D10: Randomness in Computation

The eternal struggle of choosing between control and efficiency

Sec 101: MW 3:00-4:00pm DOW 1018
IA: Eric Khiu

MATH 525 for EECS 376

Announcement

- HW 5 due Monday June 24
  - Will have problems on cryptography (but not Zero Knowledge proofs)
  - Solution will be released 10pm
- Course evaluation due Tuesday June 25
- Cryptography crash course and Final Exam Review
  - June 24 3pm-4pm DOW1018 Cryptography Crash Course (Eric)
  - June 24 4pm-5pm DOW1010 Final Exam Review (Junghwan)
- Eric’s extra OH on June 25- To be confirmed

Review: NP-Completeness

- What do we need to show to prove that a language is NP-Complete?
  - The language is in NP, by providing an efficient verifier
  - The language is NP-hard, by reducing (poly-time reduction) from an NP-hard language

Starter: Matching Pennies

- Consider a game with two players Alice and Bob
- Each player has a penny and choose heads or tails
  - Alice wins the round if both choose the same outcome
  - Bob wins the round if both choose different outcome

		Alice	
		H	T
Bob	H	Alice wins	Bob wins
	T	Bob wins	Alice wins

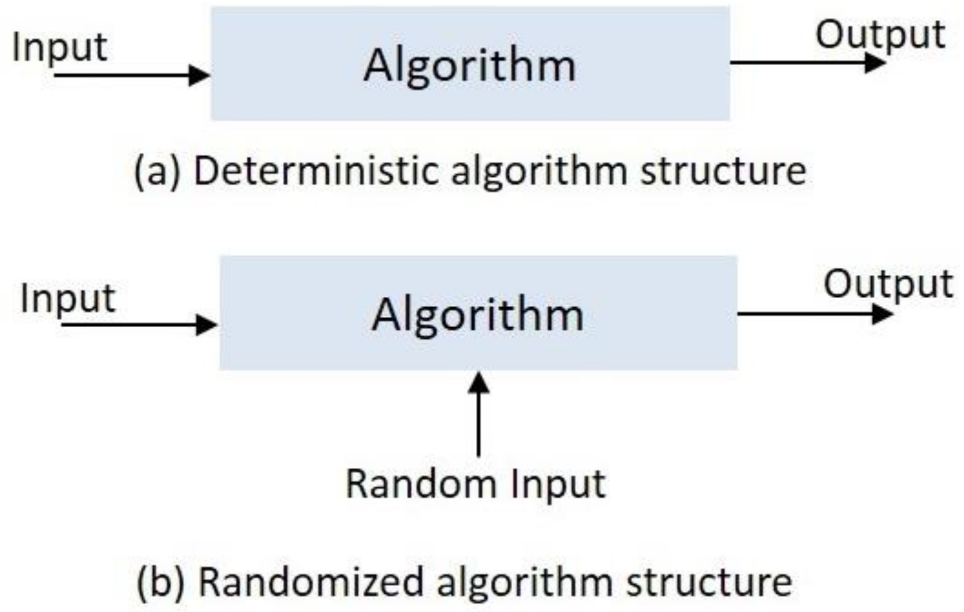
- They will play the game for 10 rounds the final winner is whoever wins the most rounds
- Consider the following algorithms:
  - Here, RAND(S) is a function that output a random element in set S

```
ALG1 (roundNum):
    if roundNum is odd then return H
    else return T
```

```
ALG2 (roundNum):
    num ← RAND({0,1})
    if num is odd then return H
    else return T
```

Discuss: If you were Alice, which algorithm would you choose and why?

Unit 4: Randomness in Computation



Sec 101: MW 3:00-4:00pm DOW 1018
IA: Eric Khiu

Motivation: Randomness in computation

- The algorithms we have seen thus far have been deterministic
  - Execute the same steps each time they are run and produce the same result
- If we use deterministic algorithm in Matching Pennies, the opponent would be able to observe the program's strategy once and defeat it every single time thereafter
  - How to prevent the opponent from predicting our moves? Make moves randomly!
- In this unit, we consider how **randomness** can be applied to computation
- We will start with reviewing/ introducing some tools to analyze randomness

Agenda

- Tools for analyzing randomized algorithms
- Probability Bounds
  - Markov's inequality
  - Chernoff bounds
  - Union Bounds

Tools for Analyzing Randomness

Warning: Math ahead!

Notes

Expected Values

- Let X be a discrete random variable (RV) over the set of events Ω, each with some probability in range $[0,1]$
- The **expected value** of X is

$$\mathbb{E}[X] = \sum_{\omega \in \Omega} \omega \cdot \Pr[X = \omega]$$

- Example: Consider a fair 6-sided die with RV D being the result of the roll.

$$\Pr[D = 1] = \Pr[D = 2] = \cdots = \Pr[D = 6] = \frac{1}{6}$$

$$\mathbb{E}[D] = 1\left(\frac{1}{6}\right) + 2\left(\frac{1}{6}\right) + \cdots + 6\left(\frac{1}{6}\right) = \frac{7}{2}$$

Linearity of Expectations

- Let X_1 and X_2 be two RVs and $X = c_1X_1 + c_2X_2$, then
$$\mathbb{E}[X] = c_1\mathbb{E}[X_1] + c_2\mathbb{E}[X_2]$$
- More generally, if we have RVs $X_1, \ldots, X_n$ and $X = c_1X_1 + \cdots + c_nX_n$, then
$$\mathbb{E}[X] = c_1\mathbb{E}[X_1] + \cdots + c_n\mathbb{E}[X_n] = \sum_{i=1}^{n} c_i\mathbb{E}[X_i]$$
- **Exercise:** Let X_1 be the result of a fair coin toss where $X_1 = 1$ if heads and $X_1 = 0$ if tails; X_2 be the results of a fair six-sided die roll. What is the expected value of $X = X_1 + X_2$?
  - $\mathbb{E}[X_1] = 0\left(\frac{1}{2}\right) + 1\left(\frac{1}{2}\right) = \frac{1}{2}$, $\mathbb{E}[X_2] = \frac{7}{2}$ from previous
  - By linearity of expectation, $\mathbb{E}[X] = \frac{1}{2} + \frac{7}{2} = 4$

Indicator Random Variable

- An **indicator RV** for an event A is defined as follows:

$$\mathbb{1}_A = [\![A]\!] = \begin{cases} 1 & \text{if } A \text{ happens} \\ 0 & \text{otherwise} \end{cases}$$

- Consider an event A that happens with probability $\Pr[A]$. Let X be an indicator random variable for A. What is $\mathbb{E}[X]$?

$$\mathbb{E}[X] = 1 \cdot \Pr[X = 1] + 0 \cdot \Pr[X = 0] = \Pr[A]$$

- If X is a discrete RV, it is sometimes useful to write $X = X_1 + \cdots + X_n$ to compute $E[X]$

Example: Are you a *peak*?

- Take integers $1, \dots, n$ and permutate them randomly as a sequence $a_1, \dots, a_n$. We say a_i is a *peak* if it is greater than all previous numbers, i.e., $a_i > a_j$ for all $j < i$. For example:

$$\underline{2}, 1, \underline{3}, \underline{5}, 4 \rightarrow \text{three peaks}$$

- Let X be the number of peaks in the sequence. Find $\mathbb{E}[X]$. You may leave your answer as a sum without simplifying it.
  - Let X_i be an indicator RV such that $X_i = 1$ if a_i is a peak, 0 otherwise
  - **Obs:** $\Pr[X_1 = 1] = 1$ (no previous), $\Pr[X_2 = 1] = 1/2$ (either $a_2 > a_1$ or $a_2 < a_1$)
  - In general, a_i is a peak $\Rightarrow a_i = \max\{a_1, \dots, a_i\}$, since all i numbers are distinct and only one max, so $\Pr[X_i = 1] = \Pr[a_i \text{ is } \textit{that} \max] = 1/i$
  - $\mathbb{E}[X] = \mathbb{E}[X_1] + \cdots + \mathbb{E}[X_n] = \Pr[X_1 = 1] + \cdots + \Pr[X_n = 1] = \sum_{i=1}^{n} \frac{1}{i}$

Exercise: Increasing Subarray

- Let A be a array of length n of a random permutation of n distinct integer. Compute the expected number of increasing subarrays in A of length k.
  - Hint: First define an indicator RV that consider whether a particular subarray of length k is increasing, then determine that probability
  - Let $X_i = 1$ if $A[i, \dots, i+k-1]$ is increasing and 0 otherwise
  - Since we only consider subarrays of length k, set $X_i = 0$ for $i = n-k+2, \dots, n$
  - For any array of length k, since all k numbers are distinct, we $k!$ permutations, but only one is increasing, so $\Pr[X_i = 1] = \Pr[A[i, \dots, i+k-1] \text{ is } \textit{that} \text{ increasing permutation}] = 1/k!$
  - $\mathbb{E}[X] = \mathbb{E}[X_1] + \cdots + \mathbb{E}[X_{n-k+1}] = \sum_{i=1}^{n-k+1} \frac{1}{k!} = \frac{n-k+1}{k!}$

Recap: Approximation Algorithms

- We can define how *good* an approximation is in terms of an approximation ratio α
  - Let $val(y)$ be a function that maps the output of a function to some value
  - Let OPT be the value of an optimal solution for some search problem
- An approximate solution y is said to be an **α-approximation** if

$$\alpha \cdot OPT \leq val(y) \quad \text{for maximization problem}$$

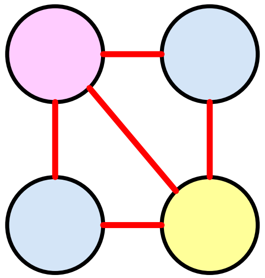
$$val(y) \leq \alpha \cdot OPT \quad \text{for minimization problem}$$

Discuss: Can we prove that the output of a randomized algorithm is an α-approximation?

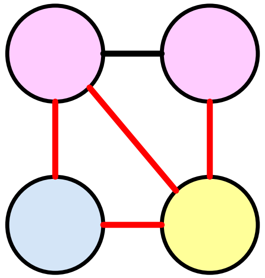
- Yes, but only in expectation-sometimes we got unlucky/ lucky and exit the bound!
  - Use $\mathbb{E}[val(y)]$ instead of $val(y)$

Example: 3-painting

- In an undirected graph, a *3-painting* is an assignment of one of three colors to each vertex. (Adjacent vertices do not *necessarily* need to have different colors). Given a 3-painting of an undirected graph, and edge is called *colorful* if its endpoints are assigned different colors.



#colorful edges = 5



#colorful edges = 4

Example: 3-painting

- In an undirected graph, a *3-painting* is an assignment of one of three colors to each vertex. (Adjacent vertices do not *necessarily* need to have different colors). Given a 3-painting of an undirected graph, and edge is called *colorful* if its endpoints are assigned different colors.
- Consider the following algorithm:

```
PAINTING(G=(V,E)):
 for v in V:
     num ← RAND({1,2,3}) // uniformly choose between {1,2,3} with prob. 1/3 each
     if num = 1 then v.color ← pink
     else if num = 2 then v.color ← blue
     else v.color ← yellow
```

- Prove that PAINTING is 2/3 approximation in expectation.
  - Hint: First compute $E[val(y)]$, then prove the bound

Example: 3-painting

```
PAINTING(G=(V,E)):
    for v in V:
        num ← RAND({1,2,3}) // uniformly choose between {1,2,3} with prob. 1/3 each
        if num = 1 then v.color ← pink
        else if num = 2 then v.color ← blue
        else v.color ← yellow
```

- Step 1: Compute $\mathbb{E}[val(y)]$
  - For each $e \in E$, let X_e be an indicator RV such that $X_e = 1$ if e is colorful and 0 otherwise
  - For each e, there are $3 \cdot 3 = 9$ possible paintings, 3 of them have same colors on both ends (6 of them have different colors), so $\Pr[X_e = 1] = \frac{6}{9} = \frac{2}{3}$
  - $\mathbb{E}[X] = \sum_{e \in E} \mathbb{E}[X_e] = \sum_{e \in E} \Pr[X_e = 1] = \frac{2}{3}|E|$

Example: 3-painting

```
PAINTING(G=(V,E)):
    for v in V:
        num ← RAND({1,2,3}) // uniformly choose between {1,2,3} with prob. 1/3 each
        if num = 1 then v.color ← pink
        else if num = 2 then v.color ← blue
        else v.color ← yellow
```

- Step 2: Prove bound
  - Now we have $\mathbb{E}[val(y)] = \frac{2}{3}|E|$
  - Let OPT be the optimum number of colorful edges. By definition, $OPT \leq |E|$
  - Therefore, $\mathbb{E}[val(y)] = \frac{2}{3}|E| \geq \frac{2}{3}OPT$, as desired.

TL; DPA

- We reviewed/ introduced tools to analyze randomness: expected values, linearity of expectations, and indicator RV
- It is sometimes useful to express a discrete RV as a sum of indicator RV when computing expectations
- For randomized algorithm, use $\mathbb{E}[val(y)]$ to prove approximation in expectation

Probability Bounds

Warning: Math ahead!

Notes

Starter: Search Algo Optimization

- Suppose you are optimizing a search algorithm for a large, constantly updating database system. The user can't wait for more than 1 second in general.
- You have the following two options

Option A	Option B
• Average search time: 0.05s • Potentially take more than 2s on a search during high-demand periods	• Average search time: 0.15s • Rarely take more than 0.6s on any search even under heady load

Discuss: Which one would you choose and why? What additional information you think will help you make the decision?

- $\Pr[A \text{ takes more than } 1\text{s}]$ and $\Pr[B \text{ takes more than } 1\text{s}]$
- **Obs:** They are both probabilities that the RV deviates from the expectation by some amount

Markov Inequality

- **Motivation:** Find an upper bound on the probability that a random variable X deviates from its expected value by some amount
- **Markov's Inequality:** Let X be a positive RV and $a > 0$, then

$$\Pr[X \geq a] \leq \frac{\mathbb{E}[X]}{a}$$

- Rearranging, we get

$$\Pr\left[X \geq a \cdot \mathbb{E}[X]\right] \leq \frac{1}{a}$$

$$\Pr[X \geq a] \leq \frac{\mathbb{E}[X]}{a}$$

$$\Pr[X \geq a \cdot \mathbb{E}[X]] \leq \frac{1}{a}$$

Example: Hash Table

- Suppose we have a hash table of size n^2 and a hash function h that chooses the mapping address uniformly at random from $0, \dots, n^2 - 1$.
- Let $S = \{s_1, \dots, s_n\}$ be the set of inserted elements and X be the RV indicating the number of collisions after performing n insertion.
- Find an upper bound on the probability that there is at least one collision ($h(s_i) = h(s_j)$) after inserting n distinct elements. (You may use $\frac{n-1}{2n} < \frac{1}{2}$ for any $n \in \mathbb{N}$)
  - First, compute $\mathbb{E}[X]$

$$\mathbb{E}[X] = \sum_{\substack{\text{all pairs } (i,j) \\ i \neq j}} \Pr[h(s_i) = h(s_j)] = \sum_{\substack{\text{all pairs } (i,j) \\ i \neq j}} \frac{1}{n^2} = \binom{n}{2} \cdot \frac{1}{n^2} = \frac{n-1}{2n}$$

  - Using Markov's inequality

$$\Pr[X \geq 1] \leq \frac{\mathbb{E}[X]}{1} = \frac{n-1}{2n} < \frac{1}{2}$$

$$\Pr[X \geq a] \leq \frac{\mathbb{E}[X]}{a}$$

$$\Pr[X \geq a \cdot \mathbb{E}[X]] \leq \frac{1}{a}$$

Search Algo Optimization Revisit

Option A	**Option B**
• Average search time: 0.05s • Potentially take more than 2s on a search during high-demand periods	• Average search time: 0.15s • Rarely take more than 0.6s on any search even under heady load

- Let A be the search time using option A and B be the search time using option B. Using Markov's inequality and $a = 1$, we have

$$\Pr[A \geq 1] \leq 0.05 \quad \text{and} \quad \Pr[B \geq 1] \leq 0.15$$

Discuss: Does this result change your decision?

- The upper bounds of the chance of option A taking at least 1 second is lower than that of option B- maybe A is better?
- WAIT: We haven't considered the "*rarely* take more than 0.6s"! Who knows $\Pr[B \geq 1]$ is actually 0.0001?
- **Takeaway:** Markov's inequality is a weak bound, but still applicable to many cases

"Reverse" Markov Inequality

- We can also find the lower bound on the probability that a RV X deviates from its expected value by some amount, if we know some upper bound for X
- If X is positive RV that is never larger than B and $a < B$, then

$$\Pr[X > a] \geq \frac{\mathbb{E}[X] - a}{B - a}$$

- **Example:** Suppose we have a biased coin were $\Pr[H] = 0.3$. Find a lower bound on the probability that there are strictly more than 10 heads after 100 tosses.
  - Let X be the number of heads after 100 tosses
  - We have $\mathbb{E}[X] = 0.3 \cdot 100 = 30$ and $B = 100$, so

$$\Pr[X > 10] \geq \frac{30 - 10}{100 - 10} = \frac{2}{9}$$

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
Chebyshev's Inequality: $$\Pr\left[\left\|\frac{1}{n}X - \mu\right\| \geq \varepsilon\right] \leq \frac{\sigma}{\varepsilon^2 n}$$	• X_i's i.i.d. s.t. $\mathbb{E}[X_i] = \mu$ and $\mathrm{Var}(X_i) = \sigma$ • $X = \sum_{i=1}^{n} X_i$ • Any $\varepsilon > 0$
(Combined) Chernoff-Hoeffding Bound: $$\Pr\left[\left\|\frac{1}{n}X - \mu\right\| \geq \varepsilon\right] \leq 2e^{-2\varepsilon^2 n}$$	• X_i's i.i.d. s.t. $X_i \in [0,1]$ and $\mathbb{E}[X_i] = \mu$ • $X = \sum_{i=1}^{n} X_i$ • Any $\varepsilon > 0$
Union bounds: $$\Pr[A_1 \cup A_2 \cup \cdots \cup A_n] \leq \sum_{i=1}^{n} \Pr[A_i]$$	N/A

Law of Large Numbers

- **Theorem** (Informal): If $X_1, X_2, \ldots$ are **independent, identically distributed (i.i.d.)** RVs with expectation μ, then

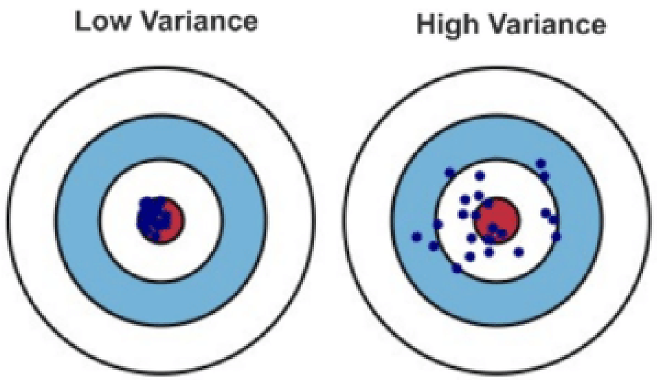
$$\lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} X_i = \mu$$

- Limitation of LLN: LLN says distribution of sum is "concentrated" around its expectation as $n \to \infty$. (However, it doesn't say how quickly it happens or what the distribution looks like.)

Variance

- **Definition:** The variance of a RV X is the average squared-distance of X from its mean, i.e.,

$$\mathrm{Var}(X) = \mathbb{E}[(X - \mathbb{E}[X])^2]$$



- **Proposition:** Alternate form of variance

$$\mathrm{Var}(X) = \mathbb{E}[X^2] - (\mathbb{E}[X])^2$$

Properties of Variance

- Scalar Multiplication:

$$\mathrm{Var}(cX) = c^2\mathrm{Var}(X)$$

- Sum of independent RV:

$$\mathrm{Var}\left(\sum_{i=1}^{n} X_i\right) = \sum_{i=1}^{n} \mathrm{Var}(X_i)$$

- Warning: This is different from linearity of expectation because the scalar will get squared!

Chebyshev's Inequality

- **Theorem:** For any RV X any scalar $a > 0$

$$\Pr[|X - \mathbb{E}[X]| \geq a] \leq \frac{\mathrm{Var}(X)}{a^2}$$

Absolute difference from mean

- **Exercise:** Prove this inequality
  - Hint: Square both sides of Markov's Inequality
- **Chebyshev's + Law of Large Numbers:** If $X = X_1 + X_2 + \cdots + X_n$ is the sum of n **i.i.d.** RVs with $\mathbb{E}[X_i] = \mu$ and $\mathrm{Var}(X_i) = \sigma$ for all $i = 1, .., n$, then for any $\varepsilon > 0$

$$\Pr\left[\left|\frac{1}{n}X - \mu\right| \geq \varepsilon\right] \leq \frac{\sigma}{\varepsilon^2 n}$$

Summary of Probabilities Bounds

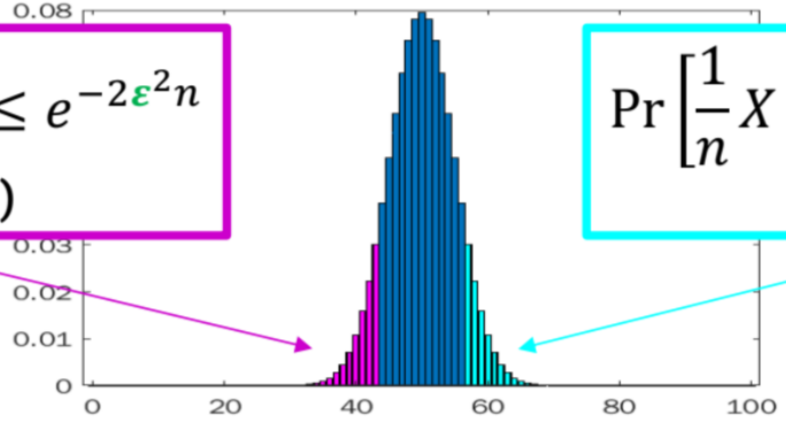
Probability Bounds	Constraints
Markov's inequality: $$\Pr[X \geq a] \leq \frac{\mathbb{E}[X]}{a}$$	• X is a positive RV
"Reverse" Markov's inequality: $$\Pr[X > a] \geq \frac{\mathbb{E}[X] - a}{B - a}$$	• X is a positive RV • X is upper bounded by some B
Chebyshev's Inequality: $$\Pr\left[\left\lvert\frac{1}{n}X - \mu\right\rvert \geq \varepsilon\right] \leq \frac{\sigma}{\varepsilon^2 n}$$	• X_i's i.i.d. s.t. $\mathbb{E}[X_i] = \mu$ and $\mathrm{Var}(X_i) = \sigma$ • $X = \sum_{i=1}^{n} X_i$ • Any $\varepsilon > 0$
(Combined) Chernoff-Hoeffding Bound: $$\Pr\left[\left\lvert\frac{1}{n}X - \mu\right\rvert \geq \varepsilon\right] \leq e^{-2\varepsilon^2 n}$$	• X_i's i.i.d. s.t. $X_i \in [0,1]$ and $\mathbb{E}[X_i] = \mu$ • $X = \sum_{i=1}^{n} X_i$ • Any $\varepsilon > 0$
Union bounds: $$\Pr[A_1 \cup A_2 \cup \cdots \cup A_n] \leq \sum_{i=1}^{n} \Pr[A_i]$$	N/A

Chernoff-Hoeffding Bounds

- **Theorem:** If $X = X_1 + X_2 + \cdots + X_n$ is the sum of n **i.i.d.** RVs with each $X_i \in [0,1]$, then, for any $\varepsilon > 0$

$$\Pr\left[\frac{1}{n}X \le \mathbb{E}[X_i] - \boldsymbol{\varepsilon}\right] \le e^{-2\boldsymbol{\varepsilon}^2 n}$$

(lower tail bound)



$$\Pr\left[\frac{1}{n}X \ge \mathbb{E}[X_i] + \boldsymbol{\varepsilon}\right] \le e^{-2\boldsymbol{\varepsilon}^2 n}$$

(upper tail bound)

- **Combined Chernoff-Hoeffding:**

$$\Pr\left[\left|\frac{1}{n}X - \mathbb{E}[X_i]\right| \ge \varepsilon\right] \le 2e^{-2\varepsilon^2 n}$$

Summary of Probabilities Bounds

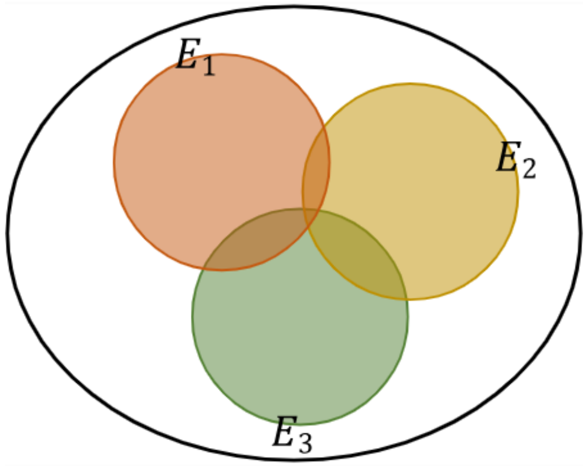
Probability Bounds	Constraints
Markov's inequality: $$\Pr[X \geq a] \leq \frac{\mathbb{E}[X]}{a}$$	• X is a positive RV
"Reverse" Markov's inequality: $$\Pr[X > a] \geq \frac{\mathbb{E}[X] - a}{B - a}$$	• X is a positive RV • X is upper bounded by some B
Chebyshev's Inequality: $$\Pr\left[\left\lvert\frac{1}{n}X - \mu\right\rvert \geq \varepsilon\right] \leq \frac{\sigma}{\varepsilon^2 n}$$	• X_i's i.i.d. s.t. $\mathbb{E}[X_i] = \mu$ and $\mathrm{Var}(X_i) = \sigma$ • $X = \sum_{i=1}^{n} X_i$ • Any $\varepsilon > 0$
(Combined) Chernoff-Hoeffding Bound: $$\Pr\left[\left\lvert\frac{1}{n}X - \mu\right\rvert \geq \varepsilon\right] \leq 2e^{-2\varepsilon^2 n}$$	• X_i's i.i.d. s.t. $X_i \in [0,1]$ and $\mathbb{E}[X_i] = \mu$ • $X = \sum_{i=1}^{n} X_i$ • Any $\varepsilon > 0$
Union bounds: $$\Pr[A_1 \cup A_2 \cup \cdots \cup A_n] \leq \sum_{i=1}^{n} \Pr[A_i]$$	N/A

Union Bound

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

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

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



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

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
Chebyshev's Inequality: $$\Pr\left[\left\|\frac{1}{n}X - \mu\right\| \geq \varepsilon\right] \leq \frac{\sigma}{\varepsilon^2 n}$$	• X_i's i.i.d. s.t. $\mathbb{E}[X_i] = \mu$ and $\mathrm{Var}(X_i) = \sigma$ • $X = \sum_{i=1}^{n} X_i$ • Any $\varepsilon > 0$
(Combined) Chernoff-Hoeffding Bound: $$\Pr\left[\left\|\frac{1}{n}X - \mu\right\| \geq \varepsilon\right] \leq 2e^{-2\varepsilon^2 n}$$	• X_i's i.i.d. s.t. $X_i \in [0,1]$ and $\mathbb{E}[X_i] = \mu$ • $X = \sum_{i=1}^{n} X_i$ • Any $\varepsilon > 0$
Union bounds: $$\Pr[A_1 \cup A_2 \cup \cdots \cup A_n] \leq \sum_{i=1}^{n} \Pr[A_i]$$	N/A