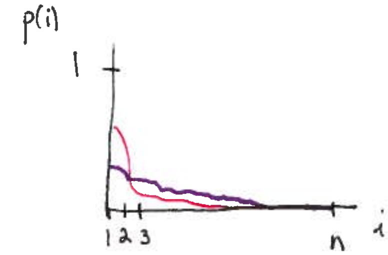
Testing & Learning Monotone Distributions (over totally ordered domain)



Def. p over $[n]$ is "monotone decreasing"
if $\forall\, i \in [n-1]$ $\quad p(i) \geq p(i+1)$

Monotonicity Tester:

- if p monotone increasing, Pass with prob $\geq 3/4$
- if p ε-far in L_1 dist from mon increasing, Fail with prob $\geq 3/4$

Useful tool: "Birge Decomposition"

(note: this is a different decomposition than in homework
in particular, it is <u>oblivious</u>!)

decompose domain $1..n$ into $\ell = \Theta\left(\frac{\log \varepsilon n}{\varepsilon}\right) \approx \Theta\left(\frac{\log n}{\varepsilon}\right)$ intervals

$$I_1^\varepsilon, I_2^\varepsilon, \ldots, I_\ell^\varepsilon \quad \text{s.t.}$$

$$|I_{k+1}^\varepsilon| = \lfloor (1+\varepsilon)^k \rfloor$$

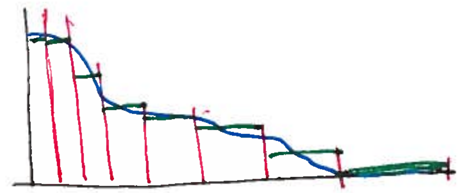
will drop ε in notation once it is fixed

$|I_1^\varepsilon| = |I_2^\varepsilon| = \ldots = 1$

$|I_a^\varepsilon| = |I_{a+1}^\varepsilon| = \ldots = 2$

$\vdots$

but then at some point the sizes grow exponentially define "flattened distribution"



$\forall\ 1 \le j \le \ell$

$\forall\ i \in I_j$ $\quad \tilde{q}_\varepsilon(i) = \frac{q(I_j)}{|I_j|}$

← assign all elements in same interval the same probability

note: $q(I_j) = \tilde{q}_\varepsilon(I_j)$

Birgé's Thm if q mon decreasing then $\|\tilde{q}_\varepsilon - q\|_1 < \varepsilon$

Corr: if q ε-close to mon decreasing then $\|\tilde{q}_\varepsilon - q\|_1 < O(\varepsilon)$

Testing Algorithm:

how can we even do this? $\tilde{q}$ isn't monotone, if q exactly uniform. See problem 2(c) from HW set due today.

Take samples of q

do uniformity test for each partition (using samples that fell in it) fail if any partition fails (if not enough samples then pass)

$w_j \leftarrow$ # samples that fell in partition j

use LP to verify w close to monotone

↑ note this is LP on $O(\log n)$ vars

How many samples?

for each partition with enough weight, say $\frac{\varepsilon}{\log n}$, need $\frac{\sqrt{n}}{\varepsilon^2}$ samples

↖ need $\frac{\sqrt{n} \cdot \log n}{\varepsilon^3}$ for each one

need another $\log\log n$ for union bound

$\approx O(\sqrt{n}\ \text{polylog}\ n \cdot \text{poly}\ \frac{1}{\varepsilon})$

(note: this can be improved!!)

Last step:

difficulty

purple is not monotone but is close

good thing: only $\frac{\log n}{\sqrt{\varepsilon}}$ variables!

Can be solved via brute force
LP (actually quite efficient)
⋮

Slightly changing perspective...

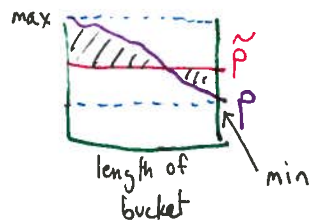
What if we know dist q is monotone, can we learn it?

Yes! use sampling to estimate $\tilde{q}_\varepsilon(I_j)$'s

Birge's Thm $\Rightarrow$ Can learn monotone distributions to w/in $\le \varepsilon$ L_1 error in $\Theta(\frac{1}{\varepsilon^3} \log n)$ samples.

Proof of Birge's Thm :

Error in bucket



gross upper bound on error:
$\leq (\text{max}-\text{min}) \cdot \text{bucket length}$

Partition of Intervals:

- Size 1 Intervals $|I_j| = 1$ ← if we have any short intervals, there are $\Omega(1/\varepsilon)$ of these; if not, we can learn the distribution
- Short Intervals $|I_j| < 1/\varepsilon$
- Long Intervals $|I_j| \geq 1/\varepsilon$

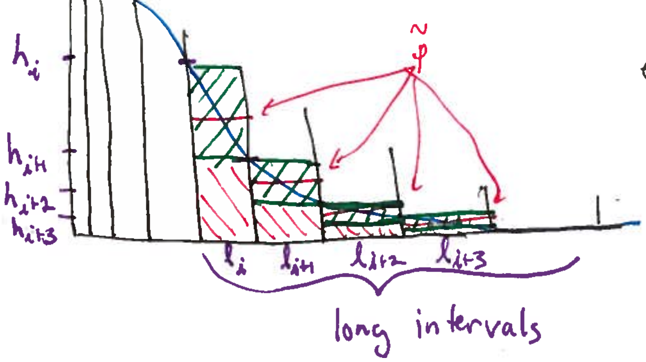
if we have these then max prob $\leq \varepsilon$ (since # size 1 intervals is $\Omega(1/\varepsilon)$)

↑ therefore min size 1 interval has prob $\leq \varepsilon$ which upper bounds later probabilities too since p is monotone

$$\text{total error} \leq \sum_{j=1}^{\ell} |I_j| \cdot (\text{max prob in } I_j - \text{min prob in } I_j)$$

$$= \underbrace{\sum_{\substack{\text{size 1} \\ \text{intervals}}} 1 \cdot 0}_{\substack{0 \\ \text{since no difference}}} + \underbrace{\sum_{\substack{\text{short} \\ \text{intervals}}} |I_j| (\text{max}-\text{min})}_{\text{omitted: idea is bound similarly to the long intervals but need to group together intervals of same size}} + \underbrace{\sum_{\substack{\text{long} \\ \text{intervals}}} |I_j| (\text{max}-\text{min})}_{\text{see below}}$$

Picture for long intervals:



green rectangles = upper bnd on error

$$\text{error} \leq (h_i - h_{i+1}) \cdot l_i + (h_{i+1} - h_{i+2}) l_{i+1} + (h_{i+2} - h_{i+3}) l_{i+2} + \dots$$

$$= h_i l_i + h_{i+1}(l_{i+1} - l_i) + h_{i+2} \underbrace{(l_{i+2} - l_{i+1})}_{\text{positive}} + h_{i+3}(l_{i+3} - l_{i+2})$$

all h_j's $\leq \varepsilon$!

positive, + $\approx \varepsilon \cdot l_{i+1}$ by way that we partitioned

$$\leq \varepsilon \Big[l_i + \underbrace{\sum h_i l_{i+1}}_{} \Big]$$

get rid of this when bounding short intervals

this is area of red rectangles, which is upper bounded by p so sum is ≤ 1

A useful tool: Hypothesis Testing

via complete description

Given collection of distributions, $\mathcal{H}$, at <u>least one</u> has high accuracy for describing p ⟸ given via samples

output one of collection that is close to p.

How many samples in terms of $|\mathcal{H}|$ + domain size?

Why is this different than testing closeness, uniformity?

Do we have the same lower bounds?

NO

Since p is guaranteed to be close to some $q \in \mathcal{H}$, all bets are off!!

A "subtool": allows comparing two hypothesis

Thm Given sample access to p

Given h_1, h_2 hypothesis distributions (fully known to algorithm)

Given accuracy parameter ε', confidence δ'

Algorithm "Choose" takes $O(\log(1/\delta')/(\varepsilon')^2)$ samples + outputs $h \in \{h_1, h_2\}$. If one of h_1, h_2 has $\|h_i - p\|_1 < \varepsilon'$

Then with prob $\geq 1-\delta'$, output h_j has $\|h_j - p\|_1 \leq 12\varepsilon'$