Michael Forbes
Mforbes@illinois.edu

2019-09-17.2 ⇄ 2019-09-17.1
cs473

cs473 Algorithms: Lecture 7

logistics: - pset 2 due W10

last time: - dynamic programming, optimized
- edit distance
- longest increasing subsequence

today: - probability review
- randomized algo
- quicksort

---

intro:

Q: what is the "most realistic" model of computation? - can run on actual computers
- captures abilities of actual computers
↳ rand()?

randomized algo: rand(k) - $O(1)$ op
- returns $0 \le i < k$ w/p $\frac{1}{k}$ [[uniform distribution]]

[[how does this help?]]

Q: who will win the next US election? [["computational" problem]]

A: ask everyone
~328 million people
~235 million eligible voters
~138 million actual voters
a lot [[expensive! do better!]]
[[semi-legit numbers]]

[[faster!]]

fact: a poll of 738 uniformly random voters will
- estimate actual vote to ±5% [[error]]
- with probability ≥ 95% [[Chernoff]]

[[is actually related to algo]]
[[hard to get uniform voters who respond]]
[[real life is more complicated]]

Q: why rand algo?

A: - can be simpler, but analysis can be more complicated
- " faster, but may fail with some probability
↳ sometimes only known efficient algo is randomized, eg primality testing [[before AKS]]

Q: what is randomness?

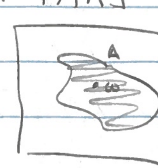
def. Ω finite/countably-infinite set, $Pr: \Omega \to [0,1]$. Then (Ω, Pr) is a discrete probability space if $\sum_{\omega \in \Omega} Pr[\omega] = 1$.

Rmk. - can define general probability space, eg $\Omega = \mathbb{R}$, but is tricky
- discrete spaces suffice here

eg: - $\Omega = \{HH, HT, TH, TT\}$, $Pr[\omega] = 1/4$ ⟵ two fair dice
- $\Omega = \{0, 1, 2, \ldots\}$ [[infinite]], $Pr[i]$ = probability algo takes i steps

def. (Ω, Pr) probability space. An event A is subset $A \subseteq \Omega$. $Pr[A] := \sum_{\omega \in A} Pr[\omega]$

ex: $A = \{HH, HT\}$ ⟵ first coin is heads
$Pr[A] = 1/2$



Rmk. events are sets, so can use set operations
- $\bar{A} = \Omega \setminus A$
- $A \cap B$, $A \cup B$

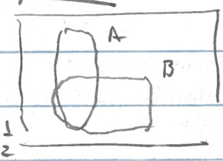
Michael Forbes
mforbes@illinois.edu
2019-09-17.2 ← 2019-09-17.1
→ 2019-09-17.3
CS473

<u>def.</u> $(\Omega, \Pr)$ prob. space. A, B events. A and B are <u>independent</u> if $\Pr[A \cap B] = \Pr[A]\Pr[B]$, otherwise <u>dependent</u>.

<u>ex:</u> $\Omega = \{HH, HT, TH, TT\}$ $\Pr[\omega] = 1/4$

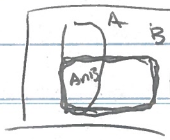
$A = \{HT, HH\}$, $B = \{TT, HT\}$

$\Pr[A] = \Pr[B] = 1/2$, $\Pr[A \cap B] = \Pr[HT] = 1/4 = \frac{1}{2}\cdot\frac{1}{2}$



<u>def:</u> $(\Omega, \Pr)$ prob space. $A \subseteq \Omega$ event w/ $\Pr[A] \neq 0$. Define the <u>conditional probability</u> of $\omega \in \Omega$ to be $\Pr[\omega \mid A] := \frac{\Pr[\omega \wedge A]}{\Pr[A]} = \begin{cases} \Pr[\omega]/\Pr[A] & \omega \in A \\ 0 & \text{else} \end{cases}$

<u>lem.</u> $(\Omega, \Pr)$ prob space. A, B events, $\Pr[B] \neq 0$. Then
- $(B, \Pr[\cdot \mid B])$ is a probability space
- $\Pr[A \cap B] = \Pr[A \mid B] \cdot \Pr[B]$
- A, B independent iff $\Pr[A \mid B] = \Pr[A]$



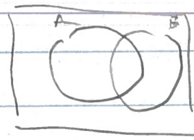
<u>pf:</u> exercise

<u>lem:</u> $\Pr[A \cup B] = \Pr[A] + \Pr[B] - \Pr[A \cap B] \leq \Pr[A] + \Pr[B]$

↳ union bound

[very simple, but powerful]

<u>pf:</u>



<u>def:</u> $(\Omega, \Pr)$. A <u>random variable</u> is $X : \Omega \to \mathbb{R}$

$X : \Omega \to \mathbb{R}$, $Y : \Omega \to \mathbb{R}$ are <u>independent</u> if all $\alpha, \beta \in \mathbb{R}$ the events $\mathbb{1}\{X = \alpha\}$ and $\mathbb{1}\{Y = \beta\}$ are independent

<u>equiv.</u> $\Pr[X = \alpha \wedge Y = \beta] = \Pr[X = \alpha] \cdot \Pr[Y = \beta]$

<u>ex:</u> $\Omega = \{HH, HT, TH, TT\}$ $\Pr[\omega] = 1/4$

$X = 1 \quad 1 \quad 0 \quad 0$
$Y = 0 \quad 1 \quad 0 \quad 1$
} X = first coin, Y = second coin } ⇒ independent

<u>def:</u> $(\Omega, \Pr)$, X rand. var. The <u>expectation</u> of X is $\mathbb{E}[X] := \sum_{\omega \in \Omega} X(\omega) \cdot \Pr[\omega]$

$A \subseteq \Omega$ event, $\Pr[A] \neq 0$. The <u>conditional expectation of X conditioned on A</u> is

$$\mathbb{E}[X \mid A] := \sum_{\omega \in \Omega} X(\omega) \cdot \Pr[\omega \mid A] = \sum_{\omega \in A} \frac{X(\omega) \cdot \Pr[\omega]}{\Pr[A]}$$

<u>lem:</u> X, Y independent ⇒ $\mathbb{E}[XY] = \mathbb{E}[X] \cdot \mathbb{E}[Y]$ [exercise]

<u>lem:</u> X, Y $\boxed{\text{arbitrary}}$ $\mathbb{E}[X + Y] = \mathbb{E}[X] + \mathbb{E}[Y]$ [exercise]

<u>def:</u> if X rand var is $X : \Omega \to \{0,1\}$ it is a <u>binary random variable</u>

If $A \subseteq \Omega$ and $X(\omega) = \mathbb{1}[\omega \in A]$ then X is an <u>indicator random variable</u>

<u>lem:</u> $X = \mathbb{1}_A$. Then $\mathbb{E}[X] = \Pr[A]$

<u>pset0:</u> to compute $\mathbb{E}[X]$ should write $X = \sum X_i$ ⇒ $\mathbb{E}[X] = \sum \Pr[A_i]$

$= \mathbb{1}_{A_i}$

Michael Forbes
miforbes@illinois.edu
2019-09-17.2 → 2019-09-17.3
2019-09-17.4
cs473

rand(k) → {0, 1, ..., k-1} uniform dist

Q: how to model randomized algo?

models: deterministic: deterministic algo f

worst case input x

$x \mapsto f(x)$

complexity: $T(n) = \max_{|x|=n} T(x)$

probabilistic: deterministic f

$x \leftarrow \mathcal{D}_n$ input [distribution] $n=|x|$

$x \mapsto f(x)$

complexity: $T(n) = \mathbb{E}_{x \leftarrow \mathcal{D}_n} T(x)$

rmk: - $\mathcal{D}$ often unknown in real life

- theory is brittle

randomized algo: randomized algo f ← algo creates randomness via rand(·)

worst case input: x [not random]

$x \mapsto f(x)$

complexity $T(n) = \max_{|x|=n} \mathbb{E}[T(x)]$ ← random var

rmk: "right" worst case notion

def: f randomized algo, the error probability of f

- at x: $E(x) := \Pr_{\text{randomness of } f}[f(x) \text{ is incorrect}]$

- worst case: $E(n) := \max_{|x|=n} E(x)$

def: A las vegas algo - $E(n) = 0$ [no error]

- $T(n) \le n^{O(1)}$ [fast expected runtime]

def: A monte carlo algo - $\forall x \; T(x) \le |x|^{O(1)}$ w/p 1 [always fast]

- $E(n)$ is "small" $\le 1/3$ [okay]

$\le 1/n$ [pretty good]

$\le 1/2^n$ [essentially always right but often overkill]

Rmk: - las vegas is "almost" deterministic

- real power comes from $E(n) > 0$

- $E(n) \le 1/2^{100}$ is "deterministic" in practice.

fact: any efficient randomized algo is wlog Monte Carlo [it's all about the error]

Q: example?

Quicksort$(a_1, \ldots, a_n) = \bar{a}$ [sort $\bar{a}$]

pick pivot a_{i_0} [somehow]

partition $\bar{a} = (\bar{b}, a_{i_0}, \bar{c})$ w/ $b_j < a_{i_0} < c_k$

return (quicksort$(\bar{b})$, a_{i_0}, quicksort$(\bar{c})$)

correctness: clear

Michael Forbes
miforbes@illinois.edu
2019-09-17.4 ← 2019-09-17.3
cs473

complexity: $j = \text{rank}(a_{i_0}) \Rightarrow |\bar{b}| = j-1 \quad |\bar{c}| = n-j$

$$T(n) \le O(n) + T(j-1) + T(n-j)$$
$$\le \cdots \le O(n^2)$$

and: if $j=1$ always $\quad T(n) \ge \Omega(n^2)$

main issue: choose pivot a_{i_0} so $\text{rank}(a_{i_0}) \le n/2$ [[live a balanced life]]

lem: pick a_{i_0} via $i_0 = \text{rand}(n)+1 \in [1, \ldots, n]$ [[will not be so formal w/ rand(·) in future]]

$\Rightarrow \text{rank}(a_{i_0}) \in [\frac{1}{6}n, \frac{5}{6}n]$ w/p $2/3$

"$\Rightarrow$" $T(n) \le T(\frac{5}{6}n) + T(n/6) + O(n)$ "with probability 2/3"

$\le \cdots \le O(n \lg n)$ ← not a formal proof

fact: can select a_{i_0} w/ $\text{rank}(a_{i_0}) = \lfloor n/2 \rfloor$ ← median, in $O(n)$ time

Rmk: - $\Rightarrow$ $O(n \lg n)$ deterministic quicksort
- but is not as efficient / simple as randomized case

thm: randomized quicksort w/ $i_0 = \text{rand}(n)+1$
- error probability 0 [[correctness]]
- for all $\bar{a}$, $\mathbb{E}[T(a_1, \ldots a_n)] \le O(n \lg n)$ [[complexity]] [[worst case]] [[may run in n^2 time in worst case]]

pf: $T(\bar{a})$ = # comparisons [[main bulk of work]] [[rand var]]

Ω = all calls to rand(·) [[complicated]] [[how to decompose?]]

$$\bar{a} \mapsto (\underbrace{\bar{b}}, a_{i_0}, \underbrace{\bar{c}})$$
rand var

[[linearity]]
$$\mathbb{E}[T(\bar{a})] \le O(n) + \mathbb{E}[T(\bar{b})] + \mathbb{E}[T(\bar{c})]$$
$$= O(n) + \sum_{i=1}^{n} \underbrace{\mathbb{E}[T(\bar{b}) \wedge \mathbb{1}[\text{rank}(a_{i_0})=i]]}_{= \underbrace{\mathbb{E}[T(\bar{b}) \mid \text{rank}(a_{i_0})=i]}_{\le T(i-1)} \cdot \underbrace{\Pr[\;]}_{=1/n}} + \sum_{i=1}^{n} \underbrace{\mathbb{E}[T(\bar{c}) \mid \text{rank}=i]}_{\le T(n-i)} \cdot \underbrace{\Pr[\text{rank}=i]}_{=1/n}$$

$$= O(n) + \frac{1}{n} \sum_{i=1}^{n} \left(T(i-1) + T(n-i)\right) \quad \leftarrow \text{any } \bar{a}$$

$\Rightarrow$ $T(n) \le O(n) + \frac{1}{n}$ "

$\le \cdots$ ← pset 0

$\le O(n \lg n)$

today: - probability
- rand algo
- quicksort

next time: - more quicksort
- concentration bounds