6.896 Sublinear Time Algorithms March 9th, 2017

Lecture 8

Lecturer: Ronitt Rubinfeld *Scribe: Anne Kelley*

1 Introduction

In the last lecture we gave an algorithm for uniformity testing which, given samples from a distribution, estimated the collision probability to decide whether it was likely to be uniform. We noted that that our analysis was not tight and in this lecture we will show a corresponding lower bound. As with many proofs of lower bounds, we'll give an example of a distribution and argue that it cannot be distinguished from uniform without using at least a minimum number of observations. Again, we'll make use of *collision probabilities*, and we'll also use of the idea of *mutual information*, which is a technique somewhat recently borrowed from Information Theory and published for the first time in this context in Spring 2016. We'll also need to use *Poissonization*, which is an older technique which allows us to avoid assumptions of independence when sampling. To begin, recall the following definition:

Definition 1 *Uniformity Tester*
Given samples from a distribution p on $[n], \epsilon$:

1. *If $p = U_{[n]}$ output PASS with probability $\geq \frac{3}{4}$*
2. *If $\|p - U_{[n]}\|_1 > \epsilon$ output FAIL with probability $\geq \frac{3}{4}$*

Today we will ultimately prove the following lower bound which is our main theorem:

Theorem 2 *Any uniformity tester requires $\Omega(\frac{\sqrt{n}}{\epsilon^2})$ samples*

2 Background Facts

Let X and Y be random variables. In particular, we'll have the random variable X be the input distribution to our algorithm which is randomly chosen to either be uniform or far from uniform and let the random variable Y be the output of the algorithm, either PASS or FAIL.

Recall the following definitions and facts from Information Theory:

1. Entropy $H(X) = -\sum_x p(x) \log p(x)$
   (a) $H(X) \geq 0$
2. Conditional Entropy $H(Y|X) = \sum_x p(x) \big[\sum_{ys.t.p(y) \neq 0} p(y|x) \log \frac{1}{p(y|x)} \big]$
   Intuitively the conditional entropy tells us whether x and y are correlated.
   (a) $H(Y|X) = 0$ if and only if Y is completely determined by X. In other words the algorithm is either always right or always wrong.
   (b) $H(Y|X) = H(Y)$ if and only if Y is independent of X.
   (c) $H(Y|X) \leq H(Y)$
   Intuitively this is because knowing the value of X can provide additional information and decrease the entropy but not take away information we already have about Y.
3. Chain Rule for Entropy $H(X, Y) = H(X) + H(Y|X)$
4. Mutual Information $I(X, Y) = H(X) + H(Y) - H(X, Y) = H(X) - H(X|Y) = H(Y) - H(Y|X)$
   The second two equalities follow from the chain rule. Intuitively mutual information tells us how much X allows us to predict Y.
5. Chain Rule for Mutual Information $I(X; (Y, Z)) = I(X; Z) + I(X; Y|Z)$

3 Lower Bound Proof

3.1 Constructing a Worst Case Distribution

Since we want to prove that no uniformity tester can exist that makes less than $\Omega(\frac{\sqrt{n}}{\epsilon^2})$ queries, we present a particular distribution that requires at least this many queries regardless of algorithm to decide whether it is close to uniform. We construct this distribution according to the following procedure, where we let S be a distribution uniform on $\frac{n}{2}$ elements (so each element has probability $\frac{2}{n}$) chosen from the subset $[n]$.

Algorithm 1 Construct Distribution

```
$X \leftarrow$ Flip a fair coin
if heads then
    return $k$ samples from distribution $\epsilon$-close to $U_{[n]}$
return $k$ samples from $S$

```

Note that if the coin flip produces tails, then the l_1 distance of the resulting distribution is one, which is comparatively large. A uniformity tester, however, will never get to see X, only the samples returned from the distribution.

3.2 Lower Bound Assuming Independence

For simplicity, we first assume independence between samples drawn from the distribution. Of course, **this assumption is false** and we will remove it in the next section. Our goal is to show that if we choose k, the number of samples, to be too small then $I(X, Y) = o(1)$. That is, the mutual information between X and Y is smaller than any constant and you have no information to figure out which case, heads or tails, you're in since the set of collisions you're likely to see are the same regardless. Combined with the contrapositive of the following lemma which we accept without proof, this then shows that no such algorithm can exist as a uniformity tester.

Lemma 3 *If f is any function (i.e. a deterministic algorithm to distinguish the distributions) such that $P_{x,samples}(f(samples) = x) \geq 51\%$ then the mutual information $\geq 2 * 10^{-4}$.*

Now we solve for the required value of k to force $I(X, Y) = o(1)$.

Let a_i be the number of times that the element i appears in the sample of size k. By definition we know that

$$I(x, samples) = I(x, \{a_i\}_{i=1}^n)$$

Using our assumption of independence between samples and the chain rule we have

$$I(x, \{a_i\}) = \sum_{i=1}^{n} I(x, a_i)$$

Every element in $[n]$ has an equal probability of being included in the sample k since the distribution is either uniform or S where they each have a $\frac{n}{2}$ chance of being included in the distribution S and then an equal chance of being chosen for the sample. Therefore, $I(x, a_i)$ is the same for all i. Given the fact that $I(x, a_1) = O(\frac{k^2}{n^2})$, which we take without proof,

$$I(x, a_i) = nI(x, a_1) = O(\frac{k^2}{n})$$

Choosing $k = o(\sqrt{n})$ then gives $I(x, y) = o(1)$ as desired.

3.3 Lower Bound Without Independence Assumption

Now we want to remove the assumption of independence between samples. To do this, we use a technique called Poissonization. The key idea behind this technique is that instead of drawing a fixed number of samples, we pull a random number from a Poisson distribution and take a corresponding number of samples. The intuition is that before knowing what some of the samples were changed the probabilities associated with other elements being represented since there were less spots left in the sample they could possibly occupy. Now the unknown number of samples prevents the probabilities associated with other elements from decreasing. Recall, the following information about a Poisson distributon:

Definition 4 *For a Poisson distribution, $\Psi(\lambda)$, with parameter λ: $P_k = \frac{\lambda^k e^{-\lambda}}{k!}$. Let $X \leftarrow \Psi(\lambda)$, then $E[X] = Var(X) = \lambda$.*

As suggested above, we choose $k \sim \Psi(\lambda)$ and then pick k samples. Importantly, the number of occurrences of distinct elements are independent with Poisson sampling. Also, the number of occurrences of element $i \sim \Psi(kp_i)$ which means that the expected value and variance of number of occurrences of element i are both kp_i. Choosing samples in this way gives us independence between samples as we used it above, and therefore, our proof from above is valid without the assumption of independence as long as the sample size is chosen according to a Poisson distribution with a sufficiently large parameter.

4 Conclusion

The final step in proving our lower bound is choosing λ for our poisson distribution. We choose λ to be $c\frac{\sqrt{n}}{\epsilon^2}$ where c is a large constant such that the expected value of the number of samples we draw from the distribution is much larger than $\frac{\sqrt{n}}{\epsilon^2}$. Then, the probability that the number of samples drawn is less than $\frac{\sqrt{n}}{\epsilon^2}$ is tiny, so the overall probability the algorithm succeeds is still large. Therefore, any algorithm for uniformity testing using poissonization requires $O(\Psi(c * \frac{\sqrt{n}}{\epsilon^2}))$ samples and any algorithm for uniformity testing must require $O(\frac{\sqrt{n}}{\epsilon^2})$.