Lecture 12:

Testing properties of strings

Lower bounds via Yao's method

Property Testing of Strings:

different than edit distance, which is also reasonable to consider.

<u>def</u> w ε-far from P_n if $\forall\, y \in P_n$

w & y differ on $\geq \varepsilon \cdot n$ locns

Property tester for P_n: on input w

- if $w \in P_n$, pass (with prob $\geq 3/4$)
- if w ε-far from P_n, fail (with prob $\geq 3/4$)

Palindromes

Let $P_n = \{ w \mid w$ is an bit string & $w = vv^R \}$

Query complexity of prop tester? $O(1)$

Algorithm:

Do $O(1/\varepsilon)$ times:

Pick random i, test if $w_i = w_{n-i+1}$

Contrapositive:
$p \Rightarrow q$ $\Leftrightarrow$ equivalent
$\neg q \Rightarrow \neg p$

Why does it work? if ε-far $\Rightarrow$ likely to fail

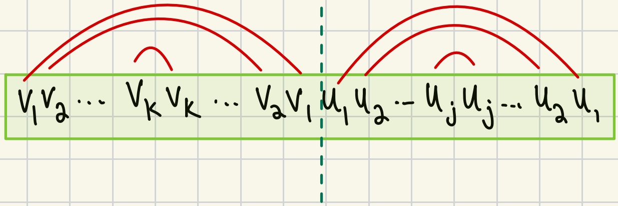
equivalent to

if not likely to fail $\Rightarrow$ ε-close

If test passes whp, then $\leq \varepsilon n$ "pairs" $(i, n-i+1)$ don't match. Can fix each one with $\leq \varepsilon n$ changes.

Concatenations of Palindromes

$L_n = \{ w \mid w \text{ is } n\text{-bit string } w = vv^R uu^R \}$



def w ε-far from L_n if $\forall y \in L_n$

w & y differ on $\geq \varepsilon \cdot n$ locns

(different than edit distance, which is also a reasonable distance to consider)

Thm if algorithm $\mathcal{A}$ satisfies

$\forall x \in L_n \quad \Pr[\mathcal{A}(x) = \text{Pass}] \geq 2/3$

$\forall x$ ε-far from $L_n \quad \Pr[\mathcal{A}(x) = \text{Fail}] \geq 2/3$

then $\mathcal{A}$ makes $\Omega(\sqrt{n})$ queries.

How does one prove lower bounds?

a difficulty: property testing algorithms are randomized

how do you argue about their behavior?

Useful tool for lower bounding randomized algorithms:

Yao's Principle:

If there is a probability distribution D on union of "positive" ("yes"/"pass") & "negative" ("no"/"fail") inputs, s.t. any deterministic algorithm of query complexity $\leq t$ outputs incorrect answer with probability $\geq 1/3$ on inputs chosen according to D, then t is a lower bound on the randomized query complexity.

moral: average case deterministic l.b. $\Downarrow$ randomized worst case l.b.

principle works for all types of randomized algorithms

why? proof omitted

Game theoretic view:

Alice selects deterministic algorithm A

Bob selects input x

payoff = cost of $A(x)$

Von Neuman's minimax $\Rightarrow$ Bob has randomized strategy which is as good when A randomized.

Proof of Theorem

Plan: give distribution on inputs that is hard for all deterministic algorithms with $o(\sqrt{n})$ queries.
Then Yao $\Rightarrow$ randomized l.b. of $\Omega(\sqrt{n})$

without loss of generality

wlog assume $6 | n$

Distribution on negative inputs:

N = random string of distance $\geq \varepsilon n$ from L_n

should output "Fail" on these

Distribution on positive inputs:

$$P = \begin{cases} 1. \text{ pick } k \in_R \left[\frac{n}{6}+1, \frac{n}{3}\right] \\ 2. \text{ pick random } v, u \text{ s.t. } |v| = k, \ |u| = \frac{n-2k}{2} \\ 3. \text{ output } vv^R uu^R \end{cases}$$

should output "Pass" on these

note:
choice of k
$\Rightarrow |vv^R| + |uu^R|$
are both $\geq n/3$
$\& \leq 2n/3$

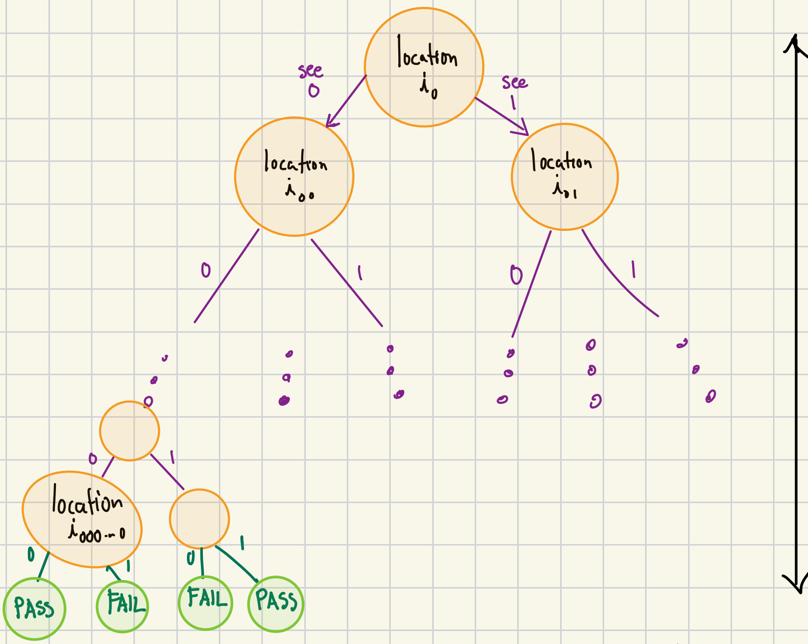
note: some strings can be generated by ≥ 1 k
e.g. $00000000 \cdots 0$

Distribution $\mathcal{D}$:

- flip coin
- H: output according to N
  T: output according to P

Assume (for contradiction) that there is a deterministic algorithm $\mathcal{A}$ using $\leq t = o(\sqrt{n})$ queries

Consider query tree of $\mathcal{A}$:



depth t

$\leq 2^t$ root-leaf paths

wlog, all leave have depth t

Output leaves labelled with $\mathcal{A}$'s answer when follows path

hopefully inputs in P reach "pass" leaves
" " N " "fail" leaves

Note: we can calculate probability of reaching leaf since we <u>know</u> input distribution $\mathcal{D}$

Error of leaf ℓ: if ℓ is labelled:

should fail

Pass: $E^-(\ell) = \{ \text{inputs } w \in \{0,1\}^n \mid w \ \varepsilon\text{-far} + w \text{ reaches leaf } \ell \}$

Fail: $E^+(\ell) = \{ \text{inputs } w \in \{0,1\}^n \mid w \in L + w \text{ reaches leaf } \ell \}$

should pass

Total error of A on $\mathcal{D}$:

$$= \sum_{\substack{\ell \\ \text{"pass"}}} \Pr_{w \in \mathcal{D}} [w \in E^-(\ell)] + \sum_{\substack{\ell \\ \text{"fail"}}} \Pr_{w \in \mathcal{D}} [w \in E^+(\ell)]$$

should fail but reach passing leaf

should pass but reach failing leaf

Why is this big?

will show lots of input from both N+P end up at <u>all</u> leaves.

Main Claims

Claim 1 if $t = o(n)$, $\forall\, \ell$ at depth t

$$\Pr_{\mathcal{D}}[\, w \in E^-(\ell)\,] \geq \left(\tfrac{1}{2} - o(1)\right) 2^{-t}$$

Claim 2 if $t = o(\sqrt{n})$, $\forall\, \ell$ at depth t

$$\Pr_{\mathcal{D}}[\, w \in E^+(\ell)\,] \geq \left(\tfrac{1}{2} - o(1)\right) 2^{-t}$$

So total error of $\mathcal{A}$ on $\mathcal{D}$ is

$$= \sum_{\substack{\ell \\ \text{PASS}}} \left(\tfrac{1}{2} - o(1)\right) 2^{-t} + \sum_{\substack{\ell \\ \text{FAIL}}} \left(\tfrac{1}{2} - o(1)\right) 2^{-t} \geq \tfrac{1}{2} - o(1) \gg \tfrac{1}{3}$$

$\blacksquare$

still need to prove the claims ...

Pf of Claim 1

idea: $\mathcal{N}$ is close to $\mathcal{U}$

& $\mathcal{U}$ would end up uniformly distributed at each leaf

$$\Rightarrow \Pr_{w \in \mathcal{U}}[w \in E^-(\ell)] = \frac{2^{n-t}}{2^n} = 2^{-t}$$

but how much can distribution change by using $\mathcal{N}$ instead of $\mathcal{U}$?

$$|L_n| \le \underset{\substack{\uparrow \\ \text{Choice of } u,v}}{2^{n/2}} \cdot \underset{\substack{\uparrow \\ \text{Choice of } i}}{n/2}$$

words at dist $\le \varepsilon$ from L_n:

$$\le 2^{n/2} \cdot n/2 \cdot \sum_{i=0}^{\varepsilon n} \binom{n}{i} \le 2^{n/2 + 2\varepsilon \log(\frac{1}{\varepsilon}) n}$$

so $$E^-(\ell) \ge \underset{\substack{\uparrow \\ \text{\# strings that reach } \ell}}{2^{n-t}} - \underset{\substack{\uparrow \\ \text{\# strings that allowed to pass}}}{2^{n/2 + 2\varepsilon \log(1/\varepsilon) n}} = (1 - o(1)) 2^{n-t}$$

← assume worst case! they all come to leaf ℓ

so $$\underset{\substack{\uparrow \\ \text{needs to come from } \mathcal{N}}}{\Pr_{\mathcal{D}}[w \in E^-(\ell)] \ge \frac{1}{2} \Pr_{\mathcal{N}}[w \in E^-(\ell)]} \ge \frac{1}{2} \frac{|E^-(\ell)|}{\#\text{strings in } \mathcal{N}} \ge \frac{1}{2} \frac{|E^-(\ell)|}{\underset{\substack{\uparrow \\ \text{upper bnd on \# strings in } \mathcal{N}}}{2^n}} \ge \left(\frac{1}{2} - o(1)\right) 2^{-t}$$

∎

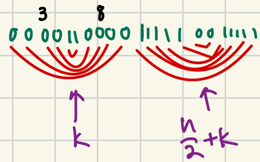
Proof of claim 2

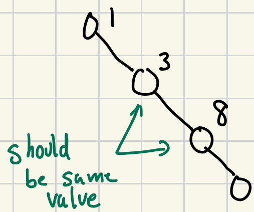
will show: for every fixed set of $o(\sqrt{n})$ queries, lots of strings in L_n follow that path!

count # strings agreeing with t queries of leaf: 2^{n-t}

count # strings in L_n " " " " " : $\geq 2^{n-t} - ?$

Main difficulty:





Fix $k = 10$

Should see same value at

1, 10	11, n
2, 9	12, n−1
3, 8	13, n−2
4, 7	⋮
5, 6	

☹ maybe no string in L_n follows the path?

☺ that's why we pick k randomly in $[n/6 \,..\, n/3]$!

not all queries bad e.g. for most strings, 3 + 8 are not correlated

does choice of k correlate the queries?

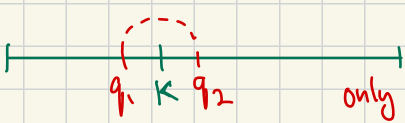
For given leaf ℓ, let $Q_\ell \leftarrow$ indices queried on path to ℓ

For each of $\binom{t}{2}$ pairs of queries $q_1, q_2 \in Q_\ell$,

at most 2 choices of k for which

q_1, q_2 symmetric around either k or $\frac{n}{2} + k$

e.g.



need $k = \frac{q_1 + q_2}{2}$

only 1 choice in this case

$\Longrightarrow$ # choices of k ("good k") s.t. $\underline{\underline{\text{no}}}$ $\underline{\underline{\text{pair}}}$ in Q_ℓ symmetric around k or $\frac{n}{2}+k$ is

$$\geq \frac{n}{6} - 2\binom{t}{2} = (1-o(1))\frac{n}{6}$$

using that t is $o(\sqrt{n})$

for these good k, # strings that follow path is $2^{n/2-t}$

So $\Pr_p[w \in E^t(\ell)] = \sum_w \sum_k \underbrace{\Pr_p(w|k)}_{2^{-n/2}} \underbrace{\Pr[\text{choose } k]}_{6/n} \cdot \mathbb{1}_{w \in E^t(\ell)}$

$$\geq \frac{1}{\left(\frac{n}{6}\right)\left(2^{n/2}\right)} \left[(1-o(1)) \cdot \frac{n}{6}\right] \cdot 2^{n/2-t}$$

good k

strings following path on good k

$$= (1-o(1)) \cdot 2^{-t}$$

$\blacksquare$