EECS 376 Discussion 12

Sec 27: Th 5:30-6:30 DOW 1017

IA: Eric Khiu

Slide deck available at course drive/Discussion/Slides/Eric Khiu

Agenda

- Probability Bounds
  - Chernoff bounds
  - Union Bounds
- Fingerprinting
- Fast Modular Exponentiation (if time)
- Diffie-Hellman Key Exchange

Probability Bounds

Review: Markov Inequality

- **Markov's Inequality:** Let X be a positive RV and $a > 0$, then
$$\Pr[X \geq a] \leq \frac{E[X]}{a}$$
  - Rearranging, we get
$$\Pr\big[X \geq a \cdot E[X]\big] \leq \frac{1}{a}$$
- **"Reverse" Markov's Inequality:** Let X be a positive RV upper-bounded by B, then
$$\Pr[X > a] \geq \frac{E[X] - a}{B - a}$$

Large Deviation Chernoff Bound

- Let $X = X_1 + X_2 + \cdots + X_n$ be the sum of n independent indicator RV with expected value $\mathbb{E}[X] = \mu$ (WARNING: IT'S NOT $\mathbb{E}[X_i]$!)

Sanity Check: Can X be negative? What about μ?

- **Large deviation Chernoff bound** says that the probability of X exceeding μ by some $\lambda \geq 1$ is

$$\Pr[X - \mu \geq \lambda\mu] \leq e^{-\frac{\lambda u}{3}}$$

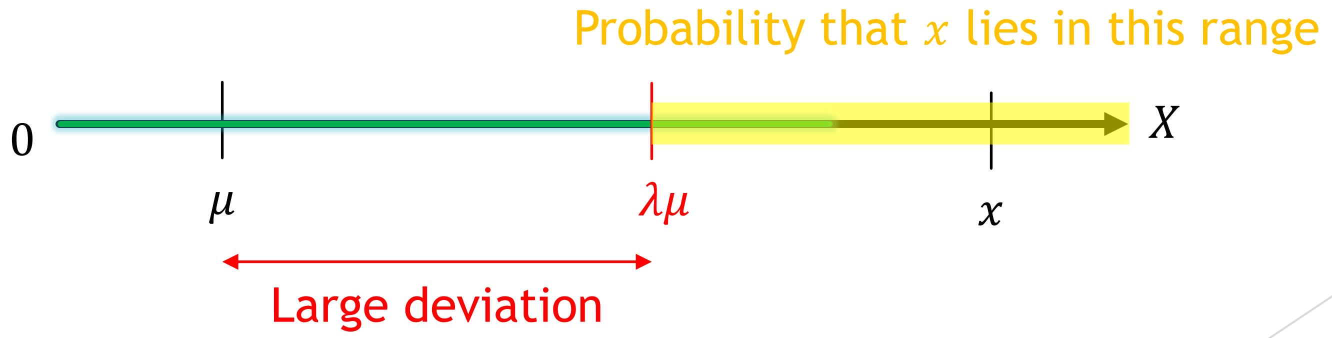
Large Deviation Chernoff Bound

- Let $X = X_1 + X_2 + \cdots + X_n$ be the sum of n independent indicator RV with expected value $\mathbb{E}[X] = \mu$ (WARNING: IT'S NOT $\mathbb{E}[X_i]$!)

Sanity Check: Can X be negative? What about μ?

- **Large deviation Chernoff bound** says that the probability of X exceeding μ by some $\lambda \geq 1$ is

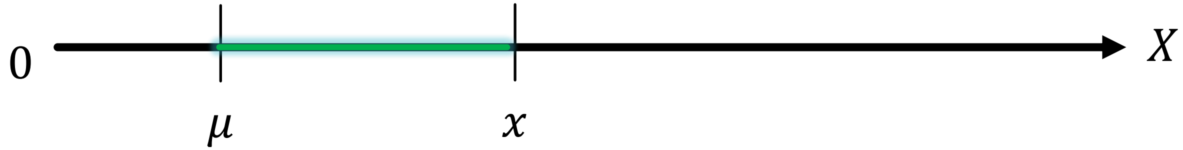
$$\Pr[X - \mu \geq \lambda\mu] \leq e^{-\frac{\lambda u}{3}}$$



Small Deviation Chernoff Bounds

- **Small deviation Chernoff bounds** says that the probability of X exceeding μ by some at least some $\lambda \in [0,1]$ is

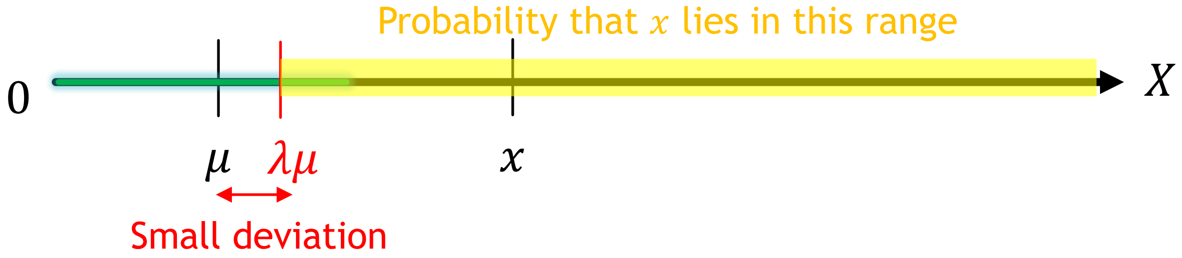
$$\Pr[X - \mu \geq \lambda\mu] \leq e^{-\frac{\lambda^2 u}{3}}$$



Small Deviation Chernoff Bounds

- **Small deviation Chernoff bounds** says that the probability of X exceeding μ by some at least some $\lambda \in [0,1]$ is

$$\Pr[X - \mu \geq \lambda\mu] \leq e^{-\frac{\lambda^2 u}{3}}$$



- And the probability of X below μ by at least some $\lambda \in [0,1]$ is

$$\Pr[\mu - X \geq \lambda\mu] = \Pr[X - \mu \leq -\lambda\mu] \leq e^{-\frac{\lambda^2 u}{3}}$$

Large deviation ($\lambda \geq 1$):

$$\Pr[X - \mu \geq \lambda\mu] \leq e^{-\frac{\lambda u}{3}}$$

Small deviation ($\lambda \in [0,1]$):

$$\Pr[X - \mu \geq \lambda\mu] \leq e^{-\frac{\lambda^2 u}{3}}$$

$$\Pr[X - \mu \leq -\lambda\mu] \leq e^{-\frac{\lambda^2 u}{3}}$$

Chernoff Bounds Exercise

- Suppose we roll a fair 4-sided die 24 times. Let X be the random variable representing the number of 1's obtained

- Using Chernoff bounds, give an upper bound to the probabilities
  - $\Pr[X \geq 18]$
  - $\Pr[X \geq 9]$

  Hint: First compute $\mu = \mathbb{E}[X]$

  Let $X_i = 1$ if the ith roll is a 1, $\mathbb{E}[X] = 24 \cdot \frac{1}{4} = 6$

Chernoff Bounds Exercise 1

Large deviation ($\lambda \geq 1$):

$$\Pr[X - \mu \geq \lambda\mu] \leq e^{-\frac{\lambda u}{3}}$$

Small deviation ($\lambda \in [0,1]$):

$$\Pr[X - \mu \geq \lambda\mu] \leq e^{-\frac{\lambda^2 u}{3}}$$

$$\Pr[X - \mu \leq -\lambda\mu] \leq e^{-\frac{\lambda^2 u}{3}}$$

- Suppose we roll a fair 4-sided die 24 times. Let X be the random variable representing the number of 1's obtained

- Using Chernoff bounds, give an upper bound to $\Pr[X \geq 18]$

  Let $X_i = 1$ if the ith roll is a 1, $\mathbb{E}[X] = 24 \cdot \frac{1}{4} = 6$

  - Step 1: Rewrite the expression as $X - \mu$

    $$\Pr[X \geq 18] = \Pr[X - 6 \geq 18 - 6] = \Pr[X - 6 \geq 12]$$

  - Step 2: Find λ

    $$12 = 6\lambda \Rightarrow \lambda = \frac{12}{6} = 2 \geq 1$$

  - Step 3: Apply the large deviation Chernoff bound

    $$\Pr[X - 6 \geq 2(6)] \leq e^{-\frac{2(6)}{3}} \approx 0.0183$$

Chernoff Bounds Exercise 2

Large deviation ($\lambda \geq 1$):

$$\Pr[X - \mu \geq \lambda\mu] \leq e^{-\frac{\lambda u}{3}}$$

Small deviation ($\lambda \in [0,1]$):

$$\Pr[X - \mu \geq \lambda\mu] \leq e^{-\frac{\lambda^2 u}{3}}$$

$$\Pr[X - \mu \leq -\lambda\mu] \leq e^{-\frac{\lambda^2 u}{3}}$$

- Suppose we roll a fair 4-sided die 24 times. Let X be the random variable representing the number of 1's obtained

- Using Chernoff bounds, give an upper bound to $\Pr[X \geq 9]$

  Let $X_i = 1$ if the ith roll is a 1, $\mathbb{E}[X] = 24 \cdot \frac{1}{4} = 6$

  - Step 1: Rewrite the expression as $X - \mu$

    $$\Pr[X \geq 9] = \Pr[X - 6 \geq 9 - 6] = \Pr[X - 6 \geq 3]$$

  - Step 2: Find λ

    $$3 = 6\lambda \Rightarrow \lambda = \frac{3}{6} = \frac{1}{2} \leq 1$$

  - Step 3: Apply the small deviation Chernoff bound

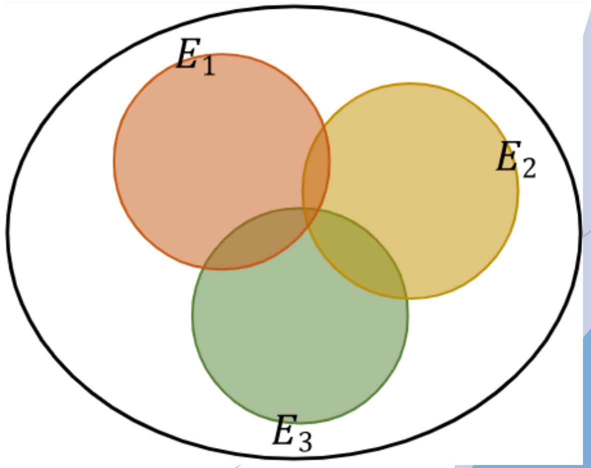
    $$\Pr\left[X - 6 \geq \frac{1}{2}(6)\right] \leq e^{-\frac{\left(\frac{1}{2}\right)^2 (6)}{3}} \approx 0.6065$$

Union Bound

- The probability of any one of many events occurring is less than the sums of the probabilities of each event

- Let $A_1, A_2, \ldots, A_n$ be a set of (possible dependent) events, then

$$\Pr[A_1 \cup A_2 \cup \cdots \cup A_n] \leq \sum_{i=1}^{n} \Pr[A_i]$$



$$\Pr[A_1 \cup A_2 \cup \cdots \cup A_n] \le \sum_{i=1}^{n} \Pr[A_i]$$

Union Bound Exercise

- In a computer system equipped with 50 processors, each engaged in concurrent multithreading tasks, there exists a probability of 0.001 for an individual processor to experience failure. Determine the probability that **at least one** processor encounters failure.

- Let $X_i = 1$ if the i-th processor fails and 0 otherwise, so $\Pr[X_i] = 0.001$
- $\Pr[\text{at least one fails}] = \Pr[X_1 \cup X_2 \cup \cdots \cup X_{50}]$
- Apply Union Bound

$$\Pr[X_1 \cup X_2 \cup \cdots \cup X_{50}] \le \sum_{i=1}^{50} \Pr[X_i] = 50(0.001) = 0.05$$

Summary of Probabilities Bounds

Probability Bounds	Constraints
Markov's inequality: $$\Pr[X \geq a] \leq \frac{\mathbb{E}[X]}{a}$$	• X is a positive RV
"Reverse" Markov's inequality: $$\Pr[X > a] \geq \frac{\mathbb{E}[X] - a}{B - a}$$	• X is a positive RV • X is upper bounded by some B
Chernoff large deviation bound: $$\Pr[X - \mu \geq \lambda\mu] \leq e^{-\frac{\lambda u}{3}}$$	• X is a sum of independent IRV • $\mathbb{E}[X] = \mu$ • $\lambda > 1$
Chernoff small deviation bounds: $$\Pr[X - \mu \geq \lambda\mu] \leq e^{-\frac{\lambda^2 u}{3}}$$ $$\Pr[X - \mu \leq -\lambda\mu] \leq e^{-\frac{\lambda^2 u}{3}}$$	• X is a sum of independent IRV • $\mathbb{E}[X] = \mu$ • $\lambda \in [0,1]$
Union bounds: $$\Pr[A_1 \cup A_2 \cup \cdots \cup A_n] \leq \sum_{i=1}^{n} \Pr[A_i]$$	N/A

Fingerprinting

Setup

- Suppose Alice wants to communicate some large n-bits number x to Bob
- She wants to send as few bits as possible, so instead she uploads this number to a server for him to download
- The server is untrusted, so once Bob downloads the number y, he and Alice will need to confirm that x and y are the same

Randomized Fingerprinting

- Alice randomly chooses a prime p from the first $10n$ primes and sends Bob the message $(p, x \bmod p)$

- Bob computes $y \bmod p$
  - If $x \bmod p = y \bmod \mathrm{p}$, Bob concludes that $x = y$
  - Otherwise, Bob concludes that $x \neq y$

- When $x = y$, this protocol is correct for all choices of p
- When $x \neq y$, this protocol is correct for at least 90% of the choices of p
  - Example: $p = 3$; $x = 1, y \in \{7,10,13, \dots\}$ breaks the protocol

Randomized Fingerprinting Exercise

- Alice randomly chooses a prime p from the first $10n$ primes and sends Bob the message $(p, x \bmod p)$

- Bob computes $y \bmod p$
  - If $x \bmod p = y \bmod p$, Bob concludes that $x = y$
  - Otherwise, Bob concludes that $x \neq y$

- State whether the algorithm is correct for all choices of p, or give all examples of p that cause the protocol to result in the incorrect decision
  - $x = 70, y = 64$
  - $x = 342, y = 342$

  Hint: If $x \bmod p = y \bmod p$, what is $(x - y) \bmod p$?

Randomized Fingerprinting Exercise

- Recall that a and b are congruent modulo n, written as $a \equiv b (\mathrm{mod}\, n)$ if
  - $a \bmod n = b \bmod \mathrm{n}$, or equivalently,
  - $\exists k \in \mathbb{Z}$ such that $a = b + kn$, or equivalently
  - $a - b$ is a multiple of n
- So if $x \bmod p = y \bmod p$, then $(x - y) \bmod p = 0$
- This means we just need to find the prime divisors of $x - y$
- State whether the algorithm is correct for all choices of p, or give all examples of p that cause the protocol to result in the incorrect decision
  - $x = 70, y = 64$
  - $x = 342, y = 342$

Unit 5: Cryptography

Fast Modular Exponentiation

Exponentiation in Modular Arithmetic

- Recall that if $a \equiv b (\mathrm{mod}\ n)$, then for any $k \in \mathbb{Z}$,
  - $a + k \equiv b + k\ (\mathrm{mod}\ n)$
  - $ak \equiv bk\ (\mathrm{mod}\ n)$
- **Property 215:** Suppose $a = a' (\mathrm{mod}\ n)$ and $b = b' (\mathrm{mod}\ n)$, then
$$ab \equiv a'b'\ (\mathrm{mod}\ n)$$
  - Proof idea: Let $a - a' = kn$ and $b - b' = mn$ for some integers k and m, then
  $$ab = (kn + a') \cdot (mn + b') = \cdots = (kmn + a'm + b'k)n + a'b'$$
  - Therefore, $ab - a'b' = (kmn + a'm + b'k)n$, so $ab \equiv a'b'\ (\mathrm{mod}\ n)$
- **Corollary:** If $a \equiv b (\mathrm{mod}\ n)$, then $a^k \equiv b^k\ (\mathrm{mod}\ n)$
  - Proof idea: $a^k = \underbrace{a \cdot a \cdot \ldots \cdot a}_{k \text{ times}}$, use property 215 and induction

Property 215: Suppose $a = a' (\mathrm{mod}\, n)$ and $b = b' (\mathrm{mod}\, n)$, then

$$ab \equiv a'b' \ (\mathrm{mod}\, n)$$

Fast Modular Exponentiation

- Suppose we want to compute $a^b \bmod n$
- Consider the binary representation of b

$$b = b_r \cdot 2^r + b_{r-1} \cdot 2^{r-1} + \cdots + b_0 \cdot 2^0$$

  - Here, b_i is either 0 or 1
  - $r = \lfloor \log_2 b \rfloor$
- Then, we can represent a^b as

$$\begin{aligned} a^b &= a^{b_r \cdot 2^r + b_{r-1} \cdot 2^{r-1} + \cdots + b_0 \cdot 2^0} \\ &= a^{b_r \cdot 2^r} \times a^{b_{r-1} \cdot 2^{r-1}} \times \cdots \times a^{b_0 \cdot 2^{r-1}} \end{aligned}$$

- Thus, we can compute $a^{2^i} \bmod n$ for each $a \leq i \leq r$ and include those whose $b_i = 1$ in the product

Property 215: Suppose $a = a'(\text{mod } n)$ and $b = b'(\text{mod } n)$, then

$$ab \equiv a'b' \ (\text{mod } n)$$

Fast Modular Exponentiation

14	×	1	=	14
14	×	2	=	28
14	×	3	=	42
14	×	4	=	56
14	×	5	=	70
14	×	6	=	84
14	×	7	=	98
14	×	8	=	112
14	×	9	=	126
14	×	10	=	140

- Example: $3^5 \bmod 14$
- Step 1: Express $b = 5$ in binary

$$5 = 101$$

- Step 2: Compute $3^{2^i} \bmod 14$ for $i = 0 \leq i \leq \lfloor \log_2 5 \rfloor = 2$

$$3^{2^0} = 3^1 = 3 \equiv 3 \ (\text{mod } 14)$$
$$3^{2^1} = 3^2 = 9 \equiv 9 \ (\text{mod } 14)$$
$$3^{2^2} = 9^2 = 81 \equiv 11 \ (\text{mod } 14)$$

- Step 3: Multiply and simplify

$$\begin{aligned} 3^5 &= 3^4 \cdot 3^1 \\ &\equiv 11 \cdot 3 \ (\text{mod } 14) \\ &\equiv 33 \ (\text{mod } 14) \\ &\equiv 5 \ (\text{mod } 14) \end{aligned}$$

Your turn: Compute $3^{57} \bmod 14$ Ans: 13

Fast Modular Exponentiation Algorithm

- **(Take home) exercise:** Complete the following DP algorithm for fast modular exponentiation and analyze the runtime:

```
FastModExp(a,b,n):
    r ← ⌊log b⌋
    allocate an empty array DP[0, …, r]
    DP[0] ← a
    for i = 1, …, r do
        [                    ]
    ans ← 1
    for i = 0, …, r do
        if [        ] then
            [                    ]
    return ans
```

See Algorithm 220 on course notes for solution

Congruent Class and Generator

Congruent class

- **Congruent class:** For any $n \in \mathbb{N}$, we define $\mathbb{Z}_n = \{0,1,2,\dots,n-1\}$ as the set of congruence class modulo n.

- The group $\mathbb{Z}_n^* \subseteq \mathbb{Z}_n$ is the set of nonzero elements of $\mathbb{Z}_n$ that have an inverse in modulo n, i.e.,

$$\mathbb{Z}_n^* = \{x \in \mathbb{Z}_n : \gcd(x,n) = 1\}$$

Discuss: What if n is prime?

  - A prime number is coprime to all natural numbers smaller than it
  - $\mathbb{Z}_n^* = \{1,2,\dots,n-1\}$

Generator

- **Generator:** Let p be a prime. $g \in \mathbb{Z}_p^*$ is a *generator* if for every $x \in \mathbb{Z}_p^*$, there exists some $i \in \mathbb{N}$ such that $x = g^i \bmod p$

- Example: $g = 2$ is a generator of $\mathbb{Z}_5^*$:
  - $\mathbb{Z}_5^* = \{1,2,3,4\}$
  - $2^0 = 1 \equiv 1 \pmod 5$
  - $2^1 = 2 \equiv 2 \pmod 5$
  - $2^2 = 4 \equiv 4 \pmod 5$
  - $2^3 = 8 \equiv 3 \pmod 5$

- But $g = 2$ is a not generator of $\mathbb{Z}_7^*$:
  - $\mathbb{Z}_7^* = \{1,2,3,4,5,6\}$
  - $2^0 = 1 \equiv 1 \pmod 7$
  - $2^1 = 2 \equiv 2 \pmod 7$
  - $2^2 = 4 \equiv 4 \pmod 7$
  - $2^3 = 8 \equiv 1 \pmod 7$
  - $2^4 = 16 \equiv 2 \pmod 7$
  - …

Concept Check

Generator: Let p be a prime. $g \in \mathbb{Z}_p^*$ is a *generator* if for every $x \in \mathbb{Z}_p^*$, there exists some $i \in \mathbb{N}$ such that $x = g^i \bmod p$

- Is $g = 3$ a generator of $\mathbb{Z}_{11}^*$?
  - $\mathbb{Z}_{11}^* = \{1,2, \dots, 10\}$
  - $3^0 = 1 \equiv 1 \pmod{11}$
  - $3^1 = 3 \equiv 3 \pmod{11}$
  - $3^2 = 9 \equiv 9 \pmod{11}$
  - $3^3 = 27 \equiv 5 \pmod{11}$
  - $3^4 = 81 \equiv 4 \pmod{11}$
  - $3^5 = 3 \cdot 3^4 \equiv 3 \cdot 4 = 12 \equiv 1 (\bmod 11)$
  - …

11	x	1	=	11
11	x	2	=	22
11	x	3	=	33
11	x	4	=	44
11	x	5	=	55
11	x	6	=	66
11	x	7	=	77
11	x	8	=	88
11	x	9	=	99
11	x	10	=	110

Another Definition of Generator

- We had the following definition for a generator:
  - Let p be a prime. $g \in \mathbb{Z}_p^*$ is a *generator* if for every $x \in \mathbb{Z}_p^*$, there exists some $i \in \mathbb{N}$ such that $x = g^i \bmod p$
- The following definition is equivalent:
  - Let p be a prime. $g \in \mathbb{Z}_p^*$ is a *generator* if for every $x \in \mathbb{Z}_p^*$, there exists some $y \in \{0, \dots, p-2\}$ such that $x = g^y \bmod p$
  - So instead of defining congruent class of a prime number as
$$\mathbb{Z}_p^* = \{x \in \mathbb{Z}_p : \gcd(x, n) = 1\} = \{1,2, \dots, p-1\}$$
  - The following definition is equivalent:
$$\mathbb{Z}_p^* = \{g^y \bmod p : y \in \{0,1, \dots, p-2\}\}$$
- **Main takeaway:** g generates $\mathbb{Z}_p^*$ iff $g^0 \bmod p$ through $g^{p-2} \bmod p$ hit all elements of $\mathbb{Z}_p^*$ (exactly once)

Diffie-Hellman Key Exchange

Diffie-Hellman Protocol

- Alice and Bob need a shared secret key in order to encrypt and send messages, but there's an eavesdropper Eve on their communication channel
- Public information: a prime number p and a generator g

Send $A = g^a \bmod p$

Send $B = g^b \bmod p$

Choose $a \in \mathbb{Z}_p^*$

Compute $B^a \bmod p$

Secret key: $g^{ab} \bmod p = B^a \bmod p = A^b \bmod p$

Choose $b \in \mathbb{Z}_p^*$

Compute $A^b \bmod p$

Diffie-Hellman Protocol Example

Diffie-Hellman Protocol:

- Alice chooses some secret $a \leftarrow \mathbb{Z}_p^*$ at random
- Bob chooses some secret $b \leftarrow \mathbb{Z}_p^*$ at random
- Alice sends $A = g^a \mod p$ to Bob
- Bob sends $B = g^b \mod p$ to Alice
- Alice computes $B^a = g^{ab} \mod p$ as the secret key
- Bob computes $A^b = g^{ab} \mod p$ as the secret key

- Suppose the prime is $p = 7$ and the generator is $g = 3$
- Suppose you were Alice and you pick $a = 3$, what do you send to Bob?
  - $A = 3^3 \bmod 7 = 27 \bmod 7 = 6$
- After sending A to Bob, suppose you receive $B = 2$, what is the shared key?
  - $B^a = 2^3 = 8 \equiv 1 \pmod{p}$