# CS-GY 6763: Lecture 2 Chebyshev's Inequality, Union Bound, Exponential Tail Bounds

NYU Tandon School of Engineering, Prof. Christopher Musco

It can be hard to know how formal to be. We will try to provide feedback on first problem set for anyone who is either too rigorous or too loose. It's a learning process.

### Things that are generally fine:

- Can assume input size n is > C for some constant c. E.g. n > 2, n > 10.
- Similarly can assume  $\epsilon < c$  for constant c. E.g.  $\epsilon < .1, \epsilon < .01$ .
- If I write O(z), you are free to choose the constant. E.g., it's fine if your analysis of CountSketch only works for tables of size  $1000 \cdot m$ .
- Derivatives, integrals, etc. can be taken from e.g. WolframAlpha without working through steps.
- Basic inequalities can be used without proof, as long as you verify numerically. Don't need to include plot on problem set.

### EXAMPLE INEQUALITY



### EXAMPLE INEQUALITY

$$1 - \epsilon \leq \left(\frac{1}{1 + \epsilon}\right) \leq 1 - .5\epsilon \text{ for } \epsilon \in [0, 1].$$

Proof by plotting:



**Tip:** When confronted with a complex expression, try to simplify by using big-Oh notation, or just rounding things off. Then clean-up your proof after you get to a solution.

### Examples:

- $(m-1) \approx m$
- $\cdot \frac{1}{n} \frac{1}{n^2} \approx \frac{1}{n}$
- ·  $\log(n/2) \approx \log(n)$

Let *h* be a <u>random function</u> from  $|\mathcal{U}| \to \{1, ..., m\}$ . This means that *h* is constructed by an algorithm using a seed of random numbers, but then the function is fixed. Given input  $x \in \mathcal{U}$ , it always returns the same output, h(x).

**Definition:** Uniformly Random Hash Function. A random function  $h : \mathcal{U} \rightarrow \{1, \dots, m\}$  is called uniformly random if:

- $\Pr[\underline{h(x)} = i] = \frac{1}{m}$  for all  $x \in \mathcal{U}$ ,  $i \in \{1, \dots, m\}$ .
- $h(x), h(y), h(z), \ldots$  are mutually independent random variables for all  $x, y, z, \ldots \in \mathcal{U}$ .

• Which implies that 
$$\Pr[h(x) = h(y)] = \sqrt{2}$$

$$\mathcal{U}=$$
 universe of possible keys,  $m=$  number of values hashed to

Why are uniformly random hash functions impractical to



### NOTE ON RANDOM HASH FUNCTIONS

\@

For the application to CountMin from last class we can weaken our assumption that *h* is uniformly random.

Definition (Universal hash function)

A random hash function  $h : U \to \{1, ..., m\}$  is <u>universal</u> if, for any fixed  $x, y \in U$ ,

$$\Pr[h(x) = h(y)] \overleftarrow{\leq}_m^1.$$

Claim: A uniformly random hash-function is universal.

**Efficient alternative:** Let p be a prime number between  $|\mathcal{U}|$  and  $2|\mathcal{U}|$ . Let a, b be random numbers in  $0, \ldots, p, a \neq 0$ .

$$(h(x) = (0, x + b) (mod p)$$
 (mod m)

is universal. Lecture notes with proof posted on website.

Another definition you might come across:

**Definition (Pairwise independent hash function)** A random hash function  $h : U \to \{1, ..., m\}$  is pairwise independent if, for any fixed  $x, y \in U, i, j \in \{1, ..., m\}$ ,

$$\Pr[h(x) = i \cap h(y) = j] = \frac{1}{m^2}.$$

Can we naturally extended to k-wise independence for k > 2, which is strictly stronger, and needed for some applications.

#### LECTURE ROAD MAP

Last week we saw the power of <u>Linearity of Expectation</u> + <u>Markov's</u>. This week we will discuss four more tools:

- Linearity of Variance + Chebyshev's Inequality
- Union Bound + Exponential Tail Bounds



These six simple tools combined are surprising powerful and flexible. They form the cornerstone of randomized algorithm design.

# CHEBYSHEV'S INEQUALITY

A new concentration inequality:

# Lemma (Chebyshev's Inequality)

Let X be a random variable with expectation  $\mathbb{E}[X]$  and variance

 $\sigma^{2} = \operatorname{Var}[X]. \text{ Then for any } k > 0,$   $Pr[|X - \mathbb{E}[X]| \ge k \cdot \sigma] \le \frac{1}{k^{2}}$   $\mathcal{G} = \left[ \operatorname{Var}(k) \right]$ 



 $\sigma = \sqrt{Var[X]}$  is the <u>standard deviation</u> of X. Intuitively this bound makes sense: it is tighter when  $\sigma$  is smaller.

Properties of Chebyshev's inequality:

- Good: No requirement of non-negativity. X can be anything.
- **Good:** Two-sided. Bounds the probability that  $|X \mathbb{E}X|$  is large, which means that *X* isn't too far above <u>or</u> below its expectation. Markov's only bounded probability that *X* exceeds  $\mathbb{E}[X]$ .
- Bad/Good: Requires a bound on the variance of of X.

No hard rule for which to apply! Both Markov's and Chebyshev's are useful in different settings.

# PROOF OF CHEBYSHEV'S INEQUALITY

**Idea:** Apply Markov's inequality to the (non-negative) random variable  $S = (X - \mathbb{E}[X])^2$ .

# Lemma (Chebyshev's Inequality)

Let X be a random variable with expectation  $\mathbb{E}[X]$  and variance  $\sigma^2 = \text{Var}[X]$ . Then for any k > 0,

**Markov's inequality**: for non-negative r.v. S,  $Pr[S \ge t] \le \mathbb{E}[S]/t$ .

# If I flip a fair coin 100 times, show that with 93% chance I get between 30 and 70 heads?

Let  $C_1, \ldots, C_{100}$  be independent random variables that are 1 with probability 1/2, 0 otherwise.

Let  $H = \sum_{i=1}^{100} C_i$  be the number of heads that get flipped.

$$\mathbb{E}[H] = \mathbb{E}\left(\sum_{i=1}^{100} C_i\right) = \sum_{i=1}^{100} \mathbb{E}(C_i) = \sum_{i=1}^{100} Y_2 = 0$$

$$Var[H] = V_{0r} \left( \sum_{i=1}^{100} C_{i} \right) = \sum_{i=1}^{100} V_{0r} \left( C_{i} \right)^{2} = \sum_{i=1}^{100} C_{2r} = \sum_{i=1$$

**Fact:** For <u>pairwise</u> independent random variables  $X_1, \ldots, X_m$ ,

 $\operatorname{Var}[X_1 + X_2 + \ldots + X_m] = \operatorname{Var}[X_1] + \operatorname{Var}[X_2] + \ldots + \operatorname{Var}[X_m].$ 

I.e., we require that for any  $i, j X_i$  and  $X_j$  are independent. This is strictly weaker than <u>mutual independence</u>, which requires that for all possible values  $v_1, \ldots, v_k$ ,

$$\Pr[X_1 = v_1, \dots, X_k = v_k] = \Pr[X_1 = v_1] \cdot \dots \cdot \Pr[X_k = v_k].$$

#### 50-30:20 70-50:20

# If I flip a fair coin 100 times, show that with 93% chance I get between 30 and 70 heads?

Let  $C_1, \ldots, C_{100}$  be independent random variables that are 1 with probability 1/2, 0 otherwise.

Let  $H = \sum_{i=1}^{100} C_i$  be the number of heads that get flipped. Var[H] = 25. Chebyshev's:  $P_{c} \left[ |H - 50| \ge (k \cdot 5) \right] \le \frac{1}{k^2}$  $k \cdot M$   $P_{c} \left[ |H - 50| \ge 220 \right] \le \frac{1}{k^2} \le \frac{1}{16} = 6 \dots \%$  Recall the set up from last lecture:

Draw items  $x_1, \ldots, x_m$  uniformly at random from a set of size n and count the number of collisions:

$$D = \sum_{\substack{i,j \in \{1,\ldots,m\}\\i < j}} \mathbb{1}[x_i == x_j].$$

We showed that  $\mathbb{E}[D] = \binom{m}{2} \frac{1}{n}$ .

**Exercise:** Show that  $Var[D] \le {\binom{m}{2}} \frac{1}{n}$  and use to prove the claim on Slide 32 from Lecture 1.

# Abstract architecture of a streaming algorithm:

Have massive dataset  $X = x_1, ..., x_n$  with *n* pieces of data that arrive in a sequential stream. There is far too much data to store or process it in a single location.

- Still want to analyze the data: i.e. fit a model or (approximately) compute some function f(X).
- To do so, we must compress data "on-the-fly", storing some smaller data structure which still contains interesting information.
- Often can only take a <u>single-pass</u> over the data.

**Count-Min** was our first example of a streaming algorithm for the  $(\epsilon, k)$ -frequent items problem.

**Sensor data:** GPS or seismometer readings to detect geological anomalies, telescope images, satellite imagery, highway travel time sensors.

**Web traffic and data:** User data for website, including e.g. click data, web searches and API queries, posts and image uploads on social media.

**Training machine learning models:** Often done in a streaming setting when training dataset is huge, often with multiple passes.



Lots of software frameworks exist for easy development of streaming algorithms.

#### DISTINCT ELEMENTS PROBLEM

**Input:**  $x_1, \ldots, x_n \in \mathcal{U}$  where  $\mathcal{U}$  is a huge universe of items.

Output: Number of distinct inputs.

Example: f(1, 10, 2, 4, 9, 2, 10, 4) - 5

# Applications:

- Distinct users hitting a webpage.
- Distinct values in a database column (e.g. for estimating the size of group by queries)
- Number of distinct queries to a search engine.
- Distinct motifs in DNA sequence.

Implementations widely used at Google (Sawzall, Dremel, PowerDrill), Yahoo, Twitter, Facebook Presto, etc.

**Input:**  $X_1, \ldots, X_n \in \mathcal{U}$  where  $\mathcal{U}$  is a huge universe of items.

**Output:** Number of distinct inputs. **Example:**  $f(1, 10, 2, 4, 9, 2, 10, 4) \rightarrow 5$ 

Flajolet-Martin (simplified):

• Choose random hash function  $h: \underline{\mathcal{U}} \to [\underline{0}, \underline{1}]$ .

$$\cdot S = 1$$

• For  $i = 1, \ldots, n$ 



The hash function *h* maps from  $\mathcal{U}$  to a random point in [0, 1]?  $\{2, 1, \dots, m\}$ 

### Hashing to real numbers:

- Impossible to implement h(x) in reality, but you can replace it with  $\frac{g(x)}{k}$ , where g is a hash function that maps to  $\{0, 1, \dots, k\}$  for sufficiently large k.
- All results hold if this "discrete" hash is used instead, but the analysis is simpler if we assume access to *h*.
- Just like when we assumed uniform random hash functions, this is a useful abstraction which makes understanding and analyzing the underlying algorithms easier.

### VISUALIZATION

# Flajolet-Martin (simplified):

- Choose random hash function  $h: \mathcal{U} \rightarrow [0, 1]$ .
- S = 1
- For i = 1, ..., n•  $S \leftarrow \min(S, h(x_i))$
- Return:  $\tilde{D} = \frac{1}{S} 1$





Let D equal the number of distinct elements in our stream.



### D unique locations after hashing

**Intuition:** When *D* is larger, *S* will be smaller. Makes sense to return the estimate  $\tilde{D} = \begin{pmatrix} 1 \\ \frac{1}{5} & -1 \end{pmatrix}$ 

#### **FM ANALYSIS**



Let *D* equal the number of distinct elements in our stream.



#### THE CALCULUS PROOF



# PROOF "FROM THE BOOK"



# PROOF "FROM THE BOOK"



### **FM ANALYSIS**

(

$$\mathbb{E}S = \frac{1}{D+1}. \text{ Estimate: } \tilde{D} = \frac{1}{5} - 1. \text{ We have for } \epsilon < \frac{1}{2}: \qquad \downarrow + \alpha \leq |-\alpha|$$

$$\mathbb{E}S = \frac{1}{D+1}. \text{ Estimate: } \tilde{D} = \frac{1}{5} - 1 \qquad (\text{If } (1-\epsilon)\mathbf{\hat{5}} \leq \mathbf{\hat{4}}S \leq (1+\epsilon)\mathbf{\hat{5}}, \text{ then:} \qquad \downarrow + \alpha \leq |-\alpha|$$

$$\mathbb{D} = \frac{1}{5} - 1 \qquad (\text{If } (1-\epsilon)\mathbf{\hat{5}} \leq \mathbf{\hat{4}}S \leq (1+\epsilon)\mathbf{\hat{5}}, \text{ then:} \qquad \downarrow + \alpha \leq |+\alpha|$$

$$\mathbb{D} = \frac{1}{5} - 1 \qquad (1-4\epsilon)\mathbf{\hat{D}} < \mathbf{\hat{D}} (1+4\epsilon)\mathbf{\hat{D}}. \qquad \downarrow + \alpha \leq |+\alpha|$$

$$\mathbb{D} = \frac{1}{5} - 1 \qquad (1-4\epsilon)\mathbf{\hat{D}} < \mathbf{\hat{D}} (1+4\epsilon)\mathbf{\hat{D}}. \qquad (1+2\epsilon)\mathbf{\hat{5}} = (1+2\epsilon)\mathbf{\hat{5}} \qquad (\frac{1}{1-\alpha})\mathbf{\hat{5}} \leq 2\epsilon(\mathbf{\hat{5}})^{2\epsilon}$$

$$(1-2\epsilon)\mathbf{\hat{5}} = \frac{1}{5} - 1 \leq (1+2\epsilon)\mathbf{\hat{5}} = \frac{1}{5} \leq (1+2\epsilon)\mathbf{\hat{5}} = 1 \qquad 2\epsilon(\mathbf{\hat{5}})^{2\epsilon} \leq 1\epsilon(\mathbf{\hat{5}})^{2\epsilon}$$

$$(1-2\epsilon)\mathbf{\hat{5}} = 1 \leq (1+2\epsilon)\mathbf{\hat{5}} = 1 \qquad 2\epsilon(\mathbf{\hat{5}})^{2\epsilon} = 1 \qquad \epsilon(\mathbf{\hat{5}})^{2\epsilon}$$

$$(1-2\epsilon)\mathbf{\hat{5}} = 1 \leq (1+2\epsilon)\mathbf{\hat{5}} = 1 \qquad \epsilon(\mathbf{\hat{5}})^{2\epsilon} = 1 \qquad \epsilon(\mathbf{\hat{5}})^{2\epsilon}$$

$$(1-2\epsilon)\mathbf{\hat{5}} = 1 \leq (1+2\epsilon)\mathbf{\hat{5}} = 1 \qquad \epsilon(\mathbf{\hat{5}})^{2\epsilon} = 1 \qquad \epsilon(\mathbf{\hat{5}})^{2\epsilon}$$

$$(1-2\epsilon)\mathbf{\hat{5}} = 1 \leq (1+2\epsilon)\mathbf{\hat{5}} = 1 \qquad \epsilon(\mathbf{\hat{5}})^{2\epsilon} = 1 \qquad \epsilon(\mathbf{\hat{5}})^{2\epsilon}$$

$$(1-2\epsilon)\mathbf{\hat{5}} = 1 \leq (1+2\epsilon)\mathbf{\hat{5}} = 1 \qquad \epsilon(\mathbf{\hat{5}})^{2\epsilon} = 1 \qquad \epsilon(\mathbf{\hat{5}})^{2\epsilon}$$

$$(1-2\epsilon)\mathbf{\hat{5}} = 1 \leq (1+2\epsilon)\mathbf{\hat{5}} = 1 \qquad \epsilon(\mathbf{\hat{5}})^{2\epsilon} = 1 \qquad \epsilon(\mathbf{\hat{5}})^{2\epsilon}$$

$$(1-2\epsilon)\mathbf{\hat{5}} = 1 \leq (1+2\epsilon)\mathbf{\hat{5}} = 1 \qquad \epsilon(\mathbf{\hat{5}})^{2\epsilon} = 1 \qquad \epsilon(\mathbf{\hat{5}})^{2\epsilon}$$

$$(1-2\epsilon)\mathbf{\hat{5}} = 1 \leq (1+2\epsilon)\mathbf{\hat{5}} = 1 \qquad \epsilon(\mathbf{\hat{5}})^{2\epsilon} = 1 \qquad \epsilon(\mathbf{\hat{5}})^{2\epsilon}$$

$$(1-2\epsilon)\mathbf{\hat{5}} = 1 \leq (1+2\epsilon)\mathbf{\hat{5}} = 1 \qquad \epsilon(\mathbf{\hat{5}})^{2\epsilon} = 1 \qquad \epsilon(\mathbf{\hat{5})^{2\epsilon} =$$

#### CALCULUS PROOF

Lemma  

$$\underbrace{\operatorname{Var}[S]}_{\mathbb{Z}} = \underbrace{\mathbb{E}[S^2]}_{\mathbb{E}} - \underbrace{\mathbb{E}[S]}_{\mathbb{E}}^2 = \underbrace{\binom{2}{(D+1)(D+2)} - \frac{1}{(D+1)^2}}_{\mathbb{E}} \leq \frac{1}{(D+1)^2}. \quad \mathcal{L} = \underbrace{\mathbb{E}[\zeta]}_{\mathbb{E}}^2$$
Proof:  

$$\mathbb{E}[S^2] = \int_{0}^{1} \Pr[S^2 \geq \lambda] d\lambda \quad (D+1)(D+1)$$

$$= \int_{0}^{1} \Pr[S \geq \sqrt{\lambda}] d\lambda$$

$$= \int_{0}^{1} (1 - \sqrt{\lambda})^D d\lambda$$

$$= \frac{2}{(D+1)(D+2)}$$

www.wolframalpha.com/input/?i=integral+from+0+to+1+
of+%281-sqrt%28x%29%25ED

# PROOF "FROM THE BOOK"

 $\mathbb{E}[S^2] = ??.$ 



$$(1-\varepsilon) \not\in [S] \leq S \leq (1+\varepsilon) \not\in [S]$$
  
 $(1-\varepsilon) \not\in [S] \leq \varepsilon \mid E(S)$ 

- $\cdot \mathbb{E}[S] = \frac{1}{D+1} = \mu.$
- $\operatorname{Var}[S] \leq \underline{\mu}^2$ . Standard deviation:  $\underline{\sigma} \leq \underline{\mu}$ .
- Want to bound  $\Pr[|\underline{S} \underline{\mu}| \ge \epsilon \mu] \le \delta$ .

Chebyshev's:  $\Pr[|S - \mu| \ge \epsilon \mu] \le \Pr[|S - \mu| \ge \epsilon \sigma] \le \frac{1}{\epsilon^2}$ .

Vacuous bound. Our variance is way too high! モニレ ノ レ



**Trick of the trade:** Repeat many independent trials and take the mean to get a better estimator.

Given i.i.d. (independent, identically distributed) random variables  $X_1, \ldots, X_k$  with mean  $\mu$  and variance  $\sigma^2$ , what is:

$$\mathbb{E}\left[\frac{1}{k}\sum_{i=1}^{k}X_{i}\right] = \frac{1}{k}, \overset{\mu}{\underset{i=1}{\overset{k}{\underset{i=1}{\underset{i=1}{\overset{k}{\underset{i=1}{\underset{i=1}{\overset{k}{\underset{i=1}{\overset{k}{\underset{i=1}{\overset{k}{\underset{i=1}{\underset{i=1}{\overset{k}{\underset{i=1}{\underset{i=1}{\overset{k}{\underset{i=1}{\underset{i=1}{\overset{k}{\underset{i=1}{\underset{i=1}{\overset{k}{\underset{i=1}{\atopi=1}{\underset{i=1}{\underset{i=1}{\underset{i=1}{\underset{i=1}{\underset{i=1}{\underset{i=1}{\underset{i=1}{\underset{i=1}{\underset{i=1}{\underset{i=1}{\underset{i=1}{\atopi=1}{\underset{i=1}{\underset{i=1}{\underset{i=1}{\underset{i=1}{\underset{i=1}{\underset{i=1}{\atopi=1}{\underset{i=1}{\atopi}{\underset{i=1}{\atopi=1}{\underset{i=1}{\atopi}{\underset{i=1}{\atopi=1}{\atopi=1}{\underset{i=1}{\atopi=1}{\underset{i=1}{\atopi=1}{\atopi=1}{\atopi=1}{\underset{i=1}{\atopi=$$

$$\cdot \operatorname{Var} \begin{bmatrix} \frac{1}{k} \sum_{i=1}^{k} X_i \end{bmatrix} = \frac{1}{k^2} \operatorname{Var} \begin{bmatrix} \frac{2}{k} & X_i \end{bmatrix} = \frac{1}{k^2} \cdot 6^2$$

$$K \cdot 6^2$$

#### **FM ANALYSIS**

Using independent hash functions, maintain k independent sketches  $S_1, \ldots, S_k$ .



Flajolet-Martin:

• Choose <u>k</u> random hash function  $\underline{h_1, \ldots, h_k} : \mathcal{U} \to [0, 1]$ . •  $\underline{S_p} = 1, \ldots, \underline{S_p} = 1$ (For  $i = 1, \ldots, n$ •  $S_j \leftarrow \min(S_j, h_j(x_j))$  for all  $j \in 1, \ldots, k$ . •  $\underline{S} = (\underline{S_1} + \ldots + \underline{S_k})/k$ • Return:  $\frac{1}{5} - 1$ 

#### **FM ANALYSIS**

1 estimator:  

$$\begin{array}{c}
1 \text{ estimator:} \\
\vdots \mathbb{E}[S] = \frac{1}{D+1} = \mu.
\\
\vdots \text{ Var}[S] = \mu^{2}
\end{array}$$

$$\begin{array}{c}
1 \text{ for } & W = \frac{1}{C_{T}} & (\mu = C G) \\
\vdots & U = C G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots & U = G & (\mu = C G) \\
\vdots &$$

up to error  $\epsilon$  with success probability  $1 - \delta$ .

35

#### NOTE ON FAILURE PROBABILITY

.05

,01 10000

- Achieves any accuracy desired  $(1/\epsilon^2)$  dependence cannot be improved.
  - No dependence on number of distinct elements <u>D</u>. Naive algorithm takes O(D) space.
  - But...  $1/\delta$  dependence is not ideal. For 95% success rate, pay a  $\frac{1}{5\%} = 20$  factor overhead in space.

We can get a better bound depending on  $O(\log(1/\delta))$  sing exponential tail bounds.
In practice, we cannot hash to real numbers on [0, 1]. Instead, map to bit vectors.

Real Flajolet-Martin / HyperLogLog







- Estimate # distinct elements based on maximum number of trailing zeros **m**.
- The more distinct hashes we see, the higher we expect this maximum to be.

37

#### LOGLOG SPACE

**Total Space:** 
$$O\left(\frac{\log \log D}{\epsilon^2} + (\log D)\right)$$
 for an  $\epsilon$  approximate count.

"Using an auxiliary memory smaller than the size of this abstract, the LogLog algorithm makes it possible to estimate in a single pass and within a few percents the number of different words in the whole of Shakespeare's works." – Flajolet, Durand.

Using HyperLogLog to count 1 billion distinct items with 2% accuracy:

space used = 
$$O\left(\frac{\log \log D}{\epsilon^2} + \log D\right)$$
  
=  $\left(\frac{1.04 \cdot \lceil \log_2 \log_2 D \rceil}{\epsilon^2} + \lceil \log_2 D \rceil\right)$  bits  
=  $\frac{1.04 \cdot 5}{.02^2} + 30 = \underline{13030}$  bits  $\approx \underline{1.6 \ kB!}$ 

#### DISTRIBUTED DISTINCT ELEMENTS



Distinct elements summaries are "mergeable". No need to share lists of distinct elements if those elements are stored on different machines. Just share minimum hash value. Implementations: Google PowerDrill, Facebook Presto, Twitter Algebird, Amazon Redshift.

**Use Case:** Exploratory SQL-like queries on tables with 100's of billions of rows.

- Count number of distinct users in Germany that made at least one search containing the word 'auto' in the last month.
- Count number of distinct subject lines in emails sent by users that have registered in the last week, in comparison to number of emails sent overall (to estimate rates of spam accounts).

Answering a query requires a (distributed) linear scan over the database: <u>2 seconds</u> in Google's distributed implementation.

Google Paper: "Processing a Trillion Cells per Mouse Click"

# BREAK

# Load balancing problem:

Suppose Google answers map search queries using servers  $A_1, \ldots, A_q$ . Given a query like "new york to rhode island", common practice is to choose a random hash function  $h \rightarrow \{1, \ldots, q\}$  and to route this query to server: Why use a hash function instead of just distributing requests randomly?

**Goal:** Ensure that requests are distributed evenly, so no one server gets loaded with too many requests. We want to avoid downtime and slow responses to clients.

Suppose we have *n* servers and *m* requests,  $x_1, \ldots, x_m$ . Let  $s_i$  be the number of requests sent to server  $i \in \{1, \ldots, n\}$ :

$$s_{i} = \sum_{j=1}^{m} \mathbb{1}[h(\underline{x}_{j}) = i].$$

$$If[s_{i}] = \sum_{j=1}^{m} \sum_{j=1}^{m} I[h(\underline{x}_{j}) = i].$$

Formally, our goal is to understand the value of maximum load on any server, which can be written as the random variable:

$$S = \max_{i \in \{1, \dots, n\}} S_i.$$

A good first step in any analysis of random variables is to first think about expectations. If we have *n* servers and *m* requests, for any  $i \in \{1, ..., n\}$ :

$$\mathbb{E}[s_i] = \sum_{j=1}^m \mathbb{E}\left[\mathbb{1}[h(x_j) = i]\right] = \underbrace{\frac{m}{n}}_{n} \mathbb{E}\left[\mathbb{E}\left[\mathbb{1}[h(x_j) = i]\right] = \underbrace{\frac{m}{n}}_{n} \mathbb{E}\left[\mathbb{E}\left[\mathbb{1}[h(x_j) = i]\right]\right] = \underbrace{\frac{m}{n}}_{n} \mathbb{E}\left[\mathbb{E}\left[\mathbb{E}\left[\mathbb{1}[h(x_j) = i]\right]\right] = \underbrace{\frac{m}{n}}_{n} \mathbb{E}\left[\mathbb{E}\left[\mathbb{E}\left[\mathbb{E}\left[\mathbb{1}[h(x_j) = i]\right]\right]\right] = \underbrace{\frac{m}{n}}_{n} \mathbb{E}\left[\mathbb{E}$$

But it's very unclear what the expectation of  $S = \max_{i \in \{1,...,n\}} s_i$ is... in particular,  $\mathbb{E}[S] \neq \max_{i \in \{1,...,n\}} \mathbb{E}[s_i]$ .

**Exercise:** Convince yourself that for two random variables A and B,  $\mathbb{E}[\max(A, B)] \neq \max(\mathbb{E}[A], \mathbb{E}[B])$  even if those random variable are independent.

Number of servers: To reduce notation and keep the math simple, let's assume that  $\underline{m} = \underline{n}$ . I.e., we have exactly the same number of servers and requests.

**Hash function:** Continue to assume a fully (uniformly) random hash function *h*.

Often called the "balls-into-bins" model.

 $\mathbb{E}[s_i] = \text{expected number of balls per bin} = \frac{m}{n} = 1$ . We would like to prove a bound of the form:  $\mathcal{L} = \mathcal{H}$ 

$$\Pr[\max s_i \ge C] \le \frac{1}{10}.$$
for as tight a value of C. I.e., something much better than  $C = n$ .

44

#### **BOUNDING A UNION OF EVENTS**

**Goal:** Prove that for some *C*,  $\Pr[\max_{i \in I} s_i \ge C] \le \frac{1}{10}.$   $\cup \text{ means "or". } \cap \text{ means "and".}$  **Equivalent statement:** Prove that for some *C*,

 $\Pr[(\underline{s_1 \ge C}) \cup (\underline{s_2 \ge C}) \cup \dots \cup (\underline{s_n \ge C})] \le \frac{1}{10}.$ 

Need to bound the probability of a <u>union</u> of different events.

These events are not independent!!

n = number of balls and number of bins.  $s_i$  is number of balls in bin *i*. C = upper bound on maximum number of balls in any bin.

100

#### USE A UNION BOUND

# Lemma (Union Bound)

For <u>any</u> random events  $A_1, \ldots, A_k$ :

 $\Pr[A_1 \cup A_2 \cup \ldots \cup A_k] \leq \Pr[A_1] + \Pr[A_2] + \ldots + \Pr[A_k].$ 





#### NEW GOAL

Prove that for some C,

$$\Pr[s_i \ge C] \le \frac{1}{10n}.$$

This should look hard! We need to prove that  $s_i < C$  (i.e. the *i*<sup>th</sup> bin has a small number of balls) with very high probability (specifically  $1 - \frac{1}{10n}$ ).

Markov's inequality is too weak of a bound for this.

n = number of balls and number of bins.  $s_i$  is number of balls in bin *i*. C = upper bound on maximum number of balls in any bin.

**Goal:** Prove that  $\Pr[\underline{s_i} \ge C] \le \frac{1}{10n}$ .

• Step 1. To apply Chebyshev's inequality, we need to understand  $\sigma^2 = Var[s_i]$ .

Use <u>linearity of variance</u>. Let  $s_{i,j}$  be a {0, 1} indicator random variable for the event that ball *j* falls in bin *i*. We have:

$$(s_i) = \sum_{j=1}^n s_{i,j}$$

And  $s_{i,1}, \ldots, s_{i,n}$  are independent so:

$$Var[s_i] = \sum_{j=1}^{n} Var[s_{i,j}]$$

n = number of balls and number of bins.  $s_i$  is number of balls in bin i.  $\mathbb{E}[s_i] = 1$ . C = upper bound on max number of balls in bin.

#### VARIANCE ANALYSIS

So:

$$s_{i,j} = \begin{cases} 1 \text{ with probability } \frac{1}{n} \\ 0 \text{ otherwise.} \end{cases}$$

$$\begin{bmatrix} \mathbb{E}[s_{i,j}] = 1 \cdot \frac{1}{n} + 6 \cdot (1 \cdot \frac{1}{n}) = \frac{1}{n} \\ \mathbb{E}[s_{i,j}^2] = \frac{1}{n} \end{cases}$$

$$far[s_{i,j}] = \mathbb{E}[s_{i,j}^2] - \mathbb{E}[s_{i,j}]^2 = \frac{1}{n} - \frac{1}{n^2} \approx \frac{1}{n} \leq \frac{1}{n} \end{cases}$$

n = number of balls and number of bins.  $s_{i,j}$  is event ball j lands in bin i.

# APPLYING CHEBYSHEV'S

**Goal:** Prove that  $\Pr[s_i \ge C] \le \frac{1}{10n}$ . Step 1. To apply Chebyshev's inequality, we need to  $\operatorname{Mappi,}_{nd \sigma^{2} = \operatorname{Var}[s_{i}]}_{nd \sigma^{2} = \sum_{i=1}^{n} \operatorname{Var}[s_{i,i}]} = \sum_{j=1}^{n} \frac{1}{n} - \frac{1}{n^{2}} = 1 - \frac{1}{n} \underbrace{(1, 1)}_{\sigma^{2} = 1} = 1 - \frac{1}{n} \underbrace{(1, 1)$ understand  $\sigma^2 = \text{Var}[s_i]$ . Step 2. Apply Chebyshev's inequality:  $\Pr\left[|s_i - \mathbb{E}[s_i]| \ge k \cdot 1\right] \le \frac{\epsilon}{k^2} \quad \mathbf{J}$ which implies  $\Pr[|s_i - 1| \ge k \cdot 1] \le \frac{1}{k^2}$ .  $\Pr[S; \ge from] \le \sqrt{10}m$ 

n = number of balls and number of bins.  $s_i =$  number of balls in bin *i*.  $s_{i,j}$  is event ball *j* lands in bin *i*.  $\mathbb{E}[s_i] = 1$ .

### APPLYING CHEBYSHEV'S

**Goal:** Prove that  $\Pr[s_i \ge C] \le \frac{1}{10n}$ . We just proved:  $\Pr[|s_i - 1| \ge k] \le \frac{1}{k^2}$ . Setting  $k = \sqrt{10n}$  gives:  $\Pr[|s_i - 1| \ge \sqrt{10n}] \le \frac{1}{10n}$ .

So, we have that:

$$\Pr[s_i \ge \sqrt{10n} + 1] \le \frac{1}{10n}.$$

By the union bound argument from earlier, it thus holds that:

$$\Pr[\max_{i \in \{1,...,n\}} s_i \ge \sqrt{10n} + 1] \le \frac{1}{10}.$$

n = number of balls and number of bins.  $s_i$  is number of balls in bin *i*. C = upper bound on maximum number of balls in any bin.

O(14)

#### FINAL RESULT



When hashing *n* balls into *n* bins, the maximum bin contains  $\hat{O}(\sqrt{n})$  balls with probability  $\frac{9}{10}$ .



Much better than the trivial bound of *n*!

#### PROOF A UNION BOUND

Lemma (Union Bound)  
For any random events 
$$\underline{A_1, \ldots, \underline{A_k}}$$
:  
 $\Pr[\underline{A_1 \cup A_2 \cup \ldots \cup A_k}] \leq \Pr[A_1] + \Pr[A_2] + \ldots + \Pr[A_k]$ .

Let 
$$X_i = \mathbb{1}[A_i]$$
 and apply Markov's to  $S = \sum_{i=1}^{k} X_i$ .  
 $\mathbb{1}[A_i] \qquad S = \sum_{i=1}^{k} X_i$   
 $\mathbb{E}[S] = \sum_{i=1}^{k} \mathbb{E}[X_i] = \sum_{i=1}^{k} \Pr(A_i]$   
 $\Pr(A_i] \qquad \Pr(A_i] \qquad \Pr(A_i]$   
 $\Pr(S \ge 1] \le \mathbb{E}[S] = \sum_{i=1}^{k} \Pr(A_i]$   $\square$ 

54

# 0(54)

# Techniques used that will appear again:

- Union bound to control the <u>maximum</u> of many random variables.
- Chebyshev's inequality to bound a variable whose variance we can compute.
- To compute the variance, break down random variable into smaller pieces and apply linearity of variance.

But... For this problem we can actually use even stronger tools to prove a better bound of  $O(\log n)$  for the most loaded bin.

#### **BEYOND CHEBYSHEV**



68-95-99 rule for Gaussian bell-curve.  $X \sim N(0, \sigma^2)$ 

#### Chebyshev's Inequality:

#### Truth:

$$\begin{aligned} &\Pr\left(|X - \mathbb{E}[X]| \ge 1\sigma\right) \le 100\% \\ &\Pr\left(|X - \mathbb{E}[X]| \ge 2\sigma\right) \le 25\% \\ &\Pr\left(|X - \mathbb{E}[X]| \ge 3\sigma\right) \le 11\% \\ &\Pr\left(|X - \mathbb{E}[X]| \ge 4\sigma\right) \le 6\%. \end{aligned}$$

$$\begin{aligned} & \Pr\left(|X - \mathbb{E}[X]| \ge 1\sigma\right) \approx \underline{32\%} \\ & \Pr\left(|X - \mathbb{E}[X]| \ge 2\sigma\right) \approx \underline{5\%} \\ & \Pr\left(|X - \mathbb{E}[X]| \ge 3\sigma\right) \approx \underline{1\%} \\ & \Pr\left(|X - \mathbb{E}[X]| \ge 4\sigma\right) \approx .01\% \end{aligned}$$

#### GAUSSIAN CONCENTRATION

For 
$$X \sim \mathcal{N}(\mu, \sigma^2)$$
:  

$$\Pr[X = \mu \pm x] = \frac{1}{\sigma\sqrt{2\pi}} e^{-x^2/2\sigma^2}$$
Lemma (G&Assian Tail Bound)  
For  $X \sim \mathcal{N}(\mu, \sigma^2)$ :  

$$\Pr[|X - \mathbb{E}X| \ge k \cdot \sigma] \le 2e^{-k^2/2}$$

$$\int_{\mathbb{E}^{4}} e^{-k^2/2} (Gaussian tail)$$

$$\int_{\mathbb{E}^{4}} e^{-k^2/2} (Gaussian$$

57

# **Takeaway:** Gaussian random variables concentrate much tighter around their expectation than variance alone predicts.

Why does this matter for algorithm design?

#### **CENTRAL LIMIT THEOREM**

#### Theorem (CLT – Informal)

Any sum of mutually independent, (identically distributed) r.v.'s  $X_1, \ldots, X_k$  with mean  $\mu$  and finite variance  $\sigma^2$  converges to a Gaussian r.v. with mean  $k \cdot \mu$  and variance  $k \cdot \sigma^2$ , as  $k \to \infty$ .



(a) Distribution of # of heads after 10 coin flips, compared to a Gaussian.

(b) Distribution of # of heads after 50 coin flips, compared to a Gaussian.

Recall:

# Definition (Mutual Independence)

Random variables  $X_1, \ldots, X_k$  are <u>mutually independent</u> if, for all possible values  $v_1, \ldots, v_k$ ,

$$\Pr[X_1 = v_1, \dots, X_k = v_k] = \Pr[X_1 = v_1] \cdot \dots \cdot \Pr[X_k = v_k]$$

Strictly stronger than pairwise independence.

#### EXERCISE

793% E[...]= 50 Var[...]=25 (let gue If I flip a fair coin 100 times, lower bound the chance I get between ζ= ζ= μ → ν<sub>2</sub> 30 and 70 heads? 20 awy from 50 ρ<sub>1</sub> (|ζ. τ0 | τ 20] ≤ 1 × 6. τ. For this problem, we will assume the CLT holds exactly for a sum of independent random variables – i.e., that this sum looks exactly like a Gaussian random variable. Lemma (Guassian Tail Bound) For  $\mathfrak{S} \sim \mathcal{N}(\mu, \sigma^2)$ : S. 50 7, 4.5  $\leq 2e^{-\frac{4^2}{2}} = 2e^{-\frac{4^2}{2}}$  $\Pr[|\mathbf{S} - \mathbb{E}\mathbf{S}| \ge k \cdot \sigma] \le 2e^{-k^2/2}$ 

Exporential corcertations These back-of-the-envelop calculations can be made rigorous! Lots of different "versions" of bound which do so.

- Chernoff bound
   Bernstein bound
   Hoeffding bound

Different assumptions on random varibles (e.g. binary vs. bounded), different forms (additive vs. multiplicative error), etc. Wikipedia is your friend.

QUANTITATIVE VERSIONS OF THE CLT

$$V_{\text{br}}(s) : \sum_{i=1}^{k} V_{\text{or}}(X_{i}) \cong \sum_{i=1}^{k} \mathbb{E}(X_{i}^{*}] : \mu \sum_{i=1}^{k} \sum_{i=1}^{k} \mathbb{E}(X_{i}^{*}] : \mu \sum_{i=1}^{k} \sum_{i=1}^{k} \mathbb{E}(X_{i}) = \mathbb{E}[X_{i}]$$

$$Let X_{1}, X_{2}, \dots, X_{k} be independent \{0, 1\} - valued random \quad 6 \neq \mathbb{I}_{k}$$

$$variables and let p_{i} = \mathbb{E}[X_{i}], where \ 0 \leqslant p_{i} \leqslant 1. \text{ Then the sum}$$

$$S = \sum_{i=1}^{k} X_{i}, \text{ which has mean } \mu = \sum_{i=1}^{k} p_{i}, \text{ satisfies}$$

$$X_{i} = \sum_{i=1}^{k} \mathbb{E}(X_{i})^{*} \qquad (\Pr[S \ge (1+\epsilon)\mu] \le e^{\frac{-\epsilon^{2}\mu}{2+\epsilon}})$$

$$= \mathbb{E}[X_{i}]^{*} \quad (\Pr[S \le (1-\epsilon)\mu] \le e^{\frac{-\epsilon^{2}\mu}{2+\epsilon}}) \le e^{-\frac{a^{2}\mu}{2+\epsilon}})$$

$$= 2e^{-\frac{a^{2}\mu}{2}} \quad (\Pr[S \le (1-\epsilon)\mu] \le e^{\frac{-\epsilon^{2}\mu}{2}}) = 2e^{-\frac{a^{2}\mu}{2}}$$

$$= 2e^{-\frac{a^{2}\mu}{2}}$$

# Theorem (Bernstein Inequality)

Let  $X_1, X_2, ..., X_k$  be independent random variables with each  $X_i \in [-1, 1]$ . Let  $\mu_i = \mathbb{E}[X_i]$  and  $\sigma_i^2 = \operatorname{Var}[X_i]$ . Let  $\mu = \sum_i \mu_i$  and  $\sigma^2 = \sum_i \sigma_i^2$ . Then, for  $k \leq \frac{1}{2}\sigma_i$ ,  $S = \sum_i X_i$  satisfies  $\Pr[|S - \mu| > k \sigma] \leq 2 \exp\left(-\frac{k^2}{4}\right).$ 

#### QUANTITATIVE VERSIONS OF THE CLT

# Theorem (Hoeffding Inequality)

Let  $X_1, X_2, ..., X_k$  be independent random variables with each  $X_i \in [\underline{a_i, b_i}]$ . Let  $\mu_i = \mathbb{E}[X_i]$  and  $\mu = \sum_i \mu_i$ . Then, for any  $\alpha > 0$ ,  $S = \sum_i X_i$  satisfies:

{o, 13

$$\Pr[|\underline{S-\mu}| > \underline{\alpha}] \le 2 \exp\left(-\frac{\alpha^2}{\sum_{i=1}^k (b_i - a_i)^2}\right)$$

#### HOW ARE THESE BOUNDS PROVEN?

Variance is a natural measure of central tendency, but there [[((x- Ex)]]] are others.

 $q^{\text{th}}$  central moment:  $\mathbb{E}[(X - \mathbb{E}X)^q]$  k = 2 gives the variance. Proof of Chebyshev's applies Markov's sources inequality to the random variable  $(X - \mathbb{E}X)^2$ ).

172

Idea in brief: Apply Markov's inequality to  $\mathbb{E}[(X - \mathbb{E}X)^q$  for larger q, or more generally to  $f(X - \mathbb{E}X)$  for some other non-negative function f. E.g., to  $\underline{\exp(X - \mathbb{E}X)}$ .

We will explore this approach in the problem set.

#### CHERNOFF BOUND APPLICATION

Sample Application: Flip biased coin k times: i.e. the coin is heads with probability (b) As long as  $k \ge O\left(\frac{\log(1/\delta)}{\epsilon^2}\right)$  $k = k \circ f = P^2$   $\Pr[|\# \text{ heads } -\underline{b \cdot k}| \ge \epsilon k] \le \delta$   $\mathcal{O}\left(\frac{1}{\epsilon^2 \cdot \delta}\right)$ 

**Setup:** Let  $X_i = \mathbb{1}[i^{\text{th}} \text{ flip is heads}]$ . Want bound probability that  $\sum_{i=1}^{k} X_i$  deviates from it's expectation.

Corollary of Chernoff bound: Let  $S = \sum_{i=1}^{k} X_i$  and  $\mu = \mathbb{E}[S]$ . For  $0 < \epsilon < 1$ ,  $\Pr[|S - \mu| \ge \epsilon \mu] \le 2e^{-\epsilon^2 \mu/3}$   $|\# \underset{K}{\text{heods}} - \frac{1}{5} \cdot \frac{1}{5}| \le \frac{\epsilon}{5}$   $|\# \underset{K}{\text{heods}} - \frac{1}{5}| \le \frac{\epsilon}{5}$  $|\# \underset{K}{\text{heods}} - \frac{1}{5}| \le \frac{\epsilon}{5}$  **Sample Application:** Flip biased coin k times: i.e. the coin is heads with probability b. As long as  $k \ge O\left(\frac{\log(1/\delta)}{\epsilon^2}\right)$ ,

$$\Pr[|\# \text{ heads} - b \cdot k| \ge \epsilon k] \le \delta$$

Pay very little for higher probability – if you increase the number of coin flips by 2x,  $\delta$  goes from  $1/10 \rightarrow 1/100 - 1/10000$ 

#### LOAD BALANCING

We are going to see a more interesting application of exponential concentration bounds to the randomized load balancing problem. *n* jobs are distributed randomly to *n* servers using a hash function. Let  $S_i$  be the number of jobs sent to server *i*. What's the smallest **B** for which we can prove:  $Pr[max_iS_i \ge B] \le 1/10$ 

**Recall:** Suffices to prove that, for any *i*,  $\Pr[S_i \ge B] \le \frac{1}{10}$ 

$$\Pr[\underline{max_iS_i \ge B}] = \Pr[\underline{r[S_1 \ge B \text{ or } \dots \text{ or } S_n \ge B}]$$

$$\leq \Pr[\underline{S_1 \ge B}] + \dots + \Pr[\underline{S_n \ge B}] \quad (\text{union bound}).$$

$$\varphi \not \leq \gamma_{l_{D_n}}$$

#### LOAD BALANCING

For N7,10, THE TOM Theorem (Chernoff Bound) Let  $X_1, X_2, \ldots, X_n$  be independent  $\{0, 1\}$ -valued random variables and let  $p_j = \mathbb{E}[X_j]$ , where  $0 < p_j < 1$ . Then the sum  $S = \sum_{i=1}^{n} X_i$ , which has mean  $\mu = \sum_{i=1}^{n} p_i$ , satisfies 5.7:2 [(Si)=1=1 1082 (X) 1080 (X)  $\Pr[X \ge (1+\epsilon)\mu] \le e^{\frac{1}{2+\epsilon}}$ ~ Y = 10 g ((x) . 10 2 (e) (0) (x) S; 5;- 5x; Consider a single bin. Let  $X_j = \mathbb{1}[\text{ball } j \text{ lands in that bin}]$ .  $\mathbb{E}[X_i] = \frac{1}{2}$ , so  $\mu = 1$ .  $\mathbb{E}[\mathsf{C}_{S_i}] = \frac{1}{2} \mathbb{E}[\mathsf{C}_{S_i}] = \frac{1}{2} \mathbb{E}[\mathsf{C}_{S_i}$  $\mathbb{E}[X_i] = \frac{1}{n}$ , so  $\mu = 1$ .  $\Pr[S \ge (1 + c \log n)\mu] \le e^{\frac{-c^2 \log^2 n}{2 + c \log n}} \le e^{\frac{-c \log^2 n}{2 \log n}} \le e^{-.5c \log n} < e$ for sufficiently large c 2+ clos n = 2 clos (4) 2103 (4) = 13(2) E= closin e-log(n2) 1 70

So max load for randomized load balancing is  $O(\log n)!$  Best we could prove with Chebyshev's was  $O(\sqrt{n})$ .

**Power of 2 Choices:** Instead of assigning job to random server, choose 2 random servers and assign to the least loaded. With probability 1/10 the maximum load is bounded by:

(a)  $O(\log n)$ (b)  $O(\sqrt{\log n})$ (c)  $O(\log \log n)$ (d) O(1)