09/11/21

BPP = two-sided error.

One-Sided error

Defn:- RTIME$(T(n))$ to be the class of languages L s.t. there is a PTM M running in time $T(n)$ such that

1) $x \in L \Rightarrow \Pr[M(x) = 1] \geq 2/3$.
2) $x \notin L \Rightarrow \Pr[M(x) = 0] = 1$

Alternative Defn:- $\exists$ DTM M' and random string $r \in \{0,1\}^{p(|x|)}$

1) $x \in L \Rightarrow \Pr_{r \in \{0,1\}^{p(|x|)}}[M'(x,r) = 1] \geq 2/3$
2) $x \notin L \Rightarrow \Pr_{r \in \{0,1\}^{p(|x|)}}[M'(x,r) = 0] = 1.$

Defn:- $RP := \bigcup_c RTIME(n^c)$

Q: Is $RP \subseteq BPP$? YES

$$\begin{cases} x \in L \Rightarrow \Pr[M(x)=1] \geq 2/3 \\ x \notin L \Rightarrow \Pr[M(x)=0] \geq 2/3 \end{cases}$$

Q: is $RP \subseteq NP$? YES!

NP requires.

$$\begin{cases} \cancel{x \in L}\ \Pr_r[N(x,r)=1] \geq \frac{1}{2^{p(|x|)}} \\ x \notin L\ \Pr_r[N(x,r)=0] = 1. \end{cases}$$

Obs: $P \subseteq RP \subseteq BPP$
$\subseteq NP$

OPEN :- BPP $\subseteq$ NP ?

Defn : coRP := $\{L \mid \bar{L} \in RP\}$

or, $x \in L \Rightarrow \Pr_r[M'(x,r) = 1] = 1$.

$x \notin L \Rightarrow \Pr_r[M'(x,r) = 0] \geq 2/3$

where M' is a DTM running in polynomial time.

PRIMES $\in$ RP ? $\in$ coRP ?

PRIMES $\in$ coRP via the algo. we saw in last class.

PIT $\in$ coRP. via the algo we saw in last class.

OPEN : PIT $\in$ P ? (Probably, PIT $\in$ RP ?).

Q: Does coRP $\subseteq$ BPP ? YES.

Suppose you have a RP algorithm for a language L.

And also you have a coRP algorithm for the same language L.

Q. Can you devise a poly-time algorithm that never errs*? (but it runs in expected poly-time)

when an RP algorith on an input says accept then you know for sure that the input is in the language.

Similarly when coRP algorithm says reject then you know for sure that the input is not in the language.

$x \in L? \longrightarrow$ RP-algo gave reject.
coRP-algo gave accept.

Suppose $x \in L$,

1st time:- RP-algo gave reject.

Pr of error $\leq 1/3$

coRP-algo gave accept

Pr of error $\leq \frac{1}{3}$

2^{nd} time:- RP-algo gave reject

Pr of error $\leq \frac{1}{3}$

coRP-algo gave accept.

Pr of error $\leq \frac{1}{3}$

After two runs, Pr of error $\leq \frac{1}{9}$

if you run both algo. K- times

Pr of error $\leq \left(\frac{1}{3}\right)^K$

what is the runtime $K \cdot 2 \cdot$ poly-time.

if K is polynomial then the whole run time is poly.

Defn:- $ZPP := \bigcup_c ZTIME(n^c)$
(zero-sided error)

when $ZTIME(T(n))$ is the class of languages

L s.t. $\exists$ a PTM M that

runs in expected time $O(T(n))$

such that for every input x,

if M halts on x, then it outputs the correct answer.

Expected time on input x

$$= \sum_{\substack{\text{random string} \\ r}} \left[\text{Prob. that the random string is } r\right] \cdot \left[\begin{matrix}\text{Time taken} \\ \text{on input} \\ x, r\end{matrix}\right]$$

<u>Thm</u>:- $RP \cap coRP \subseteq ZPP$ ← (Exercise)

<u>infact</u>, $ZPP = RP \cap coRP$

<u>Proof</u>:- $ZPP \subseteq RP \cap coRP$.

<u>$(ZPP \subseteq RP)$</u> $\cap$ co-RP

$L \in ZPP$, $\Rightarrow$ $\exists$ a PTM M with expected running time $q(|x|)$ on input x, where $q(\cdot)$ is polynomial.

<u>(co) RP algorithm</u>

i) On input x, Run the ZPP machine M for at most $\underbrace{2 \cdot q(|x|)}_{3 \cdot q(|x|)}$ time.

2) If M stops within this time, output M's answer.

3) otherwise output reject.
(accept)

when $x \in L \Rightarrow$ there is a possibility of error.

$x \notin L \Rightarrow$ Pr of error $= 0$.

Pr. of error $\leq$ Pr. that M does not stop is $\dfrac{2 \cdot q(|x|)\text{ time}}{3 \cdot q(|x|)}$

Markov's Inequality:
For any non negative random Variable X and $a > 0$.
$$\Pr[X \geq a \cdot E[X]] \leq \frac{1}{a}$$

X = runtime of M on input x.
(different random choices leads to different run time).

$E[X] \leq q(|x|)$.

Pr that M doesn't stop in $\underbrace{2 \cdot q(|x|)}_{3 \cdot q(|x|)}$ time

$= \Pr[X \geq 2 \cdot q(|x|)]$

$= \Pr[X \geq 2 \cdot E[X]]$

by Markov's Inequality

$\leq \frac{1}{2}$.

Pr of error $\leq \frac{1}{3}$.

Error-reduction

Defn :- BPP_δ for $0 < \delta < \frac{1}{2}$ defines the class BPP s.t. Prob. of error $\leq \delta$.

Defn :- $BPP_{\frac{1}{2} - \frac{1}{n^c}}$ for $c > 0$ defines the class BPP s.t Prob. of error $\leq \frac{1}{2} - \frac{1}{n^c}$

Thm :- $BPP_{\frac{1}{2} - \frac{1}{n^c}} = BPP_{\frac{1}{2^{n^d}}}$ for all $c, d > 0$.

Proof :- $BPP_{\frac{1}{2} - \frac{1}{n^c}} \subseteq BPP_{\frac{1}{2^{n^d}}}$; $BPP_{\frac{1}{2^{n^d}}} \subseteq BPP_{\frac{1}{2} - \frac{1}{n^c}}$ (easy)

By defn :- if Prob. of error $\leq \frac{1}{2^{n^d}}$

$\leq \frac{1}{2} - \frac{1}{n^c}$

$$BPP_{\frac{1}{2}-\frac{1}{n^c}} \subseteq BPP_{\frac{1}{2^{n^d}}}$$

$L \in BPP_{\frac{1}{2}-\frac{1}{n^c}} \Rightarrow \exists$ a PTM M with run time $q(|x|)$.

s.t.

$$x \in L \Rightarrow Pr[M(x) = 1] \geq \frac{1}{2} + \frac{1}{n^c}$$

$$x \notin L \Rightarrow Pr[M(x) = 0] \geq \frac{1}{2} + \frac{1}{n^c}.$$

where $|x| = n$.

Algo :-

(1) Run M on input x independently k times.

Let $z_1, \ldots, z_k$ be the output of M on x on these k-runs.

where $z_i \in \{0,1\}$.

(2) Output the Majority of

$Z_1, \ldots, Z_K$.

Define a random variable

$$Y_i = \begin{cases} 1 & \text{if } Z_i \text{ is the correct answer.} \\ 0 & \text{otherwise.} \end{cases}$$

for $1 \leq i \leq K$.

$\Pr[Y_i = 1]$ = Prob. that M gives the correct answer.

$\rho \geq \frac{1}{2} + \frac{1}{n^c}$

Define $Y = \sum_{i=1}^{K} Y_i$

Define $\mu := \mathbb{E}[Y] = \sum_{i=1}^{k} \mathbb{E}[Y_i]$

$$= \sum_{i=1}^{K} 1 \cdot \Pr[Y_i = 1] + 0 \cdot \Pr[Y_i = 0]$$

$$= \sum_{i=1}^{k} \Pr[Y_i = 1]$$

$$= K\rho \geq K\left(\frac{1}{2} + \frac{1}{n^c}\right).$$

Chernoff-Bound:

Let $Y_1, \dots, Y_k$ be independent 0-1 random variables.

s.t. $\Pr[Y_i = 1] = p_i$.

Let $Y = \sum_{i=1}^{k} Y_i$ and $\mu := E[Y]$

Then for $0 < \delta < 1$,

$$\Pr[Y \le (1-\delta)\mu] \le e^{-\frac{\delta^2 \mu}{2}}$$

$$\Pr[Y \ge (1+\delta)\mu] \le e^{-\frac{\delta^2 \mu}{3}}$$

the machine is wrong whenever the majority is wrong.

we want.

$$(1-\delta)\, k \cdot \rho \geq \frac{k}{2}$$

$$\Rightarrow \quad 1-\delta \geq \frac{1}{2\rho}$$

$$\Rightarrow \quad \delta \leq 1 - \frac{1}{2\rho}.$$

Prob. of error. $\leq \text{Prob}\left[Y \leq \frac{k}{2}\right]$

$$\leq \text{Prob}\left[Y \leq (1-\delta)\, k \cdot \rho\right]$$

$$\leq \Pr\left[Y \leq (1-\delta)\, \mathbb{E}[Y]\right]$$

$$\leq e^{-\frac{\delta^2 \mathbb{E}[Y]}{2}}$$

$$= e^{-\frac{\delta^2 \cdot k \cdot \rho}{2}} \leq \frac{1}{2^{n^d}}$$

$$\Rightarrow e^{-\frac{\left(1-\frac{1}{2\rho}\right)^2 \cdot k \cdot \rho}{2}} \leq \frac{1}{2^{n^d}}$$

$$\Rightarrow \left(1-\frac{1}{2\rho}\right)^2 \cdot k \cdot \rho \cdot 2 \geq n^d$$

$$\Rightarrow \quad k \geq \frac{n^d}{2\cdot\rho\left(1-\frac{1}{2\rho}\right)^2}$$

$$\Rightarrow \quad k \geq \frac{n^d}{2\cdot\rho\cdot\frac{1}{n^{2c}}} \geq \frac{n^d\cdot n^{2c}}{2}$$

$\Rightarrow$ # of runs you need is n^{d+2c}.