Distribution Testing

- lower bound idea for uniformity testing
- closeness testing: via two techniques
  - Poissonization
  - reduction to low ℓ_2-norm case

Last time:

- estimate $\|p\|_2^2$ via $\hat{C} \leftarrow \frac{\sum_{i<j} \delta_{ij}}{\binom{s}{2}}$
  - Variance of estimator is $O\left(\frac{\|p\|_2^2}{s^2} + \frac{\|p\|_2^3}{s}\right)$
  - additive $\frac{\varepsilon^2}{2}$ error using $O(1/\varepsilon^4)$ samples
  - $\underbrace{\text{multiplicative } (1 \pm \varepsilon^2/3)}_{}$ error using $O(\sqrt{n}/\varepsilon^4)$ samples

$\Rightarrow$ distinguish $p = U$ from $\|p - U\|_1 > \varepsilon$

what about runtime?

Next homework:

- property test for uniformity in L_1-distance requires $\Omega(\sqrt{n})$ samples

idea: distinguish (1) $p = U$

(2) Choose p via

1. pick $S \subseteq [n]$ s.t. $|S| = n/2$ randomly
2. $p \leftarrow U_S$

Generalizations: given distributions p, q

is $p = q$ or is p "far" from q?

1. "Identity Testing":

q known to algorithm, no samples needed

in "DNA"

hardcoded into algorithm

} focus on sample complexity, but can make runtime similar

L_1: $\Theta(\sqrt{n})$ samples

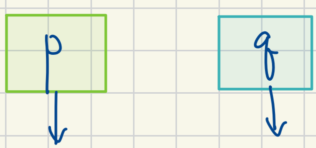
(see homework)

2. "Closeness testing":

q is given via samples

L_1: $\Theta(n^{2/3})$ samples

(today)



3. tolerant versions: is $\|p-q\|_1 < \varepsilon$ or $\|p-q\|_1 > \varepsilon'$?

L_1: $\Theta(n/\log n)$ samples

Uniformity Testing algorithms

- estimate # collisions
- estimate # distinct elements
- similar to χ^2-based tester
- plug-in tester

all optimal in terms of n & ε

best in terms of δ

Identity Testing algorithms

- reduce to uniformity (several ways) ← see next pset
- similar to χ^2-based

Lots of ways to approach closeness testing

as in uniformity testing, lets consider L_2-distance

$$\|p-q\|_2^2 = \sum (p_i - q_i)^2$$
$$= \sum p_i^2 - 2\sum p_i q_i + \sum q_i^2$$

estimate as before

"cross collision" probability - estimate analogously

note $\sum p_i q_i \leq \frac{\sum p_i^2 + \sum q_i^2}{2}$

difficulty:

when $q = U$ we had upper bound on $\sum q_i^2$ & $\sum p_i q_i$

now we don't ...

possible solutions:

handle high probability elements separately?

see below for another approach.

Poissonization

A difficulty in analyzing distribution testers:

typical algorithm:

take m samples $\{s_1 \dots s_m\} = S$

let $X_i \leftarrow$ # times elt $i \in D$ appears in S

problem: X_i's not independent.

e.g. if $X_i = \frac{m}{2} + 5$ then $X_j < m/2$

Can we make X_i's independent?

$\boxed{\begin{array}{l} Poi(\lambda): \Pr[X=k] = \dfrac{e^{-\lambda}\lambda^k}{k!} \\ E[X] = Var[X] = \lambda \end{array}}$

Poissonization

new algorithm:

$\hat{m} \leftarrow Poi(m)$

take $\hat{m}$ samples to get $\hat{S}$

let $X_i \leftarrow$ # times elt i appears in $\hat{S}$

(1)

equivalent

$\Longleftrightarrow$

For each $i \in [n]$

$X_i \leftarrow Poi[m \cdot p_i]$

add X_i copies of i to sample

randomly permute sample

(2)

X_i's are independent now!

why equivalent?

$$\Pr[X_i = c \text{ according to } ①] = \sum_{k=c}^{\infty} \Pr[\hat{m}=k] \cdot \binom{k}{c} p_i^c (1-p_i)^{k-c}$$

$$= \sum_{k=c}^{\infty} \frac{e^{-m} m^k}{\cancel{k!}} \cdot \frac{\cancel{k!}}{c!\,(k-c)!} \cdot p_i^c (1-p_i)^{k-c}$$

$$= \frac{e^{-m} m^c p_i^c}{c!} \cdot \underbrace{\sum_{k=c}^{\infty} \frac{m^{k-c} (1-p_i)^{k-c}}{(k-c)!}}_{= \sum_{k'=0}^{\infty} \frac{m^{k'} (1-p_i)^{k'}}{k'!} = e^{m(1-p_i)}}$$

$$= \frac{\cancel{e^{-m}} m^c p_i^c}{c!} \cdot e^{m(\cancel{1}-p_i)}$$

$$= \frac{e^{-mp_i} (mp_i)^c}{c!}$$

$$= \Pr_{X_i \sim \mathrm{Poi}[mp_i]}[X_i = c]$$

$$= \Pr[X_i = c \text{ according to } ②]$$

Taylor series expansion for e^{-x}:

$$e^{-x} = \sum_{k=0}^{\infty} \frac{(-x)^k}{k!}$$

(also need to check joint distributions are same)

Reduction to low L_2-norm case

recall uniformity test statistic $\hat{C}$:

$$\mathrm{Var}[\hat{C}] = O\left(\frac{\|p\|_2^2}{s^2} + \frac{\|p\|_2^3}{s}\right)$$

recall # samples $\sim \frac{\mathrm{Var}}{\alpha^2}$ where $\alpha \sim \frac{1}{f(n)}$

so need $\|p\|_2$ small (in terms of n) & s big, to "kill" $f(n)$

Problem $\|p\|_2$ can be large $\Rightarrow$ need lots of samples?

Goal: transform p, q into p', q' s.t. $\|p'\|_2$ & $\|q'\|_2$ small

&

$$p = q \Rightarrow p' = q'$$

$$\|p-q\|_1 > \varepsilon \Rightarrow \|p'-q'\|_1 > \varepsilon$$

gives reduction to small L_2 norm case

recall that $\|p'\|_2$ & $\|q'\|_2$ small $\Rightarrow \sum p'_i q'_i$ small

reduction will work both when

- q known
- q given via samples

"Flattening"

Transformation of p: (assume original algorithm uses m samples)

$S \leftarrow$ draw $m' = Poi(m)$ samples from p over domain $[n]$

$b_i \leftarrow$ # times i appears in S $\forall i \in [n]$ (so $b_i \sim Poi(p(i) \cdot m)$)

$\forall i$, add $b_i + 1$ elts to new domain

(i, j) where $j \in [b_i + 1]$

new distribution p':

pick $i \in_r p$

pick $j \in_u [b_i + 1]$

output (i, j)

size of new domain $= n + m'$

$$p'(i,j) = \frac{p(i)}{b_i + 1}$$

Example

domain of p is $[5]$

$S \leftarrow \{2, 5, 3, 2, 3\}$

$b_2 = b_3 = 2$

$b_5 = 1$

all other b_i's $= 0$

domain of p':	prob
$\{ (1,1)$	$p(1)$
$(2,1)\ (2,2)\ (2,3)$	$\frac{p(2)}{3}\ \frac{p(2)}{3}\ \frac{p(2)}{3}$
$(3,1)\ (3,2)\ (3,3)$	$\frac{p(3)}{3}\ \frac{p(3)}{3}\ \frac{p(3)}{3}$
$(4,1)$	$p(4)$
$(5,1)\ (5,2) \}$	$\frac{p(5)}{2}\ \frac{p(5)}{2}$

Properties needed by reduction hold:

(ie. if $p=q$ then $p'=q'$ & if $\|p-q\|_1 > \varepsilon$ then $\|p'-q'\|_1 > \varepsilon$)

If transform $p \to p'$, $q \to q'$ using $\underline{\underline{\text{same}}}$ S & $\underline{\underline{\text{same}}}$ b_i's:

$$\|p-q\|_1 = \sum_x |p(x)-q(x)|$$

$$= \sum_x \sum_{y=1}^{b_x+1} \frac{|p(x)-q(x)|}{b_x+1} \qquad \text{algebra}$$

$$= \sum_x \sum_{y=1}^{b_x+1} |p'(x,y)-q'(x,y)| \qquad \text{def of } p', q'$$

$$= \|p'-q'\|_1$$

Will show p' has low $\|p'\|_2^2$ (next)

But ... what about q' ???

if $p'=q'$ then q' also has low $\|p'\|_2^2$

but if $\|p'-q'\|_1 > \varepsilon$ maybe q' has big $\|p'\|_2^2$?

will show how to deal with this case

<u>Claim</u> $E[\|p'\|_2^2] \leq \frac{1}{m}$

<u>Why?</u>

$$E[\|p'\|_2^2] = E\left[\sum_{i=1}^{n} \sum_{j=1}^{b_i+1} p'(i,j)^2\right]$$

$$= E\left[\sum_{i=1}^{n} \sum_{j=1}^{b_i+1} \frac{p(i)^2}{(b_i+1)^2}\right]$$

$$= E\left[\sum_{i=1}^{n} \frac{p(i)^2}{(b_i+1)}\right] = \sum_{i=1}^{n} p(i)^2 \cdot E\left[\frac{1}{1+b_i}\right]$$

$$\overset{*}{\leq} \sum_{i=1}^{n} \frac{p(i)^2}{m \cdot p(i)} = \frac{1}{m} \sum p(i) = \frac{1}{m} \quad \blacksquare$$

$*$: $b_i+1 \sim 1 + \text{Poi}(m \cdot p(i))$

<u>Known</u>: If $Y \sim \text{Poi}(\lambda)$ then $E[z^Y] \sim e^{\lambda(z-1)}$

So, $E\left[\frac{1}{1+b_i}\right] = E\left[\int_0^1 z^{b_i}\, dz\right] = \int_0^1 E[z^{b_i}]\, dz$

$$= \int_0^1 e^{(m \cdot p(i))(z-1)}\, dz = \frac{1}{mp(i)} e^{mp(i)(z-1)}\Big]_0^1$$

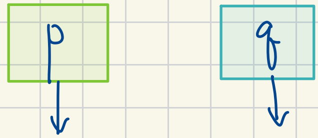
$$= \frac{1}{mp(i)} \cdot \left[1 - e^{-mp(i)}\right] \leq \frac{1}{mp(i)}$$

Check 1st equality:

$$\int_0^1 z^x\, dz = \frac{z^{x+1}}{x+1}\Big]_0^1$$

$$= \frac{1}{x+1} - \frac{0}{x+1}$$

L_2-distance between p & q:
(multiplicative estimate)



<u>Thm</u> (*) given samples of dists p, q over $[n]$

s.t. $b \geq \max\{\|p\|_2, \|q\|_2\}$,

can distinguish $p = q$ from $\|p-q\|_1 > \varepsilon$

in $O(bn/\varepsilon^2)$ samples

proof is similar to uniformity $\Rightarrow$

<u>Corr</u> if $b = \min\{\|p\|_2, \|q\|_2\}$

can distinguish $p = q$ from $\|p-q\|_1 > \varepsilon$ in $O(bn/\varepsilon^2)$ samples

<u>Pf idea for corr</u>

1. estimate $\|p\|_2$ & $\|q\|_2$ to mult factor of c with $O(\sqrt{n})$ samples

2. if differ by $> c$ mult factor, infer $p \neq q$ & reject

3. else use thm * with $b' = cb$

<u>Conclude</u>: $\|p\|_2$ small because of flattening
$\|q\|_2$ small because w/in mult factor c of $\|p\|_2$
$\sum p_i q_i$ small because $\|p\|_2, \|q\|_2$ small

Algorithm for testing Closeness of p, q

- let $k \leftarrow n^{2/3} \varepsilon^{-4/3}$
- $S \leftarrow \mathrm{Poi}(k)$ samples from p
- use S to "flatten" p, q (use same b_i's for q)
- run tester of corollary on p', q' wrt S

behavior?

Closeness on p, q vs. p', q' equivalent

\# samples?

whp $|S| = O(k)$

$E[\|p'\|_2^2] = O(1/k)$ so whp $\|p'\|_2 = O\left(\frac{1}{\sqrt{k}}\right)$

total: $O\Big(\underbrace{k}_{\text{pick } S} + \underbrace{\frac{1}{\sqrt{k}} \cdot n \cdot \frac{1}{\varepsilon^2}}_{\text{run tester on } p', q' \text{ via corollary}}\Big) = O\left(n^{2/3} \varepsilon^{-4/3} + \frac{1}{n^{1/3} \varepsilon^{-2/3}} \cdot n \cdot \frac{1}{\varepsilon^2}\right)$

$= O\left(n^{2/3} \varepsilon^{-4/3}\right)$ ∎