Announcements

Apply for TA before 28 Nov
Sign up for courses, 30 Nov

Ind. var. weekly Note 4

(KT) Chernoff p. 759
(Cormen) Universal hash p. 265
(KT) universal hash p. 740
#of onto func p. 543

Today

Leftover exercises from last time
Using ind var. and chernoff
Universal hashing

Exercise on weekly note: Multiple choicese

n Questions

4 Answers to each, one of which is correct

Students can answer one of the 4, or leave blank

+1 for correct answer, $-\frac{1}{3}$ for incorrect, 0 if blank

Pass: $\geq \frac{n}{2}$ score.

Def: A challenged student knows $\leq 40\%$ of answers, and guess the rest ($\frac{1}{4}$ for each choice)

Def: A test is good if the prob that a challenged student passes is $\leq 5\%$

Let $m = \frac{3}{5}n$ ← The amount a chall stud guesses.

Let $X_i = \begin{cases} 1 & \text{if } i\text{'th guess correct} \\ 0 & \text{o.w.} \end{cases}$, $X = \sum_{i=1}^{m} X_i$

a) $E(X)$?

▷ First of all, what is $E(X_i)$?

↳ $E(X_i) = p(X_i = 1) = \frac{1}{4}$

$E(X) = \underset{\text{lin of exp}}{\sum} E(X_i) = \sum_{i=1}^{m} \frac{1}{4} = \frac{1}{4}m = \frac{3}{20}n$

Note this is not the exp score of 1 Q. It would be $S_i = \begin{cases} 1 \\ -\frac{1}{3} \end{cases}$ which is not an ind. var.

$E(S_i) = \frac{3}{4}\left(-\frac{1}{3}\right) + \frac{1}{4} \cdot 1$

$= -\frac{1}{4} + \frac{1}{4} = 0$

b) Show that a chall stud only passes if $X \geq \frac{3}{2}E(X)$

Continued

Exercise on weekly note: continued

what he actually answers correct

b) show chall stud only pass if $X \geq \frac{3}{2}E(X)$

i.e. show the corresponding score is $\geq \frac{n}{2}$.

▷ what could be the first thing to do?

↳ Expand $\frac{3}{2}E(X) = \frac{3}{2} \cdot \frac{3}{20} n = \frac{9}{40} n$

▷ Next?

compute score where $X \geq \frac{9}{40} n$

First: The stud gets $\frac{2}{5} n$ points for known answers.

And the stud gets $\frac{9}{40} n$ for the guesses.

But $\frac{3}{5} n - \frac{9}{40} n = \frac{24}{40} n - \frac{9}{40} n = \frac{15}{40} n = \frac{3}{8} n$ are incorrect.

The score for these: $-\frac{1}{3} \cdot \frac{3}{8} n = -\frac{1}{8} n$

Total score:

$$\frac{2}{5} n + \frac{9}{40} n - \frac{1}{8} n = \frac{16}{40} n + \frac{9}{40} n - \frac{5}{40} n = \frac{20}{40} n = \frac{1}{2} n$$

c) Use chernoff to find n where prob of passing chall stud is $\leq 0{,}05$ (5%). I.e. we want to bound $P(X > \frac{3}{2}E(X))$

If we choose $\delta = \frac{1}{2}$ (as $1 + \frac{1}{2} = \frac{3}{2}$) and $\mu = E(X) = \frac{3}{20} n$

we get

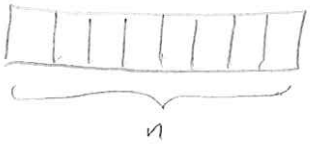
$$P\left(X > \frac{3}{2} \cdot \frac{3}{20} n\right) < \underbrace{\left(\frac{e^{\frac{1}{2}}}{\left(\frac{3}{2}\right)^{\left(\frac{3}{2}\right)}} \right)^{\frac{3}{20} n}}_{\text{just a constant}} \approx 0{,}897^{\frac{3}{20} n} \approx 0{,}984^n$$

Solve $0{,}984^n = 0{,}05$ for n gives $\rightarrow$ $n = 185$ (rounded up)

So we need 185 Qs to ensure $\leq 5\%$ prob.

Cormen 5.2-5

Let $A[1..n]$, array of distinct numbers



If for some $i<j$, $A[i] > A[j]$, then (i,j) is an inversion.

A contains random perm of $\{1,\dots,n\}$

Use ind. vars. to compute # of inversions.

▷ what should we create ind. vars. for? $X_{i,j} = \begin{cases} 1 & \text{if } \\ 0 & \end{cases}$

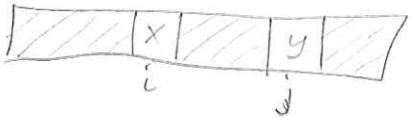
we try $X_{ij} = \begin{cases} 1 & \text{if } (i,j) \text{ is an inv} \\ 0 & \end{cases}$,

▷ How many X_{ij} do we have?

we only create X_{ij}'s for $i<j$, so for $i=1$ we have $n-1$, for $i=2$ we have $n-2$ ..., upto $i=n-1$ where we have 1. This can be expressed as $\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} 1 = \frac{n\cdot(n-1)}{2} = \binom{n}{2}$

▷ For each X_{ij}, what is $p(X_{ij}=1) = E(X_{ij})$?

↳ consider some i,j



$x, y \in \{1,\dots,n\}$. If $x>y$, (i,j) is an inv.

x can take on n different values, and
y can take on $n-1$ of the remaining values. } $\Rightarrow n\cdot(n-1)$ total outcomes

We are interested in those cases where $x>y$ which there are $\frac{n\cdot(n-1)}{2}$ of (Same logic as above).

$E(X_{ij}) = \frac{\frac{n\cdot(n-1)}{2}}{n(n-1)} = \frac{1}{2}$

▷ $X = \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} X_{ij}$, counts all inversions

▷ $E(X) = \sum\sum E(X_{ij}) = \frac{1}{2}\sum\sum 1 = \frac{1}{2}\cdot\frac{n\cdot(n-1)}{2} = \frac{n\cdot(n-1)}{4}$

▷ Fun fact: The running time of insertion sort and bubble sort is the same as the number of inversions in A as they ondo all of these step by step.

Cormen 5.3-7 Random Sample
p. 129

▷ We want random sample of size m of $\{1, 2, \ldots, n\}$.
Each subset of size m should be likely. $\leftarrow \frac{1}{\binom{n}{m}}$

▷ We could: RANDOM-IN-PLACE, take first m elems.
This is n calls to random()
If m is much smaller than n, this could be expensive.

▷ Suggested alg:

```
RAND-SAMPLE(m, n):        ← note {1,...,n} is implicit
  if m = 0:
    return ∅              } base case
  else:
    S = RAND-SAMPLE(m-1, n-1)
    i = random(1, n)
    if i ∈ S:
      S' = S ∪ {n}        ← S cannot contain n already as
    else: // i ∉ S
      S' = S ∪ {i}
  return S'
```

▷ Understanding alg: we assume we can create a sample of size $m-1$ (each one eq likely). We pick a new candidate to be included in S (prob $\frac{1}{n}$). If it is already present, we add the only elem we know can be present: n. Otherwise we add the candidate.

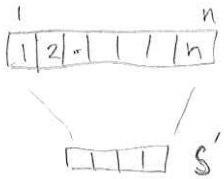
▷ Show alg creates subset with $\frac{1}{\binom{n}{m}}$ chance
Induction on m: base case $m = 0$: $\frac{1}{\binom{n}{0}} = \frac{1}{1} = 1$
There is only 1 subset of $\emptyset$, namely $\emptyset$.
Now assume we have sample of $\{1 \ldots n-1\}$ of size $m-1$ where each sample has prob $= \frac{1}{\binom{n-1}{m-1}}$. continued

Cormen 5.3-7 continued

First of all, let us write out $\binom{n-1}{m-1} = \frac{(n-1)!}{(m-1)!\,(n-1-(m-1))!}$

$= \frac{(n-1)!}{(m-1)!\,(n-m)!}$, so prob would be $\frac{1}{\binom{n-1}{m-1}} = \frac{(m-1)!\,(n-m)!}{(n-1)!}$

▷ Now fix some S' (set of m elements) that we want to show has prob $\frac{1}{\binom{n}{m}}$.



We consider different cases for how Alg could gen S. We can ask if $n \in S'$ or not

▷ Case 1: $n \notin S'$:

Consider all the elements in S'. For a given $a \in S'$ what is the prob that $S' \setminus \{a\}$ was created recursively?

↳ $\frac{1}{\binom{n-1}{m-1}}$, by induction hyp

There are m elem in S' out of n in total, so the prob for S

$$\frac{m}{n}\,\frac{1}{\binom{n-1}{m-1}} = \frac{m}{n}\,\frac{(m-1)!\,(n-m)!}{(n-1)!} = \frac{m!\,(n-m)!}{n!} = \frac{1}{\binom{n}{m}}$$

▷ Case 2: $n \in S'$:

Now we ask ourselves what the prob of $S' \setminus \{n\}$ is.

↳ $\frac{1}{\binom{n-1}{m-1}}$, again

And n was included in S' if i hit some already present elem or turned out to be n.

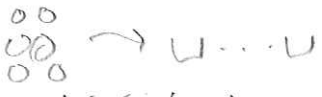
There are $m-1+1 = m$ out of n choices So, prob of S'

$$\frac{m}{n}\,\frac{1}{\binom{n-1}{m-1}} = \frac{1}{\binom{n}{m}}$$

Jan 13.2

5 tasks
10 people

Every person can help solve 1 task

a) First, we can see as  obj into boxes

What are obj and what are boxes?

↳ People are obj
Tasks are boxes, if a person is put "in" a task he/she helps solve it.

▷ # of ways to distro People if okay that not all tasks solved?

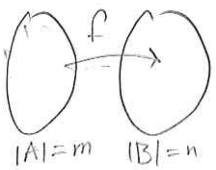
(Let's say both tasks and people are distinguishable.)

Then we have 5 choices for first person etc

$\Rightarrow 6^{10}$

b) What if all tasks must be solved (i.e. ≥ 1 on each task)?

↳ No boxes empty: # Onto functions / stirling



what is m? $\Rightarrow$ 10
what is n? $\Rightarrow$ 5

$$5^{10} - \binom{5}{1}(5-1)^{10} + \binom{5}{2}(5-2)^{10} - \binom{5}{3}(5-3)^{10} + \binom{5}{4}(5-4)^{10}$$

$$= 5^{10} - 5 \cdot 4^{10} + 10 \cdot 3^{10} - 10 \cdot 2^{10} + 5 \cdot 1^{10}$$

$$= 5103000$$

Jan 13.2 continued

c) Now we use a rand alg to distribute people to tasks with $\frac{1}{5}$ for each task

▷ what is prob that all tasks solved? (i.e. prob that no task is "empty"?)

↳ $\frac{b)}{a)} = \frac{5^{10} - 5 \cdot 4^{10} + 10 \cdot 3^{10} - 10 \cdot 2^{10} + 5}{5^{10}}$

$= \frac{5103000}{9765625} \approx 0,523$

d) Exp number of runs before all tasks solved? (not "empty")

↳ Each try is indep trial with success $\approx \frac{1}{2}$

⇒ Geometric distribution!

exp # $= \frac{1}{p} = \frac{1}{\frac{1}{2}} = 2$ (1,9 if we use 0,523)

e) For task j (↑ fixed, for instance task 3), what is exp # of people assigned to it?

↳ Let's use ind. var! Let $X_i = \begin{cases} 1 & \text{if person } i \text{ assigned to task 3} \\ 0 & \end{cases}$

$E(X_i) = \frac{1}{5}$, $E(X) = \frac{1}{5} \cdot 10 = 2$ $\quad X = \sum_{i=1}^{10} X_i$

f) For a given task show $P(X > 6) < 0,1$ $\quad E(X) = \mu$

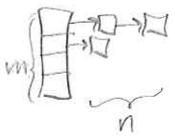
↳ use chernoff: $P(X > (1+\delta)2) < \left(\frac{e^2}{3^3}\right)^2 = 0,075$

(δ ↑)

11-4 Hashing and auth

Cormen
p. 284

Universal hashing: Recall hashing from DMS07

for instance chaining 

, m is size of table, n is the number of keys.

Load factor: $\alpha = \frac{n}{m}$, if m is some factor of $n \Rightarrow \underset{E}{O}(1)$ ops.

when we do this we have one hash func and assume that input is uniformly random – But input never is!
There is always some input that is bad for some hash func.

The solution: Have a family (set) H of hash func that we pick our hash func h from at random!

[It is sort of like some QuickSort impl shuffle the array first to not depend on the user input – now we are in control of how the input looks.]

We say a family H is universal if

$$\forall k, l \in U,\ k \neq l \Rightarrow \left| \{ h \in H \mid h(k) = h(l) \} \right| \leq \frac{|H|}{m}$$

U: universe of keys, for instance all ints

$\#$ of h's that cause collision

This is similar to saying $P(h(k) = h(l)) \leq \frac{1}{m}$ (chance of collision) where h chosen randomly from H.

One can show that this gives the desired load factor $\alpha = \frac{n}{m}$.

How to create such H:

Choose large prime $p > m$

Then for all $a \in \{1, 2, \ldots, p-1\}$ and $b \in \{0, 1, 2, \ldots, p-1\}$ we can design the family to be

$$h_{a,b}(k) = ((a \cdot k + b) \bmod p) \bmod m$$

So $|H| = \underbrace{(p-1)}_{a} \cdot \underbrace{p}_{b}$ | one can the show H is universal.

continued

11-4 continued

there one of course other ways to create H.

For instance imagine that all key are tuples of size n where each entry is from $\{0,1,\ldots,p-1\}$.

Now we can create a family by for each $a = (a_0, a_1, \ldots, a_{n-1})$ (also just a n-tuple key), create

$$h_a(K) = \left(\sum_{i=0}^{n-1} a_i k_i\right) \bmod p$$

(K: n-tuple key)

Now for the exercise:

First: A family H is k-universal if for all seq (tuple) of k distinct keys (eg. if $U = \mathbb{N}$ and $k = 3$ we can look at $(1,2,3), (1,2,4), \ldots$) and a hash func h chosen randomly from, then applying h to each entry (eg. $(h(1), h(2), h(3))$ or $(h(1), h(2), h(4))$) should have equal prob of hitting all sequences/tuples of len k from $\{0,1,2,\ldots,m-1\}$ ← the output domain for each h. we have m^k of these tuples we can hit.

a) Show if H is 2-universal $\Rightarrow$ H is universal (see prev page)

So now $k=2$.

So we have that all tuples (k_1, k_2) when applied h as $(h(k_1), h(k_2))$ has equal prob of being any $(\_, \_)$ (we have m^2 of these tuples). This prob is $\frac{1}{m^2}$ ($\_ \in \{0,1,2,\ldots,m-1\}$)

we must show $p(h(x) = h(y)) \leq \frac{1}{m}$, for random $h \in H$.

continued

11-4 continued

▷ What can we do? How can we use the 2-universal-ness?

Suppose wrant to determine $p(\underbrace{h(x) = h(y)}_{\text{collision}})$.

We can construct tuple (x, y) and apply h: $(h(x), h(y))$. We are interested in the cases where the result is (z, z) for $z \in \{0, 1, \dots, m-1\}$
In other words: when are $h(x)$ and $h(y)$ equal?

▷ How many (z, z)-pairs do we have?
↳ m, $(0,0), (1,1), \dots (m-1, m-1)$

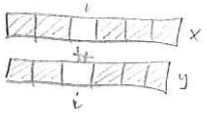
▷ Each $(-,-)$ has prob $\frac{1}{m^2}$ and we are interested in m of these
↳ $\frac{m_1}{m^2} = \frac{1}{m}$

b) We now have $h_a(k)$ hash family as described before

▷ First we have to show that this H is universal.
Jørgen showed this (or something similar) in lecture.
Also found in KT p. 740.

Note: $m \neq p$!

the idea is that we have $x = (x_0, \dots x_{n-1})$, $y = (y_0, \dots, y_{n-1})$
If $h(x) = h(y)$, but $x \neq y$ then $\exists\, x_i, y_i : x_i \neq y_i$



One can then argue that the choices for a_j's (where $j \neq i$) doesn't matter and a_i is what makes a difference.
As we have p different choices for a_i the prob is $\frac{1}{p}$

▷ we also have to show that $h_a(k)$ family is not 2-universal.

continued

11-4 continued

b)

▷ Show $h_a(k)$ not 2-universal

i.e. for $k = (k_1, k_2)$, the result $(h_a(k_1), h_a(k_2))$ does not have equal chance of hitting all $(\_, \_)$ ($\_ \in \{0, 1, \ldots, p-1\}$) using random h_a.

▷ If all h_a would produce the same output of $h_a(k)$ then something is bad.

Why? Suppose I picked some h_a and computed $h_a(k)$ and it gave a result I was unhappy with then I would like to pick some $h_{a'}$ and only have $\frac{1}{m} = \frac{1}{p}$ prob of getting some bad value.

▷ What key would give same result for all h_a?

↳ $0 = (0, 0, \ldots, 0)$, which would give $0 \bmod p = 0$

c) To deal with this we consider

$$h_{a,b}(x) = \left(\sum_{i=0}^{n-1} (a_i x_i) + b \right) \bmod p$$

(x: n-tuple)

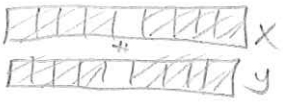
Here we have
a: n-tuple, like key
b: number $\in \{0, 1, 2, \ldots, p-1\}$

Show $h_{a,b}$ family is 2-universal.

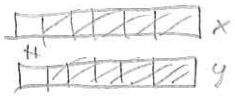
▷ Now $(a_0, a_1, \ldots, a_{n-1})$ and b one chosen randomly and we want to show that any key pair (x, y), $x \neq y$ ($x = (x_0, x_1, \ldots x_{n-1})$, $y = (y_0, y_1, \ldots, y_{n-1})$) gets mapped to (x', y') by $(h_{a,b}'(x), h_{a,b}'(y))$ equally likely.

continued

c) Firstof all, as $x \neq y$, some $x_i \neq y_i$



we can WLOG say it happens at $i = 0$



First some notation:

$$x' = h_{a,b}(x) \quad , \quad y' = h_{a,b}'(y)$$

$$A = \sum_{i=1}^{n-1} a_i x_i \quad , \quad B = \sum_{i=1} a_i y_i$$

$h_a(x)$ (no b) without x_0 term, $h_a(y)$ (no b) without y_0 term.

We can create equation

$$x' = a_0 x_0 + A + b \pmod p$$

$$y' = a_0 y_0 + B + b \pmod p$$

Now, as $x_0 \neq y_0$ and p is prime it turns out that the equations have a unique solution for a fixed x' and y' by tweaking a_0 and b_0. In other words we can make x' and y' whatever we want by chosing a_0 and b We have p^2 choices for a_0 and b and therefore p^2 choices for (x', y'). We can now choose $a_1, a_2, \ldots a_{n-1}$ of which there are $n-1 \Rightarrow p^{n-1}$ choices. All of these create the same (x', y') (we didn't really care about them), so all pairs equally likely.

continued

11-4 continued

d) Alice and Bob have agreed secretly on a random 2-universal hash func $h: U \to \mathbb{Z}_p$

(U: universe, e.g. all ints; $\mathbb{Z}_p$: $\{0, 1, \dots, p-1\} \pmod p$)

▷ Later $A \xrightarrow{m} B$ and she wants to make sure

(m: message, in U)

B knows it is her. A also sends tag $t = h(m)$

So $A \xrightarrow{m,t} B$

B now takes m and check if $h(m) = t$.

▷ Eve intercepts m, t and replace with m', t' trying to fool B.

▷ Argue that no matter how much computational power E has the prob that B accepts is' $\leq \frac{1}{p}$

↳ E must create m' s.t. $h(m') = t'$

and E does not know h, [illegible]

E could try all different h's to try and guess which h has used for m and t.

E has a problem with this strategy though:

Any (m, m') is equally likely to be any $(h(m), h(m'))$

(By being 2-universal) ($h(m) = t$)

t is fixed, so all $(t, h(m'))$ are equally likely.

There are p of these.

$\Rightarrow \frac{1}{p}$ guess