, Thm. 18	$O(\frac{\gamma^2 \log^2 n}{h^2(\gamma-1)^2})$, Thm. 14

Table 1: Our results for γ-approximation, where $\gamma > 1$.

1.2 Related work

The work of Goldreich and Vadhan [5] considers the complexity of approximating the entropy in a different model in which a distribution Y is encoded as a circuit C such that $Y = C(X)$, where the input X to the circuit is uniformly distributed; in this model, they show that a version of the problem is complete for statistical zero-knowledge. Their version of the problem could be viewed as an additive approximation to the entropy.

The work of [2] and [1] considers algorithms for testing other properties of distributions in the generation oracle model. The properties considered are whether two input distributions are close or far, and whether a joint distribution is independent, respectively. Both works give algorithms whose sample complexity is sublinear in the domain size as well as lower bounds showing the algorithms to be nearly optimal.

1.3 Organization

In Section 2, we introduce the basic definitions used in this paper. In Section 3, we give algorithms and lower bounds for the case when the input is presented in the generation oracle model. In Section 4, we provide a lower bound for the case when the input is presented in the evaluation oracle model. In Section 5, we give algorithms and lower bounds for the case when the input is presented in the combined oracle model. We give more efficient algorithms for two families of distributions in Section 6 and Section 7.

2 Preliminaries

We consider discrete distributions over a domain of size n, which we denote by $[n] \stackrel{\text{def}}{=} \{1, \ldots, n\}$. Let $\mathbf{p} = \langle p_1, \ldots, p_n \rangle$ be such a distribution where $p_i \geq 0, \sum_{i=1}^{n} p_i = 1$. An algorithm is said to have an *evaluation oracle* access to the distribution $\mathbf{p}$ if oracle query i is answered by p_i. An algorithm is said to have a *generation oracle* access to the distribution $\mathbf{p}$ if it is given a source which generates samples independently according to $\mathbf{p}$. An algorithm is said to have a *combined oracle* access to the distribution $\mathbf{p}$ if it is given both an evaluation and a generation oracle access to $\mathbf{p}$. We say the algorithm is in model $\mathcal{O}$ if it has oracle access of type $\mathcal{O}$ to the distribution.

The entropy of the distribution $\mathbf{p}$ is defined as

$$H(\mathbf{p}) \stackrel{\text{def}}{=} -\sum_{i=1}^{n} p_i \log p_i$$

where all the logarithms are to the base 2. For a set $S \subseteq [n]$, we define $w_{\mathbf{p}}(S) \stackrel{\text{def}}{=} \sum_{i \in S} p_i$, and we define the contribution of S to the entropy as

$$H_S(\mathbf{p}) \stackrel{\text{def}}{=} -\sum_{i \in S} p_i \log p_i.$$

Notice that $H_S(\mathbf{p}) + H_{[n]\setminus S}(\mathbf{p}) = H(\mathbf{p})$.

For a distribution $\mathbf{p}$, we denote its L_2-norm by $\|\mathbf{p}\| \stackrel{\text{def}}{=} \sqrt{\sum_{i=1}^{n} p_i^2}$ and its L_∞-norm by $\|\mathbf{p}\|_\infty \stackrel{\text{def}}{=} \max_{i=1}^{n} p_i$. For distributions $\mathbf{p}, \mathbf{q}$, we denote the L_1-distance between them by $|\mathbf{p} - \mathbf{q}| \stackrel{\text{def}}{=} \sum_{i=1}^{n} |p_i - q_i|$.

Let $\gamma > 1$ and let $\mathcal{D}$ be a family of distributions. We say that $\mathcal{A}$ is an algorithm in model $\mathcal{O}$ for γ-*approximating* the entropy of a distribution in $\mathcal{D}$, if for every $\mathbf{p} \in D$, given oracle access to $\mathbf{p}$ in model $\mathcal{O}$, algorithm $\mathcal{A}$ outputs $\mathcal{A}(\mathbf{p})$ such that $H(\mathbf{p})/\gamma \leq \mathcal{A}(\mathbf{p}) \leq \gamma H(\mathbf{p})$ with probability at least $3/4$. (This probability can be made $1-\delta$ by running the algorithm $\log(1/\delta)$ times and outputting the median of the values.) The time complexity of $\mathcal{A}$ is specified as a function of γ and n. We will use the notation $\mathcal{D}_h$ for $h > 0$ to denote the family of distributions with entropy at least h.

3 The generation oracle model

3.1 Upper bounds

In this section we obtain an algorithm for estimating the entropy of a large class of distributions in the generation oracle model. We prove the following theorem:

Theorem 1 *For any $\gamma > 1$ and $0 < \epsilon_o < 1/2$, there exists an algorithm in the generation oracle model that can approximate the entropy of a distribution on $[n]$ in $\mathcal{D}_{4\gamma/(\epsilon_o(1-2\epsilon_o))}$ to within a multiplicative factor of $(1+2\epsilon_o)\gamma$ with probability at least $3/4$ in $O((n^{\frac{1}{\gamma^2}}/\epsilon_o^2) \cdot \log n)$ time.*

Given $\eta > 0$ and $\gamma' > 1$, one can set $\gamma = \gamma'/(1+2\epsilon_o)$ in the above algorithm and choose ϵ_o small enough to yield a γ'-approximation algorithm whose running time is $O(n^{(1+\eta)/\gamma'^2} \log n)$ when the entropy is $\Omega(\gamma/\eta)$. Note that choosing η to be small affects both the running time and the family of distributions to which the algorithm can be applied.

The main idea behind the algorithm is the following. We classify elements in $[n]$ as big or small based on their probability mass. For a fixed $\alpha > 0$ and a distribution $\mathbf{p}$, the set of indices with high probabilities is defined as:

$$B_\alpha(\mathbf{p}) \stackrel{\text{def}}{=} \{i \in [n] \mid p_i \geq n^{-\alpha}\}.$$

We then approximate the contribution of the entropy of the big and small elements separately. Section 3.1.1 shows how to approximate the entropy of the big elements, Section 3.1.2 shows how to approximate the entropy of the small elements, and Section 3.1.3 combines these approximations to yield Theorem 1.

3.1.1 Approximating the entropy of the big elements

To estimate the entropy of big elements, we approximate the probability of each of the big elements by sampling the generation oracle sufficiently many times.

Lemma 2 *For every $0 < \alpha, \epsilon_o \leq 1$ and sufficiently large n, there is an algorithm that uses $O((n^\alpha/\epsilon_o^2) \cdot \log n)$ samples from $\mathbf{p}$ and outputs $\mathbf{q}$ such that with probability at least $1 - n^{-1}$, the following hold for all i:*

1. *if $i \in B_\alpha(\mathbf{p})$, then $|p_i - q_i| \leq \epsilon_o p_i$, and*
2. *if $p_i \leq \frac{1-\epsilon_o}{1+\epsilon_o} n^{-\alpha}$, then $q_i \leq (1-\epsilon_o) n^{-\alpha}$*

Proof. Let $m = O((n^\alpha/\epsilon_o^2) \cdot \log n)$. Fix i and let X_j be the indicator variable that the j-th sample is i. Let $q_i = \sum X_j/m$. By Chernoff bounds, if $p_i \geq n^{-\alpha}$, then

$$\Pr[q_i > (1+\epsilon_o)p_i] \leq \exp\left(-\frac{\epsilon_o^2 p_i m}{3}\right) \leq \exp\left(-\frac{\epsilon_o^2 n^{-\alpha} m}{3}\right) \leq \frac{1}{n^2}.$$

Using a similar argument for the other direction, we can bound the probability that any element i such that $p_i \geq n^{-\alpha}$ is not estimated within $1+\epsilon_o$. Using Chernoff bounds again, we can show that for i such that $p_i < \frac{1-\epsilon_o}{1+\epsilon_o} n^{-\alpha}$,

$$\Pr\left[q_i > (1-\epsilon_o)n^{-\alpha}\right] \leq n^{-2}.$$

Hence, if $i \in B_\alpha(\mathbf{p})$ then $|p_i - q_i| \leq \epsilon_o p_i$. Now, (1) and (2) of the lemma follow from a union bound over all i. ■

The following lemma shows that the contribution to entropy of the elements in $B_\alpha(\mathbf{p})$ can be approximated well using $\mathbf{q}$ instead of $\mathbf{p}$.

Lemma 3 *For any set $B \subseteq [n]$ such that for each $i \in B$, $|p_i - q_i| \leq \epsilon_o p_i$ for $0 < \epsilon_0 \leq 1$,*

$$|H_B(\mathbf{q}) - H_B(\mathbf{p})| \leq \epsilon_o H_B(\mathbf{p}) + 2\epsilon_o w_{\mathbf{p}}(B).$$

Proof. For $i \in B$, let $q_i = (1+\varepsilon_i)p_i$ such that $|\varepsilon_i| \leq \epsilon_o$.

$$\begin{aligned}
H_B(\mathbf{q}) - H_B(\mathbf{p}) &= -\sum_{i\in B}(1+\varepsilon_i)p_i \log((1+\varepsilon_i)p_i) \;+\; \sum_{i\in B} p_i \log p_i \\
&= -\sum_{i\in B}(1+\varepsilon_i)p_i \log p_i - \sum_{i\in B}(1+\varepsilon_i)p_i \log(1+\varepsilon_i) + \sum_{i\in B} p_i \log p_i \\
&= -\sum_{i\in B}\varepsilon_i p_i \log p_i - \sum_{i\in B}(1+\varepsilon_i)p_i \log(1+\varepsilon_i).
\end{aligned}$$

By the triangle inequality,

$$\begin{aligned}
|H_B(\mathbf{q}) - H_B(\mathbf{p})| &\leq \left|-\sum_{i\in B}\varepsilon_i p_i \log p_i\right| + \left|\sum_{i\in B}(1+\varepsilon_i)p_i \log(1+\varepsilon_i)\right| \\
&\leq \sum_{i\in B} -|\varepsilon_i| p_i \log p_i + \sum_{i\in B} p_i \left|(1+\varepsilon_i)\log(1+\varepsilon_i)\right| \\
&\leq \epsilon_o H_B(\mathbf{p}) + 2\epsilon_o w_{\mathbf{p}}(B).
\end{aligned}$$

The last step above uses the fact that for $|\varepsilon| \leq \epsilon_o \leq 1$, $|(1+\varepsilon)\log(1+\varepsilon)| \leq 2|\varepsilon| \leq 2\epsilon_o$. ■

3.1.2 Approximating the entropy of the small elements

Now, we obtain an estimate on the entropy contribution of the small elements. Suppose set S is such that $S \subseteq [n] \setminus B_\alpha(\mathbf{p})$.

First of all, if $w_{\mathbf{p}}(S) \leq n^{-\alpha}$, the contribution of entropy from S is below any constant and can be ignored. So, we can assume without loss of generality that $w_{\mathbf{p}}(S) \geq n^{-\alpha}$. Let $\hat{w}$ be the normalized frequency of the samples from S (i.e., the number of samples from S divided by the total number of samples) in $O((n^\alpha/\epsilon_o^2)\log n)$ samples.

Lemma 4 *With probability at least $1 - n^{-1}$, $(1-\epsilon_o)w_{\mathbf{p}}(S) \leq \hat{w} \leq (1+\epsilon_o)w_{\mathbf{p}}(S)$.*

Proof. Let X_j be the indicator random variable that takes value 1 with probability $w_{\mathbf{p}}(S)$ for $j = 1, \ldots, m = O((n^\alpha/\epsilon_o^2)\log n)$, and $X = \sum X_j$. Note that X/m corresponds to $\hat{w}$, and $\mathrm{E}[X] = m w_{\mathbf{p}}(S)$. The lemma follows by Chernoff bounds and $w_{\mathbf{p}}(S) \geq n^{-\alpha}$. ■

The following lemma shows how to approximate the entropy of the small elements.

Lemma 5 *For $S \subseteq [n] \setminus B_\alpha(\mathbf{p})$, $\alpha w_{\mathbf{p}}(S) \log n \leq H_S(\mathbf{p}) \leq w_{\mathbf{p}}(S) \log n + (\log e)/e$.*

Proof. Since $H_S(\mathbf{p})$ is a symmetric concave function, it is is maximized when the probability mass $w_{\mathbf{p}}(S)$ is spread evenly over all the elements in S, i.e., when $p_i = w_{\mathbf{p}}(S)/|S|$ for all i. In this case,

$$H_S(\mathbf{p}) = w_{\mathbf{p}}(S) \log(|S|/w_{\mathbf{p}}(S)) = w_{\mathbf{p}}(S) \log |S| - w_{\mathbf{p}}(S) \log w_{\mathbf{p}}(S) \leq w_{\mathbf{p}}(S) \log n + (\log e)/e.$$

Here, we used $\sup_{x\in[0,1]} x \log(1/x) = (\log e)/e$, which follows from simple calculus.

Again, since $H_S(\mathbf{p})$ is symmetric concave, it will take its minimum value when as many as possible of its variables are at their extreme points, namely, 0 and 1. So, the minimum value is achieved when $n^\alpha w_{\mathbf{p}}(S)$ of p_i's have the value $n^{-\alpha}$, and the rest are 0. In this case, $H_S(\mathbf{p}) = \alpha w_{\mathbf{p}}(S) \log n$. ■

3.1.3 Putting it together

In this section we describe our approximation algorithm for $H(\mathbf{p})$ and prove Theorem 1. The following is our algorithm for obtaining a γ-approximation to the entropy:

Algorithm ApproximateEntropy(γ, ϵ_o)

1. $\alpha = 1/\gamma^2$.
2. Get $m = O((n^\alpha/\epsilon_o^2) \cdot \log n)$ samples from $\mathbf{p}$.
3. Let $\mathbf{q}$ be the normalized frequencies of $[n]$ in the sample (q_i is the frequency of i in the sample normalized by m) and $B = \{i \mid q_i > (1-\epsilon_o)n^{-\alpha}\}$ (Notice $B_\alpha(\mathbf{p}) \subseteq B$).
4. Output $H_B(\mathbf{q}) + \frac{w_{\mathbf{q}}([n]\setminus B) \log n}{\gamma}$.

Proof. (of Theorem 1) Let $S = [n]\setminus B$. Using Lemma 3 and Lemma 5,

$$\begin{aligned}
H_B(\mathbf{q}) + \frac{w_{\mathbf{q}}(S)\log n}{\gamma} &\leq (1+\epsilon_o)H_B(\mathbf{p}) + 2\epsilon_o + \frac{1+\epsilon_o}{\gamma} w_{\mathbf{p}}(S)\log n \\
&\leq (1+\epsilon_o)(H_B(\mathbf{p}) + \gamma H_S(\mathbf{p})) + 2\epsilon_o \\
&\leq (1+\epsilon_o)\gamma H(\mathbf{p}) + 2\epsilon_o \\
&\leq (1+2\epsilon_o)\gamma H(\mathbf{p}),
\end{aligned}$$

if $H(\mathbf{p}) \geq 2/\gamma$. Similarly,

$$\begin{aligned}
H_B(\mathbf{q}) + \frac{w_{\mathbf{q}}(S)\log n}{\gamma} &\geq (1-\epsilon_o)H_B(\mathbf{p}) - 2\epsilon_o + \frac{1-\epsilon_o}{\gamma} w_{\mathbf{p}}(S)\log n \\
&\geq (1-\epsilon_o)\left(H_B(\mathbf{p}) + \frac{(H_S(\mathbf{p}) - e^{-1}\log e)}{\gamma}\right) - 2\epsilon_o \\
&= (1-\epsilon_o)(H_B(\mathbf{p}) + H_S(\mathbf{p})/\gamma) - \frac{1-\epsilon_o}{\gamma} e^{-1}\log e - 2\epsilon_o \\
&\geq H(\mathbf{p})/((1+2\epsilon_o)\gamma),
\end{aligned}$$

if $H(\mathbf{p}) \geq \frac{4\gamma}{\epsilon_o(1-2\epsilon_o)} \geq 2/\gamma$. ■

3.2 Lower bounds

In this section we prove lower bounds on the number of samples needed to approximate the entropy of a distribution to within a multiplicative factor of $\gamma > 1$. All of our lower bounds are shown by giving pairs of distributions that are hard to distinguish and have entropy ratio at least γ^2. The lower bounds follow since an algorithm which γ-approximates the entropy would allow one to distinguish the distributions.

3.2.1 Impossibility of approximating entropy in general

First, we show that because distributions could have zero entropy, there is no algorithm that can γ-approximate the entropy of *every* distribution.

Theorem 6 *For any $\gamma > 1$, there is no algorithm that γ-approximates the entropy of every distribution in the generation oracle model.*

Proof. Assume that $\mathcal{A}$ is an algorithm that approximates the entropy of any distribution. Let $t(n)$ be an upper bound on the runtime of $\mathcal{A}$ on distributions over $[n]$. Consider the two distributions $\mathbf{p}$ and $\mathbf{q}$ where $\mathbf{p} = \langle 1, 0, \ldots, 0\rangle$ and $\mathbf{q} = \langle 1 - 1/(10t(n)), 1/(10t(n)), 0, \ldots, 0\rangle$. Any algorithm which uses only $t(n)$ samples cannot distinguish between $\mathbf{p}$ and $\mathbf{q}$ with probability higher than $1/10$. Since the entropy of $\mathbf{p}$ is 0, any algorithm which gives a multiplicative approximation should output 0. On the other hand, any algorithm which approximates the entropy of $\mathbf{q}$ to within a multiplicative factor of γ should output a positive value. Thus, any algorithm which γ-approximates the entropy would be able to distinguish between $\mathbf{p}$ and $\mathbf{q}$. By contradiction, the theorem follows. ∎

3.2.2 Lower bounds on approximating the entropy of high-entropy distributions

We will show (Lemma 11) a lower bound of $\Omega(n^{\frac{2}{5\gamma^2-2}})$ samples to γ-approximate the entropy for any distribution in $\mathcal{D}_{(5\log n)/(10\gamma^2-4)}$. Before we show this lower bound, we show a simpler lower bound of $\Omega(n^{1/2\gamma^2})$ samples.

Theorem 7 *For any $\gamma > 1$ and sufficiently large n, any algorithm in the generation oracle model that γ-approximates the entropy of a distribution in $\mathcal{D}_{(\log n)/\gamma^2}$ is required to make $\Omega(n^{1/2\gamma^2})$ samples from it.*

Proof. Consider two distributions $\mathbf{p}$ and $\mathbf{q}$ on n elements where $\mathbf{p}$ is uniform on the set $[n]$ and $\mathbf{q}$ is uniform on a set S that is a randomly chosen subset of $[n]$ of size n^{1/γ^2}. It is easy to see that $H(\mathbf{p})/H(\mathbf{q}) = \gamma^2$. By the Birthday Paradox, with probability $1/3$, we do not see any repetitions in the sample before we take $\Omega(n^{1/2\gamma^2})$ samples from either distribution. Hence, the samples from $\mathbf{p}$ and $\mathbf{q}$ look identical. Thus, we can conclude that $\Omega(n^{1/2\gamma^2})$ samples are needed to distinguish these distributions. ∎

We now prove a better lower bound for the case of $\gamma < \sqrt{2}$. Our lower bound proof borrows the outline of the arguments in [2, 1]. We start by giving a canonical form for entropy approximation algorithms. In particular, we describe an aggregate representation of the samples such an algorithm takes. We then prove that we can assume, without loss of generality, that the algorithm is given this representation of the samples as input instead of the samples themselves.

Let $S = \{x_1, \ldots, x_s\}$ be a set of samples from a distribution over $[n]$.

Definition 8 (Fingerprint [2, 1]) *The* fingerprint *d_S of a sample set S is the function $d_S : [s] \to [n] \cup \{0\}$ such that $d_S(i)$ is the number of elements x such that x appears i times in S.*

We will just use $d(i)$ when S is clear from the context. Next, we show that the fingerprint of a set of samples is just as useful to approximate the entropy as the samples themselves.

Definition 9 (Canonical algorithm [2, 1]) *An algorithm in the generation oracle model is called* canonical *if the input to the algorithm is only the fingerprint of the samples.*

Lemma 10 ([2, 1]) *Given algorithm $\mathcal{A}$ that γ-approximates the entropy of a distribution in the generation oracle model using s samples, there exists a canonical algorithm $\mathcal{A}'$ that γ-approximates the entropy using the fingerprint of a sample set of size s with error probability upper bounded by that of $\mathcal{A}$.*

Proof. We will construct a canonical algorithm $\mathcal{A}'$ using calls to $\mathcal{A}$. In particular, algorithm $\mathcal{A}'$, given a fingerprint of a sample from a distribution $\mathbf{p}$, will construct a set of samples with the same fingerprint and pass this set of samples to algorithm $\mathcal{A}$. One can view this new set of samples as a relabeling (according to a permutation of the labels) of the original sample that the fingerprint was constructed from. Thus, the set of samples passed to algorithm $\mathcal{A}$ can be viewed as samples from a distribution $\mathbf{p}'$ (obtained from $\mathbf{p}$ by permuting the labels of the elements) such that $H(\mathbf{p}) = H(\mathbf{p}')$. The procedure is as follows. Upon getting the fingerprint of a sample S, $\mathcal{A}'$ chooses $d_S(i)$ elements at random from $[n]$ without replacement for each i. Then, $\mathcal{A}'$ passes the multiset S' such that an element chosen for i appears exactly i times in S'. Finally, $\mathcal{A}'$ outputs the value that $\mathcal{A}$ outputs on S'.

The crucial observation is that the algorithm $\mathcal{A}$ is invariant under relabeling of the elements. But, our construction of $\mathcal{A}'$ actually simulates some randomly chosen permutation π of the domain elements (denoted by $\pi(\mathbf{p})$) for $\mathcal{A}$. Therefore,

$$\Pr\left[\mathcal{A}' \ \gamma\text{-approximates } H(\mathbf{p})\right] = \mathrm{E}_\pi[\Pr\left[\mathcal{A} \ \gamma\text{-approximates } H(\pi(\mathbf{p}))\right]],$$

which is at least the correctness probability of $\mathcal{A}$ since the correctness probability of $\mathcal{A}$ is determined by the worst possible distribution.

■

Lemma 11 *For any $\gamma > 1$, sufficiently large n, and any canonical algorithm $\mathcal{A}$ using $o(n^{2/(5\gamma^2-2)})$ samples, there exist two distributions $\mathbf{p}$ and $\mathbf{q}$ over $[n]$ with $H(\mathbf{p})/H(\mathbf{q}) \geq \gamma^2$, such that $\mathcal{A}$ cannot distinguish between $\mathbf{p}$ and $\mathbf{q}$ with probability at least $2/3$.*

Proof idea. Let $\alpha \stackrel{\text{def}}{=} 2/(5\gamma^2-2)$ and $\beta \stackrel{\text{def}}{=} 3\alpha/2$ and let $\mathcal{A}$ be a canonical algorithm that uses $o(n^\alpha)$ samples. To prove this lemma, we define the distributions $\mathbf{p}$ and $\mathbf{q}$ as follows:

$$p_i \stackrel{\text{def}}{=} \begin{cases} n^{-\alpha} & 1 \leq i \leq n^\alpha/2 \\ n^{-1} & n/2 < i \leq n \\ 0 & \text{otherwise} \end{cases}$$

$$q_i \stackrel{\text{def}}{=} \begin{cases} n^{-\alpha} & 1 \leq i \leq n^\alpha/2 \\ n^{-\beta} & n/2 < i \leq (n+n^\beta)/2 \\ 0 & \text{otherwise} \end{cases}$$

Call i *big* if $1 \leq i \leq n^\alpha/2$ and *small* if otherwise and $p_i > 0$.

Note that $H(\mathbf{p}) = (\alpha+1)/2 \cdot \log n$ and $H(\mathbf{q}) = (\alpha+\beta)/2 \cdot \log n$. Therefore,

$$\frac{H(\mathbf{p})}{H(\mathbf{q})} = \frac{\alpha+1}{\alpha+\beta} = \frac{5\gamma^2}{5\gamma^2-2} \cdot \frac{5\gamma^2-2}{5} = \gamma^2.$$

First, note that when restricted to the big elements, distributions $\mathbf{p}$ and $\mathbf{q}$ are identical; the only distinguishing aspect between them is in the contribution of the small elements to the fingerprint. But since $\mathcal{A}$ is canonical, it cannot distinguish the contribution of the small elements from the contribution of the big elements to the fingerprint. For entries of the fingerprint that correspond to three or more collisions, $\mathcal{A}$ can assume that only the big elements contribute to the value of the entry, since we show that it is rare for small elements to occur more than twice when only $o(n^\alpha)$ samples are taken (see Lemma 27). Thus, those entries for three or more collisions have almost identical distributions whether they come from $\mathbf{p}$ or $\mathbf{q}$. However, for the other entries (those corresponding to elements that occur at most twice), we must also show that no algorithm can distinguish the two distributions. At a high level, the reason for this is that when only $o(n^\alpha)$ samples are taken, the variance of double occurrences of *big* elements is large enough to swamp the difference in the expected number of double occurrences of *small* elements between $\mathbf{p}$ and $\mathbf{q}$ (Lemma 29) and thus hides any difference in their distribution. This intuition alone would not be enough to conclude the proof—what if the distributions of the double occurrences had essentially the same expectation and variance, but were distributed very differently? For example, what if the number of double occurrences of elements in $\mathbf{p}$ is always even, whereas in $\mathbf{q}$ it is always odd? We show that the distributions on the entries have nice properties which essentially say that they are "smooth" and can in fact be approximated by independent Gaussian-like distributions. Using these properties, we are able to show that the statistical difference between the fingerprints generated according to $\mathbf{p}$ and the fingerprints generated according to $\mathbf{q}$ is small. The technical details of this proof essentially follow those in [2, 1]; for completeness, we present them in the Appendix. Thus, we obtain:

Theorem 12 *For any $\gamma > 1$ and sufficiently large n, any algorithm in the generation oracle model that γ-approximates the entropy of a distribution over $[n]$ in $\mathcal{D}_{(\log n)/2\gamma^2}$ needs $\Omega\left(n^{\max\{1/(2\gamma^2),2/(5\gamma^2-2)\}}\right)$ samples from it.*

4 The evaluation oracle model: a lower bound

In the introduction, we mentioned that a linear number of queries is necessary to approximate the entropy in the evaluation oracle model in general. Since the total number of possible queries is n, the complexity of entropy approximation in this model is settled. Next, we present a lower bound on the number of queries needed to approximate the entropy in the evaluation oracle model, when a lower bound on the entropy of the distribution can be assumed.

Theorem 13 *For $\gamma > 1$, $h > 0$, and sufficiently large n, any algorithm in the evaluation oracle model that γ-approximates the entropy of a distribution over $[n]$ in $\mathcal{D}_h$ is required to make $\Omega(n2^{-\gamma^2(h+1)})$ queries.*

Proof. We will define two distributions $\mathbf{p}$ and $\mathbf{q}$ in $\mathcal{D}_h$ with entropy ratio at least γ^2 and show that $\Omega(n2^{-\gamma^2(h+1)})$ queries are required to distinguish them.

Let R be a uniformly chosen subset of $[n]$ of size $2^{\gamma^2(h+1)}$. The distribution $\mathbf{p}$ is defined to be uniform over R. Let S be a uniformly chosen subset of $[n]$ of size $2^{\gamma^2(h+1)}/(\gamma^2(1+(1/h)))$ and let s be a uniformly chosen element in $[n]\setminus S$. The distribution $\mathbf{q}$ assigns probability $2^{-\gamma^2(h+1)}$ to each of the elements in S and assigns the rest of the probability mass, namely, $1-(\gamma^{-2}/(1+1/h))$, to s.

We have $H(\mathbf{p}) = \gamma^2(h+1)$, whereas $h \le H(\mathbf{q}) \le h+1$ (since $H_S(\mathbf{q}) = \frac{1}{\gamma^2\cdot(1+(1/h))}\cdot\gamma^2\cdot(h+1) = h$). Hence, $H(\mathbf{p})/H(\mathbf{q}) \ge \gamma^2$. The number of queries that are required to distinguish $\mathbf{p}$ and $\mathbf{q}$ is at least the reciprocal of the fraction of the elements with nonzero probabilities, which is $\Omega(n/2^{\gamma^2(h+1)})$. ■

5 The combined oracle model

In this section we consider the combined oracle model where an algorithm is given access to both an evaluation and a generation oracle access to the same distribution.

5.1 Upper bound

Our algorithm for this case hinges on an alternate interpretation of the entropy: the entropy of the distribution $\mathbf{p}$ is the expected value of $-\log p_i$ where i is distributed according to $\mathbf{p}$. The algorithm, provided that the entropy value is not too small, needs only a polylogarithmic number of samples and probes.

Theorem 14 *For any $\gamma > 1$, there exists an algorithm in the combined oracle model that can approximate the entropy of a distribution on $[n]$ in $\mathcal{D}_h$ to within a multiplicative factor of γ with probability at least $3/4$ in $O\left(\frac{\gamma^2 \log^2 n}{h^2(\gamma-1)^2}\right)$ time.*

Proof. Let $m = \Theta\left(\frac{\gamma^2 \log^2 n}{h^2(\gamma-1)^2}\right)$. The algorithm takes m samples, say $i_1, \ldots, i_m$ from $\mathbf{p}$. The output of the algorithm is $X = (1/m)\sum_{j\in[m]} - \log p_{i_j}$, where p_{i_j} values are obtained from the evaluation oracle.

Define the random variable $X_j \stackrel{\text{def}}{=} -\log p_{i_j}$ for $j = 1, \ldots, m$. Clearly, $\mathrm{E}[X_j] = H(\mathbf{p})$. All that remains to show is that the variance of the summation is not too large, so that we get a γ-approximation to the entropy value. Since the X_j's are independent, it will suffice to bound the variance of an individual X_j. We bound this variance in the following lemma.

Lemma 15 $\mathrm{Var}[X_j] \leq \log^2 n$.

Proof. For $n = 2$, maximizing $\mathrm{Var}[X_j] = p\log^2 p + (1-p)\log^2(1-p) - (p\log p + (1-p)\log(1-p))^2$ subject to $0 \leq p \leq 1$ yields $\mathrm{Var}[X_j] < 1 = \log^2 n$. Therefore, let $n \geq 3$. Since $\mathrm{Var}[X_j] \leq \mathrm{E}\left[X_j^2\right]$, it suffices to show an upper bound on $\mathrm{E}\left[X_j^2\right] = \sum_i p_i \log^2 p_i$.

Note that the function $f(x) = x\log^2 x$ is concave for $0 < x < e^{-1}$. Hence, conditioned on $p_i < e^{-1}$ for all i, $\sum_i f(p_i)$ is maximized when $\mathbf{p}$ is the uniform distribution (this argument is analogous to the proof that the entropy takes its maximum value for the uniform distribution). Thus, we get an upper bound of $\log^2 n$.

Next, we show that the sum $\sum_i f(p_i)$ is maximized when $p_i < e^{-1}$ for all i. Suppose $p_j \geq e^{-1}$ for some j. Since $n \geq 3$, there exists k such that $p_k \leq (1-p_j)/(n-1)$. Consider the derivative $f'(x) = \log^2 x + 2\log(e)\log x$, at points p_j and p_k. By simple calculus, $f'(p_k) > f'(p_j)$. Hence, we can increase the sum by simultaneously decreasing p_j and increasing p_k. By combining with the argument above, we conclude that $\sum_i f(p_i) \leq \log^2 n$.

■

By the independence of the X_j's, $\mathrm{Var}[X] = \mathrm{Var}[X_j]/m \leq (\log^2 n)/m$.

At this point, we use Chebyshev's inequality, which states that for $\rho > 0$, $\Pr[|X - \mathrm{E}[X]| \geq \rho] \leq \mathrm{Var}[X]/\rho^2$, to bound the error probability of the algorithm.

$$
\begin{aligned}
\Pr[\text{A does not } \gamma\text{-approximate } H(\mathbf{p})] &= \Pr[X \leq H(\mathbf{p})/\gamma \text{ or } X \geq \gamma H(\mathbf{p})] \\
&\leq \Pr[|X - H(\mathbf{p})| \geq (\gamma-1)H(\mathbf{p})/\gamma] \\
&\leq \frac{\gamma^2 \log^2 n}{m \cdot H(\mathbf{p})^2(\gamma-1)^2} \leq 1/3,
\end{aligned}
$$

where the last inequality follows from the choice of m. ■

Corollary 16 *There exists an algorithm $\mathcal{A}$ in the combined oracle model which γ-approximates $H(\mathbf{p})$ in $O((\frac{\gamma}{\gamma-1})^2)$ time when $H(\mathbf{p}) = \Omega(\log n)$,*

5.2 Lower bounds

The next theorem gives a lower bound for the combined oracle model when the entropy of the distribution is not lower bounded. The distribution used to show this lower bound has very small entropy and thus does not fall into the family of distributions for which the above upper bound applies.

Theorem 17 *For $\gamma > 1$ and sufficiently large n, any algorithm in the combined oracle model that γ-approximates the entropy of a distribution over $[n]$ (with non-zero entropy) is required to take $\Omega(n^{\frac{1-o(1)}{\gamma^2}})$ samples from it.*

Proof. Let $\alpha = \frac{1-o(1)}{\gamma^2} < 1$. Consider distributions $\mathbf{p}$ and $\mathbf{q}$ defined as follows:

$$p_i \stackrel{\text{def}}{=} \begin{cases} 1 - n^{-\alpha} & i = 1 \\ n^{-\alpha} & i = 2 \\ 0 & \text{otherwise} \end{cases}$$
$$q_i \stackrel{\text{def}}{=} \begin{cases} 1 - n^{-\alpha} & i = 1 \\ n^{-1} & 2 \le i \le n^{1-\alpha} + 1 \\ 0 & \text{otherwise} \end{cases}$$

It can be calculated that $H(\mathbf{p}) = O(n^{-\alpha}) + \alpha n^{-\alpha} \log n$ and $H(\mathbf{q}) > n^{-\alpha} \log n$. By the choice of α, $H(\mathbf{q})/H(\mathbf{p}) \ge \gamma^2$.

Let $\mathcal{P}$ be the family of distributions obtained from $\mathbf{p}$ by permuting the labels of the elements. Define $\mathcal{Q}$ similarly for $\mathbf{q}$. It is simple to verify that any algorithm taking $o(n^\alpha)$ samples and making $o(n^\alpha)$ probes will fail to distinguish between a randomly chosen member of $\mathcal{P}$ and a randomly chosen member of $\mathcal{Q}$ with high probability. ■

The next theorem gives a lower bound on the complexity of approximating the entropy in the combined oracle model for distributions with entropy above some specific threshold. The proof generalizes the counterexample in Theorem 17.

Theorem 18 *For $\gamma > 1$, $h > 0$, and sufficiently large n, any algorithm in the combined oracle model that γ-approximates the entropy of a distribution over $[n]$ in $\mathcal{D}_h$ is required to take $\Omega(\log n/(h(\gamma^2 - 1))$ samples from it.*

Proof. Let $\alpha < 1$ (to be determined later) and $k \stackrel{\text{def}}{=} \lceil 2^{h/(1-n^{-\alpha})} \rceil$. Consider the following distributions $\mathbf{p}$ and $\mathbf{q}$ defined as follows:

$$p_i \stackrel{\text{def}}{=} \begin{cases} (1 - n^{-\alpha})/k & 1 \le i \le k \\ n^{-\alpha} & i = k+1 \\ 0 & \text{otherwise} \end{cases}$$
$$q_i \stackrel{\text{def}}{=} \begin{cases} (1 - n^{-\alpha})/k & 1 \le i \le k \\ n^{-1} & k+1 \le i \le k + n^{1-\alpha} \\ 0 & \text{otherwise} \end{cases}$$

It can be calculated that $H(\mathbf{p}) = h + O(n^{-\alpha}) + \alpha n^{-\alpha} \log n$ and $H(\mathbf{q}) > h + n^{-\alpha} \log n$.

Let $\mathcal{P}$ be the family of distributions obtained from $\mathbf{p}$ by permuting the labels of the elements. Define $\mathcal{Q}$ similarly for $\mathbf{q}$. It is simple to verify that any algorithm taking $o(n^\alpha)$ samples and making $o(n^\alpha)$ probes will fail to distinguish between a randomly chosen member of $\mathcal{P}$ and a randomly chosen member of $\mathcal{Q}$ with high probability.

If we choose α such that $n^\alpha = (\log n)/(2h(\gamma^2 - 1))$, then the entropy ratio

$$\frac{H(\mathbf{q})}{H(\mathbf{p})} > \frac{h + n^{-\alpha} \log n}{h + (1 + o(1))\alpha n^{-\alpha} \log n} \geq \frac{h(2\gamma^2 - 1)}{h(1 + o(1))} \geq \gamma^2.$$

This concludes the proof. ■

6 Monotone distributions

A *monotone distribution* $\mathbf{p} = \langle p_1, \ldots, p_n \rangle$ is one for which $p_i \geq p_{i+1}$ for all i. The structure of a monotone distribution makes it much easier to approximate the entropy.

6.1 The evaluation oracle model

We show that given evaluation oracle access to a monotone distribution, we can approximate the entropy in polylogarithmic time.

Theorem 19 *For $\gamma > 1$, there exists an algorithm in the evaluation oracle model that can approximate the entropy of a monotone distribution on $[n]$ in $\mathcal{D}_{\Omega(\gamma^2/(\sqrt{\gamma}-1))}$ to within a multiplicative factor of γ with probability at least $3/4$ in $O(\lceil 1/\log \gamma \rceil \log n)$ time.*

Proof. The algorithm $\mathcal{A}$ partitions the domain elements into sets such that all the elements in a set have similar probability values, and then uses the sizes of each of the sets to estimate the entropy.

The algorithm $\mathcal{A}$ constructs this partition recursively. For an interval $[\ell, u]$, values p_ℓ and p_u is obtained from the evulation oracle and the interval is partitioned into equal sub-intervals unless $p_\ell/p_u \leq \gamma$ or $p_\ell \leq n^{-2}$. Starting with the interval $[1, n]$, at the end of this procedure, a sequence of elements $i_o = 1, i_1, \ldots, i_k = n$ is obtained such that for each $j \leq k$, either $p_{i_j} \leq n^{-2}$ or $p_{i_j}/p_{i_{j+1}} \leq \gamma$. Note that $k \leq \lceil 1/\log \gamma \rceil \log n$. Let k' be the smallest j such that $p_{i_j} \leq n^{-2}$, otherwise let $k' = k$.

Consider the following partition of $[n]$. For $1 \leq j \leq k'$, let $B_j = \{i \mid i_{j-1} \leq i < i_j\}$. Let B_0 be the remaining i's and therefore for $i \in B_0$, $p_i \leq n^{-2}$. Thus, $w_\mathbf{p}(B_0) < 1/n$ and $H_{B_0}(\mathbf{p}) < 2n^{-1} \log n$.

At this point, with no further queries, $\mathcal{A}$ has a $\sqrt{\gamma}$-approximation to every p_i for $i \notin B_0$. This is because, if $i \in B_j$, then $\mathcal{A}$ can use the geometric mean of the largest and smallest values in B_j to approximate the value of p_i. Using these approximate p_i values, $\mathcal{A}$ computes the entropy values that are a γ-approximation to $H_{B_j}(\mathbf{p})$'s for $j > 0$ as follows.

Let $B = \cup_{j \geq 1} B_j$. For each $i \in B$, let $q_i \stackrel{\text{def}}{=} c_i p_i$ be the representative probability value that the algorithm uses for B_j to which i belongs. By the earlier argument, $\frac{1}{\sqrt{\gamma}} \leq c_i \leq \sqrt{\gamma}$ for all $i \in B$. For all the other i's, let $q_i = 0$.

$$\begin{aligned} H_B(\mathbf{q}) &= -\sum_{i \in B} q_i \log q_i = -\sum_{i \in B} c_i p_i \log(c_i p_i)) = -\sum_{i \in B} c_i p_i \log p_i - \sum_{i \in B} c_i p_i \log c_i \\ &\leq \sqrt{\gamma} H_B(\mathbf{p}) + \frac{w_\mathbf{p}(B) \log e}{e} \quad \leq \quad \gamma H(\mathbf{p}), \end{aligned}$$

when $H(p) \geq \log e/(e(\gamma - \sqrt{\gamma}))$. The first inequality follows from the fact that $-x \log x \leq (\log e)/e$ for all x. Similarly,

$$\begin{aligned}
H_B(\mathbf{q}) &= -\sum_{i\in B} q_i \log q_i = -\sum_{i\in B} c_i p_i \log(c_i p_i) = -\sum_{i\in B} c_i p_i \log p_i - \sum_{i\in B} c_i p_i \log c_i \\
&\geq \frac{1}{\sqrt{\gamma}} H_B(\mathbf{p}) - w_{\mathbf{p}}(B)\sqrt{\gamma}\log\sqrt{\gamma} \;\; \geq \;\; \frac{1}{\sqrt{\gamma}}\left(H(\mathbf{p}) - \frac{2\log n}{n}\right) - w_{\mathbf{p}}(B)\sqrt{\gamma}\log\sqrt{\gamma} \\
&\geq H(\mathbf{p})/\gamma,
\end{aligned}$$

when $H(\mathbf{p}) \geq (\gamma^2 + (2\sqrt{\gamma} n^{-1} \log n))/(\sqrt{\gamma} - 1)$. The second to last inequality follows from $H_{B_0}(\mathbf{p}) < 2n^{-1}\log n$.

Finally, the algorithm outputs $-\sum_{i\in B} q_i \log q_i$, which is a γ-approximation to $H(\mathbf{p})$. The number of probes that the algorithm makes is $k \leq \lceil 1/\log\gamma \rceil \log n$. ■

6.2 The generation oracle model

We show that the entropy of a monotone distribution can be approximated in polylogarithmic time also in the generation oracle model. Our algorithm rests on the following observation that is formally stated in Lemma 20: if a monotone distribution $\mathbf{p}$ over $[n]$ is such that $w_{\mathbf{p}}([n/2])$ and $w_{\mathbf{p}}([n]\backslash[n/2])$ are very close, then the distribution must be close to uniform. In such a case, we can approximate the entropy of the distribution by the entropy of the uniform distribution.

The main idea behind our algorithm is to recursively partition the domain into half, stopping the recursion if the total sum of the probabilities of the two halves are close or if they are both too small to contribute much to the total entropy. Our algorithm can be viewed as forming a tree based on the set of samples S, where the root is labeled by the range $[1, n]$, and if the node labeled by the range $[i, j]$ is partitioned, its children are labeled by the ranges $[i, (i+j)/2]$ and $[(i+j)/2+1, j]$, respectively. Once the partition tree is determined, the algorithm estimates the entropy by assuming that the distribution restricted to the ranges labeling each leaf is uniform, and combines the estimates in proportion to their total contribution to the probability. By the choice of our splitting and stopping criteria, we show that the number of leaves in the tree is at most polylogarithmic in n. This in turn allows us to bound both the running time and the probability of error.

More specifically, the procedure **BuildTree**(S, β) takes as input a parameter $\beta > 1$ and a multiset S of m samples from $\mathbf{p}$, and outputs a rooted binary tree T_S as follows: Let v be a node in the tree that is currently a leaf corresponding to the interval $[i, j]$ for some i and j. For an interval I, let S_I denote the set of samples that are in I and $|I|$ the length of the interval. We determine that v will remain a leaf if either of the following two conditions is satisfied:

- $|S_{[i,j]}| < m\beta/\log^3 n$ (call v *light*), or
- $|S_{[i,(i+j)/2]}| \leq \beta |S_{[(i+j)/2+1,j]}|$ (call v *balanced*).

Otherwise, we split v's interval by attaching two children to v, corresponding to the intervals $[i, (i+j)/2]$ (the left child) and $[(i+j)/2+1, j]$ (the right child). Let $\mathcal{I}(T_S)$ denote the set of intervals corresponding to the balanced leaves of T_S.

For each balanced interval $I \in \mathcal{I}(T_S)$, we estimate the contribution of the interval to the total entropy of the distribution. Note that if the interval I had uniform conditional distribution, then

$$H_I(\mathbf{p}) = \sum_{i\in I} \frac{w_{\mathbf{p}}(I)}{|I|} \log \frac{|I|}{w_{\mathbf{p}}(I)} = w_{\mathbf{p}}(I)\left(\log|I| - \log w_{\mathbf{p}}(I)\right) = w_{\mathbf{p}}(I)\left(\log(|I|/2) - \log(w_{\mathbf{p}}(I)/2)\right).$$

Motivated by this, we define a function $\alpha(I, \beta)$ which approximates the entropy in the balanced interval I:

$$\alpha(I, \beta) \stackrel{\text{def}}{=} \frac{|S_I|}{m} \left(\log \frac{|I|}{2} + \log \frac{m}{\beta |S_I|} \right).$$

We now give the top level description of our algorithm:

Algorithm MonotoneApproximateEntropy(γ)

1. $\beta = \sqrt{\gamma}$.
2. Get a multiset S of $m = O((\beta^5 \log^4 n)/(\beta - 1)^2)$ samples from $\mathbf{p}$.
3. $T_S = \mathbf{BuildTree}(S, \beta)$.
4. Output $\sum_{I \in \mathcal{I}(T_S)} \alpha(I, \beta)$.

The main steps in the proof are the following. First, we give a key lemma upon which the whole algorithm is based; this lemma implies that for an interval corresponding to a balanced leaf, the upper and lower bounds on the possible entropy values are fairly close (Lemma 20). The rest of the proof is devoted to showing that the domain can be split into intervals that are either balanced or small enough that they do not contribute much (in total) to the entropy of the distribution. In Lemma 21, we show that sampling can be reliably used to decide whether or not to split an interval. We then quantify the relationship between $\alpha(I, \beta)$ and $H_I(\mathbf{p})$ for each interval I corresponding to a balanced leaf, in terms of the sampling error (Lemma 23). Note that if it were possible to partition the whole domain into balanced intervals of large enough size, then it would be a simple matter to bound the the number of intervals and thus the error probability and running time of the algorithm. The most challenging part of the proof is to show that (1) the number of light intervals is at most nearly logarithmic in the size of the domain (Lemma 24) and (2) the error contribution from the light intervals is negligible and can thus be ignored. Given these two facts, one can then bound the error probability and running time of the algorithm. It is possible to get a simple polylogarithmic bound on the number of light intervals by noting that the algorithm does not split them, and therefore each of their weights is at least $\Omega(m\beta/\log^3 n)$. Unfortunately, this simple bound will not be enough to give us any bound on the error of our algorithm. Using that the splitting criterion splits unbalanced intervals, we are able to give a nearly logarithmic bound on the number of light intervals. This in turn can be used to get a nontrivial bound on the error of the algorithm on light intervals. Finally, we put these together to show that our algorithm approximates the entropy (Theorem 25).

First, we show upper and lower bounds on the entropy contribution of an interval in terms of the total weight and the weight distribution between two halves of the interval.

Lemma 20 *Let I be an interval of length $2k$ in $[n]$, I_1 and I_2 be the bisection of I, and $\mathbf{p}$ be a monotone distribution over $[n]$. Then,*

$$H_I(\mathbf{p}) \leq w_{\mathbf{p}}(I) \log k - w_{\mathbf{p}}(I_1) \log w_{\mathbf{p}}(I_1) - w_{\mathbf{p}}(I_2) \log w_{\mathbf{p}}(I_2),$$

and

$$H_I(\mathbf{p}) \geq 2w_{\mathbf{p}}(I_2) \log k - w_{\mathbf{p}}(I_2) \Big(\log w_{\mathbf{p}}(I_1) + \log w_{\mathbf{p}}(I_2) \Big).$$

In particular, the ratio of the upper bound to the lower bound on $H_I(\mathbf{p})$ is at most $w_{\mathbf{p}}(I)/2w_{\mathbf{p}}(I_2)$.

Proof. The proof of the upper bound follows from the concavity of $H(\mathbf{p})$. The maximum value is attained when the available weight is distributed uniformly over the elements.

Let $w_1 \stackrel{\mathrm{def}}{=} w_{\mathbf{p}}(I_1)$ and $w_2 \stackrel{\mathrm{def}}{=} w_{\mathbf{p}}(I_2)$. We will prove the lower bound even for functions that satisfy a relaxation of the monotonicity property: namely, the condition that for $i \leq k$, $p_i \geq w_2/k$, and for $i > k$, $p_i \leq w_1/k$. It is easy to verify that any monotone distribution will satisfy this new constraint. Again, by the concavity of the entropy function, we can plug w_2/k for the elements in I_1 and w_1/k for as many elements as possible in I_2 to give a lower bound on $H_I(\mathbf{p})$. Hence, we get

$$H_I(\mathbf{p}) \geq w_2 \log \frac{k}{w_2} + w_2 \log \frac{k}{w_1}.$$

■

For a balanced leaf corresponding to an interval I with the bisection I_1, I_2, the ratio $w_{\mathbf{p}}(I)/2w_{\mathbf{p}}(I_2)$ can be made small by choosing the parameter β appropriately.

The following lemma shows that the samples can be used to decide if an interval should be split.

Lemma 21 *Let I be an interval in $[1, n]$ such that $w_{\mathbf{p}}(I) \geq \log^{-3} n$ and I_1, I_2 be a bisection of I. Let S be a sample set drawn from $\mathbf{p}$. For $\beta > 1$,*

1. *with probability at least $1 - n^{-2}$, $mw_{\mathbf{p}}(I)/\beta \leq |S_I| \leq \beta m w_{\mathbf{p}}(I)$;*
2. *if $w_{\mathbf{p}}(I_1)/w_{\mathbf{p}}(I_2) \geq 2\beta - 1$, then with probability at least $1 - 2n^{-2}$, $|S_{I_1}| \geq \beta \cdot |S_{I_2}|$;*
3. *if $w_{\mathbf{p}}(I_1)/w_{\mathbf{p}}(I_2) \leq (1 + \beta)/2$, then with probability at least $1 - 2n^{-2}$, $|S_{I_1}| \leq \beta \cdot |S_{I_2}|$.*

Proof. Part 1 follows from a straight-forward application of the Chernoff bounds. The random variable X that corresponds to $|S_I|$ is a summation of m independent Bernoulli trials, each with success probability $w_{\mathbf{p}}(I)$. Therefore, $\mathrm{E}[X] = mw_{\mathbf{p}}(I)$, and by the choice of m in the algorithm, the probability that this summation is less than $\mathrm{E}[X]/\beta$ or more than $\beta\mathrm{E}[X]$ is at most $1/n^2$.

By Part 1, we know that with probability at least $1 - n^{-2}$, $|S_I| \geq mw_{\mathbf{p}}(I)/\beta$. Fix any $t > mw_{\mathbf{p}}(I)/\beta$. To prove Part 2, consider the ratio of the number of samples from I_1 and I_2 conditioned on the event that there are exactly t samples from I. Let Y_i, for $i = 1, \ldots, t$, be an indicator random variable that takes the value 1 if the i-th of these t samples is in I_2, and $Y = \sum_i Y_i$. Therefore, we want to show that the probability that $(t - Y)/Y < \beta$ is at most $2/n^2$.

The rest of the proof is an application of the Chernoff bounds. Note that $(t - Y)/Y < \beta$ implies $Y > t/(\beta + 1)$. Since $\mathrm{E}[Y] \leq t/(2\beta)$, we get

$$\Pr\left[Y > \frac{t}{\beta+1}\right] \leq \Pr\left[Y > \mathrm{E}[Y] + \frac{t(\beta-1)}{2\beta(\beta+1)}\right] \leq \exp\left(\frac{-t(\beta-1)^2}{\beta^2(\beta+1)^2}\right).$$

Conditioned on the event that $t \geq mw_{\mathbf{p}}(I)/\beta$, this probability is less than $1/n^2$. Combining this with Part 1, we can conclude that with probability at least $1 - 2n^{-2}$, we have $|S_{I_1}| \geq \beta \cdot |S_{I_2}|$.

Similarly, the third part of the lemma can be proved. ■

By Lemma 21, we can assume that for a "balanced interval" I, the ratio of the weights for the halves is at most $2\beta - 1$ and that intervals associated with two sibling nodes have weight ratio at least $(1 + \beta)/2$.

We now make an assumption which states that no bad event happens.

Assumption 22 *(1) For each interval we decide to split, we have that $w_p(I_1)/w_p(I_2) \geq (1+\beta)/2$; (2) for each balanced interval, we have that $w_p(I_1)/w_p(I_2) \leq 2\beta - 1$; and (3) each light leaf has weight at most $\beta^2/\log^3 n$.*

Now we can show that under the assumption above, the entropy contribution of each balanced interval is approximated well. Recall that $\mathcal{I}(T_S)$ denote the set of all balanced intervals in T_S.

Lemma 23 *Under Assumption 22, for every $I \in \mathcal{I}(T_S)$, if $w_{\mathbf{p}}(I) \geq \log^{-3} n$, then*

$$\frac{H_I(\mathbf{p})}{\beta} - 2\beta w_{\mathbf{p}}(I) \leq \alpha(I,\beta) \leq \beta^2 H_I(\mathbf{p}).$$

Proof. Let I_1, I_2 be the bisection of I. Under Assumption 22, $|S_I|/(m\beta) \leq w_{\mathbf{p}}(I) \leq |S_I|\beta/m$ and $w_{\mathbf{p}}(I_1)/w_{\mathbf{p}}(I_2) \leq 2\beta - 1$. Combining these with Lemma 20, we can conclude that the maximum and minimum possible entropy values for $H_I(\mathbf{p})$ have ratio at most β. Therefore,

$$\begin{aligned}
\alpha(I,\beta) &= \frac{|S_I|}{m}\left(\log\frac{|I|}{2} + \log\frac{m}{\beta|S_I|}\right) \\
&\leq \beta w_{\mathbf{p}}(I)\log(|I|/2) - \beta w_{\mathbf{p}}(I)\log(w_{\mathbf{p}}(I)) \\
&\leq \beta w_{\mathbf{p}}(I)\log(|I|/2) - \beta(w_{\mathbf{p}}(I_1)\log w_{\mathbf{p}}(I_1) - w_{\mathbf{p}}(I_2)\log w_{\mathbf{p}}(I_2)) \\
&\leq \beta^2 H_I(\mathbf{p}).
\end{aligned}$$

The third inequality above follows from the concavity of the function $-x\log x$, and the last inequality follows from Lemma 20. Similarly, for the other direction, we have

$$\begin{aligned}
\alpha(I,\beta) &= \frac{|S_I|}{m}\left(\log\frac{|I|}{2} + \log\frac{m}{\beta|S_I|}\right) \\
&\geq \frac{w_{\mathbf{p}}(I)}{\beta}\log(|I|/2) - \frac{w_{\mathbf{p}}(I)}{\beta}\log(w_{\mathbf{p}}(I)/2) - \frac{w_{\mathbf{p}}(I)\log(2\beta^2)}{\beta} \\
&\geq \beta^{-1}\left(w_{\mathbf{p}}(I)\log(|I|/2) - w_{\mathbf{p}}(I_1)\log w_{\mathbf{p}}(I_1) - w_{\mathbf{p}}(I_2)\log w_{\mathbf{p}}(I_2)\right) - 2\beta w_{\mathbf{p}}(I) \\
&\geq \frac{H_I(\mathbf{p})}{\beta} - 2\beta w_{\mathbf{p}}(I).
\end{aligned}$$

■

Next, we show a bound on the number of nodes in the tree.

Lemma 24 *Under Assumption 22, given $\beta > 1$, the number of nodes in T_S is at most*

$$\frac{6\log n \log\log n}{\log(\beta+1) - 1}.$$

Proof. Each node v in the tree corresponds to an interval $I_v = [i,j]$ of probabilities $p_i, \ldots, p_{j-1}$; let $w(v) = p_i + \cdots + p_{j-1}$. For each level h in the tree, let N_h (resp. L_h) denote the number of internal nodes (resp. leaves) at level h. The total number of nodes in the tree (whose depth is at most $\log n$) is at most $\sum_{h=1}^{\log n} N_h + L_h$. But, $\sum_h L_h = 1 + \sum_h N_h$. Now, we show an upper bound on N_h.

Fix a level h and let $v_1, \ldots, v_{N_h}$ be the internal nodes in level h ordered by the intervals they define. Note that the intervals are strictly non-overlapping since the nodes are from the same level. If v_i and v_{i+1} are siblings, then by Assumption 22, $w(v_i) \geq (1+\beta)w(v_{i+1})/2$. Otherwise, by monotonicity, $w(v_i) \geq w(v_{i+1})$. Since T_S is a full binary tree, every node in level h has a sibling in level h, so for at least half the i's in $[I_h]$, $w(v_i) \geq (1+\beta)w(v_{i+1})/2$. Furthermore, since $1/\log^3 n \leq w(v) \leq 1$,

$$N_h \leq 3\log_{(1+\beta)/2}\log n = \frac{3\log\log n}{(\log(1+\beta)) - 1}.$$

■

Now, we are ready to complete our proof.

Theorem 25 *For $\gamma > 1$, there is an algorithm that approximates the entropy of a monotone distribution on $[n]$ in $\mathcal{D}_{\Omega(\gamma^{5/2}/(\log(\sqrt{\gamma}+1)-1)(\sqrt{\gamma}-1))}$ to within a multiplicative factor of γ with probability at least $3/4$ in*

$$O\left(\frac{\gamma^{5/2}\log^6 n}{(\sqrt{\gamma}-1)^2(\log(\sqrt{\gamma}+1)-1)}\right) \quad \textit{time.}$$

Proof. Let Assumption 22 hold. By Lemma 23, we can assume that

$$\frac{H_I(\mathbf{p})}{\beta} - 2\beta w_{\mathbf{p}}(I) \leq \alpha(I,\beta) \leq \beta^2 H_I(\mathbf{p})$$

for each interval I that is associated with a balanced leaf and each light leaf has weight at most $\beta^2/\log^3 n$. Since the total weight of the intervals associated with light leaves is at most

$$\frac{3\beta^2\log\log n}{(\log(\beta+1)-1)\log^2 n},$$

their combined entropy contribution is

$$O\left(\frac{\gamma^2\log\log n}{(\log(\sqrt{\gamma}+1)-1)\log n}\right).$$

Let $B = \cup_{I\in\mathcal{I}(T_S)} I$. Since the algorithm's output is a at $\gamma H_B(\mathbf{p})$, it is clearly at most $\gamma H(\mathbf{p})$. We can show the other direction as follows.

$$\sum_{I\in\mathcal{I}(T_S)} \alpha(I,\beta) \geq \frac{H_B(\mathbf{p})}{\beta} - 2\beta \geq \frac{H(\mathbf{p}) - O\left(\frac{\gamma^2\log\log n}{(\log(\sqrt{\gamma}+1)-1)\log n}\right)}{\beta} - 2\beta \geq \frac{H(\mathbf{p})}{\beta^2}$$

when $H(\mathbf{p}) = \Omega(\gamma^{5/2}/(\log(\sqrt{\gamma}+1)-1)(\sqrt{\gamma}-1))$.

Consider $2n$ intervals that correspond to the nodes of a complete tree T. By Lemma 21, Assumption 22 fails to hold for a particular interval of T with probability $O(1/n^2)$. Hence, Assumption 22 fails to hold for T_S with probability $O(1/n)$ by the union bound on all the nodes. Therefore, the error probability of the algorithm is $o(1)$. The running time of the algorithm is the sample size times the size of T_S. ■

Note that the lower bound shown in Theorem 6 applies to monotone distributions. Therefore, a restriction on the entropy such as the one in the statement of Theorem 25 is necessary.

7 Subset-uniform distributions

Consider subset-uniform distributions $\mathcal{E}_k$ which are uniform over some subset $K \subset [n]$ with $|K| = k$. The entropy of this class of distributions is $\log k$. If we approximate k to within a multiplicative factor of γ, then we get a very strong additive approximation to $\log k$. Now, given a generation oracle access to a distribution which is promised to be from $\mathcal{E}_k$ for some k, the entropy estimation problem reduces to approximating k.

Theorem 26 *For any $\gamma > 1$, there exists an algorithm in the generation oracle model that, for every k and for any distribution $\mathbf{p} \in \mathcal{E}_k$, outputs ℓ such that $k/\gamma \le \ell \le \gamma k$ with probability at least $3/4$ in $O(\gamma^2\sqrt{k}/(\gamma-1)^2)$ time.*

Proof. Our algorithm, inspired by the algorithm of [4], is as follows. Let $t = 256\gamma^3/(\gamma-1)^4$. Let m be the total number of samples taken until we see t pairwise collisions. The algorithm outputs $\binom{m}{2}/t$.

For $i < j < m$, define ζ_{ij} be the indicator random variable that indicates when the i-th and j-th samples are the same element. Then, $\zeta^{(m)} \stackrel{\text{def}}{=} \sum_{1\le i<j\le m} \zeta_{ij}$ is the number of collisions. Clearly, $\mathrm{E}\left[\zeta_{ij}\right] = 1/k$ and so $E_m \stackrel{\text{def}}{=} \mathrm{E}\left[\zeta^{(m)}\right] = \binom{m}{2}/k$. It can be shown that $\mathrm{Var}\left[\zeta^{(m)}\right] \le 2(\binom{m}{2}/k)^{3/2} = 2E_m^{3/2}$. Now, we will show that our algorithm produces a reliable estimate by upper bounding the probabilities of an overestimate and an underestimate. These computations are straightforward applications of Chebyshev's inequality.

Consider the probability that the algorithm outputs a value more then γk; this implies $\binom{m}{2}/t \ge \gamma k$. We will show that the probability that the number of collisions in m samples is less than t is at most $1/8$ by upper bounding $\Pr\left[\zeta^{(m)} < t\right]$. Applying Chebyshev's inequality,

$$\Pr\left[\zeta^{(m)} < t\right] \le \Pr\left[\left|\zeta^{(m)} - E_m\right| \ge E_m - t\right] \le \frac{\mathrm{Var}\left[\zeta^{(m)}\right]}{(E_m - t)^2} \le \frac{2\gamma^2}{E_m^{1/2}(\gamma-1)^2} \le \frac{2\gamma^{3/2}}{t^{1/2}(\gamma-1)^2} \le \frac{1}{8},$$

since $t \ge 256\gamma^3/(\gamma-1)^4$.

Conversely, consider the probability that the algorithm outputs a value less then k/γ; this implies $\binom{m}{2}t^{-1} \le k/\gamma$. We will show that the probability that the number of collisions in m samples is more than t is at most $1/8$ by upper bounding $\Pr\left[\zeta^{(m)} \ge t\right]$. As before, applying Chebyshev's inequality,

$$\Pr\left[\zeta^{(m)} \ge t\right] \le \Pr\left[\left|\zeta^{(m)} - E_m\right| \ge t - E_m\right] \le \frac{\mathrm{Var}\left[\zeta^{(m)}\right]}{(t - E_m)^2} \le \frac{2\gamma^2 E_m^{3/2}}{t^2(\gamma-1)^2} \le \frac{2\gamma^{1/2}}{t^{1/2}(\gamma-1)^2} \le \frac{1}{8},$$

since $t > 256\gamma^3/(\gamma-1)^4$ and $\gamma > 1$.

So, total probability of error is at most $1/4$. Since $\binom{m}{2}t^{-1} \le k\gamma$, the number of samples (and hence the time) is $O(\sqrt{tk\gamma})$. ∎

Acknowledgment

We thank the referee for improving the presentation of the paper.

References

[1] T. Batu, E. Fischer, L. Fortnow, R. Kumar, R. Rubinfeld, and P. White. Testing random variables for independence and identity. *Proc. 42nd Annual Symposium on Foundations of Computer Science*, 2001.

[2] T. Batu, L. Fortnow, R. Rubinfeld, W. D. Smith, and P. White. Testing that distributions are close. *Proc. 41st Annual Symposium on Foundations of Computer Science*, pp. 259–269, 2000.

[3] W. Feller. *An Introduction to Probability Theory and Applications, I.* John Wiley & Sons Publishers, 1968.

[4] O. Goldreich and D. Ron. On testing expansion in bounded degree graphs. *ECCC*, TR00-020, 2000.

[5] O. Goldreich and S. Vadhan. Comparing entropies in statistical zero-knowledge with applications to the structure of SZK. *Proc. 14th IEEE Conf. on Computational Complexity*, pp. 54–73, 1999.

[6] B. Harris. The statistical estimation of entropy in the non-parametric case. *Colloquia Mathematica Societatis János Bolyai, Topics in Information Theory*, 16:323–355, 1975.

[7] M. Kearns, Y. Mansour, D. Ron, R. Rubinfeld, R. Schapire, and L. Sellie. On the learnability of discrete distributions. *Proc. 26th Annual ACM Symposium on Theory of Computing*, pp. 273–282, 1994.

[8] S.-K. Ma. Calculation of entropy from data of motion. *J. of Statistical Physics*, 26(2):221–240, 1981.

[9] S. P. Strong, R. Koberle, R. R. de Ruyter van Steveninck, and W. Bialek. Entropy and information in neural spike trains. *Phys. Rev. Lett.*, 80:197–200, 1998.

[10] D. Wolpert and D. R. Wolf. Estimating functions of probability distributions from a finite set of samples. Part I. Bayes estimators and the Shannon entropy. *Physical Review E*, 52(6):6841–6854, 1995.

A Proof of Lemma 11

In this section, we present the technical details of the proof of Lemma 11 that was outlined in Section 3.2.

Lemma 11 *For any $\gamma > 1$, sufficiently large n, and any canonical algorithm $\mathcal{A}$ using $o(n^{2/(5\gamma^2-2)})$ samples, there exist two distributions $\mathbf{p}$ and $\mathbf{q}$ over $[n]$ with $H(\mathbf{p})/H(\mathbf{q}) \geq \gamma^2$, such that $\mathcal{A}$ cannot distinguish between $\mathbf{p}$ and $\mathbf{q}$ with probability at least $2/3$.*

Proof. Let $\alpha \stackrel{\text{def}}{=} 2/(5\gamma^2 - 2)$ and $\beta \stackrel{\text{def}}{=} 3\alpha/2$ and let $\mathcal{A}$ be a canonical algorithm that uses $o(n^\alpha)$ samples. To prove this lemma, we define the distributions $\mathbf{p}$ and $\mathbf{q}$ as follows:

$$p_i \stackrel{\text{def}}{=} \begin{cases} n^{-\alpha} & 1 \leq i \leq n^\alpha/2 \\ n^{-1} & n/2 < i \leq n \\ 0 & \text{otherwise} \end{cases}$$

$$q_i \stackrel{\text{def}}{=} \begin{cases} n^{-\alpha} & 1 \leq i \leq n^\alpha/2 \\ n^{-\beta} & n/2 < i \leq (n + n^\beta)/2 \\ 0 & \text{otherwise} \end{cases}$$

Call i *big* if $1 \leq i \leq n^\alpha/2$ and *small* if otherwise and $p_i > 0$.

Note that $H(\mathbf{p}) = (\alpha+1)/2 \cdot \log n$ and $H(\mathbf{q}) = (\alpha+\beta)/2 \cdot \log n$. Therefore,

$$\frac{H(\mathbf{p})}{H(\mathbf{q})} = \frac{\alpha+1}{\alpha+\beta} = \left(\frac{5\gamma^2}{5\gamma^2-2}\right) \cdot \left(\frac{5\gamma^2-2}{5}\right) = \gamma^2.$$

Lemma 27 *The expected number of small elements that occur at least three times in the sample is $o(1)$ for both* $\mathbf{p}$ *and* $\mathbf{q}$.

Proof. For a fixed small element, the probability that at least three samples from $\mathbf{q}$ will be this element is $o(n^{3(\alpha-\beta)})$. Since there are $n^{-\beta}/2$ small elements, by the linearity of expectation, the expected number of such small elements in the sample is $o(1)$. ■

To simplify our argument, it would be useful if we could assume that the frequency of each element is independent of the frequencies of the other elements. To allow this, we assume that algorithm $\mathcal{A}$ first chooses an integer s_1 from the Poisson distribution with the mean $\lambda = s = o(n^{\alpha})$. The Poisson distribution with the mean λ has the probability mass function $p(k) = \exp(-\lambda)\lambda^k/k!$. Then, after taking s_1 samples from the input distribution, $\mathcal{A}$ decides whether to accept or reject the distribution. We will show that when $\mathcal{A}$ chooses s_1 this way, the frequencies of the elements are independent.

Next, we show that $\mathcal{A}$ cannot distinguish $\mathbf{p}$ from $\mathbf{q}$ with success probability at least 2/3. Since s_1 will have a value larger than $s/2$ with probability at least $1 - o(1)$ and we will show an upper bound on the statistical distance of the distributions of two random variables (i.e., the distributions on the fingerprints), it will follow that no symmetric algorithm with sample complexity $s/2$ can distinguish $\mathbf{p}$ from $\mathbf{q}$.

Let F_i be the random variable corresponding to the number of times the element i appears in the sample. The following fact is well-known (c.f., [3], pp. 216).

Fact 28 *For each i, F_i is distributed identically to the Poisson distribution with the mean $\lambda = sr$, where r is the probability of element i. Furthermore, all F_i's are mutually independent.*

Using this fact, the total number of samples from the big elements and the total number of samples from the small elements are independent.

Let $D_{\mathbf{p}}$ and $D_{\mathbf{q}}$ be the distributions on all possible fingerprints when samples are taken from $\mathbf{p}$ and $\mathbf{q}$, respectively. The rest of the proof proceeds as follows. We first construct two processes $D'_{\mathbf{p}}$ and $D'_{\mathbf{q}}$ that generate distributions on fingerprints such that $D'_{\mathbf{q}}$ is statistically close to $D_{\mathbf{p}}$ and $D'_{\mathbf{q}}$ is statistically close to $D_{\mathbf{q}}$. Then, we prove that the distributions $D'_{\mathbf{p}}$ and $D'_{\mathbf{q}}$ are statistically close. Hence, the theorem follows by the indistinguishability of $D_{\mathbf{p}}$ and $D_{\mathbf{q}}$.

Each process has two phases. The first phase is the same in both processes. They choose integers s' and s'' independently from the Poisson distribution with the mean $\lambda = s/2$. The integers s' and s'' fix the number of samples from the big and the small elements, respectively. By the properties of the Poisson distribution mentioned above, $s' + s''$ is distributed identically to s_1. Then, they randomly generate the frequency counts for each big element i using the random variables F_i defined above. The processes know which elements are big and which elements are small, although any distinguishing algorithm does not. At the end of the phase, the processes check whether the number of samples from the big elements is equal to s'. If not, they start generating the frequency counts from scratch. This concludes the first phase of the processes.

In the second phase, processes $D'_{\mathbf{p}}$ and $D'_{\mathbf{q}}$ determine the frequency counts of each small element according to $\mathbf{p}$ and $\mathbf{q}$, respectively (using F_i's). If any small element is given a total frequency count of at least three or the total number of samples generated in this phase is not exactly s'', the second phase of the process is restarted from scratch.

Since the frequency counts for all elements are determined at this point, both processes output the fingerprint of the sample they have generated.

Now, we treat the outputs of $D'_{\mathbf{p}}$ and $D'_{\mathbf{q}}$ as distributions on $[n]$. The following lemmas conclude the proof.

Lemma 29 $\left|D'_{\mathbf{p}} - D_{\mathbf{p}}\right| = o(1)$ *and* $\left|D'_{\mathbf{q}} - D_{\mathbf{q}}\right| = o(1)$.

Proof. The distribution that $D'_{\mathbf{p}}$ generates is the distribution $D_{\mathbf{p}}$ conditioned on the event that all small elements appear at most twice in the combined sample. Since this conditioning holds true with probability at least $1 - o(1)$ by Lemma 27, $|D'_{\mathbf{p}} - D_{\mathbf{p}}| \le o(1)$. A similar argument applies to $D'_{\mathbf{q}}$ and $D_{\mathbf{q}}$. ■

Lemma 30 $\left|D'_{\mathbf{p}} - D'_{\mathbf{q}}\right| \le 1/6$.

Proof. By the generation process, the L_1 distance between $D'_{\mathbf{p}}$ and $D'_{\mathbf{q}}$ can only arise from the second phase. We show that the second phases of the processes do not generate an L_1 distance larger than $1/6$.

Let G (respectively, H) be the random variable that corresponds to the values $d(2)$ when the input distribution is $\mathbf{p}$ (respectively, $\mathbf{q}$). Let $d' \stackrel{\text{def}}{=} \langle d(3), d(4), \ldots, d(s)\rangle$. We will use the fact that for any d', $\Pr\left[D'_{\mathbf{p}} \text{ generates } d', s', s''\right] = \Pr\left[D'_{\mathbf{q}} \text{ generates } d', s', s''\right]$ in the following calculation.

$$
\begin{aligned}
|D'_{\mathbf{p}} - D'_{\mathbf{q}}| &= \sum_{d} \left|\Pr\left[D'_{\mathbf{p}} \text{ generates } d\right] - \Pr\left[D'_{\mathbf{q}} \text{ generates } d\right]\right| \\
&= \sum_{d', s', s''} \Pr\left[D'_{\mathbf{p}} \text{ generates } d', s', s''\right] \cdot \\
&\qquad \sum_{k, l \ge 0} |\Pr\left[D'_{\mathbf{p}} \text{ generates } (d(1), d(2)) = (k, l) \;\mid\; d', s', s''\right] \\
&\qquad\qquad - \Pr\left[D'_{\mathbf{q}} \text{ generates } (d(1), d(2)) = (k, l) \;\mid\; d', s', s''\right]| \\
&= \sum_{d', s', s''} \Pr\left[D'_{\mathbf{p}} \text{ generates } d', s', s''\right] \cdot \\
&\qquad \sum_{k \ge 0} \left|\Pr\left[D'_{\mathbf{p}} \text{ generates } d(2) = k \;\mid\; d', s', s''\right] - \Pr\left[D'_{\mathbf{q}} \text{ generates } d(2) = k \;\mid\; d', s', s''\right]\right| \\
&= \sum_{k \ge 0} \left|\Pr\left[D'_{\mathbf{p}} \text{ generates } d(2) = k\right] - \Pr\left[D'_{\mathbf{q}} \text{ generates } d(2) = k\right]\right| \\
&= |G - H|,
\end{aligned}
$$

where the third equality follows since $s', s'', d', d(2)$ determine $d(1)$.

Consider the composition of G and H in terms of big and small elements. In the case of $\mathbf{p}$, let G_h be the number of big elements that contribute to $d(2)$ and G_l be the number of such small elements. Hence, $G = G_h + G_l$. Define H_h, H_l analogously. Then, G_h and H_h are distributed identically. In the rest of the proof, we show that the fluctuations in G_h, H_h dominate the magnitude of G_l, H_l.

Let ξ_i be the indicator random variable that takes value 1 when element i has been sampled twice. Then, $G_h = \sum_{\text{big } i} \xi_i$. By the assumption about the way samples are generated, the ξ_i's are independent. Therefore, G_h is distributed identically to the binomial distribution on the sum of n^α Bernoulli trials with success probability $\Pr[\xi_i = 1] = \exp(-s/n^\alpha)(s^2/2n^{2\alpha})$. An analogous argument shows that G_l is distributed identically to the binomial distribution with parameters $n/2$ and $\exp(-s/n)(s^2/2n^2)$. Similarly, H_l is distributed identically to the binomial distribution with parameters $n^\beta/2$ and $\exp(-s/n^\beta)(s^2/2n^{2\beta})$.

As n and m grow large enough, both G_h and G_l can be approximated well by normal distributions. Therefore, by the independence of G_h and G_l, G is also approximated well by a normal distribution. That is,

$$
\Pr[G = t] \to \frac{1}{\sqrt{2\pi}\sigma_G} \exp\left(-\frac{(t - \mathrm{E}[G])^2}{2\mathrm{Var}[G]}\right)
$$

as $n \to \infty$. Similarly, H is approximated well by a normal distribution.

Thus, $\Pr[G = t] = \Omega(1/\sigma_G)$ over an interval I_1 of length $\Omega(\sigma_G)$ centered at $\mathrm{E}[G]$. Similarly, $\Pr[H = t] = \Omega(1/\sigma_H)$ over an interval I_2 of length $\Omega(\sigma_H)$ centered at $\mathrm{E}[H]$. Since

$$\begin{aligned} \mathrm{E}[H] - \mathrm{E}[G] &= \mathrm{E}[H_l] - \mathrm{E}[G_l] \\ &\leq \mathrm{E}[H_l] \leq \exp\left(\frac{-s}{n^\beta}\right)\left(\frac{s^2}{4n^\beta}\right) = o(\sigma_H), \end{aligned}$$

$I_1 \cap I_2$ is an interval of length $\Omega(\sigma_{H_h})$. Therefore,

$$\sum_{t \in I_1 \cap I_2} |\Pr[G = t] - \Pr[H = t]| \leq o(1)$$

because for $t \in I_1 \cap I_2$, $|\Pr[G = t] - \Pr[H = t]| = o(1/\sigma_G)$. We can conclude that $\sum_t |\Pr[G = t] - \Pr[H = t]|$ is less than $1/6$ after accounting for the probability mass of G and H outside $I_1 \cap I_2$. ■

It can be seen that Lemma 11 follows from Lemma 29 and Lemma 30. ■