Lecture 4: Property Testing

Daniel Kane
Scribe: Sankeerth Rao

April 12, 2017

Abstract

We prove an upper bound on the number of samples needed to test if a distribution is uniform or ϵ away from it, given the guarentee that it is one of these two cases.

1 Setup

Given we answered the number of samples required to learn an unstructured distributions we now switch to Property testing where you don't need to know the whole distribution but just if the distribution satisfies a property. A naive method to do would be to take enough samples to learn the distribution and then check if it has the given property but we hope that testing requires fewer samples than learning.

Few questions that occur naturally are:

- Is $p = U_n$?
- Is $p = q$?
- Are p and q independent ?

However we would require infinitely many samples to separate say the uniform distribution from distributions that are arbitrarily close to it. To overcome this we assume the following:

Separation Guarantee:

- Either the property holds (OR)
- It is at least ϵ far (in TV distance) from any circumstance where the property holds.

What is the probability of failure ?

- We want an algorithm that gets 2/3 probability of success if the Guarantee holds.
- Arbitrary behavior if the guarantee doesn't hold.

2 Uniformity Testing:

A natural approach would be take N samples and approximate the Binomial number of samples that fall in a bin by the appropriate Gaussian. So if X_i samples fall in bin i then $X_i \approx \mathcal{N}(p_i N, \sqrt{p_i(1-p_i)}N)$. Then we compute

$$\sum_i \frac{(X_i - N/n)^2}{\sqrt{N\frac{1}{n}\frac{n-1}{n}}}$$

and compare its tails to the χ^2 statistic (the distribution you get by taking a sum of squares of $n-1$ independent Gaussians, which is approximately what you would expect to see if p were uniform). This only works well asymptotically because for the Normal approximation to work we need enough samples in each bin $X_i >> 1$ thus $N >> n$. But if this is the case we might as well learn the whole distribution.

Another Idea: Lets take two samples from a distribution and evaluate the probability that they are equal(collision).

$$\Pr[Same] = \sum_i p_i^2 = ||p||_2^2.$$

Note that this is precisely minimized when $p = U_n$. In fact if p is far from U_n then p will have many more collisions. Thus we have the following algorithm.

Algorithm:

- Take N samples
- Count # of collisions
- Compare to $\mathbb{E}[\#]$ under uniform distribution.

Note that we need $\sqrt{n}$ samples to see a collision under the uniform distribution. So we could take $10\sqrt{n}$ samples and see the number of collisions and compare to the uniform hypothesis and reject it with atleast 90% probability.

Let us work on an algorithm that is in between the χ^2 algorithm and the collision counting algorithm.

Is $p = q$? Take m samples from p and m samples from q. Note that the order in which the samples come doesn't matter much because any permutation of the samples would still come with the same probability so the best information we can process from the sample is:

- X_i = Number of samples from p in the ith bin.
- Y_i = Number of samples from q in the $i$the bin.

Poisson RV: A random variable T is Poisson(λ) distributed if

$$\Pr[T = t] = e^{-\lambda}\frac{\lambda^t}{t!}.$$

The mean and variance of T are λ.

Note: I think of Poisson(λ) as a limit of Binomial(n, p) distributions such that $np = \lambda$. Let B_n have the Binomial (n, p) distribution with $np = \lambda$. Then

$$\Pr[B_n = t] = \binom{n}{t}p^t(1-p)^{n-t} = \frac{n(n-1)\dots(n-t+1)}{t!}\frac{\lambda^t}{n^t}\Big(1-\frac{\lambda}{n}\Big)^{n-t}$$
$$\lim_{n\to\infty}\Pr[B_n = t] = \Pr[T = t].$$

Poissonization: Now note that we need X_i and X_j to be pairwise independent, but this is not the case when we take a fixed number of samples so we use the trick of Poissonisation where in:

- Take $Poi(m)$ samples from p and Independently
- Take $Poi(m)$ samples from q.

Note that we have the following properties

- whp. we pick only $\Theta(m)$ samples which follows from tight concentration of a Poisson random variable.
- $X_i \quad \forall i$ are pairwise independent. $Y_i \quad \forall i$ are pairwise independent. X^n, Y^n are independent.
- $X_i \sim Poi(mp_i)$ and $Y_i \sim Poi(mq_i)$.

We justify these properties in the following lemmas.

Lemma 2.1. $X_i \sim Poi(mp_i)$

Proof.

$$\begin{aligned}\Pr(X_i = k) &= \sum_{l\geq k}\Pr(X_i = k, N = l)\\ &= \sum_{l\geq k} e^{-m}\frac{m^l}{l!}\binom{l}{k}p_i^k(1-p_i)^{l-k}\\ &= \frac{e^{-m}(p_i m)^k}{k!}\sum_{l\geq k}\frac{(m(1-p_i))^{l-k}}{(l-k)!}\\ &= e^{-mp_i}\frac{(mp_i)^k}{k!}.\end{aligned}$$

□

Lemma 2.2. *$X_i \quad \forall i$ are pairwise independent.*

Proof.

$$\begin{aligned}
\Pr(X_i = k, X_j = r) &= \sum_{l \geq k+r} \Pr(X_i = k, X_j = r, N = l) \\
&= \sum_{l \geq l+r} e^{-m} \frac{m^l}{l!} \binom{l}{k, r, l-k-r} p_i^k p_j^r (1 - p_i - p_j)^{l-k-r} \\
&= \frac{e^{-m} m^{k+r} p_i^k p_j^r}{k! r!} \sum_{l \geq l+r} \frac{(m(1 - p_i - p_j))^{l-k-r}}{(l-k-r)!} \\
&= \left(e^{-mp_i} \frac{(mp_i)^k}{k!}\right)\left(e^{-mp_j} \frac{(mp_j)^r}{r!}\right).
\end{aligned}$$

□

Now one statistic we could use is $Z = \sum_i |X_i - Y_i|$, but its a pain to get rid of the $|\cdot|$, so we work with $Z = \sum_i (X_i - Y_i)^2$ instead.

$$\begin{aligned}
\mathbb{E}[Z] &= \sum_i \mathbb{E}[X_i^2 - 2X_i Y_i + Y_i^2] \\
&= \sum_i m^2 (p_i - q_i)^2 + m(p_i + q_i).
\end{aligned}$$

To fix the linear term above we modify Z to

$$Z = \sum_i (X_i - Y_i)^2 - X_i - Y_i.$$

Thus

$$\mathbb{E}[Z] = m^2 |p - q|_2^2$$

Note that this statistic knows collisions are what are important because it removes the noise from bins that have only one sample in them. $X_i = 1, Y_i = 0 \to (1-0)^2 - 1 - 0 = 0$. Now we evaluate the variance of Z. We can use the pairwise independence to say:

$$Var(Z) = \sum_i Var((X_i - Y_i)^2 - X_i - Y_i).$$

To evaluate these variances we need the following higher moments of Poisson random variable.

Higher Moments The higher moments of the Poisson distribution are Touchard polynomials in λ:

$$\mathbb{E}[T^k] = \sum_{l=0}^{k} \lambda^l \cdot S(k,l)$$

where $S(k,l)$ are the Stirling numbers of the second kind given by

$$S(k,l) = \frac{1}{l!} \sum_{j=0}^{l} (-1)^{l-j} \binom{l}{j} j^k$$

Substituting these we get that

$$\begin{aligned} Var(Z) &= \sum_i 4m^3(p_i - q_i)^2(p_i + q_i) + 2m^2(p_i + q_i)^2 \\ &\leq O(m^3|p-q|_2^2|p+q|_2 + m^2|p+q|^2) \end{aligned}$$

Now we use Chebyshev inequality

$$\Pr[|Z - \mathbb{E}[Z]| \geq 10\sqrt{Var(Z)}] \leq \frac{\mathbb{E}|Z - \mathbb{E}[Z]|^2}{100 Var(Z)} = 0.01.$$

So with 99% probability we have

$$\left| Z - m^2|p-q|_2^2 \right| < O\left(\sqrt{m^3|p-q|_2^2|p+q|_2 + m^2|p+q|^2}\right)$$

Lemma 2.3. *L^2 Tester*
If $\max(|p|_2, |q|_2) \leq b$.
If either

- $p = q$ *OR*
- $|p-q|_2 > \epsilon$,

then there is a algorithm that distinguishes with $O(b/\epsilon^2)$ samples.

Proof. The algorithm we use is:

- Compute Z
- Compare to $\frac{m^2\epsilon^2}{2}$

So for this algorithm to work we need:

- In the case that $p = q$, Z needs to be much smaller than $m^2\epsilon^2/2$ whp and
- in the case that $|p-q|_2 > \epsilon$, Z needs to be much bigger than $m^2\epsilon^2/2$ whp

We pick m so that both of these are satisfied. Lets analyse these

Case I $p = q$ We know from Chebyshev above that $|Z| < O(m|p+q|_2)$ with atleast 99% probability. So we need to pick m so that $O(m|p+q|_2) < \frac{m^2\epsilon^2}{2}$. Thus a choice of $m \geq O(\frac{b}{\epsilon^2})$ would be good for this case.

Case II $|p-q| > \epsilon$ We know from Chebyshev above that

$$Z > m^2|p-q|_2^2 - O(\sqrt{m^3|p-q|_2^2|p+q|_2 + m^2|p+q|^2}) \text{ with atleast 99\% probability.}$$

Now we need to pick m so that $m^2|p-q|_2^2 - O(\sqrt{m^3|p-q|_2^2|p+q|_2 + m^2|p+q|^2}) > \frac{m^2\epsilon^2}{2}$. Since $|p-q|_2 > \epsilon$ it suffices to pick m so that $\frac{m^2|p-q|_2^2}{2} > O(\sqrt{m^3|p-q|_2^2 b + m^2 b^2})$. So it suffices to pick m so that $(m^2|p-q|_2^2)^2 > O(m^3|p-q|_2^2 b)$ and $(m^2|p-q|_2^2)^2 > O(m^2b^2)$, but both of these conditions say that its sufficient to pick $m > O(b/|p-q|_2^2)$, but since in this case $|p-q|_2 > \epsilon$ it is sufficient to pick $m > O(b/\epsilon^2)$. □

L^1 Tester Now we derive an L^1 tester from this. We know

$$|p-q|_1 < |p-q|_2\sqrt{n}$$

Thus to test $p = q$ vs $|p-q|_1 < \epsilon$, it is sufficient to test $p = q$ vs $|p-q|_2 < \epsilon/\sqrt{n}$. Thus our algorithm needs $O(bn/\epsilon^2)$ samples.

Note that in the case when $p, q \approx U_n$, $b \approx \frac{1}{\sqrt{n}}$, thus our algorithm requires $O(\sqrt{n}/\epsilon^2)$ samples.