Randomized algorithms 3
Chernoff Bounds

CS240 Spring 2021

Rui Fan

Beating expectations

- Given a random variable X, the expected value E[X] is the value we're "supposed" to get when we take a sample of X.
- Of course, we won't always get E[X].
- But what is the probability the sample is very different from E[X]?
- Ex If we flip a fair coin 100 times, we expect to get 50 heads. What's the probability we get 60 heads? 80 heads? 100 heads?

Beating expectations

- Markov's Inequality Given a positive random variable X, $\Pr[X \geq a] \leq E[X]/a$ for any $a > 0$.
- Ex X = no. heads in 100 flips. $\Pr[X \geq 60] \leq 50/60 = 5/6$.
- Proof Suppose $\Pr[X \geq a] > E[X]/a$. By definition, we have

$$E[X] = \sum_{x} x \cdot \Pr[X = x] \geq \sum_{x \geq a} x \cdot \Pr[X = x] \geq$$

$a \cdot \Pr[X \geq a] > a \cdot E[X]/a = E[X]$, contradiction.

- Markov's inequality is weak.
  - Using the previous example, it states $\Pr[X \geq 101] \leq 50/101 \approx 0.495$, which is quite obvious!

Beating expectations

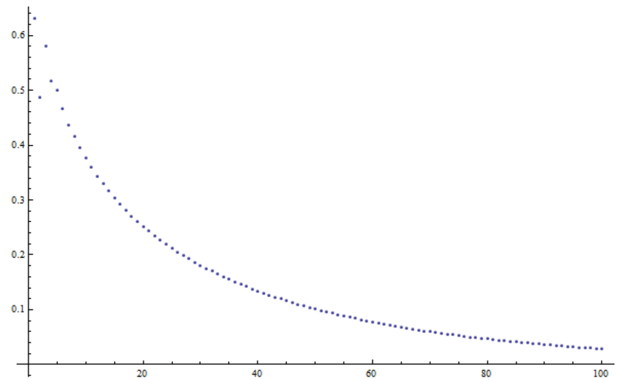
- But Markov's inequality is general.
  - X can be any positive random variable. It doesn't need to satisfy any special properties, as for some other inequalities.
  - Under some circumstances it can be used to prove stronger statements.
- Chebychev's Inequality Given a random variable X and any $a > 0$, we have $\Pr[|X - E[X]| \geq a] \leq Var[X]/a^2$.
- Proof $Var[X] = E[(X - E[X])^2]$, so by Markov's inequality,

$\Pr[|X - E[X]| \geq a] = \Pr[(X - E[X])^2 \geq a^2] \leq Var[X]/a^2$.

- Ex Let X=no. heads in 100 flips. Var[X]=100/4. So $\Pr[|X - 50| \geq 10] \leq 25/100 = 1/4$ by Chebychev.
  - Since X is symmetric about 50, $\Pr[X \geq 60] \leq 1/8$, which is much better than Markov's inequality.

Sum of independent r.v.'s

- We'll consider a special kind of random variable X that is the sum of n independent random variables, for some n. We look at how likely X is to deviate from its expectation.
- Intuitively, as n gets larger, X should approach E[X], in relative terms, very quickly.
  - In fact, the convergence is exponential.
- Consider flipping a fair coin n times. What's the probability we get at least 60% heads?
  - For n=10, the probability is 37.7%.
  - For n=20, the probability is 25.2%.
  - For n=30, the probability is 18.1%.
  - For n=40, the probability is 13.4%.
  - For n=100, the probability is 2.84%.



Chernoff bounds

- Thm 1 Let $X_1, \ldots, X_n$ be independent random variables with values in {0,1}, s.t. $\Pr[X_i = 1] = p_i$ for all i. Let $X = \sum_i X_i$ and $\mu = E[X] = \sum_i p_i$. Then

  - For $0 < \delta \leq 1$, $\mathbf{Pr}[X \geq (1+\delta)\mu] \leq e^{-\mu\delta^2/3}$.
  - For $\delta > 1$, $\mathbf{Pr}[X \geq (1+\delta)\mu] \leq e^{-\mu\delta \ln \delta/3}$.
  - For $0 \leq \delta \leq 1$, $\mathbf{Pr}[X \leq (1-\delta)\mu] \leq e^{-\mu\delta^2/2}$.

- Chernoff bounds say the probability the sum of a set of independent {0,1} valued random variables is more than $1 \pm \delta$ times its expectation μ decreases exponentially in δ and μ.

Chernoff bounds

- Chernoff bounds are very useful in the analysis of randomized algorithms.
  - Often the algorithm does things in independent stages.
  - It's often easy to compute the expected cost of each stage.
  - The total cost is the sum of the cost of all the stages.
  - We use Chernoff bounds to show this is unlikely to be too big or small compared to its expectation.
- There are some variants of Chernoff's bound that differ in the precise bounds. Some giver tighter bounds for certain values of δ. Pick the best one to use for the situation.

Chernoff bounds

- Thm 2 Let X, $X_1,\ldots,X_n$ be defined as earlier. Then for any δ>0 we have

$$\Pr[X \geq (1+\delta)\mu] \leq \left(\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right)^{\mu}$$

- Thm 3 Let $X_1,\ldots,X_n$ be independent {-1,1} valued random variables, with $\Pr[X_i=1]=\Pr[X_i=-1]=1/2$ for all i. Let $X = \sum_i X_i$. Then for any $\delta \geq 0$, $\Pr[X \geq \delta] = \Pr[X \leq -\delta] \leq e^{-\frac{\delta^2}{2n}}$.

Proof of (simplified) Thm 2

The generic bound

The generic Chernoff bound for a random variable X is attained by applying Markov's inequality to e^{tX}.[2] For every $t > 0$:

$$\Pr(X \geq a) = \Pr(e^{t \cdot X} \geq e^{t \cdot a}) \leq \frac{\mathrm{E}\left[e^{t \cdot X}\right]}{e^{t \cdot a}}.$$

When X is the sum of n random variables $X_1, \ldots, X_n$, we get for any $t > 0$,

$$\Pr(X \geq a) \leq e^{-ta} \mathrm{E}\left[\prod_i e^{t \cdot X_i}\right].$$

In particular, optimizing over t and assuming that X_i are independent, we obtain,

$$\Pr(X \geq a) \leq \min_{t>0} e^{-ta} \prod_i \mathrm{E}\left[e^{tX_i}\right]. \qquad (1)$$

Let $X_1, \ldots, X_n$ be independent Bernoulli random variables, whose sum is X, each having probability $p > 1/2$ of being equal to 1. For a Bernoulli variable:

$$\mathrm{E}\left[e^{t \cdot X_i}\right] = (1-p)e^0 + pe^t = 1 + p(e^t - 1) \leq e^{p(e^t - 1)}$$

So:

$$\mathrm{E}\left[e^{t \cdot X}\right] \leq e^{n \cdot p(e^t - 1)}$$

For any $\delta > 0$, taking $t = \ln(1+\delta) > 0$ and $a = (1+\delta)np$ gives:

$$\mathrm{E}\left[e^{t \cdot X}\right] \leq e^{\delta np} \text{ and } e^{-ta} = \frac{1}{(1+\delta)^{(1+\delta)np}}$$

and the generic Chernoff bound gives:[3]:64

$$\Pr[X \geq (1+\delta)np] \leq \frac{e^{\delta np}}{(1+\delta)^{(1+\delta)np}} = \left[\frac{e^\delta}{(1+\delta)^{1+\delta}}\right]^{np}$$

Source: https://en.wikipedia.org/wiki/Chernoff_bound

Coin flipping

- Consider again flipping n coins. Let X be the number of heads we get in n flips. What's the probability $X \geq \frac{n}{2} + \lambda$ heads, for a given λ?
- The coin flips are all independent, so using part 1 of Theorem 1, we have

$$\mathbf{Pr}[X \geq \mu + \lambda] = \mathbf{Pr}\left[X \geq \mu(1 + \frac{\lambda}{\mu})\right] \leq e^{-(\frac{\lambda}{\mu})^2 \frac{\mu}{3}} = e^{-\frac{\lambda^2}{3\mu}}$$

- We have $\mu = \frac{n}{2}$. Suppose $\lambda = O(\sqrt{n})$, so that $\lambda^2 = O(n)$. Then

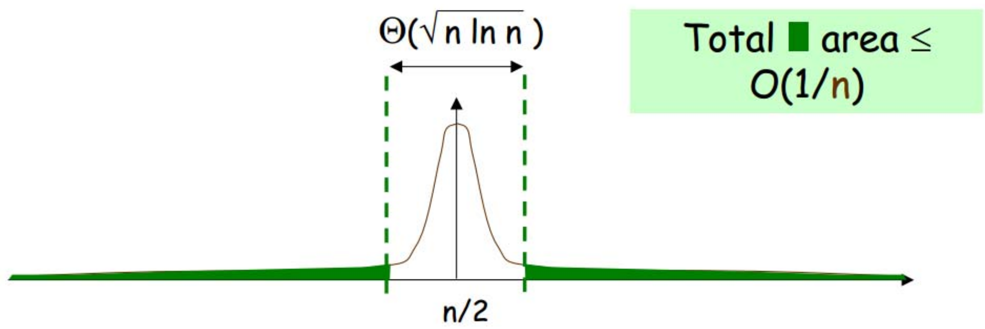
$$\mathbf{Pr}[X \geq \mu + \lambda] \leq e^{-O(1)} = O(1)$$

- Next, suppose $\lambda = O(\sqrt{n \log n})$. Then $-\frac{\lambda^2}{3\mu} = -O(\log n)$, and so

$$\mathbf{Pr}[X \geq \mu + \lambda] \leq e^{-O(\log n)} = \frac{1}{n^c}$$

  for some constant c.

- We say an event E occurs *with high probability (w.h.p.)* if $\mathbf{Pr}[E] \geq 1 - n^{-c}$ for some constant $c > 1$.
- So we have $X < \frac{n}{2} + O(\sqrt{n \log n})$ with high probability. I.e. we get at most $O(\sqrt{n \log n})$ more heads than expected, w.h.p.



Load balancing

- Suppose we have n computers. A set of equal sized jobs arrive online. We need to assign each to a computer for processing.
- To make all jobs finish fast, we want to give all computers (almost) same number of jobs. I.e. we want to balance their load.
- A simple way is to assign jobs round-robin. Keep giving next job to next computer, wrapping around if necessary.
- But this requires communicating with a centralized controller, which can be a bottleneck for large n.
- Instead, we do randomized load balancing.

- Algorithm Given a new job, assign it to a random computer.

Load balancing

- How well does this balance the load?
- If there are m jobs, then in expectation every computer gets m/n jobs.
- Let X_i be the number of jobs computer i gets, and let $X=\max_i \{X_i\}$ be max number of jobs any computer gets.
  - We'll bound probability that X_i or X are too large compared to the expectation m/n.
  - This shows the load is roughly balanced with high probability.
- Let Y_{ij} be a random variable that's 1 if the j'th job is assigned to computer i, and 0 otherwise.
  - $\Pr[Y_{ij}=1]=1/n$, since the jobs are assigned randomly.
  - $X_i=\sum_j Y_{ij}$ is the number of jobs i gets.
  - X_i is the sum of independent random variables Y_{ij}, so we can apply the Chernoff bound to X_i.

Analysis

- First consider the case $m = n$.
- We have $\mu = \mathbf{E}(X_i) = \mathbf{E}[\sum_{j=1}^{m} Y_{ij}] = \sum_{j=1}^{m} \mathbf{E}[Y_{ij}] = m\frac{1}{n} = 1$ for every i.
- We'll show $\mathbf{Pr}[X > O(\frac{\ln n}{\ln \ln n})] < \frac{1}{n}$.
  - So with high probability ($> 1 - 1/n$), every computer gets at most $O(\frac{\ln n}{\ln \ln n})$ times its expected number (=1) of jobs.
- Let's focus on X_i for some particular i. Let $\delta = O\left(\frac{\ln n}{\ln \ln n}\right) > 1$.
- By part 2 of Theorem 1, $\mathbf{Pr}[X_i > (1+\delta)\mu] \leq e^{-\delta \ln \delta / 3}$, since $\mu = 1$.
  - So $e^{-\delta \ln \delta / 3} = e^{O(\ln n)} \leq \frac{1}{n^2}$ for some choice of the constant in the big-O.
- We have $\mathbf{Pr}[X_i > O(\frac{\ln n}{\ln \ln n})] \leq \frac{1}{n^2}$ for every i. Hence by the union bound $\mathbf{Pr}[X_i > O(\frac{\ln n}{\ln \ln n})$ for any $i] \leq n\frac{1}{n^2} = \frac{1}{n}$.
- So $\mathbf{Pr}[X = \max_i X_i > O(\frac{\ln n}{\ln \ln n})] < \frac{1}{n}$.

Analysis

- We see that when there are n jobs for n computers, the load can be quite unbalanced.
- Suppose now we have more jobs, $m = 16n \ln n$. Then $\mu = 16 \ln n$.
- By Theorem 2, $\mathbf{Pr}[X_i > 2\mu] < \left(\frac{e}{4}\right)^{16 \ln n} < \left(\frac{1}{e^2}\right)^{\ln n} = \frac{1}{n^2}$.
- Thus, by the union bound $\mathbf{Pr}[\max_i X_i > 2\mu] \leq n\frac{1}{n^2} = \frac{1}{n}$.
- By part 3 of Theorem 1, $\mathbf{Pr}[X_i < \frac{1}{2}\mu] \leq e^{-(\frac{1}{2})^2 16 \ln n/2} = e^{-2 \ln n} = \frac{1}{n^2}$.
- So by the union bound, $\mathbf{Pr}[\min_i X_i < \frac{1}{2}\mu] \leq \frac{1}{n}$.
- Thus, we have that with high probability, every computer has between half and twice the average load.
- These results hold for any $m > 16n \ln n$. So the more jobs there are, the better the load balancing random allocation achieves. This is similar to the phenomenon where the more coins we flip, the more likely we are to get the expected number of heads (in relative terms).

Set balancing

- Suppose we have m sets, each a subset of {1,2,...,n}.
  - We can represent each set as an n-bit vector.
  - Ex If n=4 and S={1,3,4}, we can represent it as [1,0,1,1].
- We want to divide the sets into two groups, such that the sums of the sets in the groups are roughly equal.
- Ex S_1=[1,0,1,1], S_2=[1,1,1,0], S_3=[0,0,1,1], S_4=[0,1,1,0]. Then S_1+S_4=[1,1,2,1]=S_2+S_3.
- And exact balancing might not exist. We look for one that's as good as possible.
- Let G, G' denote the two groups. We want to minimize the max imbalance $|\sum_{i\in G} S_i - \sum_{i \in G'} S_i|_\infty$.
  - Here $|v|_\infty$ denotes the L_∞ norm, and is equal to the max component of v in absolute value. E.g. $|[1,2,-3,1]|_\infty = 3$.
  - Ex S_1=[1,1,0,1], S_2=[1,0,0,1], S_3=[1,0,1,0], S_4=[1,1,0,0].
  - There's no exact balancing, but S_1+S_3=[2,1,1,1] and S_2+S_4=[2,1,0,1], so the max imbalance is $|[2,1,1,1] - [2,1,0,1]|_\infty = |[0,0,1,0,]|_\infty$.

Set balancing

- Finding the grouping that minimizes the max imbalance is hard. Brute force would try 2^m possible groupings.
- We give a randomized algorithm that ensures the max imbalance is $\sqrt{(4m \ln n)}$ w.h.p.
  - Max possible imbalance is m.

- Algorithm Assign each set to a random group.

Analysis

- Consider an item i. Suppose i belongs to k different sets.
- For the j'th such set S, define X_j=1 if S is in the first group, and X_j=-1 if it's in the other group.
- The imbalance from item i is simply $B_i=\Sigma^k_{j=1} X_j$.
- Since sets are assigned randomly, $X_1,\ldots,X_k$ are independent {1,-1} valued r.v.'s, we can apply Thm 3 to bound $\max_i B_i$.

- If $k \le \sqrt{4m \ln n}$, then $B_i \le \sqrt{4m \ln n}$.
- If $k > \sqrt{4m \ln n}$, then by Theorem 3 we have

$$\begin{aligned}
\mathbf{Pr}[B_i > \sqrt{4m \ln n}] &\le e^{-(\sqrt{4m \ln n})^2/(2k)} \\
&= e^{-4m \ln n/(2k)} \\
&\le e^{-4m \ln n/(2m)} \\
&= e^{-2 \ln n} \\
&= \frac{1}{n^2}
\end{aligned}$$

- We showed the probability item i's imbalance is $\ge \sqrt{4m \ln n}$ is $\le \frac{1}{n^2}$. By the union bound, the probability that any of the n items' imbalance is $\ge \sqrt{4m \ln n}$ is $\le \frac{1}{n}$. So w.h.p., the max imbalance is $\le \sqrt{4m \ln n}$.