Lecture 9: Tolerant Testing

Daniel Kane
Scribe: Kuang Hsuan Lee

April 21, 2017

1 Introduction

Tolerant testing is the last topic on unstructured data sampling. In the previous lectures we assumed p is exactly q or p is far away from q. In this one, we will relax the former assumption. Because in the real world, there are no distribution is exactly the same.

2 L2 Tester

Out target is distinguishing $|p-q|_2 \leq \epsilon$ from $|p-q|_2 \geq \epsilon/2$ in $O(\frac{|q|_2}{\epsilon^2})$ samples. L_2 tester is very nice because it is unbiased and the variance is small. Hence, the estimator is close to the actual one.

3 L1 Tester

3.1 using similar L_2 idea

When we know L_2 is great, what about L_1? Let's see the identify test with L_1.
1. split ith bin into $\lceil nq_i \rceil$
2. run L_2 tester on the new distribution to distinguish between $p_s = q_s$ and$\|p_s - q_s\|_2 > \frac{\epsilon}{2\sqrt{n}}$.
And the $\|p_s - q_s\|_2$ is equal to the following result.

$$\|p_s - q_s\|_2 = \sum_i \lceil nq_i \rceil (\frac{p_i - q_i}{\lceil nq_i \rceil})^2 = \sum_i \frac{(p_i - q_i)^2}{\lceil nq_i \rceil} \leq \frac{1}{n} \sum_i \frac{(p_i - q_i)^2}{q_i} = \frac{\chi^2(p,q)}{n}$$

Then we can distinguish $|p-q| > \epsilon$ v.s. $\chi(p,q) < \frac{\epsilon}{2}$. And it is not hard to show $\chi(p,q)$ bounds $|p-q|$ with Cauchy Schwarz. If L_1 is big, we can detect that. However, even if the χ upper bounds L_1, the bound sometimes is week. For example, when q is very small and p is a little larger than it. In this case, L_1 is small but χ is very big. As a result, the tester will fail to distinguish $|p-q|_1 > \epsilon$ v.s. $|p-q|_1 < \frac{\epsilon}{2}$.

3.2 The bound of previous algorithm is week, what do we do?

We will show that it is hard to find a tester for distinguishing $|p-q|_1 > \epsilon$ v.s. $|p-q|_1 < \frac{\epsilon}{2}$ even for a constant ϵ. But we can focus on the case: $q = U_n$ because the tester can want us to approximate

$$|p-q| = \sum_i |p - \frac{1}{n}|$$

The intuition is that we cannot track the order but we can keep track of the count of how many samples land in each bin. Also, the bins are more or less interchangeable, so it does not make sense to care about which is which, only how many bins got each number of samples. How many bins have exactly two samples, three samples. These counts are only real the algorithm to get.

However, if we want to look at that how many bins have k samples. This is a Poisson distribution and the expected value is: $\sum_i e^{-mp_i} \frac{(mp_i)^k}{k!}$. And the point is if we do with a small value of k. This will give us the moment of p_i.

The easy to do is that we look at the samples, we approximate the moments of p_i or low degree polynomials of p. $\sum p_i$ is linear combination of these moments, which is basically polynomial, f is low degree of polynomial, which is easy estimate for sample. We can use the following:

$$\sum_i f(p_i) \ f \ low \ degree \ polynomial)$$

If we want to do L_2 tester, we should estimate $\sum_i (p_i - \frac{1}{n})^2$ which is easy to estimate.

Now the problem here is we want to do the L_1 norm. $|x - \frac{1}{n}|$ is not a polynomial. we maybe should not expect to easily do this. Now we cannot do it with these samples. We can learn the distribution in $\frac{n}{\epsilon^2}$ samples and use it to get ϵ approximation to $|p - q|_1$. However it needs a linear number of samples. Actually we will see that it has to take $\frac{n}{logn}$ samples when ϵ is a constant.

We are going to regime of a constant ϵ because we know the error depends on the constant ϵ. As ϵ becomes small, then finding the correct dependence on the number of samples n and ϵ is a stall open question.

We actually have the lower bound, we will go through the lower bound.

3.3 theorem 1.1

3.3.1 content

p is a distribution on $[n]$. Any tester that distinguishes between

$$|p - U_n|_1 < \frac{1}{10} \quad v.s. \quad |p - U_{\frac{n}{2}}|_1 < \frac{1}{10}$$

requires $\Omega \frac{n}{\log n}$ samples. Here $U_{\frac{n}{2}}$ is the uniform distribution over some subset of $\frac{n}{2}$ bins.

3.3.2 Note

If you have a good tolerant tester, which should be able to distinguish between these two cases: the close uniform distribution or the far apart uniform distribution.
This also excludes the possibility of being able to estimate entropy to small error.

3.3.3 Proof

We use the adversary method to prove the theorem. First, We take X a random bit and once X is fixed each p_i is picked independently according to some distribution.

$$X = 0 \quad p_i \overset{iid}{\sim} A \ = \ some \ distribution \ on \ [0, 1]$$

$$X = 1 \quad p_i \overset{iid}{\sim} B$$

We should pick A and B carefully. Lets figure out what does our adversary has to do with A, B. When $X = 0$ we want $|p - U_n|_1$ to be small. Therefore, we need $E[|A - \frac{1}{n}|] << \frac{1}{n}$. This ensures that in each bin we do not add too much mass to $|p - U_n|_1$, so that when we sum over all the bins we have $|p - U_n|_1 < \frac{1}{10}$.
When $X = 1$, we want B to be pick 0 half the time and be close to $\frac{2}{n}$ half the time. In order to quantify this for B, we use the Earth mover metric.

3.3.4 Earth mover metric

$$d_{EM}(r, s) = inf_{X \sim r, Y \sim s} E[|X - Y|]$$

For comparing different distribution on the real line. When you have two distribution 1 and distribution 2, first draw them in the x axis.
What you want to do is turn one distribution to the other.
Pick the little pieces of the probability mass and move them one place to the other and sort the total distance.
The cost is how much time you move times how much far you move.

3.3.5 Why use Earth mover metric here

Because the marginals of X and Y are fixed, we can take the infimum over all possible correlations that X and Y have. We want A to be close to $\delta_{\frac{1}{n}}$ in Earth mover metric and B to close to $\frac{1}{2}\delta_0+\frac{1}{2}\delta_{\frac{2}{n}}$
These ensures that when $X=0$, then $|p-U_n|_1<\frac{1}{10}$ and when $X=1$, then $|p-U_{\frac{n}{2}}|_1<\frac{1}{10}$
And we will see what happen to fool the tester.
There are two thing to look at.

3.3.6 When p_i is large

In this case, we can actually approximate p_i pretty well. When p_i is large, we are supposed to have more samples on these bins and the number of samples that fall in bin i are $\sim \text{Poi}(mp_i)$. The variance of Poisson is equal to mean , when mean is large, the variance become small fraction of we look at.
How large is too large?
The error we get is the difference between the empirical estimator of p_i and the actual p_i. The error in p_i we get is $\frac{p_i}{m}$. There are a bunch of bins whose probabilities p_i are all close to α. There can be at most $\frac{1}{\alpha}$ such bins. Also m is close to $\frac{n}{\log n}$, so if $\alpha \gg \frac{\log n}{n}$ then the tester will learn these p_i. When p_i is bigger than α, p_i is not hard, we just approximate the mode, which is fine. And these distributions are going to be supposed on $[0, \frac{log^2 n}{n}]$. Otherwise, if it is too big, we can have a different way to approximate it.

3.3.7 When p_i is relatively small

When p_i is relatively small, our tester is going to compute the moments of the p_i. If we can compute the moment of p_i, if we want to fool tester and the moments of A are basically the same as those of B.
The kth moment computation corresponds to looking at the number of bins with k samples. Thus the tester can compute the moments up to $O(\log n)$. Hence we only need to ensure that A and B agree on moments up to $O(\log n)$.
This time, we let $D=A-B$ be a pseudodistribution. It is supported on $[0, c\frac{\log^2 n}{n}]$ and satisfies $|D|_1 \le 2$ and $d_{EM}(D, \delta_{\frac{1}{n}} - \frac{1}{2}\delta_0 - \frac{1}{2}\delta_{\frac{2}{n}})$ is small.

$$E_{X\sim D}[X^k]=0 \quad 1\le k\le c\log n$$

Our target is finding the D. Its positive and negative parts give the required A and B. Given A, B, we will run information theory to make this thing work.

3.3.8 Lemma for construct D

p is a degree d polynomial with distinct real roots $r_1, \ldots, r_d$.
Let $a_i=\frac{1}{p'(r_i)}$, then $\sum_i a_i r_i^k=0$, for any integer $0\le k\le d-2$.
Note that the pseudo distribution $\sum_i a_i\delta_{r_i}$ has zero low degree moments. This will be the tool we will use to construct D. We get D by picking a polynomial g and using the above lemma's trick with the roots. What g do we need to pick ?
We need g to have roots at $0, \frac{1}{n}, \frac{2}{n}$ and another reasonable polynomial factor say T. We expect T to have roots in $[0, c\frac{log^2 n}{n}]$. Furthermore it needs to have nice derivatives which are not huge (Since this would ensure that A, B are close). One natural thing to try are the Chebyshev polynomials.

$$g(x)=x(x-\frac{1}{n})(x-\frac{2}{n})T_d(1-\frac{2x}{c\frac{log^2 n}{n}})$$

4 Chebyshev Polynomials

The degree-d Chebyshev polynomial is the unique polynomial that satisfies $T_d(\cos\theta)=\cos(d\theta)$
The roots of $T_d(y)$ are $y=\cos(\frac{(2m+1)\pi}{2d})$, which is near $1-\frac{1}{2}(\frac{(2m+1)\pi}{2d})^2$, where d $= c_3\log(n)$. So the first root is at $1-\frac{\pi^2}{8d^2}$. Let $c_4=\frac{\pi^2 c}{16c_3^3}$. So the corresponding x would be at approximately $c_4\frac{1}{n}$.
Note that we choose c, c_3 such that the first root of T_d as a function of x above is very far right

from $\frac{2}{n}$. Thus the three artificially implanted roots of f are well before the roots of T.
If we take c very small then $T_d(y)$ will be on $[0, \frac{2}{n}]$, in this case $T_d(y)$ will be at least $\frac{99}{100}$. Also,

$$f'(0) \approx \frac{2}{n^2},$$

$$f'(\frac{1}{n}) \approx -\frac{1}{n^2},$$

$$f'(\frac{2}{n}) \approx \frac{2}{n^2},$$

Now we need to find f' at the roots of T_d. We observe

$$T_d'(\cos\theta) = \frac{d\sin(d\theta)}{\sin\theta}$$

At the roots of T_d, $\cos(d\theta) = 0$, thus $\sin(d\theta) = \pm 1$. As a result, T_d' at the roots is $\frac{\pm d}{\sin\theta}$ where θ depends on the root. The roots of f are firstly the three artificial roots 0, $\frac{1}{n}$, $\frac{2}{n}$ and then after a gap followed by the roots $c_4\frac{1}{n}$, $c_4\frac{9}{n}$, $c_4\frac{25}{n}$. And then if we keep going on, we will find something more evenly space they work before, but not so much more. What we get, when we look at f'. Now say we evaluate $f'(c_4\frac{(2m+1)^2}{n})$, then we will get

$$f'(c_4\frac{(2m+1)^2}{n}) \leq \frac{64c_4^3m^6}{n^3}\frac{2d^2}{(2m+1)\pi}\frac{n}{log^2n} \approx c_5\frac{m^5}{n^2}$$

We can get D from this polynomial we constructed.
After we get the D, we will use D and information theory to complete the lower bound.