Lecture 1 c

Street Fighting Mathematics -

Ex - For a polynomial $p(x)$ with degree $\leq n$ that is bounded in $[-1, 1]$ for $x \in [-1, 1]$, how large can $p'(0)$ be?

1. Try small values of n.

For ex - $n = 3$,

$$p(x) = a + bx + cx^2 + dx^3$$

And $\quad -1 \leq p(x) \leq 1 \quad$ for $-1 \leq x \leq 1$

Just substitute random values of x and get a bunch of linear constraints - This can be solved with an LP.

Lecture 2 a : Asymptotics

Extension to big O : $O(g(x))$ denotes an anonymous function $f(x)$ s.t. $0 \leq f(x) \leq c \cdot g(x)$ for some c, $x > x_0$

For ex -

$$\sum i = \frac{n(n+1)}{2} = \frac{1}{2}n^2 + \frac{1}{2}n = \frac{1}{2}n^2 + O(n)$$

Here $O(n)$ is a proxy for $f(x) = \frac{1}{2}x$.

This we can further write as $= \frac{1}{2}n^2\left(1 + O\left(\frac{1}{n}\right)\right)$

goes to 0 as $n \to \infty$.

More Notation: 1) $f(x) = \text{poly}(g(x))$

equivalent to $g(x)^{O(1)}$ ($g(x)$ to a constant power).

2) $f(x) = \tilde{O}(g(x))$

eqv to $f(x) = g(x) \cdot \text{poly}(\log g(x))$

For ex - a) $n^2 \log^3 n = \tilde{O}(n^2)$

b) $n^2 3^n = \tilde{O}(3^n)$

3) when $x \to 0$, $g(x) \to 0$

$f(x) = \tilde{O}(g(x))$

$\Rightarrow f(x) \leq g(x) \text{ polylog}\left(\frac{1}{g(x)}\right)$

Standard Form

$g(n)$ is in standard form if its a product of.

i) constant

ii) constant powers of $\ln n$

iii) constant powers of n

iv) exponential functions

v) $n^{c \cdot n}$, c is a constant.

Harmonic No. Example

$$H_n = 1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n}$$

Lets try to find an upper bound,

$$\leq 1 + \frac{1}{2} + \frac{1}{2} + \frac{1}{4} + \frac{1}{4} + \frac{1}{4} + \frac{1}{4} + \frac{1}{8} + \cdots + \frac{1}{2^n}$$

$\leq 1 + 1 + 1 + 1 + \cdots$ $\log n$ times

$\leq \lfloor \log_2 n \rfloor + 1$

now lets try and find a tight lower bound,

$$\geq 1 + \frac{1}{2} + \frac{1}{4} + \frac{1}{4} + \frac{1}{8} + \frac{1}{8} + \frac{1}{8} + \frac{1}{8} + \cdots \log n \text{ times.}$$

$$\geq 1 + \frac{1}{2} + \frac{1}{2} + \cdots \log n \text{ times.}$$

$$\geq 1 + \frac{1}{2} \lceil \log_2 n \rceil$$

$\Rightarrow H_n \in O(\log n) \quad \Rightarrow H_n \in \Theta(\log n)$
$\in \Omega(\log n)$

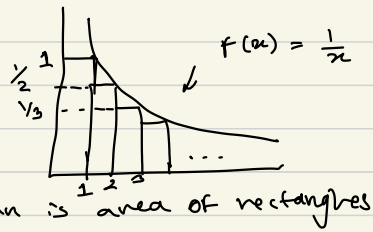
Asymptotic Equivalence. $f(x) \sim g(x)$ as $x \to \infty$ if

$$\equiv \frac{f(x)}{g(x)} = 1$$

$$\equiv f(x) = g(x)(1 \pm o(1))$$

Prop: $H_n \sim \ln(n)$

Proof: consider



Now H_n is area of rectangles.

$\Rightarrow H_n \leq 1 +$ area under curve from 1 to n.

$$\leq 1 + \int_1^n \frac{1}{x} \, dx$$

$$\leq 1 + [\ln x]_1^n = \ln n + 1.$$

Similarly, consider each bar shifted right by 1.

$H_n \geq$ area under the curve from 1 to n+1.

$H_n \geq \ln(n+1)$

Now $\ln(n+1) = \ln(n)\left(1 + \frac{1}{\ln(n)}\right) = \ln(n)\left(1 + O\left(\frac{1}{\ln(n)}\right)\right)$

$\ln(n+1) = \ln\left(n\left(1+\frac{1}{n}\right)\right) = \ln n + \ln\left(1+\frac{1}{n}\right)$

By Taylor Series, $\ln(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} + \cdots$

$\sim x$ as $x \to 0$.

$\Rightarrow \ln n + \frac{1}{n} + O\left(\frac{1}{n^2}\right)$

$= \ln n\left(1 + \frac{1}{n\ln n} + O\left(\frac{1}{n^2 \ln n}\right)\right)$

Asymptotic tricks using Taylor series

$\left.\begin{aligned} \ln(1+x) &\approx x \\ e^x &\approx 1+x \end{aligned}\right\}$ for small x.

$\frac{1}{1-\varepsilon} \approx 1+\varepsilon$

$\sqrt{1+\varepsilon} = 1 + \frac{1}{2}\varepsilon$

Ex - $\sqrt{n+1} - \sqrt{n} \sim ?$

$\sqrt{n+1} = \sqrt{n\left(1+\frac{1}{n}\right)} = \sqrt{n}\sqrt{\left(1+\frac{1}{n}\right)}$

$= \sqrt{n}\left(1 + \frac{1}{2n} \pm O\left(\frac{1}{n^2}\right)\right)$

$= \sqrt{n} + \frac{1}{2\sqrt{n}} \pm O\left(\frac{1}{n^{1.5}}\right)$

$\Rightarrow$ The reqd diff.

$= \frac{1}{2\sqrt{n}} \pm O\left(\frac{1}{n^{1.5}}\right) \sim \frac{1}{2\sqrt{n}}$

Inverting Functions

For $y = x \ln x$, $x \geq 1$

$x = f(y) = ?$

Now taking ln gives us -

$$\ln y = \ln x + \ln \ln x$$

$$\Rightarrow \ln y \sim \ln x \quad \text{as } x, y \to \infty.$$

$$\Rightarrow x \ln x \sim x \ln y$$

$$\Rightarrow x = \frac{y}{\ln y}$$

Ex: $t^2 \log t = n^3$, solve for t.

$$2 \log t \log\log t = 3 \log n$$

$$\Rightarrow \log n \sim \log t$$

$$t^2 \log n \sim t^2 \log t$$

$$t^2 = \frac{n^3}{\log n}$$

$$t = \frac{n^{3/2}}{\log^{1/2} n}$$

Minimizing cost parameter

For ex suppose running time is given by $O\left(\frac{n^3}{t}\right) + O(t \log t)$

Now we want both of these quantities to be nearly equal since $O(a+b) = O(\max(a,b))$

$\therefore$ for $\frac{n^3}{t} = t \log t$

Then we saw above that $t \sim \frac{n^{3/2}}{\log^{1/2} t}$

Lecture 3 : More Asymptotics.

Birthday Paradox.

n balls into m bins.

$\Pr(\text{no collision}) = 1 \cdot \left(1 - \frac{1}{m}\right)\left(1 - \frac{2}{m}\right) \cdots \left(1 - \frac{n-1}{m}\right)$

$1 + x \leq e^x$

$\Rightarrow P_{n,m} \leq e^{-0} \cdot e^{-1/m} \cdots e^{-\frac{n+1}{m}}$

$\leq \exp\left(-\frac{1}{m}\left(1 + 2 + \cdots + n-1\right)\right)$

$\leq \exp\left(\frac{-1}{m} \cdot \frac{(n-1)(n)}{2}\right)$

$1 - x \geq e^{-x - cx^2}$

$\Rightarrow P_{n,m} \geq \exp\left(\frac{-1}{m}\frac{(n)(n-1)}{2}\right) \cdot \exp\left(-\frac{c}{m^2}\left(1^2 + 2^2 + \cdots + (n-1)^2\right)\right)$

$\geq \exp\left(-\frac{1}{m}\frac{(n(n-1))}{2}\right) \cdot \exp\left(\frac{-c}{m^2}\,\Theta(n^3)\right)$

$\geq \exp\left(\frac{-1}{m}\frac{(n(n-1))}{2}\right) \cdot \left(1 - \frac{O(n^3)}{m^2}\right)$ (small if $n \ll m^{2/3}$)

Now $\exp\left(\frac{-1}{m}\frac{n(n-1)}{2}\right) = \exp\left(-\frac{n^2}{2m}\right) \cdot \exp\left(\frac{n}{2m}\right)$

$\geq \exp\left(\frac{-n^2}{2m}\right)\left(1 + O\left(\frac{n}{m}\right)\right)$

$\Rightarrow P_{n,m} \geq \exp\left(\frac{-n^2}{2m}\right) \cdot \left(1 + O\left(\frac{n}{m}\right)\right)\left(1 - \frac{O(n^3)}{m^2}\right)$

Which term dominates? Let's look at the equal case-

$$\frac{n}{m} = \frac{n^3}{m^2}$$

$$\Rightarrow \frac{n^2}{m} = 1 \qquad \Rightarrow \quad n = \sqrt{m}$$

$$\Rightarrow \quad 1 \pm \begin{cases} O\left(n^3/m^2\right), & n \geq \sqrt{m} \\ O\left(n/m\right), & n < \sqrt{m} \end{cases}$$

How should n be as a function of m so that $P_{n,m}$ is nearly half?

Here $\exp\left(\frac{-n^2}{2m}\right) = \frac{1}{2}$

$$\Rightarrow \quad n = \sqrt{2m}\sqrt{\ln 2} = O(\sqrt{m})$$

Here $P_{n,m} = \frac{1}{2} \pm O(\sqrt{m})$

Asymptotics of Factorials

Simple upperbound: n^n

Simple lowerbound: $\left(\frac{n}{2}\right)^{n/2}$

$$\Rightarrow \quad \frac{n}{2}\ln\left(\frac{n}{2}\right) \leq \ln(n!) \leq n\ln(n)$$

$$\Rightarrow \quad n! = 2^{\theta(n \ln n)}$$

Alternatively, from Taylor series of $e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$

$$\frac{x^n}{n!} \leq e^x \qquad \Rightarrow \qquad n! \geq \frac{x^n}{e^x}$$

Now the value of x that maximizes RHS (tightest bound)

$$\frac{d}{dx}\left(\frac{x^n}{e^x}\right) = \frac{e^x n x^{n-1} - x^n e^x}{e^{2x}} = \frac{x^{n-1}}{e^x}(n-x) = 0$$

$$\Rightarrow x = n$$

$$\therefore \quad n! \geq \frac{n^n}{e^n}$$

$$\text{i.e.} \quad n \ln\left(\frac{n}{e}\right) \leq \ln(n!)$$

$$\Rightarrow n(\ln n - 1) \leq \ln(n!)$$

Another alternative: Let $f(n) = \frac{n^n}{n!}$

Then $$\frac{f(n+1)}{f(n)} = \frac{(n+1)^{n+1}}{(n+1)!} \cdot \frac{n!}{n^n}$$

$$= \left(\frac{n+1}{n}\right)^n = \left(1 + \frac{1}{n}\right)^n$$

$$\leq \left(e^{1/n}\right)^n = e$$

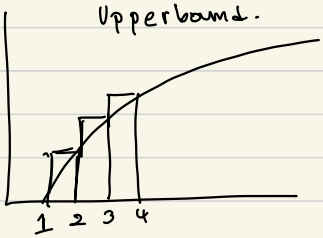
This gives the same bound: $n! \approx \frac{n^n}{e^n}$
(since $f(1) = 1$ and $f(n+1) = e\,f(n)$)

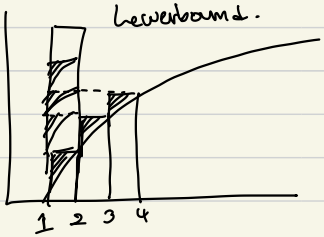
Another perspective. —

$$\ln(n!) = \sum_{i}^{n} \ln(i) \geq \int_1^n \ln x \, dx.$$

$$\geq [x \ln x - x]_1^n$$

$$\geq n \ln n - n + 1$$





Now the lowerbound is adding these "triangles" to the curve area. These are half of the area of the rectangle.

$$= \frac{1}{2} \ln(n)$$

$$\Rightarrow \leq n \ln n - n + 1 + \frac{1}{2} \ln(n)$$

$$\Rightarrow \quad \frac{n^n}{e^n} \cdot e \leq n! \leq \frac{n^n}{e^n} \cdot e \sqrt{n}$$

$$\Rightarrow n! \in \tilde{\Theta}\left(\frac{n^n}{e^n}\right)$$

Stirling's Formula -

$$n! \sim \sqrt{2\pi} \sqrt{n} \frac{n^n}{e^n}$$

Asymptotics for Binomial Coeffs -

$$\binom{n}{k} = \frac{n(n-1)\cdots(n-k+1)}{k!} \Rightarrow \text{for small } k, \quad \binom{n}{k} \approx \frac{n^k}{k!}$$

More accurately -

$$\binom{n}{k} = \frac{n^k}{k!}\left(1-\frac{1}{n}\right)\left(1-\frac{2}{n}\right)\cdots\left(1-\frac{(k-1)}{n}\right)$$

$$\leq \frac{n^k}{k!} \exp\left(\sum_i -\frac{i}{n}\right)$$

$$\leq \frac{n^k}{k!} \exp\left(\frac{-1}{n} \frac{k(k-1)}{2} \right)$$

$$\approx \frac{n^k}{k!} \left(1 - \frac{k^2}{2n} \right) \quad \text{(for small } k\text{)}.$$

$$\Rightarrow \binom{n}{k} \sim \frac{n^k}{k!} \quad \text{if} \quad k \in o(\sqrt{n})$$

Simple bounds for $\binom{n}{k}$ -

$$\binom{n}{k} \leq \frac{n^k}{k!} \leq \frac{n^k}{k^k} \cdot e^k = \left(\frac{en}{k} \right)^k$$

For the lower bound, notice $\frac{n-i}{k-i} \geq \frac{n}{k}$

$$\Rightarrow \binom{n}{k} = \frac{n}{k} \frac{n-1}{k-1} \frac{n-2}{k-2} \cdots$$

$$\geq \frac{n}{k} \cdot \frac{n}{k} \cdots$$

$$\geq \left(\frac{n}{k} \right)^k$$

Another idea

$$\binom{n}{k} = \binom{n}{n-k} \Rightarrow \text{WLOG}, \quad k \leq \frac{n}{2}$$

Write $k = pn$ for $0 \leq p \leq \frac{1}{2}$.

Let $V(n,k) = \binom{n}{0} + \binom{n}{1} + \cdots + \binom{n}{k}$

Intuitively, this is the no. of strings of length n having $\leq k$ ones.

This also the volume of the hamming ball of radius k. (points at hamming distance $\leq k$ from origin).

By the binomial theorem,

$$(1+x)^n = \binom{n}{0}x^0 + \binom{n}{1}x^1 + \cdots + \binom{n}{k}x^k + \cdots \binom{n}{n}x^n$$

$$\geq \binom{n}{0}x^0 + \cdots + \binom{n}{k}x^k$$

For $0 \leq x \leq 1$, $x^k \leq x^i$ for $i < k$

$$\Rightarrow \geq x^k \left(\binom{n}{0} + \binom{n}{1} + \cdots + \binom{n}{k} \right)$$

$$\geq x^k \, V(n,k).$$

$$\Rightarrow V(n,k) \leq \frac{(1+x)^n}{x^k}$$

For the tightest bound we want to minimize RHS.

$$\Rightarrow \frac{\partial}{\partial x}\left(\frac{1+x}{x^p}\right)^n = n\left(\frac{1+x}{x^p}\right)^{n-1} \cdot \frac{x^p - p x^{p-1}(1+x)}{x^{2p}} = 0$$

$$= (1+x)^{n-1} x^{p-1} (x - p - px) = 0$$

$$\Rightarrow x = \frac{p}{1-p}$$

Let $1-p = q$.

$$\Rightarrow V(n,k) \leq \left(x^{-p} + x^{1-p} \right)^n$$

$$\leq \left(\left(\frac{p}{q}\right)^{-p} + \left(\frac{p}{q}\right)^{q} \right)^n$$

$$\leq \left(\left(\frac{p}{q}\right)^{-p} \left(\frac{q}{q} + \frac{p}{q}\right)^{q+p=1} \right)^n$$

$$\leq \left(\left(\frac{p}{q}\right)^{-p} \left(\frac{p+q=1}{q} \right) \right)^n$$

$$\leq \left(\left(\frac{p}{q}\right)^{-p} \cdot \frac{1}{q} \right)^n = \left(\left(\frac{q}{p}\right)^{p} \cdot \frac{1}{q} \right)^n$$

$$\leq \left(\frac{1}{p^p} \cdot \frac{1}{q^{1-p}} \right)^n$$

$$\leq \left(\frac{1}{p^p} \cdot \frac{1}{q^q} \right)^n$$

$$\Rightarrow V(n,k) \leq \left(\left(\frac{1}{p}\right)^p \left(\frac{1}{q}\right)^q \right)^n$$

$$\leq 2^{nH(n)}$$

where $H(n) = -p\log p - q\log q$

which is the binary entropy.

Probability

Setup: let $X_1 \dots X_n$ be iid bin rvs with $pr[X_i = 1] = p$

let $S_n = X_1 + \dots + X_n$

$$\mathbb{E}[S] = \sum_i \mathbb{E}[X_i] = np$$

$$Var[S] = \sum_i Var[X_i] \quad \text{since } X_i \text{ are indep.}$$

$$\Rightarrow Var[X_i] = \mathbb{E}[X_i^2] - \mathbb{E}[X_i]^2$$

$$= p - p^2$$

$$= p(1-p) = pq$$

$$\Rightarrow Var[S] = npq.$$

Properties of Var -

$\text{Var}[X + X'] = \text{Var}[X] + \text{Var}[X']$ for indep X, X'

$\text{Var}[X + c] = \text{Var}[X]$

$\text{Var}[Xc] = c^2 \text{Var}[X]$

$\text{Var}[X] = \sigma_X^2$

Lifehacks: i) Set mean to zero.
ii) Set variance to 1.

In general: $Z = \frac{S - \mu}{\sigma}$

For a fair coin: $\mu = np = \frac{n}{2}$

$\sigma = \sqrt{npq} = \sqrt{\frac{n}{2}}$

$$\Rightarrow Z_n = \frac{S_n - \frac{n}{2}}{\frac{\sqrt{n}}{2}} = \frac{1}{\sqrt{n}}\left(2S_n - n\right)$$

$$= \frac{1}{\sqrt{n}} \underbrace{\sum_i 2X_i - 1}_{\text{this is an rv } Y = \begin{cases} +1 \\ -1 \end{cases}, \text{ with prob } \frac{1}{2}}$$

Now if we want $\Pr[S_n = \frac{n}{2}]$ this is equivalent to the prob $\Pr[Z = 0]$.

i.e. $$\frac{\binom{n}{n/2}}{2^n} = \frac{1}{2^n} \frac{n!}{(n/2)!(n/2)!} \leq \frac{1}{2^n} \cdot \frac{\sqrt{2\pi}\sqrt{n}\,\frac{n^n}{e^n}}{\sqrt{2\pi}\sqrt{n/2}\,\frac{(n/2)^{n/2}}{e^{n/2}} \cdot \sqrt{2\pi}\sqrt{\frac{n}{2}} \cdot \left(\frac{(n/2)^{n/2}}{e^{n/2}}\right)}$$

$$\leq \frac{1}{2^n} \frac{1}{\sqrt{2\pi}} \frac{2}{\sqrt{n}} \cdot \frac{1}{2^n}$$

$$\leq \sqrt{\frac{2}{\pi}} \frac{1}{\sqrt{n}}$$

Plotting this for various values of n gives the bell curve.

Central Limit Theorem:

Let $X_1 \cdots X_n$ be iid rvs. Then the associated Z_n tends to the Gaussian, i.e.

$\forall u$, $\Pr[Z_n \leq u] = \Pr[N \leq u] \pm o(1)$ as $n \to \infty$

Gaussian RVs

$Z \sim N(0,1)$ if it is a continuous rv distributed with p.d.f $\phi(z) = \frac{1}{\sqrt{2\pi}} e^{-z^2/2}$

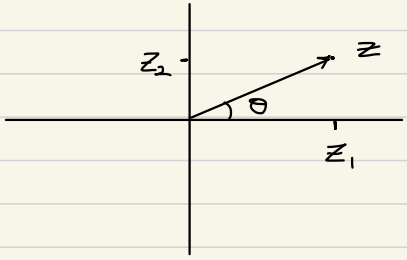
↓ Normalizing Factor $\left[\int_{-\infty}^{\infty} e^{-z^2/2} \, dz = \sqrt{2\pi} \right]$

The "most important" fact -

Let $Z = (Z_1 \cdots Z_d) \in \mathbb{R}^d$ where Z_i are iid $\sim N(0,1)$

Then Z's distribution is rotationally symmetric.

For ex, in $\mathbb{R}^2$ -



θ is uniformly distributed.

i.e. Z intersects each point on the unit circle uniformly at random.

Proof: Due to i.i.d, pdf of z = $\phi(z_1) \cdot \phi(z_2) \cdots \phi(z_d)$

$$= \left(\frac{1}{\sqrt{2\pi}}\right)^d \exp\left(\frac{1}{2}\sum_i z_i^2\right)$$

$$= \left(\frac{1}{\sqrt{2\pi}}\right)^d \exp\left(\frac{1}{2}\|z\|^2\right)$$

Thus this pdf only depends on the length of z.

corollary: sum of independent Gaussians is Gaussian.

General Gaussian rvs - Let $z \sim N(0,1)$. Let $\mu \in \mathbb{R}$, $\sigma \in \mathbb{R}^+$.
Let $Y = \mu + \sigma z$. $\Rightarrow \mathbb{E}[Y] = \mathbb{E}[\mu] + \sigma\,\mathbb{E}[z]$

$$= \mu + 0 = \mu.$$

$$\text{Var}[Y] = 0 + \sigma^2 \text{Var}[z]$$

$$= \sigma^2$$

Then $Y \sim N(\mu, \sigma)$

Tip: When dealing with these, just normalize back to a standard Gaussian.

Corollary: If $X \sim N(\mu_1, \sigma_1^2)$ and $Y \sim N(\mu_2, \sigma_2^2)$ are independent, the sum $Z = aX + bY$ is a Gaussian $N(a\mu_1 + b\mu_2,\ a^2\sigma_1^2 + b^2\sigma_2^2)$

$$\mathbb{E}[Z] = a\,\mathbb{E}[X] + b\,\mathbb{E}[Y] = a\mu_1 + b\mu_2.$$

$$\text{Var}[Z] = a^2 \text{Var}[X] + b^2 \text{Var}[Y] = a^2\sigma_1^2 + b^2\sigma_2^2$$

$$f_Z(z) = f_X(az) + f_Y(bz)$$

let $X' = \frac{X - \mu_1}{\sigma_1}$, $Y' = \frac{Y - \mu_2}{\sigma_2}$

$$\Rightarrow Z = a(\sigma_1 X' + \mu_1) + b(\sigma_2 Y' + \mu_2)$$

$= a\sigma_1 X' + b\sigma_2 Y' + a\mu_1 + b\mu_2$.

consider the vectors $(a\sigma_1, b\sigma_2)$ and (X', Y').

We know angle of (X', Y') is UAR. $\Rightarrow$ angle of dot product with $(a\sigma_1, b\sigma_2)$ is also UAR.

WLOG consider the rotation of $(a\sigma_1, b\sigma_2)$ to the x-axis $\Rightarrow$ dot product of (X', Y') with $(\sqrt{a^2\sigma_1^2 + b^2\sigma_2^2}, 0)$ is also distributed the same way.

$\Rightarrow Z = \sqrt{a^2\sigma_1^2 + b^2\sigma_2^2}\, X + a\mu_1 + b\mu_2$.

which is the required normal $N(a\mu_1 + b\mu_2, a^2\sigma_1^2 + b^2\sigma_2^2)$

Berry - Esseen Theorem

Let $X_1, \ldots X_n$ be indep rvs. Assume $\mathbb{E}[X_i] = 0$ and $\sigma_i^2 = \mathbb{E}[X_i^2]$, and $\sum_i \sigma_i^2 = 1$. Let $S = X_1 + \cdots + X_n$. Then $\forall u \in \mathbb{R}$

$$\left| \Pr[S \le u] - \Pr_{Z \sim N(0,1)}[Z \le u] \right| \le 0.56\, \beta$$

where $\beta = \sum_i \mathbb{E}[|X_i|^3]$

For ex - Coin flips -

Let $X_i = \begin{cases} +1/\sqrt{n} & \text{with prob } 1/2 \\ -1/\sqrt{n} & \text{" } 1/2 \end{cases}$

clearly $\mu_i = 0$

$\sigma_i^2 = \mathbb{E}[X_i^2] = \frac{1}{n}$

$\Rightarrow \sum \sigma_i = 1$.

$\Rightarrow \mathbb{E}[X_i^3] = \frac{1}{n^{3/2}} \Rightarrow \beta = \frac{1}{\sqrt{n}}$

$\Rightarrow \left| \Pr[S_n \le u] - \Pr[Z \le u] \right| \le \frac{0.56}{\sqrt{n}}$

(acc to the theorem).

$S_n = \frac{\#\text{Heads} - \#\text{tails}}{\sqrt{n}}$

$= \frac{H - (n-H)}{\sqrt{n}} = \frac{2H}{\sqrt{n}} - \sqrt{n}$

For $S_n \le u$, $H \le (u + \sqrt{n}) \frac{\sqrt{n}}{2}$

$\le \frac{u\sqrt{n}}{2} + \frac{n}{2}$

↓ St. Dev. ↳ mean

$\Pr[Z \le u] = \int_{-\infty}^{u} f_Z(x)\,dx = \Phi_Z(u)$

↳ CDF.

Fact: $\Phi(u) \sim \frac{f(u)}{u}$ as $u \to \infty$.

Markov and Chebyshev

i) Markov: If you know only mean and that $X \ge 0$

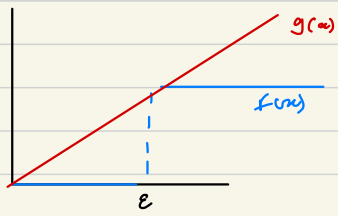
Then $\Pr[X \ge \varepsilon] \le \frac{\mathbb{E}[X]}{\varepsilon}$

alternatively

$\Pr[X \ge \varepsilon\, \mathbb{E}[X]] \le \frac{1}{\varepsilon}$

Picture Proof: $\Pr[X \geq \varepsilon] = \mathbb{E}[f(x)]$

where $f(x) = \begin{cases} 1, & x \geq \varepsilon \\ 0, & o/w \end{cases}$



let $g(x) = \frac{1}{\varepsilon} x$

$\Rightarrow \mathbb{E}[f(x)] \leq \mathbb{E}[g(x)] = \mathbb{E}\left[\frac{x}{\varepsilon}\right]$

$\leq \frac{1}{\varepsilon} \mathbb{E}[x]$

A "Markov-like" inequality -

For $0 \leq x \leq 1$, s.t. $\mathbb{E}[x] = \varepsilon$

Then $\Pr[x \geq \frac{\varepsilon}{2}] \geq \frac{\varepsilon}{2}$.

A "words" proof: Assume to the contrary that $\Pr[x \geq \frac{\varepsilon}{2}] < \frac{\varepsilon}{2}$.

$$\mathbb{E}[x] = \int_0^1 \Pr[x] \cdot x \, dx.$$

$$= \int_0^{\varepsilon/2} \Pr[x]\, x \, dx + \int_{\varepsilon/2}^1 \Pr[x]\, x \, dx$$

$$\leq \int_0^{\varepsilon/2} \Pr[x] \cdot \frac{\varepsilon}{2} + \int_{\varepsilon/2}^1 \Pr[x] \cdot 1 \, dx$$

$$\leq \frac{\varepsilon}{2} \cdot \Pr[x < \tfrac{\varepsilon}{2}] + \Pr[x \geq \tfrac{\varepsilon}{2}]$$

↓ trivial upperbound; ← assumption.

$$< \frac{\varepsilon}{2} \cdot 1 + \frac{\varepsilon}{2}$$

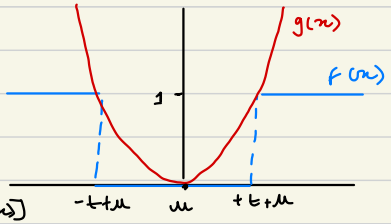
$< \varepsilon$ which is a contradiction

Chebyshev: If you know both mean and variance

$\forall\, t > 0$, $\Pr[|X - \mu| \geq t\sigma] \leq \frac{1}{t^2}$

alternatively - $\Pr[|X-\mu| \geq t] \leq \frac{\text{Var}[X]}{t^2}$

Picture Proof:



$g(x) = \frac{(x-\mu)^2}{t^2}$

$\Rightarrow \mathbb{E}[f(x)] \leq \mathbb{E}[g(x)]$

$$\leq \frac{\mathbb{E}[x^2] + \mu^2 - 2\mu\mathbb{E}[x]}{t^2}$$

$$\leq \frac{\text{Var}[x]}{t^2}.$$

Exc: For $X \geq 0$, show

$$\Pr[x=0] \leq \frac{\text{Var}[x]}{\mu^2} \leq \frac{\mathbb{E}[x^2]}{\mu^2}$$

$\Pr[X=0] = 1 - \Pr[X>0]$

Fun Fact: For an rv of the form $X_1 + X_2 + \cdots + X_n$. Chebyshev can be applied even when they're only pair-wise independent.

Chernoff: If you know $X = X_1 + \cdots + X_n$ where X_i's are all indep.

Consider the simple case where $X_i = \begin{cases} -1, & \text{wp } 1/2 \\ +1, & \text{wp } 1/2 \end{cases}$

We are interested (for example) in
$\Pr[|x| \geq 0 + 10\sqrt{\ln n}\sqrt{n}]$

Again using a picture –

consider $g(x) = \dfrac{x^4}{t^4 n^2}$

$$\Rightarrow \Pr[|x| \geq \mu + t\sqrt{n}] = \mathbb{E}[f(x)] \leq \mathbb{E}[g(x)] \leq \frac{\mathbb{E}[x^4]}{t^4 n^2}$$

Notice $x^4 = \left(\sum x_i\right)^4 = \sum x_i^4 +$ some $x_i^3 x_j$ terms
$+$ some $x_i^2 x_j^2$ terms
$+$ some $x_i x_j x_k$ terms
$+$ some $x_i x_j x_k x_l$ terms

Notice that by linearity and independence, $\mathbb{E}[x_i^3 x_j]$
$= \mathbb{E}[x_i^3] \cdot \mathbb{E}[x_j]$
$= \mathbb{E}[x_i^3] \cdot 0 = 0.$

Similarly, $x_i^2 x_j x_k$ and $x_i x_j x_k x_l$ also go to 0.

Now $\sum \mathbb{E}[x_i^4] = \sum 1 = n$

Similarly each $x_i^2 x_j^2$ also gives 1. # of such terms –
ways of choosing i and j $\leftarrow \binom{n}{2} \cdot \binom{4}{2} \rightarrow$ ways of arranging them.

$= 3n^2 - 3n$

$\Rightarrow \mathbb{E}[X^4] \leq 3n^2$

$\Rightarrow \Pr[X \geq t\sqrt{n}] \leq \frac{3n^2}{t^4 n^2} = \frac{3}{t^4}$

Lets use an exponential instead of a n^4.

Let $g(x) = \frac{e^{\lambda x}}{e^{\lambda u}}$

Now $X \geq u \Rightarrow e^{\lambda x} \geq e^{\lambda u}$

By Markov, $\Pr[e^{\lambda x} \geq e^{\lambda u}] \leq \frac{\mathbb{E}[e^{\lambda x}]}{e^{\lambda u}}$ → moment generating function

$$\begin{aligned}\mathbb{E}[e^{\lambda x}] &= \mathbb{E}[\exp(\lambda x_1)\exp(\lambda x_2)\cdots\exp(\lambda x_n)] \\ &= \mathbb{E}[\exp(\lambda X_1)]\cdots\mathbb{E}[\exp(\lambda X_n)] \\ &= \mathbb{E}[\exp(\lambda X_1)]^n \quad (\text{if i.i.d.})\end{aligned}$$

Now $\mathbb{E}[e^{\lambda X_i}] = \frac{1}{2}e^{\lambda} + \frac{1}{2}e^{-\lambda} \leq e^{\lambda^2/2}$ (by Taylor Series)

$\Rightarrow \Pr[e^{\lambda x} \geq e^{\lambda u}] \leq \frac{e^{\lambda^2/2 \cdot n}}{e^{\lambda u}} = \exp\left(\frac{n\lambda^2}{2} - \lambda u\right)$

Want to minimize this,

$\frac{2n\lambda}{2} - u = 0$

$\lambda = \frac{u}{n}$

$\Rightarrow \exp\left(\frac{u^2}{2n} - \frac{u^2}{n}\right) = \exp\left(\frac{-u^2}{2n}\right)$

Theorems that can just be plugged in

Let $X = X_1 + \cdots + X_n$ s.t X_is are indep.

1. Hoeffding's

say $a_i \le X_i \le b_i$

$$\Pr[X \ge \mu + t] \le \exp\left(\frac{-2t^2}{\sum (b_i - a_i)^2}\right) \quad \forall t > 0$$
$$\Pr[X \le \mu - t]$$

2. Chernoff's

say $0 \le X_i \le 1$. $\forall \varepsilon > 0$

$$\Pr[X \le (1-\varepsilon)\mu] \le \exp\left(\frac{-\varepsilon^2 \mu}{2}\right)$$

$$\Pr[X \ge (1+\varepsilon)\mu] \le \exp\left(\frac{-\varepsilon^2}{2+\varepsilon}\mu\right)$$

3. If you know only a bound on the mean. $\mu_L \le \mu$

$$\Pr[X \le (1-\varepsilon)\mu_L] \le \exp\left(\frac{-\varepsilon^2 \mu_L}{2}\right)$$

$$\Pr[X \ge (1+\varepsilon)\mu_H] \le \exp\left(\frac{-\varepsilon^2}{2+\varepsilon}\mu_H\right)$$

4. Sampling Theorem

say $0 \le X \le 1$ with actual mean μ.
Get n indep samples $X_1, \cdots X_n$.
Estimate $\hat{\mu} = \frac{X_1 + \cdots + X_n}{n}$

$$\text{If } n \geq \frac{3 \ln\left(\frac{1}{\delta}\right)}{\varepsilon^2} \quad \text{then } \Pr\left[|\mu - \hat{\mu}| \leq \varepsilon\right] \geq 1 - \delta$$

Turing Machines

Running Time of an algorithm depends on the computation model.

Main Advantage of the TM model is that it's 100% clear what the running time (and space usage) is.

Thm: Solving Palindromes on a 1-tape TM requires $\Omega(n^2)$ time.

This is sort of unrealistic compared to the real world where we do see an $O(n)$ algorithm.

However multi-tape TMs can solve Pals in $O(n)$, but it needs $\Omega(n)$ space. In the real world we need only $O(1)$ space.

Formally: Multitape TMs solving Pals in time T and space S need $T \cdot S = \Omega(n^2)$

RAM TMs

A TM where the head can jump to a tape cell.

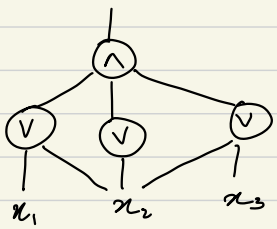
This can solve Palindromes in $O(n)$ time and $O(\log n)$ space.

Notice that in any TM model, even reading input in $O(n \log n)$

Circuits

Boolean circuits compute boolean functions of the form $f: \{0,1\}^n \to \{0,1\}$.

For ex



More formally, a circuit is a DAG where the nodes are gates.

$\mathcal{B}$ = "basis" = the set of allowed gates.

Measure of complexity - a) size : # non-input gates.
b) depth : longest input output path length.

Popular Bases -

i) $\mathcal{B} = \{\neg, \text{bin } \wedge, \text{bin } \vee\}$ aka de Morgan's gates.

ii) $\{$ all $g: \{0,1\}^2 \to \{0,1\}\}$ any binary function with fan-in 2.

iii) $\{\neg,$ n-ary $\vee,$ n-ary $\wedge\}$ de Morgan's gates but with any fan-in.

For ex - AND $\{0,1\}^n \to \{0,1\}$

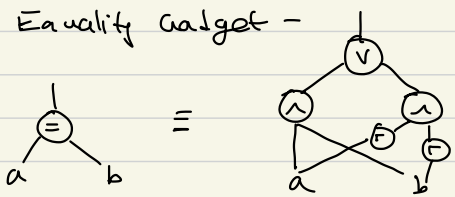
In model iii) this is a size 1 circuit.

ii) & i) this needs $\log n$ levels $\Rightarrow O(n)$ circuits.

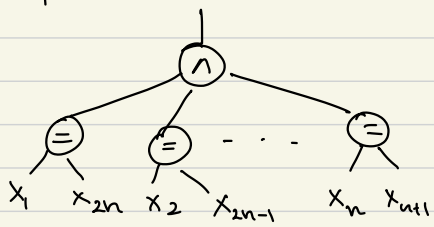
Palindrome in Circuits

Palindrome : $\{0,1\}^{2n} \to \{0,1\}$ (WLOG assume length even).

Equality Gadget -



Now a simple circuit -



This is a depth 3 circuit with size $3n$ in model 3.
(not gates are ignored when computing size / depth)

Circuit Family : One circuit for each length n.

A circuit family C decides a language L if C_n decides $L \cap \{0,1\}^n$. $\forall n$.

Complexity Classes –

AC^0: Languages decided by a circuit family of depth $O(1)$ and size $poly(n)$ (in model 3)

NC: languages decided by $polylog(n)$ depth and $poly(n)$ size. (in any model: I can convert to binary gates using depth $\log n$).

P/poly: Languages decided by $poly(n)$ size.

Thm: Multitape TMs of time $T(n)$ are simulatable by circuits of size $O(T(n)\log n)$.

"Uniformity".

Fact: Halting problem is decidable by circuit families of size $\tilde{O}(2^n)$.
This is because even the halting problem is just a boolean function mapping the program in binary to $\{0,1\}$. ⇒ the circuit can simply be the truth table of the function.

Family $C(n)$ is uniform if ∃ poly time algo for outputting C_n.

Circuit size lowerbounds.

Shannon: $\exists f \{0,1\}^n \to \{0,1\}$ needing circuit size $\Omega(2^n/n)$

Thm: $\exists L \in P$ needing $\geq 5n - o(n)$ for model 1.

Thm: $\exists L \in P$ needing $\geq \left(3 \cdot \frac{1}{86}\right) n - O(n)$

Finding good lowerbound is difficult.

Thm: consider the parity and majority functions.
(parity: outputting LSB; majority: if more than half of the bits are one.)

These are proven $\notin AC^0$
similarly multiplying 2 n bit nos $\notin AC^0$

Thus constant depth seems to be a strong restriction.

Word Ram Model –
→ Memory divided into "words" of w bits.
→ For size n input, assume $w \geq \log n$.
→ costs time 1 to do basic operations on words
→ 'Basic operations' include add, sub, bitwise, compares shifts. (multiplication not included). conditionals, random index access.

Notice these restrictions mean I can access super polynomial space. (by making w very large) and do stuff really fast. but this is pointless. Generally $w = O(\log n)$.

Ex: Given n nos find sum.

```
tot = 0
for i = 1 to n
    tot += a[i]
```

This takes time exactly as expected. $O(n)$ time and $O(1)$ space.
However tot may require more than 1 word to store.

Transdichotomous RAM model : running time indep of w.

AC^0 RAM model: Basic operations are all AC^0 operations

Now consider sorting.
say the integers are $0, \cdots, N-1$

consider $O(n)$ counting sort.
1. Allocate array of size N.
2. Count the #occurances for each no. in the input.
3. Print each with non zero occurances.

This however needs space $O(n)$. However even if the no.s are $3\log n$ bits long the space requirement becomes $O(n^3)$.
This is dependent on the size of the elements being sorted

Improvement: Radix sort.
Do counting sort on each bit. $O(w \cdot n)$
(w: word length.)

Generalized Radix -
Do counting sort on a collection of k bits $k \leq w$.
space: $O(2^k)$
time: $O\left(\frac{nw}{k}\right)$

Basic Arithmetic

Q: What is time complexity (implied word RAM model with $w = \Omega(\log n)$) to add two n-bit integers?

You can do this in $O(\frac{n}{w})$ time. Group the n bit nos in w length bits. The carry can also be at most w length.

Q: Time to mult 2 n-bit nos?

WLOG: Assume these are grouped in w bits. i.e. each word is b/w $0 \cdots N-1$.

Now the grade school algorithm runs in $O(N^2)$ (assuming word RAM allows word-multiplication in $O(1)$).

Karatsuba drops this to $O(N^{\log_2 3})$.

SOTA: Use FFT.

Thm: (Knuth) FFT in $O(N \log N)$ time in word RAM model if mult of 2 words in $O(1)$ time.

Now time to multiply: $O\left(\frac{n}{w} \cdot \log\left(\frac{n}{w}\right)\right)$

For $w = O(\log n)$

$$O\left(\frac{n}{\log n} \log n - \frac{n}{\log n} \log\log n\right)$$

$$= O(n - \varepsilon n)$$

$$= O(n)$$

On circuits and 2-tape TMs best known is still $O(n \log n)$.

Let $M(n)$ be time for multiplication.

⇒ Division: $O(M(n))$ } Iterate with Newton's method.
kth roots: $O(M(n))$ }

Modular expo: $O(k\,M(n))$
(to a k bit exponent)

GCD: $O(M(n)\log n)$ (note that Euclids is actually quadratic time) (of n bit nos)

ln, exp, sin, cos: $O(M(n)\log n)$ (Taylor series I think).

Primality Test: $O(M(n)\cdot n)$ (Rabin Miller).

DFT

$A = a_0 + a_1 n^1 + \cdots + a_{N-1} n^{N-1}$ } in base n.
$B = b_0 + b_1 n^1 + \cdots + b_{N-1} n^{N-1}$ }

Here these are two polynomials with parameter n.

We want the product polynomial. i.e. $R(x) = C_{2N-2}x^{2N-2} + \cdots + C_0$

Exc: Each $C_j < n^3$
Do carrying/addition in $O(N)$ time.
↳ Fairly simple if Division is in $O(1)$.

I can interpolate an $N-1$ degree polynomial using its evaluation on N points.

So if I have $2N$ points evaluated for both polynomials, Finding K is just pointwise multiplications.

Now the problem becomes to do evaluation and interpolation fast.

Evaluation is a matrix-vector product -

$$\begin{bmatrix} p(\omega_0) \\ \vdots \\ p(\omega_{N-1}) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & \omega_0 & \cdots & \omega_{N-1} \\ 1 & \omega_0^2 & \cdots & \omega_{N-1}^2 \\ & & \vdots & \\ 1 & \omega_0^{N-1} & \cdots & \omega_{N-1}^{N-1} \end{bmatrix}}_{(V)} \begin{bmatrix} a_0 \\ \vdots \\ a_{N-1} \end{bmatrix}$$

$\Rightarrow$ Interpolation is just multiplication by V^{-1}.

Now if I choose $\omega_j = \omega^{-j} = \exp\left(-\frac{2\pi i}{N}\right)^j$

i.e. $V[k,l] = \omega^{kl \bmod N}$

$= \exp\left(\frac{-2\pi i}{N} \cdot kl \bmod N\right)$

<u>col Properties</u>

1. Inner product of 2 distinct cols $= 0$.
2. Inner product of col with itself $= N$

<u>Proof</u>: For cols k_1, k_2

Inner Prod $= \sum_j \omega^{jk_1} \cdot \omega^{-jk_2}$

$= \sum_j \omega^{j(k_1 - k_2)}$

if $k_1 = k_2$, $= \sum_j \omega^0 = N$

else, its a GP: $\sum_j (\omega^{k_1 - k_2})^j$

$= \frac{1 - \omega^{(k_1 - k_2)N}}{1 - \omega^{k_1 - k_2}} = \frac{1 - (\omega^N)^{k_1 - k_2}}{1 - \omega^{k_1 - k_2}}$

$w^N = 1 \Rightarrow \text{Num} = 0.$

$\Rightarrow$ cols are orthogonal to each other. To make it an orthonormal basis we can always divide by $\sqrt{N}$. (the magnitude)

Cool fact: $V^{-1} = \frac{1}{N} \cdot V$ because of this property.

FFT Magic

Assume N is a power of 2.

Idea: DFT_N reduces to 2 calls of $DFT_{N/2} + O(N)$ which is a clear $O(N \log N)$ reccurence.

The "split": compute the multiplication for the even columns.

Notice that the bottom half of the result will be the same. $\Rightarrow$ This is simply a $DFT_{N/2}$ followed by a "stacking" operation (repeating the values for the remaining).

Now for odd cols, its just the previous even col multiplied by w^i for the ith row.
so do another $DFT_{N/2}$ followed by a stacking followed by an inner product.

Analysis of Boolean Functions

Useful tools in a lot of TCS Areas.

Usually a boolean fn is written as $f: \{0,1\}^n \to \{0,1\}$.

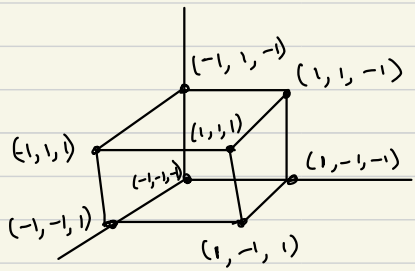
Domain can alternatively be
→ $\mathbb{F}_2^n$
→ $\{\pm 1\}^n$ where $+1 \to T$, $-1 \to F$ Multiplying same as xor

Range can be -
→ $\mathbb{F}_2$
→ $\{\pm 1\}$ or $\mathbb{R}$ (generalization).

Consider an example function -

$f = \text{Maj}_3 : \{\pm 1\}^3 \to \{\pm 1\}$ output most frequent truth value.

x_1	x_2	x_3	Maj_3
-1	-1	-1	-1
-1	-1	1	-1
-1	1	-1	-1
-1	1	1	1
1	-1	-1	-1
1	-1	1	1
1	1	-1	1
1	1	1	1



The function labels each vertex.

These are two visualizations, but here is another one -

Fit a polynomial - that interpolates these labels.

An easy way to get such a polynomial is by lagrange interpolation. But here since we have only ± 1 the expression is simpler-

For $(-1,-1,-1)$: $\left(\frac{(x_1 - 1)}{(-1-(+1))}\right)\left(\frac{(x_2-1)}{(-1-(+1))}\right)$

$$\left(\frac{(x_3-1)}{(-1-(+1))}\right) \cdot y_{-1-1-1}$$

$$= \left(\frac{1}{2} - \frac{1}{2}x_1\right)\left(\frac{1}{2} - \frac{1}{2}x_2\right)\left(\frac{1}{2} - \frac{1}{2}x_3\right)\cdot(-1)$$

Similarly repeat for each point.

After opening up and cancelling we get

$$\frac{1}{2}x_1 + \frac{1}{2}x_2 + \frac{1}{2}x_3 - \frac{1}{2}x_1x_2x_3$$

Similarly if we want XOR (x_1, x_2, x_3) we can do -

$$\text{Parity}(x) = x_1x_2x_3$$

Notice these are both multilinear (each variable has highest power 1).

Therefore Thm: Every function f mapping n bit $\{\pm 1\}^n \to \mathbb{R}$ is expressible as a multilinear polynomial. (that is unique).

General form of multilinear polynomial: $f(x) = \sum_{S \subseteq [n]} \hat{f}(S) \prod_{i \in S} x_i$, where $x_\phi = 1$ ($\hat{f}(S)$ → coefficient)

Now notice that $\prod_{i \in S} x_i$ is the XOR of some subset of input variables. $\therefore$ this can be thought of as a linear combination of XORs of subsets of variables.

We call this the Fourier expansion and the $\hat{f}(S)$ are the fourier coefficients.

For the above function $Maj_3(x)$ the fourier coeffs are —

$\widehat{Maj_3}(\{1\}) = \frac{1}{2}$ $\quad$ $\widehat{Maj_3}(\{1,2,3\}) = \frac{1}{2}$

$\widehat{Maj_3}(\{2\}) = \frac{1}{2}$

$\widehat{Maj_3}(\{3\}) = \frac{1}{2}$ $\quad$ $\widehat{Maj_3}(\text{all others}) = 0$

Similarly for $Parity_3(x)$, $\quad$ $\widehat{Parity_3}(\{1,2,3\}) = 1$

$\widehat{Parity_3}(\text{all others}) = 0$

For trivial function $f(x) = -1$: $\quad \hat{f}(\emptyset) = -1$, $\hat{f}(\text{all others}) = 0$

What if we change the domain now —

For $x \in \mathbb{F}_2^n$, $\quad f: \mathbb{F}_2^n \to \mathbb{R}$

def $\chi_S : \mathbb{F}_2^n \to \{\pm 1\}$

$$\chi_S(x) = \prod_{i \in S} (-1)^{x_i}$$ (remember -1 maps to 1, 1 maps to 0)

Now $\chi_S(x)$ computes parity.

Let's look at evaluation as a matrix multiply —

Given coefficients $\hat{f}(\emptyset), \hat{f}(\{1\}), \ldots, \hat{f}(S), \ldots \hat{f}([n])$.

Want to find the evaluations $f(00\cdots0)$, $f(000\cdots1)$, $f(x)$, $\cdots$, $f(111\cdots1)$.

$$\begin{bmatrix} f(0\cdots0) \\ f(0\cdots1) \\ \vdots \\ f(11\cdots1) \end{bmatrix} = \begin{bmatrix} \chi_\emptyset(0\cdots0) & \cdots & \chi_{[n]}(0\cdots0) \\ & \vdots & \\ \chi_\emptyset(1\cdots1) & \cdots & \chi_{[n]}(11\cdots1) \end{bmatrix} \begin{bmatrix} \hat{f}(\emptyset) \\ \hat{f}(\{1\}) \\ \vdots \\ \hat{f}([n]) \end{bmatrix}$$

This is called the Hadamard Matrix $H_N[x,S]$

where $H_N[x,S] = \chi_S(x)$ ($N = 2^n$)

$$= \prod_{i\in S} (-1)^{x_i}$$

$$= (-1)^{\sum_{i\in S} x_i} = (-1)^{\sum_{i=1}^{n} x_i s_i}$$

where sum is in $\mathbb{F}_2$.

where $s_i = \begin{cases} 1 & \text{if } i \in S \\ 0 & \text{o/w.} \end{cases}$

Lets look at $n=2$.

$N = 2^2 = 4$

$$\Rightarrow H_4 = \begin{array}{c} \\ 00 \\ 01 \\ 10 \\ 11 \\ \uparrow \text{col 2} \quad \uparrow \text{col 1} \end{array} \begin{array}{c} \begin{array}{cccc} \emptyset & \{1\} & \{2\} & \{1,2\} \end{array} \\ \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \end{array}$$

H_N has a recursive structure

$$H_2 = \begin{pmatrix} +1 & +1 \\ +1 & -1 \end{pmatrix}$$

then $H_{2^n} = H_2 \otimes H_2 \otimes \cdots \otimes H_2$ (n times)

$= H_2 \otimes H_{2^{n-1}}$

where $\otimes$ is the Kronecker Product

For ex $\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \otimes \begin{bmatrix} 0 & 5 \\ 6 & 7 \end{bmatrix} = \begin{bmatrix} 1\begin{bmatrix} 0 & 5 \\ 6 & 7 \end{bmatrix} & 2\begin{bmatrix} 0 & 5 \\ 6 & 7 \end{bmatrix} \\ 3\begin{bmatrix} 0 & 5 \\ 6 & 7 \end{bmatrix} & 4\begin{bmatrix} 0 & 5 \\ 6 & 7 \end{bmatrix} \end{bmatrix}$

$= \begin{bmatrix} 0 & 5 & 0 & 10 \\ 6 & 7 & 12 & 14 \\ 0 & 15 & 0 & 20 \\ 18 & 21 & 24 & 28 \end{bmatrix}$

Thus this matrix vector product can be done in $O(n \log n)$ using a divide and conquer similar to FFT.

Properties: i) Two distinct cols have inner prod = 0.

ii) Inner prod of col with itself = N

$(-1 \times -1 = 1, \; 1 \times 1 = 1)$.

$\Rightarrow \frac{1}{\sqrt{N}} H_N$ is unitary. $\Rightarrow \frac{1}{N} H_N^T H_N = I$.

But this is also symmetric

$\Rightarrow H_N^T = H_N$.

Thus interpolation is also $\frac{1}{N} H_N$.

Rewrite $\hat{f}(s) = \frac{1}{N} \sum_{x \in \mathbb{F}_2^n} f(x) \cdot \chi_s(x)$

as $\hat{f}(S) = \mathop{\mathbb{E}}_{x \in_R \mathbb{F}_2^n} [f(x) \cdot \chi_S(x)]$

Proof of i) : TP: $\sum_{x \in \mathbb{F}_2^n} \chi_S(x) \cdot \chi_T(x) = 0$ for $S, T \subseteq [n]$, $S \neq T$.

$$\Rightarrow \text{TP: } \frac{1}{N} \sum_{x \in \mathbb{F}_2^n} \chi_S(x)\, \chi_T(x) = 0$$

$$= \mathop{\mathbb{E}}_{x \in_R \mathbb{F}_2^n} \chi_S(x) \cdot \chi_T(x) = 0$$

Now if $i \in S \cap T$ $(-1)^{x_i} \cdot (-1)^{x_i} = (-1)^{2x_i} = 1$.

$$\Rightarrow \mathbb{E}\left[\prod_{i \in S \oplus T} (-1)^{x_i}\right]$$

$$= \begin{cases} S \oplus T = \emptyset \Rightarrow S = T \text{ (contradiction)} \\ \text{else} \quad x \text{ is chosen UAR} \\ \Rightarrow \text{each bit is } 0/1 \text{ with prob } \frac{1}{2}. \\ \text{indep at random.} \\ \Rightarrow \prod_{i \in S \oplus T} \mathbb{E}[(-1)^{x_i}] = \pi\left(\frac{1}{2}(-1) + \frac{1}{2}(1)\right) \\ = 0. \end{cases}$$

Notation

For $f, g : \{\pm 1\}^n \to \mathbb{R}$

Then $\langle f, g \rangle = 2^{-n} \sum_x f(x) g(x)$

$$= \mathop{\mathbb{E}}_{x \in_R \mathbb{F}_2^n} [f(x)\, g(x)]$$

In this notation, $\hat{f}(S) = \langle f, \chi_S \rangle$

Now for eg if $S = \phi$,

$$\hat{f}(\phi) = \langle f, 1 \rangle = \mathop{\mathbb{E}}_{x \in_R \mathbb{F}_2^n}[f(x)]$$

i.e. the fourier coefficient gives the expected value of the boolean function.

Another cool formula -

$$\langle f, g \rangle = \langle \sum_S \hat{f}(S) \cdot \chi_S , \sum_T \hat{g}(T) \cdot \chi_T \rangle$$

$$= \mathop{\mathbb{E}}_x \left[\left(\sum_S \hat{f}(S) \chi_S \right) \cdot \left(\sum_T \hat{g}(T) \cdot \chi_T \right) \right]$$

$$= \sum_S \hat{f}(S) \, \mathbb{E}[\chi_S] \cdot \sum_T \hat{g}(T) \cdot \mathbb{E}[\chi_T]$$

$$= \sum_{S,T} \hat{f}(S) \cdot \hat{g}(T) \langle \chi_S, \chi_T \rangle$$

For $S \neq T$, $\langle \chi_S, \chi_T \rangle = 0$

$$\Rightarrow \langle f, g \rangle = \sum_S \hat{f}(S) \hat{g}(S).$$

"how similar" the fourier coeffs are tells us "how similar" the actual functions are!

Corollary: $\langle f, f \rangle = \mathop{\mathbb{E}}_x [f(x)^2] = \sum_{S \subseteq [n]} \hat{f}(S)^2$

Now $f(x)^2 = 1$ if $\{\pm 1\}^n \to \{\pm 1\}$

$\Rightarrow \mathbb{E}[f(x)^2] = 1.$

$$\Rightarrow \sum_{S \subseteq [n]} \hat{f}(S)^2 = 1. \quad (\text{sum of fourier coeff}^2 = 1).$$

Corollary: $\hat{f}(\phi) = \mathop{\mathbb{E}}_{x}[f(x)]$

$\Rightarrow \hat{f}(\phi)^2 = \mathop{\mathbb{E}}_{x}[f(x)]^2$

$\Rightarrow \mathop{\mathrm{Var}}_{x}[f(x)] = \sum_{S \neq \phi} \hat{f}(S)^2$

Applications

1) Social Choice. $f: \{\pm 1\}^n \to \{\pm 1\}$ can be thought of as a voting rule in an 2-candidate, n voter election.

For ex: obvious choice is the majority function.

In Analysis of Bool Funcs, there is something called Influence of the ith coordinate.

$$\mathrm{Inf}_i[f] = \Pr_{x \in_R \{\pm 1\}^n}[f(x_1 \cdots x_n) \neq f(x_1, \cdots, -x_i, \cdots x_n)]$$

Prop (no proof): $\mathrm{Inf}_i[f] = \sum_{S \subseteq [n] \atop i \in S} \hat{f}(S)^2$

can prove with ABFs

i) Arrow's Impossibility: only dictatorship gives desirable properties for more than 2 candidate

Idea: $\Pr_{\text{3 rand votes}}$ ["condorcets paradox" using f].

$$= \frac{1}{4} + \frac{3}{4} \sum_{S} \left(\frac{-1}{3}\right)^{|S|} \hat{f}(S)^2$$

ii) KKL Thm: For any 2 candidate voting rule that has the property that $\mathbb{E}[f(x)] = 0$ (not inherently biased towards one of the candidates) we can always find $o(n)$ fraction of the voters so if you bribe them you can fix the outcome of the election.

3. Let $X_1 \cdots X_n$ be iid ± 1 with prob $1/2$.

$p(x_1, \cdots, x_n) = a_0 + a_1 x_1 + \cdots + a_n x_n$ be deg 1

$Y = p(X_1 \cdots X_n)$

$$\Pr[|Y - \text{mean}(Y)| \geq t \cdot \text{StdDev}(Y)] \leq \exp(-t^2/2)$$

Using Analysis of Bool Funcs we can get a bound even for higher degree -

$$\leq c \exp(-t^{2/k}/\text{const})$$

Quantum

Great Idea: Use probabilistic computation to speed up det. algorithms.

Belief 1: I can improve at most by a polynomial. For ex: This is true if SAT needs circuits of $\Omega(2^n)$. (which is believed to be true).

Belief 2: SAT $\notin D(1.99^n)$ time even with a prob. algo.

Quantum Algorithms: More significant speedups.

1. Shor's Algorithm: can factor n-bit integers in $\tilde{O}(n^2)$ compared to $2^{\tilde{O}(n^{1/3})}$ in classical.

2. Grover Search: SAT $\in \tilde{O}(\sqrt{2}^n)$ time.

Here both beliefs are shattered when the power of quantum is added.

Basics

Let $N = 2^n$.

Consider a 'data vector' of size N

$$[f(00\cdots0), f(0\cdots1), \cdots, f(11\cdots1)]^T$$

Where $f: \{0,1\}^n \to \mathbb{C}$ is efficiently computable by a classical algorithm.

If I multiply by DFT_N or H_N, I get a 'coefficient vector' $[c(0\cdots0), \cdots, c(11\cdots1)]^T$

Now I can't actually see this vector. However I can sample from this vector with prob proportional to the coefficient^2.

QM Axioms

1. The state of a physical qubit is a unit vector in 2 dimensions.

   i.e. $\begin{bmatrix} a \\ b \end{bmatrix} \in \mathbb{C}^2$ with $|a|^2 + |b|^2 = 1$

this is often written as - $a|0\rangle + b|1\rangle$

here $|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$, $|1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ (Braket / Dirac notation)

a, b are called "Amplitudes".

Intuitive meaning —

The actual state the particle is in is a linear combination of the basic states: $|0\rangle$ and $|1\rangle$.

For this lecture Aplitudes are $f(0)$ and $f(1)$ s.t. 1 qubits state is $\begin{bmatrix} f(0) \\ f(1) \end{bmatrix}$ and $|f(0)|^2 + |f(1)|^2 = 1$

More generally the state of n qubits is a unit vector in 2^n dimensions.

$$[f(0\cdots0), \ldots, f(1\cdots1)]^T \in \mathbb{C}^{2^n}, \sum_{x \in \{0,1\}^n} |f(x)|^2 = 1.$$

Axiom 2: physical changes to Qubits $\equiv$ linear transformation.

i.e. for n qubit system to change we multiply by an $N \times N$ unitary matrix (maps unit vectors to unit vector).

for ex - $\frac{1}{\sqrt{2}} \begin{bmatrix} +1 & +1 \\ +1 & -1 \end{bmatrix}$ (which is both H_2, DFT_2)

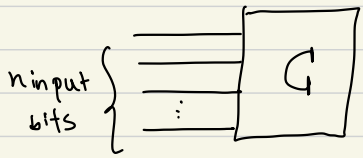
is unitary. and takes $\begin{bmatrix} a \\ b \end{bmatrix} \to \begin{bmatrix} \frac{a+b}{\sqrt{2}} \\ \frac{a-b}{\sqrt{2}} \end{bmatrix}$

another ex - $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ (rotate by 45°)

(quantum NOT)

takes $\begin{bmatrix} a \\ b \end{bmatrix} \to \begin{bmatrix} b \\ a \end{bmatrix}$. Means it takes $|0\rangle \to |1\rangle$ and vice versa.

Classical –



Is there a physical C that implements any $g: \{0,1\}^n \to \{0,1\}^n$?

Yes. But charge for # gates used.

Quantum –

Built from 1, 2 qubit gates that each do some unitary transforms.

Charge for # gates used.

Some Facts –

→ $\frac{1}{\sqrt{N}} H_N$ is implementible using n 1 qubit gates.

→ $\frac{1}{\sqrt{N}} DFT_N$ is implementible using n^2 1-2 qubit gates

Let  be a classical circuit of T gates.

Then it is easy to construct a quantum circuit of $O(T)$ gates.

Consider an $N \times N$ matrix with $-1^{C(00\cdots0)}, \ldots,$

$(-1)^{C(11\cdots1)}$ on the diagonal and 0 elsewhere.

Notiche when this is multiplied with $[f(0^n)\cdots f(1^n)]^T$ it negates some of these amplitudes.
(sum of squares is unaffected).
The amplitudes negated are the strings where C output 1.

Axiom 3: When you "measure" n qubits in the state $\sum_{x\in\{0,1\}^n} f(x)|x\rangle$ you get a classical string "x" with prob $|f(x)|^2$ and the state collapses.

For ex $\begin{bmatrix} 0.8 \\ 0.6 \end{bmatrix}$ then on measuring, with prob 0.64 we get the string 10.
and prob 0.36 we get string 01.

Grover's Search

SAT $\in \tilde{O}(\sqrt{2^n})$ time.

→ Circuit SAT: does this have an input that outputs 1.

Input: Boolean circuit C with n input, 1 output.
(assume C with poly(n) gates).

Assume: $C(x) = 1$ on unique string x^* OR C always outputs 0.

i.e. Focus: find x^* assuming it exists.

Brute Force: Try all x and check in poly(n).
$O(2^n \cdot \text{poly}(n))$

Grovers —

1. Allocate n qubits and initialize to $[1 0 0 \cdots 0]^T$
2. Apply $\frac{1}{\sqrt{N}} H_N$. Now state is $[1/\sqrt{N} \cdots 1/\sqrt{N}]^T$
3. Build QC, a quantum circuit based on the given circuit C.

   Notice this negates a single amplitude, the one corresponding to x^*.

Also notice I can't observe right now as all have equal prob $\frac{1}{\sqrt{N}}$ and so it can collapse to any state UAR.

4. 'Grover' steps —
   i) Apply $\frac{1}{\sqrt{N}} H_N$.
   ii) Apply quantumification of OR_N $\begin{bmatrix} +1 & & \\ & -1 & 0 \\ & 0 & \ddots \\ & & -1 \end{bmatrix}$
   iii) Apply $\frac{1}{\sqrt{N}} H_N$ again.

negates all coeffs except $f(0\cdots0) = f(\phi)$

Intuition: i) convert the state to its coefficients of multilinear polynomial representation (interpolation).
i.e. get $\hat{f}(\phi) \ldots \hat{f}([n])$.
ii) Negates all except $\hat{f}(\phi)$.

Now from boolean funcs, $\hat{f}(\phi)$ represents the $\mathop{\mathbb{E}}_{x \in_R \mathbb{F}_2^n}(f(x))$.

Now $f(x) = \hat{f}(\phi) - \hat{f}(s_1 3) \cdot (-1)^{x_0} - \ldots - \hat{f}([n]) \cdot \prod_i (-1)^{x_i}$

$= \mu - (f(x) - \mu)$

i.e. check how much above the average $f(x)$ was

and put it that much below the average, and vice-versa. So its sort of reflecting across the average.
Step 3 simply evaluates again.

Example: Consider the circuit with TT−

$N = 2^2 = 4$

Input	Out
00	0
01	0
10	1
11	0

After first 3 steps we have -

$$\text{state} = \left[\tfrac{1}{\sqrt{4}} \;\; \tfrac{1}{\sqrt{4}} \;\; -\tfrac{1}{\sqrt{4}} \;\; \tfrac{1}{\sqrt{4}}\right]^T$$

$$= \left[\tfrac{1}{2} \;\; \tfrac{1}{2} \;\; -\tfrac{1}{2} \;\; \tfrac{1}{2}\right]^T$$

Following the grover steps, $\mu = \frac{1}{4}\times\frac{1}{2} + \frac{1}{4}\times\frac{1}{2} + \frac{1}{4}\times\frac{1}{2} - \frac{1}{4}\times\frac{1}{2} = \frac{1}{4}$

$$\Rightarrow \text{state} = \left[0 \;\; 0 \;\; 1 \;\; 0\right]^T$$

Now if we measure, we get 10 with prob 1.
This is a satisfying assignment.

For the general case,

$\mu \approx \frac{1}{\sqrt{N}}$ $\Rightarrow$ All $x \neq x^*$ remain nearly same.
$f(x^*) \approx 3/\sqrt{N}$.

Now we can repeat the steps. $f(x^*)$ will drop to $-3/\sqrt{N}$ then to $5/\sqrt{N}$ then $-5/\sqrt{N}$ and so on.

We just have to repeat this till $f(x^*) \approx 0.1$

(then we can boost with repeated trials as needed).

$$\Rightarrow \frac{2k+1}{\sqrt{N}} \approx 0.1$$

$$\Rightarrow k \approx \frac{0.1\sqrt{N} - 1}{2} = O(\sqrt{N})$$

$$= O(\sqrt{2^n})$$
$$= O(\sqrt{2}^n)$$

Fields, Polynomials

Field: Usual ops: $+, -, \cdot, \div$. For ex - $\mathbb{R}, \mathbb{Q}, \mathbb{C}$. (not $\mathbb{Z}$, no division).

Polynomials: over a field $\mathbb{F}$ is an expr like $c_1 X_1^3 + c_2 X_2 X_3^5$ where $c_i \in \mathbb{F}$ and X_i are indeterminate.

$\mathbb{F}[X_1, \cdots X_r]$: all polynomials with coefficients from $\mathbb{F}$.

Facts:

- For prime p, ints mod p is a field $\mathbb{F}_p$.
- For $d \in \mathbb{N}^+$, prime p $\exists$ $\mathbb{F}_{p^d}$ of size p^d. that is unique.
- can construct them in polylog (q) time where q is # elements.

- deg d univariate polynomials have $\leq d$ roots.
- deg atmost d multivariate polynomials satisfy -

$$\Pr_{\text{fixed } a_1, \ldots, a_n}[P(a_1, \ldots, a_n) = 0] \leq \frac{d}{q}$$

Q Why is $\mathbb{F}_p$ a field. i.e. why is it closed under division?

A Find inverse using Extended Euclidean Algo.

EEA -

Given $b, p \in \mathbb{Z}$ finds $c, d \in \mathbb{Z}$ s.t $c \cdot b + d \cdot p = \gcd(b, p)$

For prime p this gives us $c \cdot b + d \cdot p = 1 \mod p$

$\Rightarrow c \cdot b = 1 \mod p$

$\Rightarrow c = b^{-1} \mod p$

Q How do you find large (n bit) prime no.s?

A 1. pick random no.

2. Run Rabin-Miller.

This works because the fraction of primes among n-bit no.s is $\tilde{\Omega}\left(\frac{1}{n}\right)$ (acc to the prime no. theorem).

$\Rightarrow$ with prob after $O(n)$ trials you will find it.

Non-Prime Fields -

Fact -

$\mathbb{F}_q = \mathbb{F}_{3^2} = \{a + bi \mid a, b \in \mathbb{F}_3\}$

Univariate Polynomials -

$F[X] = \{c_0 + c_1 X + \cdots + c_n X^n \mid c_i \in F\}$

Polynomials are generally not fields because no multiplicative inverse. (is a ring though).

Facts:

i) "Division"

$B(x) \div A(x) = Q(x)$ and remainder $R(x)$

i.e. $B(x) = Q(x) \cdot A(x) + R(x)$.

where $\deg(R) < \deg(A)$.

$\Rightarrow$ can do Euclids gcd.

ii) $P(x)$ is irreducible (prime) iff it can't be factored as a product of two lower deg polynomials.

iii) Thm: $\{F[x] \bmod P(x)\}$ is a field of size $|F|^{\deg(P)}$ if $P(x)$ is irreducible.

i.e. all polynomials in the field F with degree $\leq \deg(P)$

(works out combinatorially also: $|F|$ choices for each of $c_0, \cdots, c_{\deg(P)-1}$.).

For ex - x^2+1 is irreducible in $\mathbb{F}_3[x]$.

Then $\{\mathbb{F}_3[x] \bmod x^2+1\}$ is a field. ↳ all coeffs are $\{0,1,2\}$

i.e. $\{a+bx \mid a,b \in \mathbb{F}_3\}$.

For mult, $(a+bx)(a+bx) = a^2+2abx+b^2x^2$

$$x^2+1 \;\overline{)\; b^2x^2+2abx+a^2 \;}\; b^2$$

$$b^2x^2 + b^2$$

$$2abx + a^2 - b^2 \longrightarrow \text{result mod } x^2+1.$$

Notice this is same as setting $x^2 = -1$

This works in general also. Set the irreducible polynomial to 0.

Notice as a field this is (isomorphic?) to $\mathbb{F}_{|\mathbb{F}|^{\deg(p)}}$
$= \mathbb{F}_{3^2} = \mathbb{F}_q$.

in the beginning we said $\mathbb{F}_q = \{a+bi\}$
This is same as $\mathbb{F}_q = \{a+bx \mid x^2 = -1\}$

Cordlary – Given irreducible poly of deg d in $\mathbb{F}_p[x]$, arithmetic in $\mathbb{F}_{p^d}$ is doable in poly($d \log p$) time.
(d coeffs, each of $\log p$ bits.)

once you have the polynomial form FFT magic lets us do even mult efficiently.

How to get an irreducible poly of deg d?
i) Pick a random poly of deg d.
ii) Check irreducibility (doable in poly($d \log p$) time)

"Prime No. Thm" for irreducibles —
Pr [random poly irreducible] b/w $\frac{1}{2d}$ and $\frac{1}{d}$ for any d. $\Rightarrow O(d)$ trials needed to make this work whp.

Shoup: $\exists$ a poly(1,p) time deterministic algo to get an irred in $\mathbb{F}_p[x]$ of deg d.

Van Lint: x^2+x+1, x^6+x^3+1, $x^{18}+x^9+1$ ...
are all irreducible in $\mathbb{F}_2[x]$.

"Degree Mantra": Nonzero univariate polynomials of degree $\leq d$ has $\leq d$ roots.

Proof sketch: Keep factoring out irreducibles. This means you can factor out at most degree times. Any irreducible of the form $(x-\alpha)$ is a root.

Application: communication complexity.

A	B
$a \in \{0,1\}^n$	$b \in \{0,1\}^n$

Want to compute $f(a,b)$ by communicating min # bits.

For ex: equality? i.e. $f(a,b) = 1$ if $a = b$.

In fact any det. algo need at least $n+1$ bits of communication.

Modification: Allow one-sided error $\leq \frac{1}{n}$.
Claim: $O(\log n)$ bits suffice.

Idea: A sends hash(a) to B which is of smaller length.
B computes hash(b) and compares with hash(a).

Protocol –
i) A fixes $\mathbb{F}_q$ s.t. $n^2 \leq q \leq 2n^2$
ii) A sends q to B ($\log q = 2 \log n$ bits)

iii) Alice forms $P_A(x) = a_n x^n + \cdots a_1 x$
B forms $P_B(x) = b_n x^n + \cdots + b_1 x$

Since these are bit strings they are valid coeffs in any field.

Goal: is $P_A - P_B = 0$

iv) Alice chooses $\alpha \in \mathbb{F}_q$, computes $P_A(\alpha)$ and sends $\alpha, P_A(\alpha)$ to B.

v) B computes $P_B(\alpha)$ and $P_A(\alpha) - P_B(\alpha)$. If not 0 return not-equal.
Prob that it is wrong (not equal) despite this is if α is a root of $P_A - P_B$. This happens with prob $\frac{1}{n}$.

Polynomials vs Function computed by the polynomial —
→ Every polynomial computes a func $\mathbb{F}_q^n \to \mathbb{F}_q$ (n variate poly)
→ Every function $f: \mathbb{F}_q^n \to \mathbb{F}_q$ is computed by a poly.

$\exists\ q^{q^n}$ such functions, but there are infinitely many polynomials (degree can be arbit).

⇒ ∃ functions can be computed by multiple poly

⇒ ∃ non-zero polynomials that compute $f = 0$.

for ex - $X^2 - X$ in $\mathbb{F}_2[X]$. always outputs 0 for any input in $\mathbb{F}_2$.

In fact $X^q - X$ in $\mathbb{F}_q[X]$ also always outputs 0 for inputs in $\mathbb{F}_q$.

(For primes this is Fermat's little thm).

($a^p - a = a(a^{p-1} - 1) \equiv 0 \mod p$)

Even stronger : In $\mathbb{F}_q$ X^q always computes X.

Cor: Any poly $P \in \mathbb{F}_q[X_1, \cdots X_n]$ can be reduced s.t. $X_i^m \to X_i^{(m \bmod (q-1))}$ without changing the function that it computes.

i.e. the individual degree $< q-1$.

$\Rightarrow$ # reduced monomials (# terms.) : q^n (q possible powers for each X_i)

$\Rightarrow$ # reduced polynomials : $q^{q^n} = \#$ funcs.
(q possible coeffs for each term)

Schwarz - Zippel

Let $P \in \mathbb{F}_q[X_1, \cdots X_n]$ be a reduced, non-zero polynom with tot deg $\leq d$. then,

$$\Pr_{\substack{\alpha_1 \cdots \alpha_n \\ \in \mathbb{F}_q}}[P(\alpha_1, \cdots \alpha_n) \neq 0] \geq \text{func}(q, d)$$

Case i) $q = 2$: $\text{func}(q, d) = \frac{1}{2^d}$

Case ii) $q > d$: $\text{func}(q, d) = 1 - \frac{d}{q}$ → small if $q >> d$.

$$\Rightarrow \Pr[P(\alpha) = 0] \leq \frac{d}{q}$$

Even stronger : If $S \subseteq \mathbb{F}$ and $\alpha_i \in S$

$$\Pr[P(\alpha) = 0] \leq \frac{d}{|S|}$$

general case: $\text{func}(q, d) = \frac{1}{q^{\lfloor \frac{d}{q-1} \rfloor}} \left(1 - \frac{d \bmod q-1}{q}\right)$

Application: Finding perfect matching in Bip Graphs.

Open: In NC?

Lovasz - can do it in randomized NC.

Error Correcting Codes

ECC is an injective map $Enc: \Sigma^k \to \Sigma^n$ where Σ - alphabet

$q = |\Sigma|$

k = message length

n = block length

C = range(Enc) known as the code.

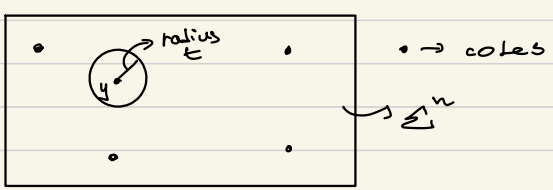
$y \in C$ are codewords.

$\frac{k}{n}$ = rate of code.

Adv Model: at most upto t errors can happen.

Err Model: errors "corrupt" one symbol to another.

Let: $\Delta(y,z) = \#\{i \mid y_i \neq z_i\}$



Now the receiver gets z, a corruption of y. z can be decoded to y as long as all these balls are disjoint.

$\Rightarrow \Delta(y,y') > 2t \quad \forall\, y, y' \in C$

Min Dis of a code C $= d = \min_{y \neq y' \in C} \Delta(y,y')$

Fact: unique decoding possible iff $t \leq \left\lfloor \frac{d-1}{2} \right\rfloor$

Idea: choose $C \subseteq \Sigma^n$ randomly.
pro: good rate vs dist tradeoff
con: Not efficient to encode/decode.

Def: Linear codes.
$\Sigma = \mathbb{F}_q$, $Enc: \mathbb{F}_q^k \to \mathbb{F}_q^n$ is a linear transform.

$$\underbrace{[\ x\ ]}_{k} \begin{bmatrix} & G & \end{bmatrix}_{k} = \underbrace{[\ y\ ]}_{n}$$

$\begin{bmatrix} & G & \end{bmatrix}$: n columns

C = rowspace(G).
i.e. a k-dim subspace of $\mathbb{F}_q^n$.
↳ (assuming each row indep).

For linear codes, encoding is efficient since it is a matrix multiply.

The general decoding problem is NP Hard, but for some codes this process is in P.

Notation: $[n,k]_q$ or $[n,k,d]_q$ represents a linear code $\mathbb{F}_q^k \to \mathbb{F}_q^n$ having min distance d.

For k-dim subspace C, define –

$$C^\perp = \{ w \in \mathbb{F}_q^n \mid w \cdot y = 0, \forall y \in C \}$$

Exc: $C^\perp$ is a subspace of $\mathbb{F}_q^n$ of dimension $n-k$.

proof: i) $0^n \in C^\perp$, since $0 \cdot x = 0 \ \forall x$.

ii) for $a, b \in C^\perp$ then $a+b \in C^\perp$ since

$$(a+b)\cdot w = a\cdot w + b\cdot w = 0 + 0 = 0.$$

iii) for $a \in C^\perp$, $\lambda a \in C^\perp$ since $\lambda a \cdot w = \lambda 0 = 0$.

Now $C^\perp$ is an $[n, n-k]_q$ code. (Dual code to C)

i.e. $Enc^\perp : \mathbb{F}_q^{n-k} \to \mathbb{F}_q^n$

$x \mapsto xH$ where

H is an $(n-k) \times n$ matrix called the parity check matrix for C.

$$\underset{n-k}{\updownarrow}\underbrace{\begin{bmatrix} & H & \end{bmatrix}}_{n}\begin{bmatrix} z \end{bmatrix}n = 0, \text{ iff } z \in C.$$

Facts - For a linear code C, $d(C)$ = min hamming weight.

- $d(C)$ = min # cols of H that are linearly dep.

Proof: $d(C) = \min\{ \overset{\text{ham wt.}}{wt(z)} \mid z \in C, z \neq 0\}$

$= \min\{ wt(z) \mid Hz = 0, z \neq 0\}$

z with lowest hamming wt w will cause Hz to be a linear combination of w columns of H.

But $Hz = 0$

$\Rightarrow$ these w columns are lin-dep.

As long as a col is not all zeros, the qty is atleast 2.
As long as two cols are not the same, the qty is atleast 3 [for $q=2$, replace same with scalar multiple for $q>2$].

$\Rightarrow$ we can have dist at least 3. If we want high rate, then $n-k$ is small.
↳ height of H.

Using this idea, the hamming code —
Defined as the code whose parity check matrix

$$H = \begin{bmatrix} 0 & 0 & \cdots & 1 \\ 0 & 0 & \cdots & 1 \\ \vdots & \vdots & & \vdots \\ 1 & 0 & \cdots & 1 \end{bmatrix}$$

r ↓ no. of rows

$n = 2^r - 1$

is a $[2^r, 2^r - r - 1, 3]_2$ code.
(n, $n - \log n$)

super high rate, but $d = 3$ $\Rightarrow$ can only correct $\left\lfloor \frac{3-1}{2} \right\rfloor = 1$ errors.

If y is a codeword, and the ith bit is flipped.
Then multiplying with parity check matrix gives —

$$H(y + I_i) = Hy + HI_i = i\text{th column of } H$$

→ This tells us exactly which bit was corrupted: (i in base 2).
→ This is also a "perfect code". i.e. the hamming balls partition all of space. This means any string is either in C or at distance 1 from some $y \in C$.

Dual of the Hamming code : Hadamard

Use H as the generator matrix.

i.e. Enc: $x \mapsto (x \cdot a)_{a \in \mathbb{F}_2^r}$

Defn: For $x \in \mathbb{F}_2^r$, define $L_x : \mathbb{F}_2^r \to \mathbb{F}_2$ by

$L_x : a \mapsto x_1 a_1 + \cdots + x_r a_r$

This is a 1 deg polynomial with coeffs x and variable a.

Now the hadamard code can be thought of as listing the truth table over all evaluations of L_x. (only the outputs)

This is a $[2^r, r, 2^r/2]_2$ code.

Why is $\delta = \frac{n}{2}$? Acc to Schwarz-Zippel, Pr of input being a root (of poly L_x) $\leq \frac{d}{q}$. Here $d = 1$ and $q = 2$.

$\Rightarrow \leq \frac{1}{2}$ the inputs give 0 in the truth table.

$\Rightarrow$ min hamming wt can have atmost $\frac{n}{2}$ 0s.

Exc: Generalize Hadamard codes to $[q^r, r, (1 - \frac{1}{q})q^r]_q$

Reed Solomon codes—

For $1 \leq k \leq n$, $q \geq n$ and $S \subseteq \mathbb{F}_q$, $|S| = n$

Enc: $\mathbb{F}_q^k \to \mathbb{F}_q^n$ works as follows—

Think of message m as the univariate polynomial with coeffs m_i. i.e. $m_0 + m_1 x_1 + \cdots + m_{k-1} x^{k-1}$

Let $Enc(m) \longmapsto m(s) \quad \forall s \in S$

i.e. evaluating the poly in n points.

this is a linear code: $\begin{bmatrix} 1 & s_{11}^1 & s_{12}^2 & \cdots & s_{1k-1}^{k-1} \\ & & \vdots & & \\ 1 & s_{n1}^1 & s_{n2}^2 & \cdots & s_{nk-1}^{k-1} \end{bmatrix}^T$ is G.

Claim: $d \geq n-(k-1)$

Proof: $d = \min$ hamm wt. $\{$(# evaluations that gives 0)$\}$ $n -$

$\geq n-(k-1)$ (at most $k-1$ roots for deg $k-1$)

$\Rightarrow$ RS is a $[n, k, n-k+1]_q$ code.

Fact: This the best possible rate-distance tradeoff for $q \geq n$.

singleton Bound: For $(n,k,d)_q$ code ($\nearrow$ round bracket $\Rightarrow$ non necessarily linear.) $k \leq n-d+1$

shows its tight.

Asymptotically "good" codes.

Goal! Fixed q.

Let $C = (C_n)$ be a sequence of $[n, k(n), d(n)]_q$ codes.

Asymptotic Rate of a family is $\lim_{n\to\infty} \frac{k(n)}{n} = R(C)$

Asymptotic relative distance is $\lim_{n\to\infty} \frac{d(n)}{n} = \delta(C)$

For ex: Hamming code is $[n, n-\log n, 3]_2$

$$R(C) = \lim_{n\to\infty} \frac{n-\log n}{n} = 1$$

$$\delta(C) = \lim_{n\to\infty} \frac{3}{n} = 0$$

Hadamard code $[n, \log n, n/2]_2$

$$R(C) = \lim_{n\to\infty} \frac{\log n}{n} = 0$$

$$\delta(C) = \lim_{n\to\infty} \frac{1}{2}\frac{h}{n} = \frac{1}{2}$$

For Rs, it doesn't really count since q grows with n but think of it as $R(C) = R_0$ and $\delta(C) = 1 - R_0$.

∴ a code is asymptotically good if q fixed, $R(C) \geq R_0 > 0$
$\delta(C) \geq \delta_0 > 0$

These actually exist. Gilbert-Varshamov bound -

$\forall q \quad \forall\, 0 \leq \delta_0 \leq 1 - \frac{1}{q}$

$\exists C$ with $\delta(C) = \delta_0$

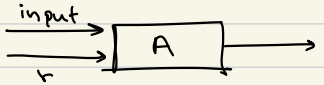
$$R(C) = 1 - h_q(\delta_0)$$

↳ q-arry entropy.

Thm: For $q=2$, $\exists$ poly(n) time encodable and decodable asymptotically good codes.

PRGs

The model: consider a randomized algorithm that takes in input and a random string r and outputs a decision yes/no.



A solves the problem (in BPP) if -

i. A is efficient

ii. $\forall x \quad \Pr_r[A(x,r) \text{ is correct}] \geq \frac{3}{4}$

Let $|r| = n$.

Naive Derandomize: Try all 2^n strings. We know at least $\frac{3}{4}$th of them will work correctly. Choose one of these.

Defn: PRGs: Let C be a class of functions $f: \{0,1\}^n \to \{0,1\}$ (known as "tests"). $G: \{0,1\}^\ell \to \{0,1\}^n$ is called an ε-PRG if

i) $n > \ell$

ii) $\forall f \in C, \quad \left| \Pr[f(r \in_R \{0,1\}^n) = 1] - \Pr[f(G(s \in_R \{0,1\}^\ell)) = 1] \right| \leq \varepsilon$

This is called "G ε-fools C"

Ex: Say $C = \{ f: \{0,1\}^n \to \{0,1\}$ computable by bool circuits of size $\leq n^{10} \}$

Notice that if $\forall x$ $A_x(r)$ runs in time $< \frac{n^{10}}{\log n}$ (approx) then it also has such a circuit representation.

Say G $\varepsilon = 0.24$ fools this C with $\ell = \log n$

-) can solve A's problem deterministically in poly time. How? Run the algo using all possible seeds. (n such seeds). with prob $\geq \frac{3}{4}$ the randomized A accepted correctly.

⇒ the algo with the PRG string accepts correctly with prob $\geq \frac{3}{4} - \varepsilon = 0.51$

⇒ The answer given by majority of the seeds will be correct.

Hardness vs Randomness –

Consider seed s of length around $\log n$
choose n subsets of the bits in the string
Now SAT is a function (that is hard) that maps strings to bits.
Apply SAT on each of these subsets to get the n bits.
(suppose the bits represented a ϕ bit is 1 if satisfiable 0 o/w)

Impagliazzo – Wigderson –
Suppose $h: \{0,1\}^* \to \{0,1\}$ where h is some hard to compute function (like SAT), such that –
i. computable in $2^{O(n)}$ time by TMs
ii. not computable by $\leq 2^{O(n)}$ sized circuits. (for suff large n)

Then BPP = P.

Most Derandomization today: Unconditional but for a specific problem. (doesn't rely on this strong CT assumption).

Nisan's PRG: $\exists$ PRG G that ε-fools randomized TM using n random bits, $s(n) = s$ space ($s(n) \geq \log n$) with $\varepsilon = 2^{-s}$ and $l = O(s \log n)$ computable in $O(l)$ space, $2^{O(l)}$ time.

Def: $G: \{0,1\}^d \to \{0,1\}^n$ is pairwise independent if $\forall i \neq j$

$$\Pr_{s \in_R \{0,1\}^d}[G(s)_i\, G(s)_j = 00] = \frac{1}{4}$$
$$01 = 11$$
$$10 = 11$$
$$11 = 11$$

Its a misnomer: it actually should be pairwise uniform.

More generally it is called k-wise independent, if for all k indices takes all $|\Sigma|^k$ outputs uniformly.

Thm: $\forall k \leq n$, $\forall$ prime powers q, $\exists$ poly(n) time computable, k-wise indep PRG with seed length $d = \lfloor \frac{k}{2} \rfloor \log_q n + O(1)$.

For ex- if $q=2$, $k=2$, $d = O(\log n)$

Ex Appl: $\frac{1}{2}$-approximating Max-Cut: Give $G(V,E)$, partition $V = (R, \bar{R})$ s.t. #edges from R to $\bar{R}$ max.

simple Algo: Pick a random partition.

$X_i = 1$ if exactly one end colored.

$$\mathbb{E}[X_i] = \frac{2}{4}$$

$$\Rightarrow \mathbb{E}[X] = \frac{|E|}{2}$$

Notice for $\frac{2}{4}$ I only needed pairwise independent coin flips.

Replace the random string with one generated by G that is 2-wise indep. By thm this can be

done with $l = O(\log n)$.

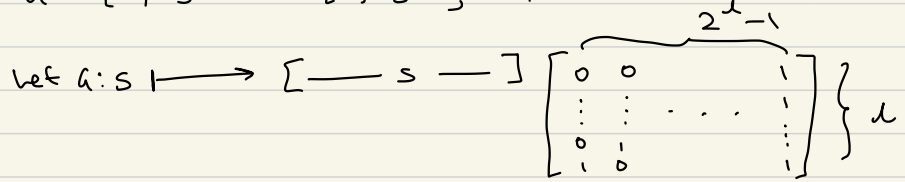
Try all n seeds. At least one of them should give a cut $\geq \frac{|E|}{2}$.

Again, Chebyshev Inequality, and variance remains the same if the bits are only pair-wise indep.

Doesn't work for Chernoff however.

Proof for $k=2$, $q=2$. (constructive).

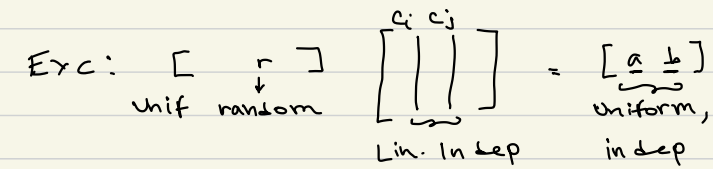
$G : \{0,1\}^l \to \{0,1\}^n$, $n = 2^l - 1$



Claim: this output is pairwise indep.

Proof: consider bits i and j.

This is $\langle s, \text{col } i \rangle$ and $\langle s, \text{col } j \rangle$



Try: $\Pr(a=0, b=0) = \Pr[\langle c_i, r\rangle = 0, \langle c_j, r\rangle = 0]$

12k.

Remember distance of code = min # dep cols (of the parity check matrix).

Min Distance of Hamming > 2.

⇒ No pair of columns are dependant.

Generalization: Good codes (with large min dist) give good k-wise generators. (the dual).

RS codes ⇒ PRGs that are k-wise indep and $l = k \log_2 n$

Epsilon-Bias Generators.

$G: \mathbb{F}_2^l \to \mathbb{F}_2^n$ is an ε-biased generator if it ε-fools all linear $f: \mathbb{F}_2^n \to \mathbb{F}_2$.

↳ $f(x_1, \cdots, x_n) = c_1 x_1 + c_2 x_2 + \cdots + c_n x_n$

i.e. the transform matrix is $[c_1 \cdots c_n]$

In other words, $\forall w \in \mathbb{F}_2^n$, $w \neq 0$

$$\left| \Pr_{s \in_R \mathbb{F}_2^l} [\langle w \cdot G(s) \rangle = 1] - \Pr_{r \in_R \mathbb{F}_2^n} [\langle w \cdot r \rangle = 1] \right| \leq \frac{\varepsilon}{2}$$

$\Pr_{r \in_R \mathbb{F}_2^n} [\langle w \cdot r \rangle = 1] = \frac{1}{2}$

$$\Rightarrow \Pr_{s \in_R \mathbb{F}_2^l} [\langle w \cdot G(s) \rangle = 1] \in \left[\frac{1}{2} - \frac{\varepsilon}{2}, \frac{1}{2} + \frac{\varepsilon}{2} \right]$$

Thm: $\exists$ efficient computable ε-biased gens with $l = O(\log n/\varepsilon)$.

In fact it has been shown that $2 \log\left(\frac{n}{\varepsilon}\right)$ is achievable.

$\Rightarrow$ Enumerating over all possible seeds takes at most quadratic time $O\left(\frac{n^2}{\epsilon^2}\right)$

Even $O\left(\frac{n}{\epsilon^2}\right)$ seeds is possible (which is also a lower bound)

Ex Appl: verifying matrix mult.

Is $AB = C$? WLOG $A, B, C \in \mathbb{F}_2^{n \times n}$.

Choose random vector. Check $(A(By)) = Cy$. $O(n^2)$.

If $AB = C$, $ABy = Cy$

If $AB \neq C$

let $D = AB - C$. has ≥ 1 non-zero row. (say d).

Consider $\Pr_{y \in_R \mathbb{F}_2^n}[\langle d \cdot y \rangle \neq 0] = \frac{1}{2}$

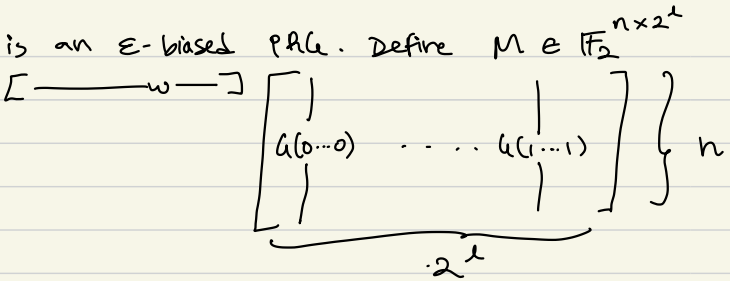
If there are more rows, this prob only reduces further since its an AND.

if y was not random but rather an output of an ϵ-biased PRG, then $\Pr[\langle d \cdot y \rangle \neq 0] \in \left[\frac{1}{2} - \frac{\epsilon}{2}, \frac{1}{2} + \frac{\epsilon}{2}\right]$

Thus we've reduced random bits needed to $O(\log n)$ while still having $O(n^2)$ run time.

ϵ-biased PRG $\leftrightarrow$ Good Binary Codes -

Say G is an ε-biased PRG. Define $M \in \mathbb{F}_2^{n \times 2^l}$



By defn of ε-biased gen, with prob $\left[\frac{1}{2} - \frac{\varepsilon}{2}, \frac{1}{2} + \frac{\varepsilon}{2}\right]$ each bit in the result is 0

$\Rightarrow$ wM has b/w $\frac{1}{2} - \frac{\varepsilon}{2}$ to $\frac{1}{2} + \frac{\varepsilon}{2}$ 1s.

$\Rightarrow$ frac Ham wt $\geq \frac{1}{2} - \frac{\varepsilon}{2}$ $\leq \frac{1}{2} + \frac{\varepsilon}{2}$

$\Rightarrow$ M is a generator for bin code with min (frac) distance $\frac{1}{2} - \frac{\varepsilon}{2}$

Spectral Graph Theory

Undirected Graph.
- finite
- parallel edges / self loops are okay.
- No vertices of degree 0.

consider $f: V \to \mathbb{R}$.
For ex - temperature, voltage etc, 0/1 indicators.

f can be represented as a vector.

$\Rightarrow$ All such vectors together form a vector space.

Key Insight in Spectral GT -

$$\mathop{\mathbb{E}}_{(u,v)\in_R E}\left[(f(u)-f(v))^2\right] \times \frac{1}{2}.$$

This quantity is notated by $\mathcal{E}[f]$.
and is called the Dirichlet / Local variance / Analytic boundary size / Laplacian quadratic form.

Facts -
i) Always ≥ 0.
ii) $\mathcal{E}[cf] = c^2\,\mathcal{E}[f]$
iii) $\mathcal{E}[f+c] = \mathcal{E}[f]$

Intuition : small $\mathcal{E}$ $\Leftrightarrow$ f's values don't vary too much along the edges.

Important ex - Let $S \subseteq V$. Let $f(u) = \begin{cases} 1, & u \in S \\ 0, & o/w \end{cases}$

Now $\mathcal{E}[f] = \frac{1}{2}\mathop{\mathbb{E}}_{(u,v)}\left[(f(u)-f(v))^2\right]$

$= \frac{1}{2}\mathbb{E}\left[\{\text{edges crossing the cut}\}\right]$

$= \mathbb{E}\left[\{\text{edges from } S \text{ to } \bar{S}\}\right]$

How to choose a random vertex?
→ choose a UAR edge (u,v)
→ output u.

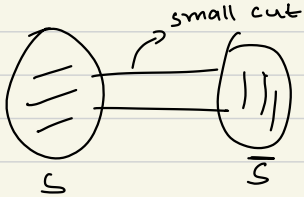
Call this distribution π.
Notice $\pi(u) = \frac{\deg(u)}{2|E|}$

Notice if G is regular, this is uniform distr.

Fact: Doing a random walk with trans prob $\frac{1}{\deg(u)}$ will give me a vertex having this distr.

Q: How fast does this converge?
A: Spectral!

Eg: What is "almost" disconnected.



let $f(u) = \begin{cases} 1, & u \in S \\ 0, & o/w \end{cases}$

We will see that Fast convergence iff $\mathbb{E}[f]$ being never small.

let $f: V \to \mathbb{R}$
For $u \in_\pi V$, $f(u)$ is a r.v.

$$\text{mean} = \mathop{\mathbb{E}}_{u \in_\pi V}[f(u)] = \mathbb{E}[f]$$

For $S \subseteq V$, f = Indicator 1_S,

$$= \Pr[u \in S]$$

= "weight" of S.

$$\text{variance} = \mathbb{E}[f(u)^2] - \mathbb{E}[f(u)]^2$$

$$\text{Claim} = \frac{1}{2} \mathop{\mathbb{E}}_{\substack{u \in_\pi V \\ v \in_\pi V \\ \text{indep}}} [(f(u) - f(v))^2]$$

Proof: $= \frac{1}{2} \mathbb{E}\left[f(u)^2 + f(v)^2 - 2f(u)f(v)\right]$

$= \frac{1}{2} \mathbb{E}[f(u)^2] + \mathbb{E}[f(v)^2] - 2\mathbb{E}[f(u)f(v)]$ (linearity)

$= \frac{1}{2}\left(2\mathbb{E}[f(u)^2] - 2\,\mathbb{E}[f(u)] \cdot \mathbb{E}[f(v)] \right)$ (independence)

$= \mathbb{E}[f(u)^2] - \mathbb{E}[f(u)]^2$

Hence $\mathrm{Var}[f] = \frac{1}{2} \mathop{\mathbb{E}}_{\substack{u \in_\pi V \\ v \in_\pi V \\ \text{indep}}} \left[(f(u) - f(v))^2\right]$

$\mathcal{E}[f] = \frac{1}{2} \mathop{\mathbb{E}}_{(u,v) \in_R E} \left[(f(u) - f(v))^2\right]$ ("local" variance of f)

Defn: Let f, g be functions $V \to \mathbb{R}$.

Let $\langle f, g \rangle_\pi = \mathop{\mathbb{E}}_{u \in_\pi V} [f(u) \cdot g(u)]$

Thm: This is a valid vector space inner product -

i. $\langle f, g \rangle = \langle g, f \rangle$

ii. $\langle cf + g, h \rangle = c\langle f, h \rangle + \langle g, h \rangle$ (linearity)

iii. $\langle f, f \rangle = \mathbb{E}[f(u)^2] \geq 0.$
with equality iff $f = 0$

Remark - For $S \subseteq V$, $f = 1_S$

Then $\|f\|_1 = \mathop{\mathbb{E}}_{u \in_\pi V} [|f(u)|]$

$= \mathbb{E}[f(u)]$ (f is 0/1 rv)

$= \Pr[u \in S]$.

$= \mathbb{E}[f(u)^2] = \|f\|_2^2$

Q: How small can $\mathcal{E}[f]$ be?

A: 0.

Q: Is there a non-trivial f with $\mathcal{E}[f] = 0$.

Trivial: Notice if $f = 0$ then $\mathcal{E}[f] = 0$.

$\Rightarrow$ for all $f + c$, $\mathcal{E}[f+c] = \mathcal{E}[f] = 0$

Prop: $\mathcal{E}[f] = 0$ iff f is constant on each connected component of f. and,

components of G = # lin. indep f with $\mathcal{E}[f] = 0$

2nd part is simple, WLOG assume $\{S_1, S_2, S_3$ are 3 components.

Now for ex look at $1_{S_1}, 1_{S_2}, 1_{S_3}$

$\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 1 \end{bmatrix}$ are clearly indep.

by part 1 these are of the form $\sum_i c_i 1_{S_i}$

Q: How big can $\mathcal{E}[f]$ be s.t. $\mathrm{Var}[f] \le 1$.

OR Max $\mathcal{E}[f]$ s.t. $\|f\|_2^2 = \mathbb{E}[f^2] \le 1$.

these two are equivalent since $\mathrm{Var}[f] = \mathbb{E}[f^2] - \mathbb{E}[f]^2$

$\Rightarrow \mathbb{E}[f^2] = \mathrm{Var}[f] + \mathbb{E}[f]^2$.

-> any f satisfying $\|f\|^2 \leq 1$ satisfies $\text{Var}[f] \leq 1$

Now notice adding subtracting constant doesn't change $\mathcal{E}[f]$. => I can make mean $[f] = 0$ while maintaining same $\mathcal{E}[f]$, so that $\mathbb{E}[f]^2 = 0$. => $\text{Var}[f] = \|f\|_2^2$.

Intuition to maximize — embed vertices onto number line keeping edge end points as far away as possible.

This intuition suggests bipartite graphs.

For $G = (V_1 \cup V_2, E)$ let $f = 1_{V_1} - 1_{V_2}$

$$\text{i.e. } f(u) = \begin{cases} +1, & u \in V_1 \\ -1, & u \in V_2 \end{cases}$$

Notice $\mathbb{E}[f^2] = \mathbb{E}[1] = 1$

also see that $\mathcal{E}[f] = \frac{1}{2}\mathbb{E}[(f(u) - f(v))^2]$

$$= \frac{1}{2}\mathbb{E}[(-1-1)^2]$$

$$= \frac{1}{2} \cdot 4 = 2.$$

Prop: $\mathcal{E}[f] \leq 2\|f\|_2^2 = 2\mathbb{E}[f^2]$

$$= \frac{1}{2}\mathop{\mathbb{E}}_{(u,v)}[(f(u) - f(v))^2]$$

$$= \frac{1}{2}\left[\mathop{\mathbb{E}}_{u \in_\pi V}[f(u)^2] + \mathop{\mathbb{E}}_{v \in_\pi V}[f(v)^2] - 2\mathop{\mathbb{E}}_{(u,v)}[f(u)f(v)]\right]$$

by Cauchy-Schwarz,

$$-\sqrt{\mathbb{E}[f(u)^2]\,\mathbb{E}[f(v)^2]} \leq \mathop{\mathbb{E}}_{(u,v)}[f(u)f(v)] \leq \sqrt{\mathbb{E}[f(u)^2]\,\mathbb{E}[f(v)^2]}$$

$$\Rightarrow \quad \mathcal{E}[f] \leq \mathop{\mathbb{E}}_{u \in_\pi V}[f^2(u)] - \left(-\sqrt{\mathop{\mathbb{E}}_{u\in_\pi V}[f^2(u)]\mathop{\mathbb{E}}_{u\in_\pi V}[f^2(u)]}\right)$$

$$\leq 2 \mathop{\mathbb{E}}_{u\in_\pi V}[f^2(u)]$$

Exc: Equality iff G is bipartite.

More Spectral GT

From before Cauchy-Schwarz above,

$$\mathcal{E}[f] = \|f^2\|_2^2 - \mathop{\mathbb{E}}_{(u,v)}[f(u)\cdot f(v)]$$

$$\mathop{\mathbb{E}}_{u\in_\pi V}\mathop{\mathbb{E}}_{(u,v)}[f(u)f(v)] = \mathop{\mathbb{E}}_{u}\, f(u)\mathop{\mathbb{E}}_{(u,v)} f(v)$$

Define $Kf: V \to \mathbb{R}$ where $Kf(u) = \mathop{\mathbb{E}}_{(u,v)} f(v)$

(avg value of f among u's neigh)

Observe that $K(f+g)(u) = \mathop{\mathbb{E}}_{(u,v)}(f+g)$

$$= \mathop{\mathbb{E}}_{(u,v)} f(v) + g(v)$$

$$= \mathop{\mathbb{E}}_{(u,v)} f(v) + \mathop{\mathbb{E}}_{(u,v)} g(v)$$

$$= Kf(u) + Kg(u)$$

Def: K is the Markov/transition operator (matrix) for G.

This matrix can be defined as —

$$K[u,v] = \begin{cases} 1/\deg[u] & \text{if } (u,v) \in E \\ 0 & \text{o/w} \end{cases}$$

Notice that this is the transition matrix for the graph random walk.

i.e. the adj mat normalized s.t. each row sum = 1
for regular $K = \frac{1}{d} A$ and K is symmetric.

fact: For $f, g : V \to \mathbb{R}$ $\quad \langle f, Kg \rangle = \mathop{\mathbb{E}}_{(u,v)} [f(u)\, g(u)]$

pf:

$$\text{LHS} = \mathop{\mathbb{E}}_{u \in_\pi V} f(u) \cdot \mathop{\mathbb{E}}_{(u,v)} g(v) = \sum_u \pi(u)\, f(u) \sum_{(u,v)} \frac{1}{\deg(u)} g(v)$$

$$= \sum_{\substack{(u,v) \\ \in E}} \frac{\pi(u)}{\deg(u)} f(u) g(v)$$

$$= \mathop{\mathbb{E}}_{(u,v) \in_R E} [f(u)\, g(u)]$$

cor: $\langle f, Kg \rangle = \langle Kf, g \rangle$
$\Rightarrow$ K is self adjoint.

What is this quantity? Suppose f and g were indicators for sets S and T. This quantity gives the prob of an edge crossing the cut.

Now $\quad \mathcal{E}[f] = \langle f, f \rangle - \langle f, Kf \rangle$

$$= \mathop{\mathbb{E}}_{u \in_\pi V} f(u) f(u) - f(u)\, Kf(u)$$

$$= \mathop{\mathbb{E}}_{u \in_\pi V} f(u)\left(f(u) - Kf(u)\right)$$

$$= \langle f, f - Kf \rangle = \langle f, If - Kf \rangle$$

$$= \langle f, \underbrace{(I-K)}_{L} f \rangle$$

Defn: $L = I - K$ is the normalized Laplacian operator for G.

If G was d-regular, $L = I - \frac{1}{d}A = \frac{1}{d}(dI - A)$

Now $Lf : V \to \mathbb{R}$ $\quad Lf(u) = f(u) - \mathop{\mathbb{E}}_{(u,v)}[f(v)]$

Sparsest Cut: $S \subseteq V$, $f = 1_S$

$\langle f, Lf \rangle = \mathcal{E}[f] = \Pr[u \in S, v \notin S]$ (frac edges crossing)

$\langle f, f \rangle = \Pr[u \in S]$ (vol of S)

Ratio: $\frac{\langle f, Lf \rangle}{\langle f, f \rangle} = \Pr[v \notin S \mid u \in S]$

$= \Pr$ [getting out of S in one step]

Defn: $\Phi[S] = \frac{\langle f, Lf \rangle}{\langle f, f \rangle}$ where $S \subseteq V$, $f = 1_S$

is the conductance of S.

Sparsest cut: Find S that minimizes $\Phi(S)$ s.t. $\text{vol}(S) \leq \frac{1}{2}$.