Lecture 5: *Probability Theory & Computing*

Advanced Algorithms

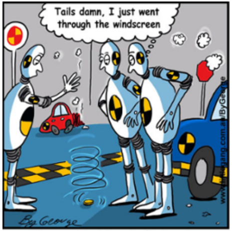
Sid Chi-Kin Chau

Australian National University

sid.chau@anu.edu.au

October 5, 2022

How to decide? Simply toss a coin ...



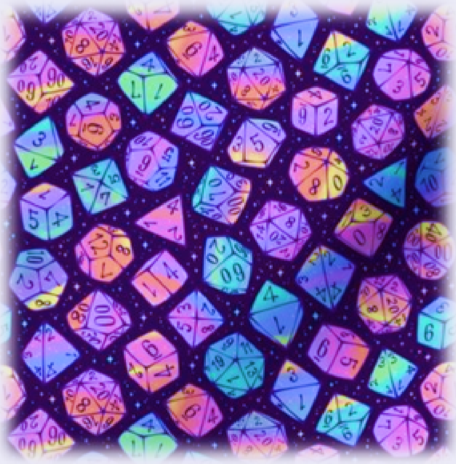
When crash test dummies flip the coin

"I'm beginning to question your business decision-making, Perkins."

"All my decisions are well thought out."

- We often make random decisions. Does it help? Did you make poor random decisions?

⋘ Probability Theory 101 ⋙



Probability Theory: Basics

- Probability theory formalizes the counting of random experimental outcomes
- Probability theory is an incredibly powerful tool for analysis, and it is a bridge between discrete math and calculus

Definition (Elements of Probability Theory)

- **Sample Space** Ω: A finite (or infinite) set of outcomes in a certain experiment
- **Probability Measure** $\mathbb{P} : \Omega \mapsto [0,1]$: A real-valued function that maps each element in sample space to a real number in $[0,1]$, such that
  - $\mathbb{P}(\omega) \in [0,1]$ for all $\omega \in \Omega$
  - $\sum_{\omega \in \Omega} \mathbb{P}(\omega) = 1$
- **Event** $E \subseteq \Omega$: A subset of outcomes, and define $\mathbb{P}(E) \triangleq \sum_{x \in E} \mathbb{P}(\omega)$
- **Random Variable** $X : \Omega \mapsto \mathbb{R}$: A mapping from an outcome to a real-valued observable quantity

Probability Theory: Example

Example (Rolling a Die Twice)

- Consider an experiment of rolling a die twice. The sample space is

$$\Omega = \{\text{⚀⚀}, \text{⚀⚁}, \text{⚀⚂}, \text{⚀⚃}, \text{⚀⚄}, \text{⚀⚅}, \text{⚁⚀}, \text{⚁⚁}, \text{⚁⚂}, \ldots, \text{⚅⚅}\}$$

- Let random variable X_{sum} be the sum of outcomes of two rolls, $X_{\text{sum}}(\cdot) : \Omega \mapsto [2, 12]$
  - E.g., $X_{\text{sum}}(\text{⚂⚅}) = 9$
- Let event E_{even} be the subset of outcomes such that the sum is even:

$$E_{\text{even}} = \{\omega \in \Omega \mid X_{\text{sum}}(\omega) \text{ is even}\}$$

- The probability of an even sum is $\mathbb{P}(E_{\text{even}}) = \frac{1}{2}$
- Note that the outcome of one roll is independent from another roll

Probability Theory: Example

Example (Drawing Two Cards)

- Consider an experiment of drawing two cards from a deck. The sample space is

$$\Omega = \{\boxed{A\diamondsuit}\boxed{A\clubsuit}, \boxed{A\diamondsuit}\boxed{A\heartsuit}, \boxed{A\diamondsuit}\boxed{A\spadesuit}, \boxed{A\clubsuit}\boxed{A\heartsuit}, \boxed{A\clubsuit}\boxed{A\spadesuit}, \boxed{A\heartsuit}\boxed{A\spadesuit}, \boxed{A\diamondsuit}\boxed{2\diamondsuit}, \boxed{A\diamondsuit}\boxed{2\clubsuit}, \ldots, \boxed{K\heartsuit}\boxed{K\spadesuit}\}$$

- Let random variable X_{pair} be the indicator whether the two cards are a pair, $X_{\text{pair}}(\cdot) : \Omega \mapsto \{0, 1\}$
  - E.g., $X_{\text{pair}}(\boxed{3\heartsuit}\boxed{3\spadesuit}) = 1$
- Let event E_{pair} be the subset of outcomes of a pair:

$$E_{\text{pair}} = \{\omega \in \Omega \mid X_{\text{pair}}(\omega) = 1\}$$

- The probability of a pair is $\mathbb{P}(E_{\text{pair}}) = \frac{2 \cdot 13 \cdot 6}{13 \cdot 4 \cdot (13 \cdot 4 - 1)}$
- Note that the outcome of the next draw is **dependent** on the previous draw

Independence

- This is the most misunderstood concept in probability theory
- Intuitively, two events are independent if the likelihood of one occurring does not depend on the other having happened
  - Examples of independent events: tossing a coin twice, winning lottery twice

Definition (Independent Events)

- Two events $E_1, E_2 \subseteq \Omega$ are independent, if $\mathbb{P}(E_1 \cap E_2) = \mathbb{P}(E_1) \cdot \mathbb{P}(E_2)$
  - If this condition does not hold, then E_1, E_2 are dependent or correlated:
    - Positively dependent (the likelihood of one event enhances the likelihood of another):

$$\mathbb{P}(E_1 \cap E_2) > \mathbb{P}(E_1) \cdot \mathbb{P}(E_2)$$

    - Negatively dependent (the likelihood of one event diminishes the likelihood of another):

$$\mathbb{P}(E_1 \cap E_2) < \mathbb{P}(E_1) \cdot \mathbb{P}(E_2)$$

- ⚠ **Mutual Exclusion:** Two events E_1, E_2 are mutual exclusive, if $E_1 \cap E_2 = \varnothing$

Expectation

- Capture the intuition of statistical average of an observable quantity in an experiment
- Expectation is a powerful tool, bridging between discrete counting and calculus

Definition (Expectation)

- Expectation: Average of a random variable: $\mathbb{E}[X] \triangleq \sum_{\omega \in \Omega} \mathbb{P}(\omega) X(\omega)$
- Note that we mostly consider Ω as a finite set (e.g. events of a finite object)

Definition (Independent Random Variables)

- Two random variables $X_1, X_2 : \Omega \mapsto \mathbb{R}$ are independent, if $\mathbb{E}[X_1 \cdot X_2] = \mathbb{E}[X_1] \cdot \mathbb{E}[X_2]$
  - Proven by letting $E_1 = \{\omega \in \Omega \mid X_1(\omega) = x\}$, $E_2 = \{\omega \in \Omega \mid X_2(\omega) = y\}$, and the fact that E_1, E_2 are independent events for any x, y

Independence: Example

Example (Drawing Two Cards)

- Consider an experiment of drawing two cards from a deck. The sample space is

$$\Omega = \{A\diamondsuit A\clubsuit,\ A\diamondsuit A\heartsuit,\ A\diamondsuit A\spadesuit,\ A\clubsuit A\heartsuit,\ A\clubsuit A\spadesuit,\ A\heartsuit A\spadesuit,\ A\diamondsuit 2\diamondsuit,\ A\diamondsuit 2\clubsuit,\ \ldots,\ K\heartsuit K\spadesuit\}$$

- Let the following random variables:
  - X_{sum} be the sum of ranks of the two cards,
  - X_{pair} be the indicator whether the two cards are a pair,
  - X_{color} be the indicator whether the two cards are of the same color
  - E.g., $X_{\text{sum}}(5\diamondsuit K\heartsuit) = 18$, $X_{\text{pair}}(5\diamondsuit K\heartsuit) = 0$, $X_{\text{color}}(5\diamondsuit K\heartsuit) = 1$
- $\mathbb{E}[X_{\text{pair}} \cdot X_{\text{color}}] = \mathbb{E}[X_{\text{pair}}] \cdot \mathbb{E}[X_{\text{color}}]$ and $\mathbb{E}[X_{\text{sum}} \cdot X_{\text{pair}}] \neq \mathbb{E}[X_{\text{sum}}] \cdot \mathbb{E}[X_{\text{pair}}]$

Union Bound & Linearity of Expectation

Lemma (Union Bound)

For any events $E_1, E_2, ..., E_n \subseteq \Omega$, *we have*

$$\mathbb{P}(E_1 \cup E_2 \cup ... \cup E_n) \leq \mathbb{P}(E_1) + \mathbb{P}(E_2) + ... + \mathbb{P}(E_n)$$

Basic Idea:

$$|E_1 \cup E_2 \cup ... \cup E_n| \leq |E_1| + |E_2| + ...|E_n|$$

Lemma (Linearity of Expectation)

$\mathbb{E}[X_1 + X_2 + ... + X_n] = \mathbb{E}[X_1] + \mathbb{E}[X_2] + ... + \mathbb{E}[X_n]$

⚠ *Note that* $X_1, X_2, ..., X_n$ *do not need to be independent*

Proof:

$$\mathbb{E}\left[\sum_{i=1}^{n} X_i\right] = \sum_{\omega \in \Omega} \mathbb{P}(\omega) \cdot \left(\sum_{i=1}^{n} X_i(\omega)\right) = \sum_{i=1}^{n} \sum_{\omega \in \Omega} \mathbb{P}(\omega) \cdot X_i(\omega) = \sum_{i=1}^{n} \mathbb{E}[X_i]$$

Linearity of Expectation: Example

Example (Drawing Two Cards)

- Consider an experiment of drawing two cards from a deck. The sample space is

$$\Omega = \{\boxed{A\diamondsuit}\boxed{A\clubsuit}, \boxed{A\diamondsuit}\boxed{A\heartsuit}, \boxed{A\diamondsuit}\boxed{A\spadesuit}, \boxed{A\clubsuit}\boxed{A\heartsuit}, \boxed{A\clubsuit}\boxed{A\spadesuit}, \boxed{A\heartsuit}\boxed{A\spadesuit}, \boxed{A\diamondsuit}\boxed{2\diamondsuit}, \boxed{A\diamondsuit}\boxed{2\clubsuit}, \ldots, \boxed{K\heartsuit}\boxed{K\spadesuit}\}$$

- Let random variable $X^1_{\textsf{rank}}$ be the rank of the first card, and $X^2_{\textsf{rank}}$ be the rank of the second card
  - E.g., $X^1_{\textsf{rank}}(\boxed{J\diamondsuit}\boxed{K\spadesuit}) = 11$, $X^2_{\textsf{rank}}(\boxed{J\diamondsuit}\boxed{K\spadesuit}) = 13$
- The expected sum of the ranks of the two cards is

$$\mathbb{E}[X_{\textsf{sum}}] = \mathbb{E}[X^1_{\textsf{rank}} + X^2_{\textsf{rank}}] = \mathbb{E}[X^1_{\textsf{rank}}] + \mathbb{E}[X^2_{\textsf{rank}}]$$

- Note that the ranks of first card and the second card are **dependent** random variables

Why Randomization?

- **Amortization:** Diversify the best and worst cases, or foil the adversary from inflicting the worst case scenario on you
- **Estimation:** Sample the outcomes and probe the possible consequences in an unknown situation for strategizing the next moves
- **Probabilistic Method:** A proof technique for proving certain combinatorial properties without explicit construction

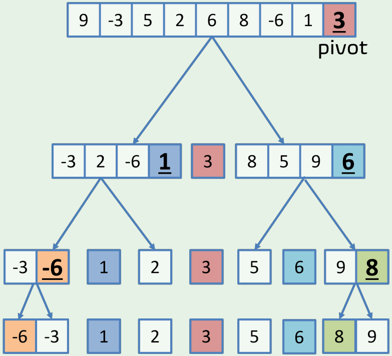
Definition

- *Las Vegas Algorithm*: A randomized algorithm that always gives correct results, but has probabilistic running time
  - Example: Randomized Quicksort
- *Monte Carlo Algorithm*: A randomized algorithm that has probabilistic accuracy
  - Example: Randomized Testing

Randomized Quicksort

Algorithm Quicksort[I: input sequence, x: pivot]

- Compare each item in I with pivot x
- Divide I into two groups:
  - $I_{<x}$ consisting of items in I that are less than x
  - $I_{\geq x}$ consisting of items in I that are greater than or equal to x
- Pick $y \in I_{<x}$
- $I_{<x} \leftarrow$ Quicksort$[I_{<x} \backslash \{y\}, y]$
- Pick $z \in I_{\geq x}$
- $I_{\geq x} \leftarrow$ Quicksort$[I_{\geq x} \backslash \{z\}, z]$
- Output $(I_{<x}, x, I_{\geq x})$



Randomized Quicksort

- Quicksort is a popular and widely used sorting algorithm
- The worst-case input for Quicksort is the case when every pivot chosen is always the smallest or largest in its group
  - E.g., choose pivot $x = 1$ or 8 from $I = \{3, 2, 5, 6, 7, 8, 1\}$
  - Then $I_{<x} = \varnothing$ or $I_{\geq x} = \varnothing$
  - The worst-case running time is $\mathsf{O}(n^2)$ – every pair of items will be compared in Quicksort
- **Randomized Quicksort**
  - Choose the pivot according to a uniform probability distribution in any group to alleviate the chance of choosing the the smallest or largest
  - Let X be the random number of comparisons performed in randomized Quicksort
  - The expected running time of randomized Quicksort is $\mathsf{O}(\mathbb{E}[X])$

Randomized Quicksort

Theorem

The expected running time of randomized Quicksort is $O(n\log(n))$

- For simplicity, we assume the set of input numbers is $\{1, ..., n\}$ in an unsorted sequence
- Let $X_{i,j}$ will be the indicator whether $i, j \in \{1, ..., n\}$ are compared in Quicksort

$$\mathbb{E}[X] = \mathbb{E}\Big[\sum_{j=2}^{n}\sum_{i=1}^{j-1} X_{i,j}\Big] = \sum_{j=2}^{n}\sum_{i=1}^{j-1}\mathbb{E}[X_{i,j}]$$

- $\mathbb{E}[X_{i,j}] = \mathbb{P}\big((i,j) \text{ are ever compared}\big) = \frac{2}{j-i+1}$, because X_i is binary random variable and
  - If any pivot is chosen from $\{1, ..., i-1, j+1, ..., n\}$, it does not affect the fact whether (i,j) will be compared
  - If the first pivot chosen from $\{i, i+1, ..., j-1, j\}$ is not i nor j, then (i,j) will never be compared, because they will be separated into two different groups since then
  - Since a pivot is chosen according to a uniform probability distribution in any group, the probability that the first pivot chosen from $\{i, i+1, ..., j-1, j\}$ is either is i or j is $\frac{2}{j-i+1}$

Randomized Quicksort

- Hence, we obtain

$$
\begin{aligned}
\mathbb{E}[X] &= \sum_{j=2}^{n}\sum_{i=1}^{j-1}\mathbb{E}[X_{i,j}] = \sum_{j=2}^{n}\sum_{i=1}^{j-1}\frac{2}{j-i+1} \\
&= 2\sum_{j=2}^{n}\sum_{k=2}^{j}\frac{1}{k} \qquad (\text{let } k = j-i+1) \\
&= 2\sum_{k=2}^{n}\sum_{j=k}^{n}\frac{1}{k} \qquad (\text{by interchanging the order of summation}) \\
&= 2\sum_{k=2}^{n}\frac{n-k+1}{k} = 2(n+1)\sum_{k=2}^{n}\frac{1}{k} - 2(n-1) = \mathrm{O}(n\log(n))
\end{aligned}
$$

- Note that the observed running time of randomized Quicksort is very close to $\mathrm{O}(n\log(n))$

Randomized Testing

- Suppose we are given three $n \times n$ matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ and want to test whether $\mathbf{AB} \stackrel{?}{=} \mathbf{C}$
  - Simply multiplying $\mathbf{A}$ by $\mathbf{B}$ takes $\mathsf{O}(n^3)$ running time. Any way better than that?

Algorithm RandomTest

- For each of t times, perform the following test
  - Choose each x_i in challenge $\mathbf{x} = (x_1, ..., x_n)^T$ independently and uniformly at random
  - We test whether $\mathbf{A}(\mathbf{Bx}) \stackrel{?}{=} \mathbf{Cx}$
- If none of the t tests fails, then we conclude $\mathbf{AB} = \mathbf{C}$

- Computing $\mathbf{A}(\mathbf{Bx})$ takes only $\mathsf{O}(n^2)$ running time, by two matrix-vector multiplications
- Totally, it takes $\mathsf{O}(t \cdot n^2)$ running time
- **False Positive**: If $\mathbf{AB} \neq \mathbf{C}$, but our test concludes $\mathbf{AB} = \mathbf{C}$
- **False Negative**: If $\mathbf{AB} = \mathbf{C}$, but our test concludes $\mathbf{AB} \neq \mathbf{C}$
  - Never happens in randomized testing

Randomized Testing

Lemma (Schwartz-Zippel Lemma)

Let $\mathbb{F}$ *be a finite field of numbers. Choose each coordinate* x_i *in challenge* $\mathbf{x} = (x_1, ..., x_n)^T$ *independently and uniformly at random from* $\mathcal{F}$*. If* $\mathbf{AB} \neq \mathbf{C}$*, then*

$$\mathbb{P}\big(\mathbf{A}(\mathbf{Bx}) = \mathbf{Cx}\big) \leq \frac{1}{|\mathbb{F}|}$$

- The probability of a false positive after t tests is less than $\left(\frac{1}{|\mathbb{F}|}\right)^t$
  - Note that each test is independent, because each challenge $\mathbf{x}$ is chosen independently and uniformly at random
- This trick is known as *probability amplification* to improve the probability of the correctness of randomized test to be very close to 1

Randomized Testing: Applications

- Third-party computation scenarios
  - Outsourcing computation in cloud computing, high-performance computers
  - Use blockchain as a public verification platform for processing confidential data
  - Problem: Unreliable/untrustworthy computation providers – How do we ensure that third-party computation is performed correctly?
- Verification of third-party computation
  - Verification should take much less computational power than the actual computation
  - A verifier challenges a prover (e.g. computation provider) who will provide a proof to show that the output is indeed computed from a given (possibly unrevealed) input according to a known computation function
- Basic idea:
  - Map the circuit of a computation function to polynomial $A(x)$, input to polynomial $B(x)$ and output to polynomial $C(x)$, where x is a random challenge
  - Run randomized tests for $A(x) \cdot B(x) \stackrel{?}{=} C(x)$ in an efficient and privacy-preserving manner

First Moment Method

Lemma (First Moment Principle)

If $\mathbb{E}[X] \leq t$*, then* $\mathbb{P}(X \leq t) > 0$

Theorem (Markov's Inequality)

For any non-negative random variable X*,*

$$\mathbb{P}(X \geq t) \leq \frac{\mathbb{E}[X]}{t}$$

Proof:

- $\mathbb{E}[X] = \sum_i i \cdot \mathbb{P}(X = i)$
- $\mathbb{E}[X] \geq \sum_{i \geq t} i \cdot \mathbb{P}(X = i) \geq t \cdot \mathbb{P}(X \geq t)$
- Markov's Inequality bounds the tail distribution by expectation

Probabilistic Method: k-SAT

Definition (k-SAT)

- For a Boolean variable x, there are two literals: x and $\bar{x}$
- Conjunctive Normal Form (CNF) is a sequence of clauses joined by "$\wedge$" (AND), where each clause consists of literals joined by "$\vee$" (OR)
  - E.g., $(x \vee y \vee \bar{z}) \wedge (\bar{x} \vee \bar{y} \vee \bar{z}) \wedge (x \vee \bar{y} \vee z)$
- A CNF formula is *satisfiable* if there is some assignment of values to its variables such that the entire formula equates to True
- An instance of k-SAT is a CNF-formula where every clause has exactly k literals

Theorem

Any instance of k-SAT with fewer than 2^k clauses is satisfiable

Probabilistic Method: k-SAT

Proof:

- Consider a random variable assignment
  - Setting each variable to be True or False with probability $\frac{1}{2}$
- For each clause C_i, define random variable X_i:

$$X_i = \begin{cases} 0, & \text{if } C_i \text{ is True} \\ 1, & \text{if } C_i \text{ is False} \end{cases}$$

- Let X be the number of unsatisfied clauses: $X = \sum_{i=1}^{m} X_i$
- Note that $\mathbb{E}[X_i] = \mathbb{P}(C_i \text{ is false}) = \frac{1}{2^k}$. Since there are $m < 2^k$ clauses,

$$\mathbb{E}[X] = \sum_{i=1}^{m} \mathbb{E}[X_i] = m \cdot \frac{1}{2^k} < 1$$

- By First Moment Principle, with positive probability $\mathbb{P}(X < 1)$, there must exist at least one satisfying assignment that the CNF formula is True

Second Moment Method

- Variance: var$[Y] \triangleq \mathbb{E}[(Y - \mathbb{E}[Y])^2] = \mathbb{E}[Y^2] - \mathbb{E}[Y]^2$
- var$\left[\sum_{i=1}^n X_i\right] = \sum_{i=1}^n$ var$[X_i]$, if $(X_1, ..., X_n)$ are independent random variables

Theorem (Chebyschev's Inequality)

For any non-negative random variable Y, the tail probability is bounded by

$$\mathbb{P}\Big(|Y - \mathbb{E}[Y]| \geq t\Big) \leq \frac{\text{var}[Y]}{t^2}$$

Proof:

- $|Y - \mathbb{E}[Y]| \geq t \;\Leftrightarrow\; (Y - \mathbb{E}[Y])^2 \geq t^2$
- Apply Markov's Inequality
- Chebyschev's Inequality is a concentration inequality
    - With high probability, a random variable is "concentrated" close to its expectation
    - Expectation is a good estimate of a random variable

Second Moment Method: Application

- Given a set of n integers S, the median of S is defined as:
  - $\lfloor \frac{n}{2} \rfloor$-th smallest $\leq$ medium $\leq (\lfloor \frac{n}{2} \rfloor + 1)$-th smallest
- Find the median requires running time $\mathrm{O}(n \log n)$ using sorting
- Can we improve the running time to $\mathrm{O}(n)$ (with high probability)?
- Basic idea:
  - Select two random elements $d, u \in S$, such that $d \leq$ medium $\leq u$
  - Determine the order of d, say d is the k-th smallest
  - Let $C = \{x \in S \mid d < x < u\}$
  - Sort C and find $(\lfloor \frac{n}{2} \rfloor - k + 1)$-th smallest in C
- How to ensure $d \leq$ medium $\leq u$, without knowing medium?
- How to ensure that $|C|$ is small enough to be sorted efficiently?

Second Moment Method: Application

Random Sampling Algorithm for Median $\mathcal{A}_{\text{rmed}}$

- Randomly pick a set $R \subseteq S$ with replacement, such that $|R| = n^{\frac{3}{4}}$
- Sort R
- Let d be the $(\lfloor \frac{1}{2} n^{\frac{3}{4}} - \sqrt{n} \rfloor)$-th smallest in R
- Let u be the $(\lfloor \frac{1}{2} n^{\frac{3}{4}} + \sqrt{n} \rfloor)$-th smallest in R
- Find $C = \{x \in S \mid d < x < u\}$
- Sort C
- Determine the order of d, say d is the k-th smallest
- Output the $(\lfloor \frac{n}{2} \rfloor - k + 1)$-th smallest in C

Theorem

The running time of $\mathcal{A}_{\text{rmed}}$ is $O(n)$, if $|C| = o(n/\log n)$ to be sorted in $O(n)$

Second Moment Method: Application

Theorem

The probability that $\mathbb{P}(\mathcal{A}_{\text{rmed}}$ *fails*$) \leq n^{-\frac{1}{4}}$

Proof:

- Let E_1 be the event that $d >$ Median and E_2 be the event that $u <$ Median
- Let E_3 be the event that $|C| > 4n^{\frac{3}{4}}$ (including the possibility $|C| \neq \mathrm{o}(n/\log n)$)
- $\mathbb{P}(\mathcal{A}_{\text{rmed}}$ fails$) \leq \mathbb{P}(E_1 \cup E_2 \cup E_3) \leq \mathbb{P}(E_1) + \mathbb{P}(E_2) + \mathbb{P}(E_3)$
- We show that $\mathbb{P}(E_1) = \mathbb{P}(E_2) \leq \frac{1}{4}n^{-\frac{1}{4}}$ and $\mathbb{P}(E_3) \leq \frac{1}{2}n^{-\frac{1}{4}}$
- R is a set of $n^{\frac{3}{4}}$ random samples with replacement. Let

$$X_i = \begin{cases} 1 & \text{if the } i\text{-th sample} \leq \text{Median} \\ 0 & \text{otherwise} \end{cases}$$

- Note that each X_i is an independent binary random variable as they are picked with replacement, and $\mathbb{E}[X_i] = \frac{\frac{n-1}{2}+1}{n}$ since there are $(\frac{n-1}{2} + 1)$ elements in $S \leq$ Median

Second Moment Method: Application

Proof (Cont.):

- Let $Y = \sum_{i=1}^{n^{\frac{3}{4}}} X_i$. Then, $\mathbb{E}[Y] = \sum_{i=1}^{n^{\frac{3}{4}}} \mathbb{E}[X_i] = \sum_{i=1}^{n^{\frac{3}{4}}} \frac{\frac{n-1}{2}+1}{n} > \frac{1}{2}n^{\frac{3}{4}}$
- Since each X_i is an independent binary random variable ($\mathbb{E}[X_i^2] = \mathbb{E}[X_i]$), we have

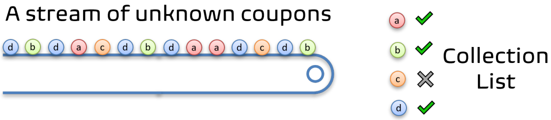
$$\mathsf{var}(Y) = n^{\frac{3}{4}}\mathsf{var}(X_i) = n^{\frac{3}{4}}(\mathbb{E}[X_i^2] - \mathbb{E}[X_i]^2) = n^{\frac{3}{4}}\Big(\frac{\frac{n-1}{2}+1}{n} - (\frac{\frac{n-1}{2}+1}{n})^2\Big) < \frac{1}{4}(n^{\frac{3}{4}})$$

- Note that d is the $(\lfloor \frac{1}{2}n^{\frac{3}{4}} - \sqrt{n} \rfloor)$-th smallest in R

$$\mathbb{P}(E_1) = \mathbb{P}\Big(Y < \frac{1}{2}n^{\frac{3}{4}} - \sqrt{n}\Big) \le \mathbb{P}\big(Y < \mathbb{E}[Y] - \sqrt{n}\big) \le \mathbb{P}\big(\big|Y - \mathbb{E}[Y]\big| > \sqrt{n}\big)$$

- By Chebyschev's Inequality, $\mathbb{P}(E_2) = \mathbb{P}(E_1) \le \mathbb{P}\big(\big|Y - \mathbb{E}[Y]\big| > \sqrt{n}\big) \le \frac{\mathsf{var}(Y)}{n} \le \frac{1}{4}n^{-\frac{1}{4}}$
- It can be shown similarly that $\mathbb{P}(E_3) \le \frac{1}{2}n^{-\frac{1}{4}}$ by
  - At least $2n^{\frac{3}{4}}$ elements of $C \ge$ Median; or at least $2n^{\frac{3}{4}}$ elements of $C \le$ Median

Coupon Collector's Problem



Definition (Coupon Collector's Problem)

- There are n different coupons
- Goal: Collect all n coupons from a sequence of independent draws
  - Each time a random coupon is drawn; each coupon appears with a uniform probability $\frac{1}{n}$
  - Sometime, a coupon drawn may have appeared before
- Let X be the number of draws required to collect all n coupons: $X = \sum_{i=1}^{n} X_i$, where X_i is number of draws to collect the i-th different coupon that has not been collected before

Coupon Collector's Problem

Definition (Geometric Random Variable)

- Geometric random variable, $\mathrm{Geom}(p)$, is a random number of steps, where each step continues with probability $1-p$, or stops with probability p
- $\mathbb{P}(\mathrm{Geom}(p) = k) = (1-p)^k p$ and $\mathbb{E}[\mathrm{Geom}(p)] = \frac{1}{p}$

- Each X_i is an independent geometric random variable, $\mathrm{Geom}(1 - \frac{i-1}{n})$ and $\mathbb{E}[X_i] = \frac{n}{n-i+1}$
- The expected number of draws required to collect all n coupons

$$\mathbb{E}[X] = \sum_i^n \mathbb{E}[X_i] = \sum_i^n \frac{n}{n-i+1} = n \sum_i^n \frac{1}{i} = n \log n + n\gamma$$

  - Define $H_n \triangleq \sum_i^n \frac{1}{i}$, called the harmonic number
  - $H_n = \log n + \gamma$, where γ is a constant called Euler's constant

Coupon Collector's Problem

- Note that $\mathsf{var}[\mathsf{Geom}(p)] = \frac{1-p}{p^2} \leq \frac{1}{p^2}$
- Since $\sum_{i=1}^{\infty} \left(\frac{1}{i}\right)^2 = \frac{(\pi)^2}{6}$, we have

$$\mathsf{var}[X] = \sum_{i=1}^{n} \mathsf{var}[X_i] \leq \sum_{i=1}^{n} \left(\frac{n}{n-i+1}\right)^2 \leq n^2 \sum_{i=1}^{n} \left(\frac{1}{i}\right)^2 \leq \frac{(\pi n)^2}{6}$$

- By Chebyshev's inequality,

$$\mathbb{P}\Big(|X - nH_n| \leq nH_n\Big) \leq \frac{(\pi n)^2}{6} \frac{1}{(nH_n)^2} = \frac{\pi^2}{6(H_n)^2} = \mathsf{O}\Big(\frac{1}{\log^2 n}\Big)$$

- This tail probability bound is not sharp. In fact, the tail probability is decaying exponentially fast in n

⋘ The Larger, The Simpler ⋙

Probability: Beyond Counting

- Random systems can consist of a very large degree of randomness
  - Large physical systems (e.g. movement of many gas molecules)
  - Large computer systems (e.g. many packets in Internet)
  - Large human systems (e.g. stock markets)
- The property of averaging-out kicks in: the expected behavior dominates
- Concentration of measure: As $n \to \infty$, system state $X_n \to \mathbb{E}[X_n]$
- Paradox: Smaller random systems may be complicated, larger systems may be simpler
- Probability theory can provide insights for large systems that cannot be counted
- Example of large systems in algorithms: randomized algorithms of large problem $n \to \infty$

Concentration of Measure

- Polynomial decay of tail probability in terms of t^{-k}
  - Markov Inequality: $\mathbb{P}(X \geq t) \leq \frac{\mathbb{E}[X]}{t}$
  - Chebyschev's Inequality: $\mathbb{P}(|Y - \mathbb{E}[Y]| \geq t) \leq \frac{\mathsf{var}[Y]}{t^2}$
  - Can be applicable to general random variables
  - But insufficient to show decaying probability with a polynomial number $P(t)$ of events:

$$P(t) \cdot t^{-k} \not\to 0 \text{ as } t \to \infty$$

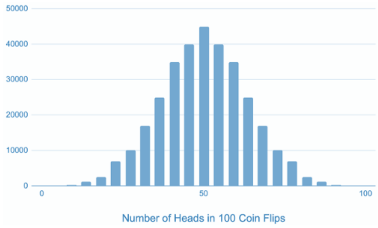
- Exponential decay of tail probability in terms of e^{-t}
  - Chernoff bound: $\mathbb{P}(|Y - \mathbb{E}[Y]| \geq t) \leq \mathsf{O}(e^{-ct \cdot \mathbb{E}[Y]})$
  - Sufficient to show decaying probability with a polynomial number $P(t)$ of events:

$$P(t) \cdot e^{-t} \to 0 \text{ as } t \to \infty$$

  - But not applicable to general random variables
  - There is a sharp decay in the tail probability for specific random variables

Chernoff Bound

- Bernoulli random variable BER(p) (e.g. head of a coin toss):
$$X = \begin{cases} 1, & \text{with probability } p \\ 0, & \text{with probability } 1-p \end{cases}$$
- Binomial random variable BIN(n, p) is a sum of independent BER(p) (e.g. the number of heads of n coin tosses),
$$S_n = \sum_{i=1}^{n} X_i$$



Theorem (Chernoff Bound for Binomial Random Variable)

Let S_n be a Binomial random variable BIN(n, p)
For any $t > 0$, the tail probability is bounded by

$$\mathbb{P}\Big(|S_n - np| \geq nt\Big) \leq 2e^{-2nt^2}$$

Chernoff Bound

Proof:

- Let $m = n(p+t)$ and $h > 0$. Consider $S_n \geq m$, by Markov's Inequality,

$$\mathbb{P}(S_n \geq m) = \mathbb{P}(e^{hS_n} \geq e^{hm}) \leq e^{-hm} \cdot \mathbb{E}[e^{hS_n}] = e^{-hm}(1-p+pe^h)^n$$

- It is because that S_n is a sum of independent binary random variables:

$$\mathbb{E}[e^{hS_n}] = \mathbb{E}\Big[\prod_{i=1}^{n} e^{hX_i}\Big] = \prod_{i=1}^{n} \mathbb{E}[e^{hX_i}] = (1-p+pe^h)^n$$

- Note that $e^{-hp}(1-p+pe^h) \leq e^{h^2/8}$ (for $0 \leq p \leq 1$ and $h > 0$). Hence,

$$\mathbb{P}(S_n - np \geq nt) \leq e^{-nht}\Big(e^{-hp}(1-p+pe^h)\Big)^n \leq e^{(-ht+h^2/8)n}$$

This attains the minimum bound, when $h = 4t$, namely, $e^{(-ht+h^2/8)n} = e^{-2nt^2}$

Chernoff Bound: Application

- Let S_n be the number of heads of n fair coin tosses
- By Chernoff Bound, we have

$$\mathbb{P}\Big(|S_n - \frac{n}{2}| \geq \frac{n}{4}\Big) \leq 2e^{-2n\frac{1}{16}} = 2e^{-\frac{n}{8}}$$

- Chebyschev's Inequality gives a much weak bound

$$\mathbb{P}\Big(|S_n - \frac{n}{2}| \geq \frac{n}{4}\Big) \leq \frac{4}{n}$$

- If we take a number of n^k samples of S_n,
  - The probability that any one of samples has $|S_n - \frac{n}{2}| \geq \frac{n}{4}$ is lesser than $n^k e^{-\frac{n}{8}}$
  - Note that $n^k e^{-\frac{n}{8}} \to 0$ as $n \to \infty$
  - Meaning that the probability of deviation is rare, when n is large

Poisson Random Variable

- Poisson random variable: $\text{Pois}(\lambda)$

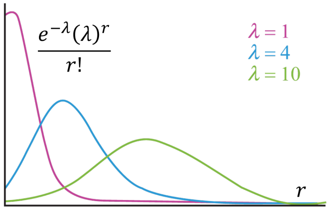
$$\mathbb{P}(\text{Pois}(\lambda) = r) = \frac{e^{-\lambda}\lambda^r}{r!}, \quad \mathbb{E}[\text{Pois}(\lambda)] = \lambda, \quad \text{var}[\text{Pois}(\lambda)] = \lambda, \quad \mathbb{E}[e^{h \cdot \text{Pois}(\lambda)}] = e^{\lambda(e^h - 1)}$$

- Poisson random variable model a given number of events in a fixed interval, occurring with a known average rate and independently of the time since the last event

- Examples:
  - Telephone calls arriving in a system
  - Customers arriving at a counter or call center
  - Cars arriving at a traffic light

- Approximate Binomial random variable:

$$\text{BIN}(n, \frac{\lambda}{n}) \to \text{Pois}(\lambda) \text{ when } n \to \infty$$



Poisson Random Variable

Theorem (Chernoff Bound for Poisson Random Variable)

Let X be a Poisson random variable Pois(λ)

- *If* $x > \lambda$,

$$\mathbb{P}(X \geq x) \leq \frac{e^{-\lambda}(e\lambda)^x}{x^x}$$

- *If* $x < \lambda$,

$$\mathbb{P}(X \leq x) \leq \frac{e^{-\lambda}(e\lambda)^x}{x^x}$$

Proof:

- We have

$$\mathbb{P}(X \geq x) = \mathbb{P}(e^{hX} \geq e^{hx}) \leq \frac{\mathbb{E}[e^{hX}]}{e^{hx}} = e^{\lambda(e^h - 1) - hx}$$

- Suppose $x > \lambda$, then $\ln(x/\lambda) > 0$
- Choose $h = \ln(x/\lambda)$, then we obtain $\mathbb{P}(X \geq x) = e^{x - \lambda - x\ln(x/\lambda)}$

Coupon Collector's Problem

Theorem

- *Let X be the number of draws required to collect all n types of coupons. Then, for any constant c,*

$$\lim_{n\to\infty} \mathbb{P}(X > n\ln n + cn) = 1 - e^{-e^{-c}}$$

Basic Ideas:

- Based on balls-and-bins model: balls = draws, bins = types of coupons
- Use Poisson approximation to model the number of balls throwing into bins, such that each bin has at least one ball, or equivalently no bin is empty
  - See the next lecture for balls-and-bins model

References

Reference Materials

- Probability and Computing (Mitzenmacher, Upfal), 2nd ed, Cambridge University Press
  - Chapters 1-3: Basics of Probability Theory
  - Chapters 4.1-4.2: Chernoff Bounds
  - Chapters 6.1-6.2: The Probabilistic Method