Announcements

- No class monday.
- Metric embedding seminar.

Review

- expectation
- notion of high probability.
- Markov.

Today: Book 4.1, 3.3, 4.2

Chebyshev.

- Remind variance, standard deviation. $\sigma_X^2 = E[(X - \mu_X)^2]$
- $E[XY] = E[X]E[Y]$ if independent
- variance of independent variables: sum of variances
- $\Pr[|X - \mu| \geq t\sigma] = \Pr[(X - \mu)^2 \geq t^2\sigma^2] \leq 1/t^2$
- So chebyshev predicts won't stray beyond stdev.
- binomial distribution. variance $np(1 - p)$. stdev $\sqrt{n}$.
- requires (only) a mean and variance. less applicable but more powerful than markov
- Balls in bins: err $1/\ln^2 n$.
- Real applications later.

Chernoff Bound

Intro

- Markov: $\Pr[f(X) > z] < E[f(X)]/z$.
- Chebyshev used X^2 in f
- other functions yield other bounds
- Chernoff most popular

Theorem:

- Let X_i poisson (ie independent 0/1) trials, $E[\sum X_i] = \mu$

$$\Pr[X > (1+\epsilon)\mu] < \left[\frac{e^\epsilon}{(1+\epsilon)^{(1+\epsilon)}}\right]^\mu .$$

- note independent of n, exponential in μ.

Proof.

- For any $t > 0$,

$$\begin{aligned}\Pr[X > (1+\epsilon)\mu] &= \Pr[\exp(tX) > \exp(t(1+\epsilon)\mu)] \\ &< \frac{E[\exp(tX)]}{\exp(t(1+\epsilon)\mu)}\end{aligned}$$

- Use independence.

$$\begin{aligned}E[\exp(tX)] &= \prod E[\exp(tX_i)] \\ E[\exp(tX_i)] &= p_i e^t + (1-p_i) \\ &= 1 + p_i(e^t - 1) \\ &\le \exp(p_i(e^t-1))\end{aligned}$$

  $\prod \exp(p_i(e^t-1)) = \exp(\mu(e^t-1))$

- So overall bound is

$$\frac{\exp((e^t-1)\mu)}{\exp(t(1+\epsilon)\mu)}$$

  True for any t. To minimize, plug in $t = \ln(1+\epsilon)$.

- Simpler bounds:
  - less than $e^{-\mu\epsilon^2/3}$ for $\epsilon < 1$
  - less than $e^{-\mu\epsilon^2/4}$ for $\epsilon < 2e - 1$.
  - Less than $2^{-(1+\epsilon)\mu}$ for larger ϵ.
- By same argument on $\exp(-tX)$,

$$\Pr[X < (1-\epsilon)\mu] < \left[\frac{e^{-\epsilon}}{(1-\epsilon)^{(1-\epsilon)}}\right]^{\mu}$$

  bound by $e^{-\epsilon^2/2}$.

Basic application:

- $cn \log n$ balls in c bins.
- max matches average
- a fortiori for n balss in n bins

General observations:

- Bound trails off when $\epsilon \approx 1/\sqrt{\mu}$, ie absolute error $\sqrt{\mu}$
- no surprise, since standard deviation is around μ (recall chebyshev)
- If $\mu = \Omega(\log n)$, probability of constant ϵ deviation is $O(1/n)$, Useful if polynomial number of events.
- Note similarito to Gaussian distribution.
- **Generalizes:** bound applies to any vars distributed in range $[0, 1]$.

Zillions of Chernoff applications.

Median finding.

First main application of Chernoff: **Random Sampling**

- List L
- median of sample looks like median of whole. neighborhood.
- analysis via Chernoff bound
- Algorithm
  - choose s samples *with replacement*
  - take fences before and after sample median
  - keep items between fences. sort.
- Analysis
  - claim (i) median within fences and (ii) few items between fences.
  - Without loss of generality, L contains $1, \ldots, n$. (ok for comparison based algorithm)
  - Samples $s_1, \ldots, s_m$ in sorted order.
  - lemma: S_r near rn/s.
    * Expected number preceding k is ks/n.
    * Chernoff: w.h.p., $\forall k$, number elements before k is $(1 \pm \epsilon_k)ks/n$, where $\epsilon_k = \sqrt{(6n \ln n)/ks}$.
    * Thus, when $k > n/4$, have $\epsilon_k \leq \epsilon = \sqrt{24 \ln n/s})$
    * Write $\epsilon = \sqrt{24 \ln n/s}$.
    * $S_{(1+\epsilon)ks/n} > k$
    * $S_r > rn/s(1 + \epsilon)$
    * $S_r < rn/s(1 - \epsilon)$.
  - Let $r_0 = \frac{s}{2}(1 - \epsilon)$
  - Then w.h.p., $\frac{n}{2}(1 - \epsilon)/(1 + \epsilon) < S_{r_0} < n/2$
  - Let $r_1 = \frac{s}{2}(1 - \epsilon)$
  - Then $S_{r_1} > n/2$

  - But $S_{r_1} - S_{r_0} = O(\epsilon n)$

- Number of elements to sort: s
- Set containing median: $O(\epsilon n) = O(n\sqrt{(\log n)/s})$.
- balance: $O(\log(n^{2/3}))$ in both steps.

Randomized is strictly better:

- Gives important constant factor improvement
- Optimum deterministic: $\geq (2+\epsilon)n$
- Optimum randomized: $\leq (3/2)n + o(n)$

Book analysis slightly different.

Routing

Second main application of Chernoff: analysis of load balancing.

- Already saw balls in bins example
- synchronous message passing
- bidirectional links, one message per step
- queues on links
- **permutation** routing
- **oblivious algorithms** only consider self packet.
- **Theorem** Any deterministic oblivious permutation routing requires $\Omega(\sqrt{N/d})$ steps on an N node degree d machine.
  - reason: some edge has lots of paths through it.
  - **homework:** special case
- Hypercube.
  - N nodes, $n = \log_2 N$ dimensions

- Nn directed edges
- bit representation
- natural routing: bit fixing (left to right)
- paths of length n—lower bound on routing time
- Nn edges for N length n paths suggest no congestion bound
- but deterministic bound $\Omega(\sqrt{N/n})$

- Randomized routing algorithm:
  - $O(n) = O(\log N)$ randomized
  - how? load balance paths.
- First idea: random destination (not permutation!), bit correction
  - Average case, but a good start.
  - $T(e_i)$ = number of paths using e_i
  - by symmetry, all $E[T(e_i)]$ equal
  - expected path length $n/2$
  - LOE: expected total path length $Nn/2$
  - nN edges in hypercube
  - Deduce $E[T(e_i)] = 1/2$
  - Chernoff: every edge gets $\leq 3n$ (prob $1 - 1/N$)
- Naive usage:
  - n phases, one per bit
  - $3n$ time per phase
  - $O(n^2)$ total
- Worst case destinations
  - Idea [Valiant-Brebner] From intermediate destination, route back!
  - routes **any** permutation in $O(n^2)$ expected time.
  - what's going in with $\sqrt{N/n}$ lower bound?

  - Adversary doesn't know our routing so cannot plan worst permutation

- What if don't wait for next phase?
  - FIFO queuing
  - total time is length plus **delay**
  - Expected delay $\leq E[\sum T(e_l)] = n/2$.
  - Chernoff bound? no. dependence of $T(e_i)$.

- High prob. bound:
  - consider paths sharing i's fixed route $(e_0, \ldots, e_k)$
  - Suppose S packets intersect route (use at least one of e_r)
  - claim delay $\leq |S|$
  - Suppose true, and let $H_{ij} = 1$ if j hits i's (fixed) route.

$$\begin{aligned} E[\text{delay}] &\leq E[\sum H_{ij}] \\ &\leq E[\sum T(e_l)] \\ &\leq n/2 \end{aligned}$$

  - Now Chernoff **does** apply (H_{ij} independent for fixed i-route).
  - $|S| = O(n)$ w.p. $1 - 2^{-5n}$, so $O(n)$ delay for all 2^n paths.

- Lag argument
  - Exercise: once packets separate, don't rejoin
  - Route for i is $\rho_i = (e_1, \ldots, e_k)$
  - charge each delay to a departure of a packet from ρ_i.
  - Packet waiting to follow e_j at time t has: **Lag** $t - j$
  - Delay of i is lag crossing e_k
  - When i delay rises to $l + 1$, some packet from S has lag l (since crosses e_j instead of i).
  - Consider last time t' where a lag-l packet exists on path

    * some lag-l packet w crosses $e_{j'}$ at t' (others increase to lag-$(l+1)$)
    * w leaves at this point (if not, then l at $e_{j'+1}$ next time)
    * charge one delay to w.

Summary:

- 2 key roles for chernoff
- sampling
- load balancing
- "high probability" results at $\log n$ means.