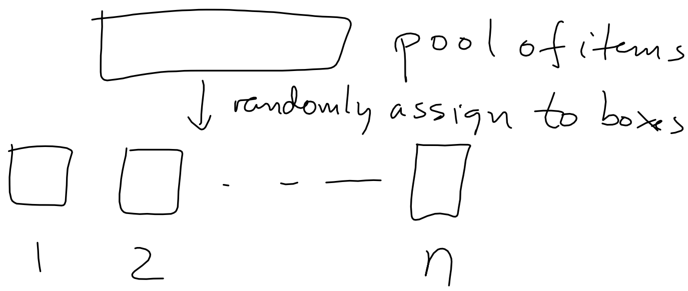
Coupon - collector



Goal: collect at least one copy of each of n different cards (above: get at least one element in each box)

Need to study the expected # of cards.

Define phases $S_0, S_1, S_2 \ldots \; S_n$

S_i: have precisely i distinct cards

Movement: $S_0 \to S_1 \to S_2 \cdots \to S_{n-1} \to S_n$

Let X_j be # cards collected in phase j

(if we get a new card, then go to phase $j+1$)

So X_j is a random variable. $X_n = 0$

and $X = \sum_{j=0}^{n-1} X_j$ is the total # of cards collected

so $E(X) = E\left(\sum_{j=0}^{n-1} X_j\right) = \sum_{j=0}^{n-1} E(X_j)$

How to find $E(X_j)$?

# of cards collected

$$0 \xrightarrow{X_0 = 1} 1 \qquad 2 \qquad j \xrightarrow{?} j+1$$

$p(\text{going from phase } S_j \text{ to } S_{j+1})$

$= p(\text{getting one of the } n-j \text{ remaining cards})$

$= \dfrac{n-j}{n} = p$ X_j is geometrically distributed.

So $E(X_j) = \dfrac{1}{p} = \dfrac{n}{n-j}$

$$E(X) = \sum_{j=0}^{n-1} E(X_j) = \sum_{j=0}^{n-1} \frac{n}{n-j} = n \sum_{j=0}^{n-1} \frac{1}{n-j}$$

$$= O(n \log n)$$

13.4 Randomized approximization alg for max 3-SAT

Problem Given a 3-SAT formula φ
Find a truth assignment which maximizes # satisfied clauses.

typical 3-SAT formula:

$$(x_1 \vee \bar{x}_2 \vee x_3) \wedge (\bar{x}_2 \vee x_3 \vee x_4) \wedge \cdots \wedge (x_1 \vee x_6 \vee \bar{x}_{11})$$

n variables $x_1, x_2 \dots x_n$
m clauses, all of size 3

literal: x_i or $\bar{x}_i$
each clause has 3 literals

Algorithm: randomly assign true or false to each x_i (call this assignment t)

Claim The expected # of the clauses that will be satisfied is $\frac{7}{8}m$

Analysis: Define indicator random variables

X_i, $i \in \{1, 2, \ldots m\}$

$$X_i = \begin{cases} 1 & \text{if } C_i \text{ is satisfied under } t \\ 0 & \text{otherwise} \end{cases}$$

$p(X_i = 1) = ?$

$p(X_i = 0) = \left(\frac{1}{2}\right)^3$ so $p(X_i = 1) = 1 - \frac{1}{8} = \frac{7}{8}$

and $E(X_i) = p(X_i = 1) = \frac{7}{8}$

So with $X = \sum X_i$ total # satisfied clauses

We get $E(X) = \sum_{i=1}^{m} E(X_i) = \sum_{i=1}^{m} \frac{7}{8} = \frac{7m}{8}$

Consequence: $\exists$ an assignment which satisfies at least $\frac{7m}{8}$ clauses

This follows from the fact that

$E(Y) \geq k \Rightarrow \exists s \in S$ s.t $Y(s) \geq k$

The algorithm will find a solution whose expected value is at least a factor of $\frac{7}{8} \cdot$ optimum.

Since optimum $\leq m$.

What do we do if the algorithm generates a truth assignment with $< \frac{7m}{8}$ satisfied clauses?

Try again!

We want to estimate how many times we should repeat the algorithm.

Can do this by estimating p = prop of success in a given round. Then E # repetitions $= \frac{1}{p}$

Estimating p

Define p_j as probability of satisfying precisely j clauses with the random truth assignment

Then using the formula for expected value, we get

$$\frac{7m}{8} = E(X) = \sum_{j=0}^{m} j \cdot p_j$$

Define m' as largest integer strictly less than $\frac{7m}{8}$

$$= \sum_{j<\frac{7m}{8}} j \cdot p_j + \sum_{j \geq \frac{7m}{8}} j \cdot p_j$$

$$\leq \sum_{j<\frac{7m}{8}} m' p_j + \sum_{j \geq \frac{7m}{8}} m \cdot p_j$$

$$= m' \sum_{j<\frac{7m}{8}} p_j + m \underbrace{\sum_{j \geq \frac{7m}{8}} p_j}_{=p}$$

$$= m'(1-p) + mp$$

$$\leq m' + mp$$

So we have seen that

$$\frac{7m}{8} \leq m' + mp$$

$$\Downarrow$$

$$mp \geq \frac{7m}{8} - m' \geq \frac{1}{8} \Rightarrow p \geq \frac{1}{8m}$$

So we expect to repeat at most

$$\frac{1}{p} = 8m \text{ times}$$

13.5 Randomized median finding

Given $n = 2k+1$ distinct numbers $a_1, a_2 \ldots a_{2k+1}$

Find the $k+1$'st number in their sorted order

Easy if we use sorting (merge sort) in $O(n \lg n)$. We want to be faster $O(n)$ in expectation. (It is possible to do it with a deterministic alg in $O(n)$ see DM553)

recall idea in Quicksort:

$$\boxed{<x} \quad \boxed{x} \quad \boxed{>x} \qquad x \text{ pivot}$$

applied to median problem:

$$\underset{I_<}{\boxed{<x}} \quad \boxed{x} \quad \underset{I_>}{\boxed{>x}}$$

- if $|I_<| = k$ then x is the median
- if $|I_<| > k$ then median is in $I_<$
- if $|I_<| < k$ then - - - - $I_>$

and now we look for element number

$$k+1-|I_<|-1 = k-|I_<|$$

Consequence: we need to solve more general problem Select(S,t): find t'th element in S

What do we need to get linear (expected) running time?

Suppose we could guarantee that

$\min\{|I_<|, |I_>|\} \geq \varepsilon n$ for constant ε

Then $T(n) = T((1-\varepsilon)n) + cn$

$\leq cn\,[1 + (1-\varepsilon) + (1-\varepsilon)^2 + \dots +]$

$= cn \frac{1}{1-(1-\varepsilon)} = \frac{cn}{\varepsilon} = O(n)$

Randomized alg : pick random pivot each time

Define phases

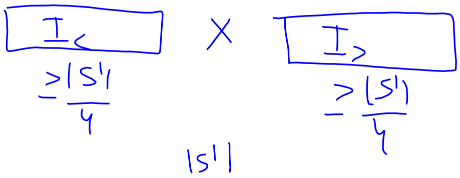
phase j: actual set S' satisfies

$$n\left(\frac{3}{4}\right)^{j+1} \leq |S'| \leq n\left(\frac{3}{4}\right)^{j}$$

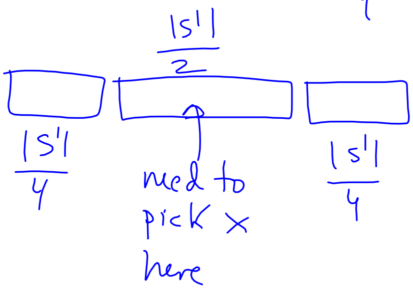
there are $O(\log n)$ phases

We will leave phase j and go to phase $j+1$ if the pivot x satisfies that

$$\min\{|I_>|, |I_<|\} \geq \frac{1}{4}|S'|$$



x is a good splitter



prob that x is a good splitter is $\frac{1}{2}$

$\Downarrow$ Expected # of pivots in phase j is $\frac{1}{\frac{1}{2}} = 2$

for all phases j

with X_i = work in phase i (excluding recursion)

we have $X = X_0 + X_1 + X_2 + \dots + X_t$ need to determine total work

$$E(X) = \sum_{j=0}^{t} E(X_j)$$

In phase j we $|S'| \leq n\left(\frac{3}{4}\right)^j$

so we do $c \cdot n\left(\frac{3}{4}\right)^j$ work (comparisons) per splitter

and we expect 2 splitters per phase

so $E(X_j) \leq 2cn\left(\frac{3}{4}\right)^j$ for all j

$$E(X) = \sum E(X) \leq \sum_{j=0}^{t} 2cn\left(\frac{3}{4}\right)^j = 2cn \sum_{j=0}^{t}\left(\frac{3}{4}\right)^j \leq 2cn \sum_{j=0}^{\infty}\left(\frac{3}{4}\right)^j = 2cn \frac{1}{1-\frac{3}{4}} = \underbrace{8c}_{\text{constant}}n$$

Similar analysis works for Quicksort:

$$\boxed{\quad I_< \quad} \; \boxed{x} \; \boxed{\quad I_> \quad}$$

modify randomness slightly:

pick new random pivot until it is a good splitter

Expect to repeat only twice as prob of getting a good splitter is $\frac{1}{2}$