Tutorial 7

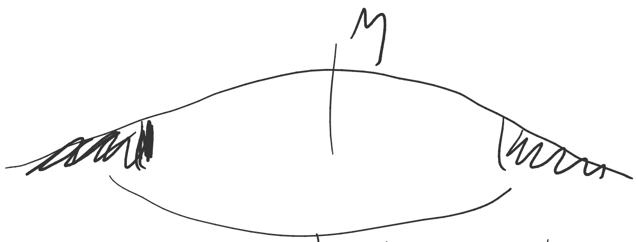
15 April 2021 16:02

Concentration bounds

- Chernoff bounds applications
  - Polling problem
  - Algorithm to estimate π

- Intro to routing on hypergraphs

Chernoff bounds



bound that X is far from mean

$$X = \sum_{i=1}^{n} X_i$$

X_i - Poisson r.v. (taking values 0 and 1)

with $Pr(X_i = 1) = p$

$E(X_i) = p$

$E(X) = n.p = \mu$ X_i are i.i.d. (identically independently distributed)

$$
\boxed{\left.\begin{aligned}
&\Pr\left(X \le (1-\delta)\cdot\mu\right) \le e^{-\frac{\mu\delta^2}{2}} \le e^{-\frac{\mu\delta^2}{3}} \\
&\Pr\left(X \ge (1+\delta)\cdot\mu\right) \le e^{-\frac{\mu\delta^2}{3}} \\
&\Pr\left(|X-\mu| \ge \delta\mu\right) \le 2e^{-\frac{\mu\delta^2}{3}}
\end{aligned}\right\}\quad 0 \le \delta \le 1}
$$

Polling problem

Two presindtial candidates A and B.

We want to estimate the number of people who will vote for A. Let's assume everyone has decided and will vote

Let's define answer of i^{th} person with r.v. X_i

Such that $\quad X_i = 1 \quad$ (i^{th} person votes for A)

$\qquad\qquad\quad\ \ X_i = 0 \quad$ (i^{th} person votes for B)

$\Pr(X_i = 1) = p \qquad$ [this is percentage of people voting for A = the number we want to estimate]

$X = \frac{X_1 + \dots + X_n}{\;}$

After asking n people, the estimate $X = \frac{x_1 + \dots + x_n}{n}$

$$\Pr\left(|X - p| \le [0.1]\cdot p\right) > 90\%$$

$$\Pr\left(|X - p| \ge \frac{p}{10}\right) \le 10\% \qquad nX = X_1 + \dots + X_n$$

Chernoff bound can be used

$$\Pr\left(|\underline{nX} - \underline{np}| \ge \frac{np}{10}\right) \le \boxed{10\%}$$

$$E(nX) = n\,E(X) = n.p$$

$$\delta = \frac{1}{10}$$

$$\Pr\left(|nX - \overset{\mu}{np}| \ge \frac{np}{10}\right) \overset{ch.b}{<} \boxed{2\cdot e^{-\frac{\overset{\mu}{n.p}}{100\cdot 3}}}$$

$$2\cdot e^{-\frac{n.p}{100\cdot 3}} \le \frac{1}{10}$$

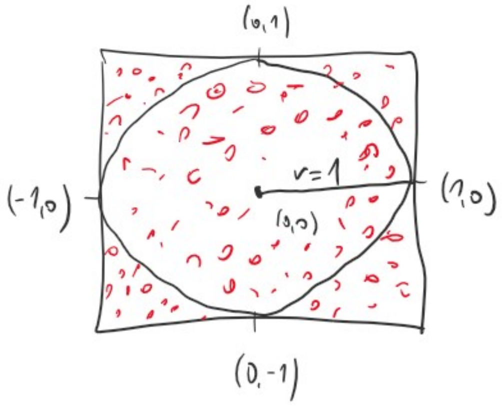
$$e^{-\frac{n.p}{300}} \le \frac{1}{20} \qquad /\ln$$

$$-\frac{n.p}{300} \le \ln\left(\frac{1}{20}\right)$$

$$n \ge -\frac{300}{p}\cdot \ln\left(\frac{1}{20}\right)$$

$$n \ge 900/p\,.$$

Algorithm to estimate $\pi \approx 3.1415\ldots$



$Z_i = 1$ if i^{th} point is inside the circle
$Z_i = 0$ if i^{th} point is outside of the circle

$$\Pr(Z_i = 1) = \frac{V(\text{circle})}{V(\text{square})} = \frac{\pi r^2}{(2r)^2} = \frac{\pi}{4}$$

After n trials we have

$$Z = \sum_{i=1}^{n} Z_i$$

$$E(Z) = \sum_{i=1}^{n} E(Z_i) = n \cdot \frac{\pi}{4}$$

$Z' = \frac{4 \cdot Z}{n}$ is our estimate of π $\quad E(Z') = \pi$

$$\Pr\left(|Z' - \pi| \le \varepsilon \cdot \pi\right)$$

$$\Pr\left(|Z' - \pi| \ge \varepsilon \cdot \pi\right)$$

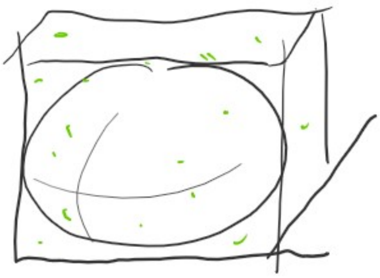
for fixed ε this exponentially decreases with n.

$$\Pr\Big(\Big|\underbrace{\tfrac{n}{4} \cdot Z'}_{Z} - \underbrace{\tfrac{n}{4} \cdot \pi}_{E(Z)}\Big| \ge \underbrace{\tfrac{n}{4}\varepsilon \cdot \pi}_{\varepsilon \cdot E(Z)}\Big) \le 2 e^{-\frac{\frac{n}{4} \cdot \pi \cdot \varepsilon^2}{3}} = e^{-\frac{n \cdot \pi \cdot \varepsilon^2}{12}}$$

$$= \frac{2}{e^{\frac{n \pi \varepsilon^2}{12}}} = \frac{2}{a^{\frac{1}{12}}} \qquad \boxed{e^{n\pi\varepsilon^2} = a}$$

Can we get better algorithm (less samples) using a cube?

Can we get better algorithm (less samples) using a cube and a sphere?



$z_i = 1$ if in a sphere

$$\Pr(z_i = 1) = \frac{V(\text{sphere})}{V(\text{cube})} = \frac{\frac{4}{3}\cdot\pi r^3}{(2r)^3} = \frac{\pi}{6}$$

$$z = \sum_{i=1}^{n} z_i \qquad E(z) = \frac{n\pi}{6}$$

$$z' = \frac{6}{n} z \qquad (\text{estimate of } \pi)$$

$$\Pr\left(\left| \frac{n\cdot z'}{6} - \frac{n\pi}{6} \right| \geq \frac{n}{6}\varepsilon.\pi \right) \leq 2e^{-\frac{\frac{\pi}{6}\cdot n\cdot\varepsilon^2}{3}}$$

$$z \qquad E(z) \qquad \varepsilon.E(z)$$

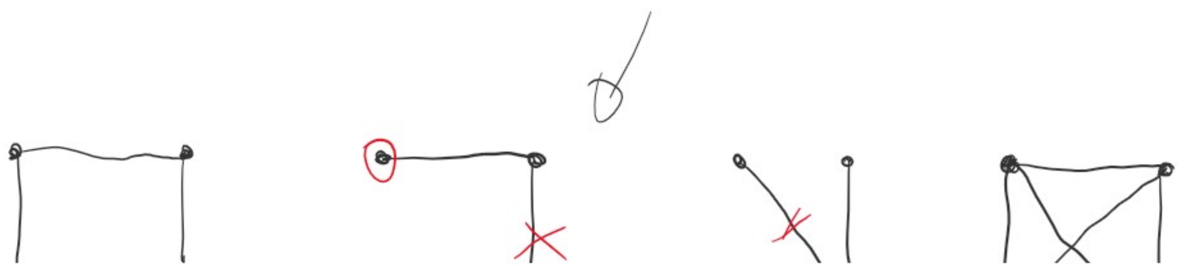
$$\leq 2.e^{-\frac{n\pi\varepsilon^2}{18}}$$

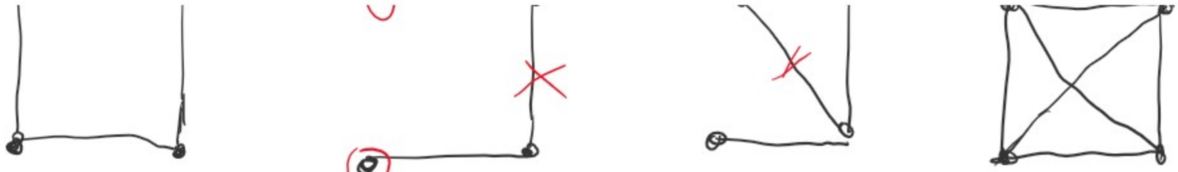
$$\left(e^{n\pi\varepsilon^2} = a\right) \qquad = \frac{2}{a^{\frac{1}{18}}} \leftarrow \text{worse bound}$$

$$a^{1/18} < a^{\frac{1}{12}} \qquad \frac{2}{a^{\frac{1}{18}}} > \frac{2}{a^{1/12}}$$

Routing in Hypercubes (Introduction)

Hypercube is a network architecture

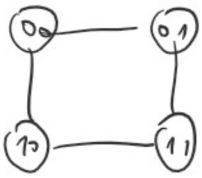


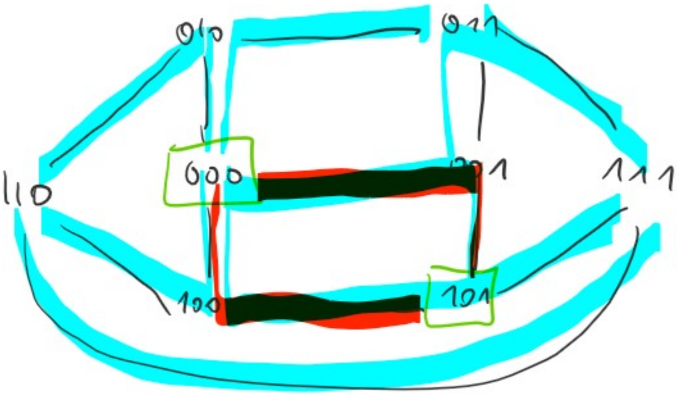


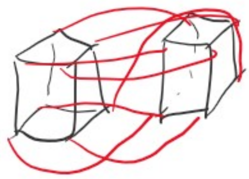
$\frac{N^2}{2}$

$N = 2^d$ nodes labeled by binary strings (of length d).

Two nodes are connected iff their labels differ in a single bit.







Total number of edges is $\frac{N \cdot d}{2}$ $(N \log N)$

Packet $(i, x, d(i))$

Source node — Data — destination

Assumption: In each timestep only 1 packet can be sent through a link.

Deterministic routing: Left to right bit fixing

$1100 \to 0100 \longrightarrow 0101$ ($k!$ different routes of length

1100 → 0100 → 0101

1101

($k!$ different routes of length k, where k is the Hamming distance)

Experiment to test throughput of routing algorithms

Each node gets a packet with a destination.

How long will it take to deliver all the packets?

The worst case

$x_1, \dots, x_d \to x_d, \dots, x_1 \to$ All nodes pass through $000\dots0$

$i = 1100 \to d(i) = 0011$

$j = 0100 \to d(j) = 0010$

$k = 1000 \to d(k) = 0001$

$L \to R$

i: $1100 \to 0100 \to 0000 \to 0010 \to 0011$

j: $0100 \to 0000 \to 0010$

k: $1000 \to 0000 \to 0001$

$$x_1 \dots x_{d/2} \overbrace{0000}^{d/2} \to \overbrace{0000}^{d/2} x_{d/2} \dots x_1$$

$101000 \to 001000 \mid \xrightarrow{\text{delay}}$.

101000 → 001 000

011000 → 001 000

$\overbrace{1000\cdots1}^{d/2}\ \overbrace{0000}^{d/2}$

$0100\cdots1\ 00$

$\vdots$

$00\cdots11\ 000\cdots0$

$\rightarrow\ \overbrace{000}^{d/2-1}\ 1\ \overbrace{000}^{d/2}$

$\overbrace{1100\cdots1}^{d/2}\,|\,\overbrace{00\cdots0}^{d/2}$

101

$\vdots$

$00\cdots111\,|\,00\cdots0$

All $2^{d/2}$ packets pass through $\vec{0}$. Node $\vec{0}$ can send at most d packets simultaneously (d-outgoing links)

$\Omega\left(\frac{2^d}{d}\right)$ timesteps needed.

$i \xrightarrow{L\to R} \sigma(i) \rightarrow d(i)$

$\sigma(i)$ chosen at random (random string of length d)

W.P. at least $1-2^{-5d}$ every packet gets delivered in time $\boxed{14d}$

w.p. … every parcel gets delivered in time $\boxed{14\,d}$.