6.896 Sublinear Time Algorithms	February 27, 2007
Lecture 7	
Lecturer: Ronitt Rubinfeld	*Scribe: Brendan Juba*

1 Recap

Recall from last time that we were considering the boolean hypercube, $\{\pm 1\}^n$. We had defined the partial ordering, for $x, y \in \{\pm 1\}^n$, $x \leq y$ if $\forall i\ x_i \leq y_i$. We said that a probability distribution p was monotone over $\{\pm 1\}^n$ if $\forall x \leq y\ p_x \leq p_y$. Recall also that we'd defined $\mathsf{bias}(X) = \sum_i X_i$.

Additive Chernoff Bound (Hoeffding)

For $X_1, \ldots, X_m$ i.i.d. random variables with range $[-a, a]$, $\hat{\mu} = \frac{1}{m}\sum_{i=1}^m X_i$, and $\mu = E[\hat{\mu}]$, $\forall \rho > 0$ $\Pr[|\hat{\mu} - \mu| > \rho] \leq 2\exp\left(-\frac{\rho^2}{2a^2}m\right)$.

We were in the middle of the analysis of the following algorithm:

Algorithm: Test Uniform

1. Pick $s = \Theta\left(\frac{n}{\varepsilon^2}\log\frac{n}{\varepsilon}\right)$ samples $X^{(1)}, \ldots, X^{(s)} \in p$.
2. If any $X^{(i)}$ has $|\mathsf{bias}(X^{(i)})| > \sqrt{2n\log(20s)}$, stop and output 'nonuniform.'
3. Let $\bar{\mu} = \frac{1}{s}\sum_{i=1}^s \mathsf{bias}(X^{(i)})$.
4. If $\bar{\mu} \leq \frac{\varepsilon}{4}$, output 'uniform.' Otherwise, output 'nonuniform.'

2 Analysis of Uniformity Test

Last time, we saw the analysis of the case where $p = U_D$. Furthermore, in the case where $|p - U_D| > \varepsilon$, we proved the following claim:

Claim 1 *If $|p - U_D| > \varepsilon$, then $E_p[\mathsf{bias}(X)] \geq \varepsilon$.*

Today we will complete the analysis of this second case, which is broken into two subcases. We wish to show that in both subcases, the algorithm outputs 'nonuniform' with probability at least 2/3.

Case 2a: $\Pr_p[|\mathsf{bias}(X)| > \sqrt{2n\log(20s)}] \geq \frac{10}{s}$

Notice that on any single sample $X^{(i)}$,

$$\Pr[\text{step 2 does not output nonuniform on } X^{(i)}] \leq 1 - \frac{10}{s}$$

where, since we have s independent samples,

$$\Pr[\text{step 2 does not output nonuniform}] \leq \left(1 - \frac{10}{s}\right)^s < \frac{1}{3}$$

(note that this probability is actually quite small, but 1/3 is sufficient)

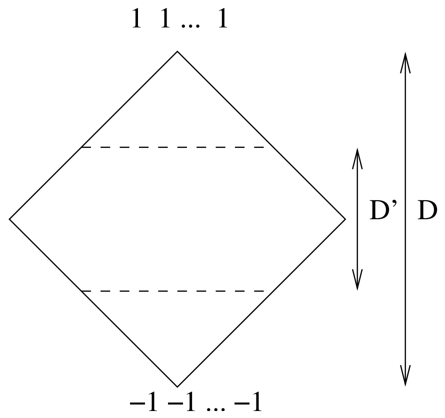

Figure 1: The set D': D without the "tails" of the distribution.

Case 2b: $\Pr_p[|\mathsf{bias}(X)| > \sqrt{2n\log(20s)}] < \frac{10}{s}$

In this case, we might stop in step 2 (and if we do, that's great for us), but it will suffice to show that step 4 will output nonuniform with sufficiently high probability otherwise.

Notice that if step 2 passes, each i satisfies $|\mathsf{bias}(X^{(i)})| \leq \sqrt{2n\log(20s)}$, so our samples are from a distribution that has been conditioned on this event: letting $D = \{\pm 1\}^n$ be our sample space, the event we are concerned with is $D' = \{x \in D : |\mathsf{bias}(X)| \leq \sqrt{2n\log(20s)}\}$. If we take $p' = p|_{D'}$, then after step 2, we know we have a sample drawn from p'.

Of course, since passing step 2 suggests that the probability of obtaining a sample from $D \setminus D'$ is very small, we anticipate that p' is close to p. We will make this precise by bounding $E_{X\in p}[|\mathsf{bias}(X)|] - E_{X\in p'}[|\mathsf{bias}(X)|]$:

Claim 2 *In this case, for sufficiently large n, $E_{X\in p}[|\mathsf{bias}(X)|] - E_{X\in p'}[|\mathsf{bias}(X)|] \leq \varepsilon/8$*

Proof It is easy to see that

$$E_{X\in p}[|\mathsf{bias}(X)|] = E_{X\in p'}[|\mathsf{bias}(X)|] \cdot \Pr[X \in D'] + E_{x\notin p'}[|\mathsf{bias}(X)|] \cdot \Pr[X \notin D']$$

Let $\alpha = 10/s$. By assumption (in case 2b) $\Pr[X \in D'] \geq 1 - \alpha$ and $\Pr[X \notin D'] \leq \alpha$. It is also immediate that

$$E_{x\in p'}[|\mathsf{bias}(X)|] \leq \sqrt{2n\log(20s)}$$

Outside D', on the other hand, we can't say anything remarkable about the bias. The best bound we get follows from the fact that for any X, $-n \leq \mathsf{bias}(X) \leq n$: $E_{x\notin p'}[|\mathsf{bias}(X)|] \leq n$.

Using these bounds in the expression above, we obtain

$$\begin{aligned}
E_{X\in p}[|\mathsf{bias}(X)|] - E_{X\in p'}[|\mathsf{bias}(X)|] &\leq \alpha \cdot E_{X\in p'}[|\mathsf{bias}(X)|] + E_{X\notin p'}[|\mathsf{bias}(X)|] \cdot \alpha \\
&\leq \alpha\sqrt{2n\log(20s)} + n\alpha \\
&\leq 2\alpha n
\end{aligned}$$

where the last part follows since $s = \Theta\left(\frac{n}{\varepsilon^2}\log\frac{n}{\varepsilon}\right)$. Moreover, this gives us that for sufficiently large n, $2\alpha n = \frac{20n}{s} = \Theta\left(\frac{\varepsilon^2}{\log(n/\varepsilon)}\right) \leq \varepsilon/8$. ■

Since, more generally, this is case 2, we are considering distributions with $E_p[\mathsf{bias}(X)] \geq \varepsilon$; we now see that after step 2 of the algorithm, we still have samples from a distribution p' with $E_{p'}[\mathsf{bias}(X)] \geq \varepsilon - \varepsilon/8 = \frac{7}{8}\varepsilon$.

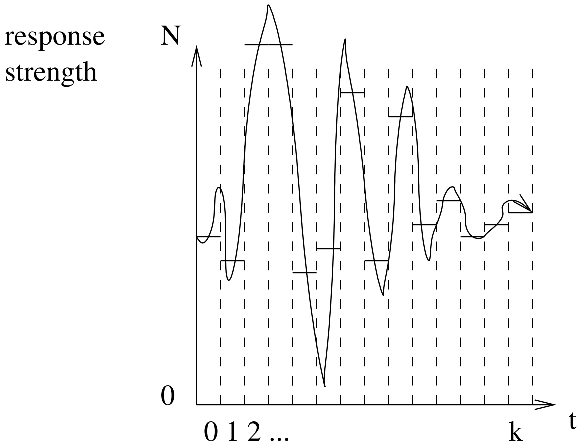

Figure 2: A hypothetical sample of neuron spike data, with discretization overlaid.

We'll show that the probability of failing to output 'nonuniform' in step 4 is sufficiently low, which will complete the analysis. We'll use the bounded range Chernoff bound with $\rho = \varepsilon/8$, $a = \sqrt{2n\log(20s)}$, and $n = s = \Theta\left(\frac{n}{\varepsilon^2}\log\frac{n}{\varepsilon}\right)$, i.e., by our previous claim,

$$\begin{aligned}
\Pr_{p'}\left[\bar{\mu} - \frac{7}{8}\varepsilon \leq -\frac{1}{8}\varepsilon\right] &\leq \Pr_{p'}[|\bar{\mu} - \mathsf{bias}(X)| \geq \varepsilon/8] \\
&\leq 2\exp\left(-\frac{\varepsilon^2}{2\cdot 8^2 \cdot 2n\log(20s)}s\right) \\
&= 2\exp\left(-\frac{\varepsilon^2}{256n\log(20s)}C\frac{n}{\varepsilon^2}\log\frac{n}{\varepsilon}\right) \\
&= 2\exp\left(-\frac{C}{256}\cdot\frac{\log(n/\varepsilon)}{\log(n/\varepsilon)+\log(C\varepsilon^{-1}\log(n/\varepsilon))}\right)
\end{aligned}$$

which can be made less than $1/5$ for sufficiently large n and a suitable choice of C, since the fraction involving n approaches 1 as n grows large.

3 Estimating the entropy of a distribution given via samples

3.1 Motivation

The entropy of a distribution is such a basic, natural quantity that people in a wide variety of fields, including learning, coding theory, physics, etc., would like to be able to calculate it. For example, at NIPS there was a workshop[1] where theoretical computer scientists, statisticians, physicists, neuroscientists and other experimentalists discussed the algorithmic and conceptual questions related to unbiased estimation of entropies and similar information-theoretic quantities in regimes where not enough samples are available. For example, a neuroscientist examining data collected from the response strengths of a particular neuron while a subject looked at sticks might want to know if the neuron was responding to the stick. To do this, the data would be discretized by breaking the responses over time into k intervals, and assigning each interval one of N values that approximates the average response strength of that interval. Thus, in one trial, the data is taken to be a symbol from an alphabet of size N^k. The problem is then to estimate the entropy of this (rather large) discretized distribution.

[1]See http://www.menem.com/ ilya/pages/NIPS03/

We have other reasons to be interested in the entropy of a distribution, too. Suppose that we are receiving symbols of an alphabet from some source, and we wish to know the information content of the source. Specifically, while we know that symbols from an alphabet of size N can be encoded in $\log N$ bits each, we may be able to do better if we use a Huffman coding of these symbols—where, in this case, the entropy of the source tells us what how compressible it is.

3.2 Multiplicative estimates for entropy

Definition 3 (Entropy) *The (Shannon) entropy of a distribution p over D is $H(p) = -\sum_{x\in D} p_x \log p_x$*

We observe a few facts about entropy. First, notice that $H(U_n) = \log n$. Second, consider the distribution q which has $q_1 = 1$ and $\forall i > 1$ $q_i = 0$; clearly, $H(q) = 0$. Now, for some $t > 0$, consider the distribution q^t given by $q^t_1 = 1 - 1/t$, $q^t_2 = 1/t$, and $\forall i > 2$ $q^t_i = 0$. It is easy to see that $H(q^t) > 0$.

However, as $t \to \infty$, we see that $H(q^t)$ is quite small:

$$H(q^t) = (1 + \log t)\frac{1}{t} + O(t^{-2})$$

This is significant for the following reason: suppose we wanted to obtain a γ-multiplicative estimate of the entropy (for $\gamma > 1$), even allowing a linear number of samples from the distribution. That is, on input p we would like our algorithm to output some value y satisfying

$$\frac{1}{\gamma}H(p) \le y \le \gamma H(p)$$

with probability at least 2/3.

This is impossible: On input q, clearly the only value of y that is satisfactory is $y = 0$. But suppose our input is q^t for a sufficiently large value of t; the output of the algorithm must be strictly positive. However, in n samples, we only expect to see $x \ne 1$ with probability at most n/t, so if (say) $t = 6n$, with probability 1/6 we only see $x = 1$. Now consider an algorithm which approximates the entropy. If, when it sees only outputs that are 1, it outputs 0, then it will give an incorrect answer to probability distribution q^t with probability at least $\frac{2}{3} - \frac{1}{6} = \frac{1}{2}$, where clearly $0 < H(q^t)/\gamma$ for any γ. Thus, we do not output a γ-multiplicative estimate for q^t with probability greater than 1/3, so the algorithm cannot be correct. On the other hand, if the algorithm outputs a number that is strictly positive when it sees only outputs that are 1, it will be wrong with probability one on input q.

So we see that we cannot obtain good multiplicative estimates for distributions with low entropy. It turns out though, that if we have a lower bound on the entropy, we can give a test. (We'll actually more generally give a test that has some additive and some multiplicative error.)

First, some more notation: given a set $S \subset D$, we write $H_S(p) = \sum_{x\in S} -p_x \log p_x$. We also consider the weight of S, $w_S(p) = \sum_{x\in S} p_x$. Finally, we let D_ρ denote the distributions with entropy at least ρ.

We'll see that we can "sort of" use the collision probabilities (i.e., as in the L_2-distance test) for the test. First, observe that since the logarithm is a concave function, by Jensen's inequality we have

$$\log ||p||_2^2 = \log \sum_x p_x^2 \ge \sum_x \log p_x^2 \ge \sum_x p_x \log p_x = -H(p)$$

(recalling $\log ||p||_2^2$ is negative)

So we see that the collision probabilities give us a lower bound on the entropy. We'd also like a lower bound, and we'll see that under some additional conditions, we can use the collision probabilities for this too. Specifically, we will assume that $\max_x p_x = ||p||_\infty \le n^{-\alpha}$. Now, notice that $\sum_x p_x^2 \le \max_x p_x \sum_x p_x = ||p||_\infty \cdot 1$. Thus, we have

$$\log ||p||_2^2 \le \log ||p||_\infty \le \log n^{-\alpha} = -\alpha \log n \le -\alpha H(p)$$

So, in summary, given that $||p||_\infty \le n^{-\alpha}$,

$$-\frac{1}{\alpha}\log ||p||_2^2 \ge H(p) \ge -\log ||p||_2^2$$

So estimating the collision probabilities could be used to give a reasonable estimate of the entropy for some distributions. However, we are going to do something even easier: we'll only need to estimate the total weight of the distribution.

Now, we would have expected to see an algorithm that approximates the entropy to distributions from D_ρ to within a γ-factor in some time such as $\tilde{O}(n^{1/\gamma^2})$, for some ρ and γ. What we actually achieve also depends on a third parameter, ε_0:

Theorem 4 *For any $\gamma > 1$, $0 < \varepsilon_0 < 1/2$, we can approximate the entropy for any distribution in $D_{\frac{4\gamma}{\varepsilon_0(1-2\varepsilon_0)}}$ to within a $(1+2\varepsilon_0)\cdot\gamma$-factor in $O\left(\frac{n^{1/\gamma^2}\log n}{\varepsilon_0^2}\right)$ time.*

We can think of this as having some constant factor in the running time and some constant blow-up of the approximation factor. We should notice also that we've only ruled out the ability to estimate the entropy of distributions with constant entropy.

The algorithm that we give should be contrasted with the naive algorithm or "plug-in estimate" (as it's referred to by statisticians): we sample the distribution enough times to obtain some value $\hat{p}_x$ approximating p_x, and use this value to compute $H(\hat{p})$ "by definition."

Algorithm

On input γ, ε_0,

1. Take $m = O\left(\frac{n^{1/\gamma^2}\log n}{\varepsilon_0^2}\right)$ samples of our distribution q.
2. $\hat{q} \leftarrow$ the normalized frequencies of the sample
3. $B \leftarrow \{i : \hat{q}_i > (1-\varepsilon_0)n^{-1/\gamma^2}\}$
4. Output $H_B(\hat{q}) + w_{([n]\setminus B)}(\hat{q})(\log n)\frac{1}{\gamma}$

i.e., we use the plug-in estimate for the heavy elements, and approximate the light elements with the uniform distribution, with some $1/\gamma$ factor "safety net."

The proof proceeds roughly as follows: let $B_{1/\gamma^2}(q) = \{x : q_x \ge n^{1/\gamma^2}\}$ be our set of "big elements." In the algorithm, we approximate this set by B; we've included some "slack" in our choice of B, so the elements that should be in B (i.e., those in B_{1/γ^2}) end up in B with very high probabilitiy. We can also show that no "really small" elements will end up in B. Thus, Chernoff bounds will tell us that the elements in B have their probabilities approximated well, in a manner similar to the analysis of the closeness tests. Therefore, we find that $H_B(\hat{q})$ is a good estimate of $H_B(q)$.

We still need to examine why the other term is a good estimate for the contributions of the rest of the elements. To this end, we have the following lemma:

Lemma 5 (Main small element lemma) *For $S = [n] \setminus B$,*

$$\frac{1}{\gamma^2} w_S(p)\log n \le H_S(p) \le w_S(p)\log n + \frac{\log e}{e}$$

(so we can safely output the geometric mean: $\frac{1}{\gamma} w_S(p)\log n$ is within a γ-factor of $H_S(p)$)

Proof Recall that we say that a function is symmetric if any two permutations on the order of the arguments in any input yield the same output, and that a function is concave if $\forall \alpha, p, q\ \alpha f(p) + (1 -$

$\alpha)f(q) \leq f(\alpha p + (1-\alpha)q)$. We know that (when the sum of the inputs is constrained) such functions are maximized when all of its inputs are equal. In this case, at $p_i = \frac{w_S(p)}{|S|}$ and so

$$H_S(p) \leq w_S(p) \log\left(\frac{|S|}{w_S(p)}\right) \leq w_S(p) \log |S| - w_S(p) \log w_S(p)$$

where it can be shown that $\max_{x \in [0,1]} x \log \frac{1}{x} = \frac{\log e}{e}$. Therefore, $H_S(p) \leq w_S(p) \log n + \frac{\log e}{e}$.

We also know that such functions are minimized when their inputs take extremal points. Recall now that we had assumed that the maximum probability was bounded by n^{1/γ^2}—so, the function is maximized when we assign weight n^{1/γ^2} to as many inputs as possible, some other input gets the remaining weight, and the rest of the inputs receive weight 0. Assume wlog that all elements received weight n^{1/γ^2}. Now notice that for this choice of inputs,

$$H_S(p) = (n^{1/\gamma^2} w_S(p))(-n^{1/\gamma^2} \log n^{1/\gamma^2}) = \frac{1}{\gamma^2} w_S(p) \log n$$

■

So we assume that every element in S has probability at most n^{1/γ^2}, which we know should hold by an earlier Chernoff bound argument—otherwise, these elements would be in B instead. In a complete proof, only verifying that $w_S(\hat{q})$ is close enough to $w_S(q)$ would remain, which only involves additional calculations and applications of Chernoff bounds.