The complexity of approximating the entropy[*]

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

April 20, 2005

Abstract

We consider the problem of approximating the entropy of a discrete distribution under several different models of oracle access to the distribution. In the evaluation oracle model, the algorithm is given access to the explicit array of probabilities specifying the distribution. In this model, linear time in the size of the domain is both necessary and sufficient for approximating the entropy.

In the generation oracle model, the algorithm has access only to independent samples from the distribution. In this case, 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 this model does not permit a multiplicative approximation to the entropy in general. For the class of distributions to which our upper bound applies, we obtain a lower bound of $\Omega\left(n^{1/(2\gamma^2)}\right)$.

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 greater efficiency 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 such distributions, we also show a lower bound of $\Omega\left(\frac{\log n}{h(\gamma^2-1)+\gamma^2}\right)$.

Finally, we consider two special families of distributions: those in which the probabilities of the elements decrease monotonically with respect to a known ordering of the domain, 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, pages 678–687, 2002.

[†] School of Computing Science, Simon Fraser University, Burnaby, BC, Canada V5A 1S6. This work was supported by ONR N00014-97-1-0505, MURI. Phone: (604) 268 7115, Fax: (604) 291 3045, Email: batu@cs.sfu.ca.

[‡] University of California, San Diego 92093. 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: (845) 432 0398, Email: ravi@almaden.ibm.com.

[¶] CSAIL, MIT, Cambridge, MA 02139. Phone: (617) 253 0884, Fax: (617) 258 9738, Email: ronitt@csail.mit.edu.

1 Introduction

The Shannon entropy is a measure of the randomness of a distribution, and plays a central role in statistics, information theory, and coding theory. The entropy of a random source sheds light on the inherent 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 whose 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 entire representation can calculate the exact entropy. However, it is also easy to see that in this model, time linear 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 that approximates the entropy to within a multiplicative factor must distinguish between these two distributions, and a randomized algorithm for distinguishing 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] that draws samples from it. This model has been considered in both the statistics and physics communities (c.f., [6, 11, 8, 10]), though none of the previous work provides 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 that do not require superlinear (in the domain size) sample complexity are those presented in [8, 10]. 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 [10] 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 $\|\mathbf{p}\|_\infty$ of $\mathbf{p}$ 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,

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

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. 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 by different methods. On the other hand, we show that one cannot get a multiplicative approximation to the entropy in general. Furthermore, even for the class of distributions to which our upper bound applies, we obtain a lower bound of $\Omega(n^{1/(2\gamma^2)})$.

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 that runs in $\tilde{O}\left(n^{\frac{1+o(1)}{4}}\right)$ time, while our lower bound demonstrates that a running time of at least $\Omega(n^{1/8})$ is necessary.

(2) The evaluation oracle model: When the distribution is given as an evaluation oracle, we show a lower bound of $\Omega(n2^{-\gamma^2(h+1)})$ on the number of oracle accesses needed 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 that 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 $\Omega\left(\frac{\log n}{h(\gamma^2-1)+\gamma^2}\right)$ for this class of distributions. For example, to achieve a constant approximation for distributions with entropy $\Omega(h)$, our algorithm runs in time $O((1/h^2)\log^2 n)$ while our lower bound is $\Omega((1/h)\log n)$, that is, 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 probabilities decrease monotonically over some known ordering of the elements (i.e., $p_i \geq p_{i+1}$). We give an $O((1+\log^{-1}\gamma)\log n)$-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. 8	?
Generation oracle:	General	∞, Thm. 6	–
	High enough entropy	$\Omega\left(n^{1/(2\gamma^2)}\right)$, $H(\mathbf{p}) > \Omega((\log n)/\gamma^2)$, Thm. 7	$\tilde{O}(n^{1/\gamma^2})$, $H(\mathbf{p}) > \Omega(\gamma)$, Thm. 2
Combined oracle:	General	$\Omega(n^{(1-o(1))/\gamma^2})$, Thm. 12	?
	When $H(\mathbf{p}) \geq h$	$\Omega\left(\frac{\log n}{h(\gamma^2-1)+\gamma^2}\right)$, Thm. 13	$O\left(\frac{\gamma^2\log^2 n}{h^2(\gamma-1)^2}\right)$, Thm. 9

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 $Y = C(X)$ whose input X 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 papers give algorithms whose sample complexity is sublinear in the domain size along with 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 generation oracle model. Section 4 describes a lower bound for the evaluation oracle model, and Section 5 gives algorithms and lower bounds for the combined oracle case. Finally, in Sections 6 and 7, we give more efficient algorithms for two families of distributions.

2 Preliminaries

We consider discrete distributions over a domain of size n, which we denote by $[n] \stackrel{\text{def}}{=} \{1, \dots, n\}$. Let $\mathbf{p} = \langle p_1, \dots, p_n \rangle$ be such a distribution where $p_i \geq 0, \sum_{i=1}^n p_i = 1$. An algorithm is said to have *evaluation oracle* access to the distribution $\mathbf{p}$ if oracle query i is answered by p_i. An algorithm is said to have *generation oracle* access to $\mathbf{p}$ if it is given a source that draws samples independently from $\mathbf{p}$. An algorithm has *combined oracle* access to $\mathbf{p}$ if it has both evaluation and 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 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 $H_S(\mathbf{p}) + H_{[n] \setminus S}(\mathbf{p}) = H(\mathbf{p})$.

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

The following lemma summarizes some upper and lower bounds on entropy that will turn out to be useful at many points in the paper.

Lemma 1 *Pick any $S \subseteq [n]$.*

- *a. The partial entropy $H_S(\mathbf{p})$ is maximized when $w_{\mathbf{p}}(S)$ is spread uniformly over $|S|$:*

$$H_S(\mathbf{p}) \;\le\; w_{\mathbf{p}}(S) \cdot \log(|S|/w_{\mathbf{p}}(S)) \;\le\; w_{\mathbf{p}}(S) \cdot \log|S| + (\log e)/e.$$

- *b. Suppose there is some $\beta \le 1/e$ such that $p_i \le \beta$ for all $i \in S$. Then $H_S(\mathbf{p}) \le \beta|S|\log(1/\beta)$.*
- *c. Suppose there is some β such $\beta \le p_i \le 1/e$ for all $i \in S$. Then $H_S(\mathbf{p}) \ge \beta|S|\log(1/\beta)$.*
- *d. Suppose there is some β such that $p_i \le \beta$ for all $i \in S$. Then $H_S(\mathbf{p}) \ge w_{\mathbf{p}}(S)\log(1/\beta)$.*

Proof. Statement (a) follows from the concavity of the logarithm function. By Jensen's inequality,

$$\frac{1}{w_{\mathbf{p}}(S)} \cdot H_S(\mathbf{p}) \;=\; \sum_{i\in S} \frac{p_i}{w_{\mathbf{p}}(S)} \log\frac{1}{p_i} \;\le\; \log\left(\sum_{i\in S}\frac{p_i}{w_{\mathbf{p}}(S)} \cdot \frac{1}{p_i}\right) \;=\; \log\frac{|S|}{w_{\mathbf{p}}(S)}.$$

Therefore,

$$H_S(\mathbf{p}) \;\le\; w_{\mathbf{p}}(S) \cdot \log\frac{|S|}{w_{\mathbf{p}}(S)} \;\le\; w_{\mathbf{p}}(S) \cdot \log|S| + \frac{\log e}{e}.$$

The last inequality comes from observing that the function $x\log(1/x)$ is zero at $x = 0, 1$ and has a single local maximum in the interval $[0, 1]$, at $x = 1/e$.

Statement (b) follows immediately from the previous observation about $x\log(1/x)$, which implies that under the given constraint, $H_S(\mathbf{p})$ is maximized by setting all the p_i to β.

The proof of Statement (c) also follows from the concavity of $x\log(1/x)$; under the given constraints, $p_i \log(1/p_i)$ is minimized when $p_i = \beta$.

For statement (d), we notice that $H_S(\mathbf{p})$ is strictly concave: therefore, over any closed domain, it is minimized at a boundary point. In particular, when the domain is $[0, \beta]^{|S|}$, the minimum point must have some coordinate with $p_i = 0$ or $p_i = \beta$. We can now restrict attention to the remaining coordinates and apply the same argument again. In this way, we find that the minimum is realized when $w_{\mathbf{p}}(S)/\beta$ of the p_i are β, and the rest are zero. ■

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 $\mathbf{p} \in \mathcal{D}$, given oracle access of type $\mathcal{O}$ to $\mathbf{p}$, algorithm $\mathcal{A}$ outputs a value $\mathcal{A}(\mathbf{p})$ such that $H(\mathbf{p})/\gamma \le \mathcal{A}(\mathbf{p}) \le \gamma \cdot H(\mathbf{p})$ with probability at least $3/4$. (This probability of success can generically be increased to $1 - \delta$ by running the algorithm $\log(1/\delta)$ times and returning 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 $\mathcal{D}_h$ 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 2 *For every $\gamma > 1$ and every ϵ_o such that $0 < \epsilon_o \leq 1/2$, there exists an algorithm in the generation oracle model that runs in time $O((n^{\frac{1}{\gamma^2}}/\epsilon_o^2) \cdot \log n)$, and with success probability at least $3/4$, returns a $(1+2\epsilon_o)\gamma$-approximation to the entropy of any distribution on $[n]$ in $\mathcal{D}_{4\gamma/(\epsilon_o(1-2\epsilon_o))}$.*

Given any $\eta > 0$ and $\gamma' > 1$, one can set $\gamma = \gamma'/(1+2\epsilon_o)$ above and choose ϵ_o small enough to yield a γ'-approximation algorithm with running time $O(n^{(1+\eta)/\gamma'^2} \log n)$, for distributions of entropy $\Omega(\gamma/\eta)$. 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 either big or small, depending on their probability mass. For a fixed $\alpha > 0$ and a distribution $\mathbf{p}$, the set of indices with high probabilities (the *big* elements) is defined as:

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

We then approximate the contributions of the big and small elements to the entropy 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 2.

3.1.1 Approximating the entropy of the big elements

To estimate the amount by which the big elements contribute to the entropy, we approximate each of their probabilities by drawing samples from the generation oracle.

Lemma 3 *For every α such that $0 < \alpha \leq 1$, every ϵ_o such that $0 < \epsilon_o \leq 1/2$, 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 a distribution $\mathbf{q}$ over $[n]$ such that with probability at least $1 - n^{-1}$, the following hold for all i:*

1. *if $p_i \geq \frac{1-\epsilon_o}{1+\epsilon_o} n^{-\alpha}$ (in particular this holds 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 $m = (18 n^\alpha/\epsilon_o^2) \cdot \log 2n$. Fix any $i \in [n]$ and define X_j to be the indicator variable that the j-th sample is i. Let $q_i = \sum X_j/m$, the average of independent, identically distributed Boolean random variables. If $p_i \geq \frac{1-\epsilon_o}{1+\epsilon_o} n^{-\alpha}$, then by Chernoff bounds

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

Moving onto smaller elements, we again can use Chernoff bounds to show that if $p_i < \frac{1-\epsilon_o}{1+\epsilon_o} n^{-\alpha}$, then

$$\begin{aligned} \Pr\left[q_i > (1-\epsilon_o)n^{-\alpha}\right] &= \Pr\left[q_i - p_i > (1-\epsilon_o)n^{-\alpha} - p_i\right] \\ &\leq \Pr\left[q_i - p_i > \frac{\epsilon_o(1-\epsilon_o)}{1+\epsilon_o} n^{-\alpha}\right] \\ &\leq \exp\left(-\left(\frac{\epsilon_o(1-\epsilon_o)n^{-\alpha}}{(1+\epsilon_o)p_i}\right)^2 \cdot \frac{p_i m}{3}\right) \end{aligned}$$

$$
\begin{aligned}
&\leq \exp\left(-\left(\frac{1-\epsilon_o}{1+\epsilon_o}\right)^2 \cdot \frac{6n^{-\alpha}}{p_i} \cdot \log 2n\right)\\
&\leq \exp(-2\log 2n) \leq 1/2n^2.
\end{aligned}
$$

Statements (1) and (2) of the lemma follow from a union bound over all i. ■

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

Lemma 4 *Pick any $B \subseteq [n]$. Let $\epsilon_o \in (0,1)$ be chosen so that each $i \in B$ satisfies $|p_i - q_i| \leq \epsilon_o p_i$. Then,*

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

Proof. For $i \in B$, write $q_i = (1+\varepsilon_i)p_i$. We know 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,

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

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

We now estimate the entropy contribution of the small elements. Let S be any subset of small elements, that is, $S \subseteq [n] \setminus B_\alpha(\mathbf{p})$.

If $w_{\mathbf{p}}(S) \leq n^{-\alpha}$, then the contribution of S to the entropy is below any constant and can be ignored for approximation purposes. So, we may assume without loss of generality that $w_{\mathbf{p}}(S) \geq n^{-\alpha}$. Let $\hat{w}(S)$ be the empirical estimate of the probability mass of S, in other words, the number of samples from S divided by the total number of samples. The following lemma bounds the accuracy of this estimate.

Lemma 5 *If $S \subseteq [n]$ satisfies $w_{\mathbf{p}}(S) \geq n^{-\alpha}$ and if $m = O((n^\alpha/\epsilon_o^2)\log n)$ samples are drawn from $\mathbf{p}$, then with probability at least $1 - n^{-1}$, the empirical estimate $\hat{w}(S)$ satisfies $(1-\epsilon_o)\cdot w_{\mathbf{p}}(S) \leq \hat{w}(S) \leq (1+\epsilon_o)\cdot w_{\mathbf{p}}(S)$. Moreover, if $w_{\mathbf{p}}(S) < n^{-\alpha}$, then $\hat{w}(S) < (1+\epsilon_o)n^{-\alpha}$.*

Proof. Let X_j be the indicator random variable that takes value 1 when the j-th sample lies in S, and let $X = \sum X_j$. Then X is $m\hat{w}(S)$ and it has expected value $\mathrm{E}[X] = m \cdot w_{\mathbf{p}}(S)$. The lemma follows by Chernoff bounds, and by the fact that $w_{\mathbf{p}}(S) \geq n^{-\alpha}$. Similar to the proof of Lemma 3, we can show that if $w_{\mathbf{p}}(S) < n^{-\alpha}$, then $\hat{w}(S) < (1+\epsilon_o)n^{-\alpha}$. ■

Since $p_i < n^{-\alpha}$ for $i \in S$, by Lemma 1(a,d), we have that

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

Hence, using estimate $\hat{w}(S)$ for $w_{\mathbf{p}}(S)$, we get an approximation to $H_S(\mathbf{p})$.

3.1.3 Putting it together

In this section we describe our approximation algorithm for $H(\mathbf{p})$ and prove Theorem 2. The following is our algorithm for obtaining a γ-approximation to the entropy:

Algorithm ApproximateEntropy(γ, ϵ_o)

```
1. Set $\alpha = 1/\gamma^2$.
2. Get $m = O((n^\alpha/\epsilon_o^2) \cdot \log n)$ samples from $\mathbf{p}$.
3. Let $\mathbf{q}$ be the empirical probabilities of the $n$ elements; that is, $q_i$ is the frequency of $i$ in the sample divided by $m$. Let $B = \{i \mid q_i > (1-\epsilon_o)n^{-\alpha}\}$.
4. Output $H_B(\mathbf{q}) + \frac{w_{\mathbf{q}}([n]\setminus B)\log n}{\gamma}$.
```

Notice that the set B is an empirically-determined substitute for $B_\alpha(\mathbf{p})$. We now prove that this algorithm satisfies Theorem 2.

Proof. (of Theorem 2) First of all, Lemma 3 assures us that with probability at least $1 - 1/n$, two conditions hold: (1) $B_\alpha(\mathbf{p}) \subseteq B$, and (2) every element $i \in B$ satisfies $|p_i - q_i| \leq \epsilon_o p_i$. For the rest of the proof, we will assume that these conditions hold.

Let $S = [n] \setminus B$. Assume for the moment that $w_{\mathbf{p}}(S) \geq n^{-\alpha}$. In this case, we know from Lemma 5 that with high probability, $|w_{\mathbf{q}}(S) - w_{\mathbf{p}}(S)| \leq \epsilon_o w_{\mathbf{p}}(S)$. Lemma 1(a,d) tells us that

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

Then by Lemma 4,

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

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

$$H_B(\mathbf{q}) + \frac{w_{\mathbf{q}}(S)\log n}{\gamma} \geq (1-\epsilon_o) \cdot H_B(\mathbf{p}) - 2\epsilon_o + \frac{1-\epsilon_o}{\gamma} \cdot w_{\mathbf{p}}(S) \log n$$

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

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

It remains to handle the case when $w_{\mathbf{p}}(S)$ is less than $n^{-\alpha}$. Lemma 5 tells us that $w_{\mathbf{q}}(S)$ is with high probability at most $(1+\epsilon_o)n^{-\alpha}$. Therefore, our estimate of the entropy from small elements, $(w_{\mathbf{q}}(S)\log n)/\gamma$, lies somewhere between zero and $((1+\epsilon_o)n^{-\alpha}\log n)/\gamma$. For any γ bounded away from one, this is only a negligible contribution to $H(\mathbf{p})$, well within the approximation bound. ■

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 exhibiting pairs of distributions that have very different entropies, with ratio at least γ^2, and yet are hard to distinguish given only a few samples.

3.2.1 Impossibility of approximating entropy in general

First we show that there is no algorithm for computing entropy that can guarantee a bounded approximation factor for all input distributions. The basic problem is that no amount of sampling can conclusively establish that a distribution has zero entropy.

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

Proof. Let $\mathcal{A}$ be any algorithm for computing entropy, and let $t(n)$ be an upper bound on its running time on distributions over $[n]$. Consider the two distributions $\mathbf{p} = \langle 1, 0, \ldots, 0\rangle$ and $\mathbf{q} = \langle 1 - 1/(100t(n)), 1/(100t(n)), 0, \ldots, 0\rangle$. Notice that $\mathbf{p}$ has zero entropy while $\mathbf{q}$ has positive entropy.

Suppose we run $\mathcal{A}$ on either $\mathbf{p}$ or $\mathbf{q}$. Since it uses at most $t(n)$ samples, its oracle calls will almost always (99% of the time) produce a succession of identical elements, regardless of whether the underlying distribution is $\mathbf{p}$ or $\mathbf{q}$. In such cases, if $\mathcal{A}$ guesses that the entropy is zero, its approximation factor on $\mathbf{q}$ will be unbounded, whereas if it guesses a positive number, its approximation factor on $\mathbf{p}$ will be unbounded. ■

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

The following theorem shows a lower bound on the number of samples required to approximate the entropy of distributions with high entropy.

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

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 randomly chosen subset $S \subseteq [n]$ 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 1/2, we will not see any repetitions if we take $n^{1/2\gamma^2}$ samples from either distribution. In such cases, the samples from $\mathbf{p}$ and $\mathbf{q}$ look identical. Thus at least $\Omega(n^{1/2\gamma^2})$ samples are needed to distinguish these distributions. ■

Recently, Ron [9] showed a lower bound of $\tilde{\Omega}(n^{2/(6\gamma^2-3)})$ for approximating the entropy. This is better than the above lower bound when $\gamma < \sqrt{3/2}$. Her proof also exhibits two distributions with entropy ratio γ^2 and shows that the two distributions are indistinguishable unless $\tilde{\Omega}(n^{2/(6\gamma^2-3)})$ samples are taken.

4 The evaluation oracle model: a lower bound

In the introduction, we mentioned that for general distributions over $[n]$, a linear number of queries is necessary to approximate the entropy in the evaluation oracle model. Since there are only n possible queries, the complexity of entropy approximation in this model is settled. Next, we study the number of queries needed when a lower bound on the entropy of the distribution can be assumed.

Theorem 8 *Let $\gamma > 1, h > 0$, and n be sufficiently large. If an algorithm $\mathcal{A}$ that operates in the evaluation oracle model achieves a γ-approximation to the entropy of distributions over $[n]$ in $\mathcal{D}_h$, then it must make $\Omega(n2^{-\gamma^2(h+1)})$ queries.*

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

Let R be a subset of $[n]$ of size $2^{\gamma^2(h+1)}$, chosen uniformly at random. Distribution $\mathbf{p}$ is defined to be uniform over R. Let S also be a uniform-random subset of $[n]$, but of smaller size $\beta \cdot 2^{\gamma^2(h+1)}$, where $\beta = 1/(\gamma^2(h+1)/h)$. In addition, pick s randomly from $[n] \setminus S$. Distribution $\mathbf{q}$ assigns probability $2^{-\gamma^2(h+1)}$ to each element in S and assigns the rest of the probability mass, namely $1-\beta$, to s.

Both these distributions belong to $\mathcal{D}_h$: $H(\mathbf{p}) = \gamma^2(h+1)$ and $H(\mathbf{q})$ is between h and $h+1$ (to see this, notice $H_S(\mathbf{q}) = h$). The ratio between their entropies is $H(\mathbf{p})/H(\mathbf{q}) \geq \gamma^2$.

In the evaluation oracle, any algorithm that distinguishes between $\mathbf{p}$ and $\mathbf{q}$ must (on at least one of these two inputs) discover some location $i \in [n]$ with nonzero probability. The number of queries required is therefore 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 in which an algorithm is given both evaluation and generation oracle access to the same distribution.

5.1 Upper bound

The entropy of a distribution $\mathbf{p}$ can be viewed as the expected value of $-\log p_i$, where i is distributed according to $\mathbf{p}$. This suggests an algorithm:

1. Draw m samples from the generation oracle (m to be defined later). Call these $i_1, \ldots, i_m$.

2. For each i_j, ask the evaluation oracle for p_{i_j}.

3. Return $-\frac{1}{m}\sum_{j=1}^{m}\log p_{i_j}$.

As we will now see, if $H(\mathbf{p})$ is not too small this algorithm needs only a polylogarithmic number of queries in order to return a good approximation.

Theorem 9 *Pick any $\gamma > 1$ and any $h > 0$. If the above algorithm is run with $m = O\left(\frac{\gamma^2 \log^2 n}{h^2(\gamma-1)^2}\right)$, then it returns a γ-approximation to the entropy of any distribution over $[n]$ in $\mathcal{D}_h$, with success probability at least $3/4$.*

Proof. Let $m \overset{\text{def}}{=} \frac{3\gamma^2 \log^2 n}{h^2(\gamma-1)^2}$. Define the random variable $X_j \overset{\text{def}}{=} -\log p_{i_j}$ for $j = 1, \ldots, m$, and let $X = (1/m)\sum_j X_j$ be the final answer returned. Clearly, $\mathrm{E}[X] = \mathrm{E}[X_j] = H(\mathbf{p})$. All that needs to be shown is that the variance of X is not too large. Since the X_j's are independent, it will suffice to bound the variance of an individual X_j.

Lemma 10 $\mathrm{Var}[X_j] \leq \log^2 n$.

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 $\sum_i f(p_i)$ is a symmetric concave function when its domain is limited to $\mathbf{p} \in (0, 1/e)^n$, and, as in Lemma 1, is maximized (on this domain) when $\mathbf{p}$ is uniform. This maximum value is $\log^2 n$.

To finish the proof, we need to show that we cannot attain higher values of $\sum_i f(p_i)$ by looking at $\mathbf{p} \notin (0, 1/e)^n$. To this end, suppose $p_j \geq e^{-1}$ for some j. Then 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. Using simple calculus, and the fact that $n \geq 3$, it is easy to check that $f'(p_k) > f'(p_j)$. Hence, we can increase the sum by simultaneously decreasing p_j and increasing p_k. By combining with the argument above, we conclude that $\sum_i f(p_i) \leq \log^2 n$.

■

Since the X_j's are identical and independent, $\mathrm{Var}[X] = \mathrm{Var}[X_j]/m \leq (\log^2 n)/m$.

To bound the error probability of our algorithm, we now use Chebyshev's inequality, which states that for any $\rho > 0$,

$$\Pr[|X - \mathrm{E}[X]| \geq \rho] \leq \mathrm{Var}[X]/\rho^2.$$

We get

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

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

Corollary 11 *There exists an algorithm $\mathcal{A}$ in the combined oracle model that γ-approximates $H(\mathbf{p})$ in $O((\frac{\gamma}{\gamma-1})^2)$ time, for distributions with $H(\mathbf{p}) = \Omega(\log n)$.*

5.2 Lower bounds

This next theorem gives a lower bound for the combined oracle model when the entropy of the distribution is allowed to be very small, so small that for instance the previous upper bound does not apply.

Theorem 12 *Pick any $\gamma > 1$ and any sufficiently large n. Then any algorithm in the combined oracle model that γ-approximates the entropy of distributions over $[n]$ (with non-zero entropy) must make $\Omega(n^{1/\gamma^2})$ oracle calls.*

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

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

Note that, by the concavity of $f(x) = -x \log x$ for $0 \le \delta < 1$, and that $f'(1) = -\log e$, we have that $-(1-\delta)\log(1-\delta) \le \delta \log e$. Hence, a quick calculation shows that $H(\mathbf{p}) = -(1-n^{-\alpha})\log(1-n^{-\alpha}) + n^{-\alpha}\log n^{\alpha} \le n^{-\alpha}(\log e + \alpha \log n)$ and $H(\mathbf{q}) > n^{-\alpha}\log n$. By the choice of α, $H(\mathbf{q})/H(\mathbf{p}) > \gamma^2$.

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. To finish, notice that $n^{\alpha} = e^{-1}n^{1/\gamma^2}$. ■

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 12.

Theorem 13 *Pick any $\gamma > 1$, any $h > 0$, and any sufficiently large n. Then any algorithm in the combined oracle model that γ-approximates the entropy of distributions over $[n]$ in $\mathcal{D}_h$ must make $\Omega(\log n/(h(\gamma^2 - 1) + 2\gamma^2))$ oracle calls.*

Proof. Let $w = (h(\gamma^2 - 1) + 2\gamma^2)/\log n$ and $k \overset{\text{def}}{=} \lceil 2^{h/(1-w)} \rceil$. Consider the following distributions $\mathbf{p}$ and $\mathbf{q}$:

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

Note that $H(\mathbf{p}) = (1-w)\log\frac{k}{1-w} - w\log w = (1-w)\log k - (1-w)\log(1-w) - w\log w$. Hence, $h \le H(\mathbf{p}) \le h + 2$. Similarly, $H(\mathbf{q}) > h + w\log n$.

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 $o(1/w)$ samples and making

$o(1/w)$ probes will fail to distinguish between a randomly chosen member of $\mathcal{P}$ and a randomly chosen member of $\mathcal{Q}$ with high probability.

Meanwhile, by the choice of w, the entropy ratio is

$$\frac{H(\mathbf{q})}{H(\mathbf{p})} > \frac{h + w \log n}{h+2} = \frac{h\gamma^2 + 2\gamma^2}{h+2} = \gamma^2.$$

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 14 *For any $\gamma > 1$, there is an algorithm in the evaluation oracle model that γ-approximates the entropy of monotone distributions on $[n]$ in $\mathcal{D}_{\Omega(\gamma^2/(\sqrt{\gamma}-1))}$, and runs in $O(\lceil 1/\log\gamma \rceil \log n)$ time.*

Proof. Algorithm $\mathcal{A}$ partitions the domain $[n]$ into intervals, and only queries the endpoints of each interval. The remaining probability values are interpolated from these queries.

The partition of $[n]$ is constructed recursively, starting with a single *active* interval $[1, n]$:

While there is some active interval $[\ell, u]$:

- Make it inactive.
- If $p_\ell > n^{-2}$ and $p_\ell / p_u > \gamma$, then split $[\ell, u]$ into two equal-sized active subintervals.

Any required probability values (i.e., p_ℓ, p_u at each iteration) are obtained from the oracle. At the end of the procedure, the algorithm has probabilities for a particular sequence of elements $1 = i_o \leq i_1 \leq \cdots \leq i_k = n$, such that for each $j < k$, either $p_{i_j} \leq n^{-2}$ or $p_{i_j}/p_{i_{j+1}} \leq \gamma$. The splitting criteria ensure that the total number of queries $k+1$ is roughly logarithmic in the number of elements; more precisely, $k \leq 1 + (1 + 4/\log\gamma)\log n$.

The algorithm then approximates $\mathbf{p}$ by a distribution $\mathbf{q}$ that is interpolated from the handful of p_i values that it queries:

- For each i_j, set $q_{i_j} = p_{i_j}$.
- For any $i \in (i_j, i_{j+1})$: if $p_{i_j} \leq n^{-2}$ then set $q_i = 0$; otherwise set $q_i = \sqrt{p_{i_j} p_{i_{j+1}}}$.

Let B_0 denote the elements whose probabilities get set to zero, and let $B = [n] \backslash B_0$ be the remaining elements. We know that for $i \in B_0$, $p_i \leq n^{-2}$. Thus, $w_{\mathbf{p}}(B_0) \leq n^{-1}$ and so by Lemma 1(b), B_0 doesn't contribute much to the entropy: $H_{B_0}(\mathbf{p}) \leq 2n^{-1}\log n$. We therefore need to focus on B.

For each $i \in B$, define $c_i \overset{\text{def}}{=} q_i/p_i$. Since the endpoints of the interval containing i have probabilities that are within a multiplicative factor γ of each other, it follows that $\frac{1}{\sqrt{\gamma}} \leq c_i \leq \sqrt{\gamma}$. This means that $H_B(\mathbf{q})$ is not too different from $H_B(\mathbf{p})$:

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

when $H(p) \geq \log e/(e(\gamma - \sqrt{\gamma}))$. The first inequality follows from the fact (see Lemma 1) that $-x\log x \leq (\log e)/e$ for all $x \in (0,1)$. Similarly,

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

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

The algorithm outputs $H(\mathbf{q}) = H_B(\mathbf{q})$, which we've shown is a γ-approximation to $H(\mathbf{p})$. ■

6.2 The generation oracle model

We show that the entropy of a monotone distribution can also be approximated in polylogarithmic time in the generation oracle model. Our algorithm rests upon the following observation that is formally stated in Lemma 15: 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 when either (1) the probability masses of two halves are very close or (2) 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 summing the contributions from each leaf, assuming that the conditional distribution within a leaf (that is, the distribution restricted to the leaf's range) is uniform. 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 $i<j$. For an interval I, let S_I denote the set of samples that fall 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
- $|S_{[i,\lfloor(i+j)/2\rfloor]}| \le \beta |S_{[\lfloor(i+j)/2\rfloor+1,j]}|$ (call v *balanced*).

Otherwise, we split v's interval by attaching two children to v, corresponding to the intervals $[i, \lfloor(i+j)/2\rfloor]$ (the left child) and $[\lfloor(i+j)/2\rfloor + 1, j]$ (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 $\alpha(I, \beta)$ that approximates the entropy in the balanced interval I:

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

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 $\sum_{I\in\mathcal{I}(T_S)} \alpha(I, \beta)$.
```

Overview of the proof.

The main steps in the proof are the following. First, we give a key lemma on 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 15). 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 16, 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, taking the sampling error into account (Lemma 18). 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 deal with the light intervals, in particular to show two properties: (1) the number of such intervals is approximately logarithmic in the size of the domain (Lemma 19) and (2) their total entropy contribution is negligible and thus can be ignored. In order to do this, we prove an interesting and non-trivial property of the tree T_S: at any level, it contains at most $O(\log \log n)$ nodes. Thereafter, (1) and (2) follow easily.

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 15 *Let I be an interval of length $2k$ in $[n]$, let I_1 and I_2 be the bisection of I, and let $\mathbf{p}$ 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

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

Notice in particular that the ratio of the upper bound to the lower bound is at most $w_{\mathbf{p}}(I)/2w_{\mathbf{p}}(I_2)$.

Proof. The upper bound follows from Lemma 1(a): the partial entropies $H_{I_1}(\mathbf{p}), H_{I_2}(\mathbf{p})$ are maximized when their weights are spread uniformly over their constituent elements.

Let $w_1 \overset{\text{def}}{=} w_{\mathbf{p}}(I_1)$ and $w_2 \overset{\text{def}}{=} w_{\mathbf{p}}(I_2)$. 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. A lower bound on $H_{I_1}(\mathbf{p})$ is given by Lemma 1(c) (plug in w_2/k for as many elements as possible), and for $H_{I_2}(\mathbf{p})$ it follows immediately from $p_i \leq w_1/k$ for $i \in I_2$. Combining, 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 bisection I_1, I_2, the error in the entropy estimate depends upon the ratio $w_{\mathbf{p}}(I)/2w_{\mathbf{p}}(I_2)$. This 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 16 *Let I be an interval in $[1, n]$ such that $w_{\mathbf{p}}(I) \geq \log^{-3} n$ and I_1, I_2 a bisection of I. Let S be a sample set of size $m = O((\beta^5 \log^4 n)/(\beta - 1)^2)$ drawn from $\mathbf{p}$. For $\beta > 1$,*

1. *with probability at least $1 - n^{-2}$, $(1/\beta) \cdot m \cdot w_{\mathbf{p}}(I) \leq |S_I| \leq \beta \cdot m \cdot w_{\mathbf{p}}(I)$;*
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 straightforward application of multiplicative Chernoff bounds. The random variable $|S_I|$ is the sum of m independent Bernoulli trials, each with success probability $w_{\mathbf{p}}(I)$. Therefore $\mathrm{E}\left[|S_I|\right] = mw_{\mathbf{p}}(I)$, and by the choice of m in the algorithm, the probability that $|S_I|$ deviates from its expectation by more than a multiplicative factor of β is at most $1/n^2$.

From Part 1, we know that with probability at least $1 - n^{-2}$, $|S_I| \geq mw_{\mathbf{p}}(I)/\beta$. Fix any $t \geq mw_{\mathbf{p}}(I)/\beta$. 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 Chernoff bounds. Note that $(t - Y)/Y < \beta$ implies $Y > t/(\beta + 1)$. Since $\mathrm{E}\left[Y\right] \leq t/(2\beta)$, we get

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

Conditioned on the event that $t \geq m w_{\mathbf{p}}(I)/\beta$, 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. ■

There are various events that we would like to count upon: for instance, that for balanced intervals I, the ratio of the weights of the two halves is at most $2\beta - 1$; and that intervals associated with two sibling nodes have weight ratio at least $(1+\beta)/2$. Lemma 16 tells us that these events hold with high probability. We now package all of them into a single assumption.

Assumption 17 *(1) For each interval I corresponding to a balanced node of the tree, $|S_I|$ lies in the range $[(1/\beta)\cdot m \cdot w_{\mathbf{p}}(I), \beta \cdot m \cdot w_{\mathbf{p}}(I)]$; (2) for each interval I we decide to split, $w_{\mathbf{p}}(I_1)/w_{\mathbf{p}}(I_2) \geq (1+\beta)/2$; (3) for each balanced interval I, we have $w_{\mathbf{p}}(I_1)/w_{\mathbf{p}}(I_2) \leq 2\beta - 1$; and (4) 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 $\mathcal{I}(T_S)$ is the set of all balanced intervals in T_S.

Lemma 18 *Under Assumption 17, 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 17, $|S_I|/(m\beta) \leq w_{\mathbf{p}}(I) \leq |S_I|\beta/m$ and $w_{\mathbf{p}}(I_1)/w_{\mathbf{p}}(I_2) \leq 2\beta - 1$. These imply that the upper and lower bounds for $H_I(\mathbf{p})$ given in Lemma 15 are within a multiplicative factor β of one another. Therefore our entropy estimate $\alpha(I,\beta)$ is not too far from $H_I(\mathbf{p})$:

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

The second inequality above is a simple consequence of $w_{\mathbf{p}}(I) = w_{\mathbf{p}}(I_1) + w_{\mathbf{p}}(I_2)$, and the expression on that line is exactly (β times) the upper bound of Lemma 15. Similarly, for the other direction,

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

The second inequality follows from the concavity of $\log x$. ■

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

Lemma 19 *Under Assumption 17, given $\beta > 1$, the number of nodes in T_S is at most*

$$\frac{12 \log n \log\log n}{\log(\beta+1) - 1}.$$

Proof. For any given level of the tree, let $v_1, \ldots, v_{2k}$ denote the internal (that is, non-leaf) nodes at that level, ordered by the intervals they define. There is an even number of these nodes because they each have a sibling at the same level. If v_i, v_{i+1} are siblings, we know from Assumption 17 that $w(v_i) \geq w(v_{i+1}) \cdot (1+\beta)/2$. And in general, by monotonicity, $w(v_i) \geq w(v_{i+1})$. Therefore, as one moves from v_1 to v_{2k}, the weight $w(v_i)$ drops by a factor of at least $(1+\beta)/2$ for every two nodes. Moreover these weights never drop below $1/\log^3 n$, by the split criterion and Assumption 17. It follows that

$$k \;\leq\; \frac{3 \log\log n}{\log(1+\beta) - 1}.$$

We now have a bound on the number of internal nodes at any level. To finish the lemma, we observe that there are at most $\log n$ levels, that the total number of nodes (internal and leaf) is twice the number of internal nodes plus one, and that we have overcounted by at least one at the root level. ■

Now, we are ready to complete our proof.

Theorem 20 *For every $\gamma > 1$, there is an algorithm that approximates the entropy of a monotone distribution on $[n]$ in $\mathcal{D}_{(6\gamma^{3/2}/(\log(\sqrt{\gamma}+1)-1)(\sqrt{\gamma}-1))}$ 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. Suppose Assumption 17 holds; we will come back and address this later. Let's start by handling the leaves. By Assumption 17, each light leaf has weight at most $\beta^2/\log^3 n$, and so by Lemma 19, the total weight of the intervals associated with light leaves is at most

$$\frac{6\beta^2 \log\log n}{(\log(\beta+1)-1)\log^2 n}.$$

Therefore, their combined entropy contribution is at most $\log n$ times this,

$$\frac{6\gamma \log\log n}{(\log(\sqrt{\gamma}+1)-1)\log n}$$

(recall $\beta^2 = \gamma$), which will turn out to be negligible for our purposes.

Now we move on to the internal nodes. By Lemma 18,

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

for each interval I associated with a balanced leaf. Let $B = \cup_{I \in \mathcal{I}(T_S)} I$. The algorithm's output is:

$$\sum_{I \in \mathcal{I}(T_S)} \alpha(I,\beta) \;\leq\; \sum_{I \in \mathcal{I}(T_S)} \beta^2 \cdot H_I(\mathbf{p}) \;=\; \gamma \cdot H_B(\mathbf{p}) \;\leq\; \gamma \cdot H(\mathbf{p}).$$

We can show the other direction as follows.

$$\sum_{I\in\mathcal{I}(T_S)} \alpha(I,\beta) \geq \frac{H_B(\mathbf{p})}{\beta} - 2\beta \geq \frac{H(\mathbf{p}) - \frac{6\gamma \log\log n}{(\log(\sqrt{\gamma}+1)-1)\log n}}{\beta} - 2\beta \geq \frac{H(\mathbf{p})}{\beta^2}$$

when $H(\mathbf{p}) \geq (6\gamma^{3/2}/(\log(\sqrt{\gamma}+1)-1)(\sqrt{\gamma}-1))$.

We now proceed to justify Assumption 17. Consider the $2n$ intervals that correspond to the nodes of a complete tree T. By Lemma 16, Assumption 17 fails to hold for a particular interval of T with probability $O(1/n^2)$. Hence, Assumption 17 fails to hold for T_S with probability $O(1/n)$ by the union bound over all the intervals. 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 20 is necessary.

7 Subset-uniform distributions

Consider the family of distributions $\mathcal{E}_k$ that are uniform over some subset $K \subseteq [n]$ 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 that is promised to be from $\mathcal{E}_k$ for some k, the entropy estimation problem reduces to approximating k.

Theorem 21 *For every $\gamma > 1$, 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 $k/\gamma \leq \ell \leq \gamma k$ with probability at least $3/4$ in $O(\gamma\sqrt{k}/(\gamma-1))$ time.*

Proof. Our algorithm, inspired by [4], is as follows.

- Let $c = 16\gamma/(\gamma-1)^2$.
- Draw samples until at least c pairwise collisions are observed.
- If M is the number of samples seen, output $\binom{M}{2}/c$.

Note that M is a random variable.

To analyze this algorithm, pick any integer m, and suppose that m samples are drawn from the distribution. For $i < j$, let X_{ij} be an indicator random variable denoting a collision between the i^{th} and j^{th} samples seen. Let $S_m = \sum_{i<j} X_{ij}$ be the total number of collisions.

For any $i < j$, $\mathrm{E}[X_{ij}] = 1/k$; therefore $\mathrm{E}[S_m] = \binom{m}{2} \cdot 1/k$. This motivates the algorithm above. To bound the chance of failure, we also need the variance of S_m. Notice that

$$\mathrm{E}\left[S_m^2\right] \;=\; \mathrm{E}\left[\left(\sum_{i<j} X_{ij}\right)\left(\sum_{a<b} X_{ab}\right)\right] \;=\; \sum_{i<j,\; a<b} \mathrm{E}\left[X_{ij}X_{ab}\right].$$

In the final summation, the various terms can be segregated according to the cardinality of the set $\{i,j,a,b\}$. If this set has cardinality 3 or 4, then $\mathrm{E}[X_{ij}X_{ab}] = 1/k^2$. If the set has cardinality

2, then $\mathrm{E}[X_{ij}X_{ab}] = 1/k$. This last possibility occurs for exactly $\binom{m}{2}$ of the $\binom{m}{2}^2$ terms in the summation. Therefore

$$\mathrm{E}\left[S_m^2\right] \;=\; \left(\binom{m}{2}^2 - \binom{m}{2}\right)\frac{1}{k^2} \;+\; \binom{m}{2}\frac{1}{k},$$

whereupon $\mathrm{Var}\left[S_m\right] = \mathrm{E}\left[S_m^2\right] - \mathrm{E}\left[S_m\right]^2 = \binom{m}{2}(1/k - 1/k^2) \leq \mathrm{E}\left[S_m\right]$.

What is the chance that the algorithm outputs a number less than k/γ? Let m_0 be the largest integer m such that $\binom{m}{2} < ck/\gamma$.

$$\Pr\left[\text{Output is} < k/\gamma\right] \;=\; \Pr\left[M \leq m_0\right] \;=\; \Pr\left[S_{m_0} \geq c\right] \;\leq\; \Pr\left[\left|S_{m_0} - \mathrm{E}\left[S_{m_0}\right]\right| \geq (c - \mathrm{E}\left[S_{m_0}\right])\right]$$

This last probability can be bounded by Chebyshev's inequality, giving

$$\Pr\left[\text{Output is} < k/\gamma\right] \;\leq\; \frac{\mathrm{Var}\left[S_{m_0}\right]}{(c - \mathrm{E}\left[S_{m_0}\right])^2} \;\leq\; \frac{\mathrm{E}\left[S_{m_0}\right]}{(c - \mathrm{E}\left[S_{m_0}\right])^2} \;\leq\; \frac{\gamma}{c(\gamma-1)^2} \;\leq\; \frac{1}{16},$$

where the last two inequalities follow from $\mathrm{E}\left[S_{m_0}\right] < c/\gamma$, and from the particular choice of c.

To bound that chance that the output is more than $k\gamma$, we proceed similarly, letting m_0 denote the smallest integer m for which $\binom{m+1}{2} > c\gamma k$. Then

$$\Pr\left[\text{Output is} > k\gamma\right] \;=\; \Pr\left[M > m_0\right] \;=\; \Pr\left[S_{m_0} < c\right] \;\leq\; \Pr\left[\left|S_{m_0} - \mathrm{E}\left[S_{m_0}\right]\right| \geq (\mathrm{E}\left[S_{m_0}\right] - c)\right]$$

Again using Chebyshev's inequality, we get

$$\Pr\left[\text{Output is} > k\gamma\right] \;\leq\; \frac{\mathrm{Var}\left[S_{m_0}\right]}{(\mathrm{E}\left[S_{m_0}\right] - c)^2} \;\leq\; \frac{\mathrm{E}\left[S_{m_0}\right]}{(\mathrm{E}\left[S_{m_0}\right] - c)^2} \;\leq\; \frac{3\gamma}{c(\gamma-1)^2} \;\leq\; \frac{3}{16}.$$

The total probability of error is therefore at most 1/4. When the algorithm succeeds, $\binom{M}{2}/c \leq k\gamma$, and so the number of samples (and the running time) is $O(\sqrt{ck\gamma})$. ■

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*, pages 442–451, 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] D. Ron. Unpublished manuscript, 2005.

[10] 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.

[11] 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.