2024/09/16 Leetcode

A BC   If we start with c's first letter, it can be:

1. C|A B   C| B A

2. C B | A

So it doesn't matter who goes first
for the first part

2024/09/17 Orthonormal & Hadamard

$$\min_x \frac{x^T Q x}{x^T x}$$ where Q is orthonormal

This characteristic number gives the most deviating direction.

The question is what kind of Hadamard H (normalized) maximizes $\min_{\|x\|_2=1} x^T H^T Q x$

Steps:

1. Greedily selected $H_1, \dots, H_n$ maximizes $\min_{\|x\|_2=1} x^T H_1^T \cdots H_n^T Q x$

   or each step grows $\min_{\|x\|_2=1} x^T H_1^T \cdots H_n^T Q x$ with reasonable magtitude

2. Figure out how to solve H for a given Q

$$\min_{\|x\|_2=1} x^T H^T Q x$$

$$\max_{P,D} \min_{x^Tx=1} x^T D^T P^T H^T Q x$$

$$\|P^T H^T Q x - D x\|_2^2$$

run Hungarian Algorithm for P and D

$$Qx - HPDx$$

2024/09/17 Reduce RoPE

$$\begin{pmatrix} \sin\theta_1 & \cos\theta_1 & & & \\ \cos\theta_1 & -\sin\theta_1 & & & \\ & & \ddots & & \\ & & & \sin\theta_n & \cos\theta_n \\ & & & \cos\theta_n & -\sin\theta_n \end{pmatrix}$$

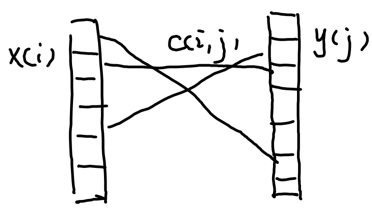
While this doesnt commute with a general matrix K

$$\begin{pmatrix} e^{-i\theta_1} & & & \\ & e^{-i\theta_2} & & \\ & & \ddots & \\ & & & e^{-i\theta_n} \end{pmatrix}$$

commutes with a complex matrix K

Can we represent K as some complex matrix ?

2024/09/17 Hungarian Algorithm for Optimal Permutation



$$\max_{\sigma} \sum_i \left(c(i, \sigma(i)) - x(i) - y(\sigma(i)) \right)$$

$$\text{s.t.} \quad c(i,j) \geq x(i) + y(j)$$

$$\min_{\sigma} \sum_i c(i, \sigma(i)) \geq \max \sum_i x(i) + \max \sum_j y(j)$$

2024/09/17 Commuting Matrices

$$\begin{pmatrix} a_1 & b_1 & & & & & \\ -b_1 & a_1 & & & & & \\ & & a_2 & b_2 & & & \\ & & -b_2 & a_2 & & & \\ & & & & \ddots & & \\ & & & & & a_n & b_n \\ & & & & & -b_n & a_n \end{pmatrix} \begin{pmatrix} c_{11} & d_{11} & \cdots & c_{1n} & d_{1n} \\ -d_{11} & c_{11} & \cdots & -d_{1n} & c_{1n} \\ \vdots & & \ddots & & \vdots \\ c_{n1} & d_{n1} & \cdots & c_{nn} & d_{nn} \\ -d_{n1} & c_{n1} & \cdots & -d_{nn} & c_{nn} \end{pmatrix}$$

commutes

2024/09/17 Hadamard Cont.

$$\begin{pmatrix} I & -I \\ I & I \end{pmatrix} \leftarrow \text{doesn't work}$$

$2048 \rightarrow 11 \times 5 \times 3 = 165$

2024/09/18 Decompose Matrix for Counting

$$A = X + YE$$

$$A = \begin{pmatrix} a_{11} & b_{11} & \cdots & a_{1n} & b_{1n} \\ c_{11} & d_{11} & & c_{1n} & d_{1n} \\ \vdots & & \ddots & & \vdots \\ a_{n1} & b_{n1} & \cdots & a_{nn} & b_{nn} \\ c_{n1} & d_{n1} & & c_{nn} & d_{nn} \end{pmatrix} \qquad E = \begin{pmatrix} & 1 & & & & \\ 1 & & & & & \\ & & & 1 & & \\ & & 1 & & & \\ & & & & \ddots & \\ & & & & & 1 \\ & & & & 1 & \end{pmatrix}$$

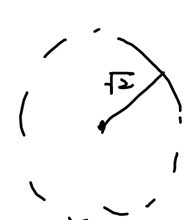
$$X = \begin{pmatrix} x_{11} & x'_{11} & \cdots & x_{1n} & x'_{1n} \\ -x'_{11} & x_{11} & & -x'_{1n} & x_{1n} \\ \vdots & & \ddots & & \vdots \\ x_{n1} & x'_{n1} & \cdots & x_{nn} & x'_{nn} \\ -x'_{n1} & x_{n1} & & -x'_{nn} & x_{nn} \end{pmatrix} \qquad E^2 = I$$

$$Y = \begin{pmatrix} y_{11} & y'_{11} & \cdots & y_{1n} & y'_{1n} \\ -y'_{11} & y_{11} & & -y'_{1n} & y_{1n} \\ \vdots & & \ddots & & \vdots \\ y_{n1} & y'_{n1} & \cdots & y_{nn} & y'_{nn} \\ -y'_{n1} & y_{n1} & & -y'_{nn} & y_{nn} \end{pmatrix}$$

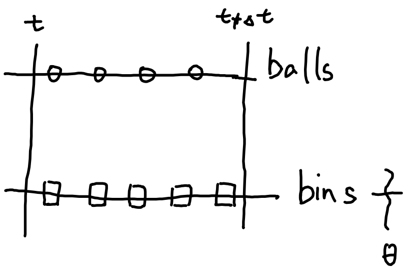
$$\begin{aligned} v_j A R_j &= v_j (X + YE) R_j \\ &= v_j R_j X + v_j Y (E R_j E) E \\ &= v_j R_j X + v_j (E R_j E) Y E \end{aligned}$$

this gives something on the $\sqrt{2}$ hypersphere

$$\text{Attention}(q) = \begin{pmatrix} q & q \end{pmatrix} \begin{pmatrix} X^T & 0 \\ 0 & (YE)^T \end{pmatrix} \begin{pmatrix} \cdots & R_j^T v_j^T & \cdots \\ & E R_j^T E v_j^T & \end{pmatrix}$$



2024/09/18 ECC for Streaming



cond. recieving bins between $[t, t+\Delta t)$

w.h.p. we can peel all code

with probability $1-\theta$ drop some bins

2024/09/18 Deterministic Scheme for Streaming ECC (No Error Case)

ball i in to bins $[a_i, b_i]$

for ball i and j, $a_j - a_i = c(j-i)$ (for example $c=2$)

$b_j - b_i = c'(j-i)$ (for example $c'=2$, too)

Therefore, i is into bins $[2i+1, 2i+m]$

(where m is the number of bins in the window)

Conditioning on known $[0 \ldots i-1]$, what kind of bound do we expect after we recieve packet $2i+m$?

$f(i,m) = \mathbb{P}\{\, i \text{ decodable} \mid [0 \ldots i-1] \text{ is decoded, recv. a bucket after } 2i+m \,\}$

$\geq 1 - (1-\theta)^m$ ← unless all the packets are dropped between $[2i+1, 2i+m]$

The probability of "$\underbrace{\text{all the packets}}_{[0 \ldots i]}$ are peelable" after $2i+m$

is at least $(1-(1-\theta)^m)^{i+1}$.

Lemma: $f(i,k) = f(i,k+1)$ when $k \geq m$

2024/09/19 Hadamard $2^n \doteq \left(\text{Permute} \cdot \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}\right)^{\times n}$

$$Q \begin{pmatrix} I_{2^{n-1}} & -I_{2^{n-1}} \\ I_{2^{n-1}} & I_{2^{n-1}} \end{pmatrix} P_1 \begin{pmatrix} I_{2^{n-1}} & -I_{2^{n-1}} \\ I_{2^{n-1}} & I_{2^{n-1}} \end{pmatrix} P_2 \cdots = \begin{pmatrix} I_{2^n} \end{pmatrix}$$

$$\begin{pmatrix} Q & \\ & Q \end{pmatrix}$$

depending on how "concentrated" the matrix is, we use different permutation to dispatch them into the following form.

$$\Big\downarrow P_i$$

Problem : How to find P_i ?

$$\begin{pmatrix} Q' & Q' \\ -Q' & Q' \end{pmatrix}$$ ← Ideally like this one

2024/09/20 Hash Functions

Definition of 2-Universal Hash Functions $a: V \to U$ $n = |U|$

For any $x_1 \neq x_2 \in V$ $P[a(x_1) = a(x_1)] \leq 1/n$

Definition of Strongly 2-Universal Hash Functions $a: V \to U$ $n = |U|$

For any $x_1 \neq x_2 \in V$ $P[a(x_1) = a_1 \wedge a(x_2) = a_2] \lesssim 1/n^2$

Existence of 2-Universal Hash Functions :

$p \in (|U|, 2|U|)$ $h_{a,b}(x) = (ax + b \bmod p) \bmod n$

prime, unif. dist.

Independence

2024/09/20 Streaming Problem

Receive n numbers in U one-by-one,
estimate $d = \#$ of distinct numbers.

Naive Plan : Takes d space

Lemma : If we need the exact number, we need $\Omega(d)$ space.

Proof sketch: After next arrival, we should be able to answer whether it's one of the d elements.

Thereom : We can find $\hat{d}$ in $\log|U|$ space s.t. $\frac{d}{2} \leq \hat{d} \leq 2d$ w.p. 3/4.

Intuition : $\mathbb{E}[\,\mathbb{E}[\min\{..\} \mid i\text{-th element is min}]] = \int_0^1 x(1-x)^{d-1}\,dx = \frac{1}{d+1}$

Throw d dots uniformly on $[0, 1]$, the min's expectation is $1/_{d+1}$

Algorithm

1) Choose a pairwise independent $h : U \to [0, 1]$

2) Select maintain the least s values of $h(x_i)$ $L_1, \dots, L_s$

3) Output : $\frac{s}{L_s}$

Theorem :

(a) $\mathbb{P}\left\{ \frac{s}{L_s} > 2d \right\} \leq \frac{3}{s}$

(b) $\mathbb{P}\left\{ \frac{s}{L_s} < \frac{d}{2} \right\} \leq \frac{3}{s}$

Proof of (a) :

We want to show $\mathbb{P}\left[L_s \text{ lands in } \left[0, \frac{s}{2d}\right] \right]$ is small

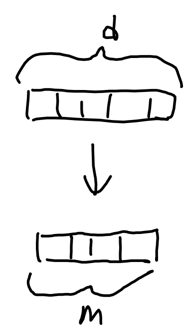
$\mathbb{P}\{ \sum_i \underline{X_i} > s \} < 2/s$ ← we try to prove this

$X_i = 1$ means i-th distinct element has $h(x_i) < \frac{s}{2d}$

$$\mathrm{Var}\{ \sum_i X_i \} = \sum_i \mathrm{Var}\{X_i\} \leq \sum_i \mathbb{E} X_i = \frac{s}{2}$$

$$\mathbb{P}\{ \sum_i X_i > s \} \leq \text{TODO}$$

2024/09/20 Dimension Reduction



Give vertices $v_1, \dots, v_n \in \mathbb{R}^d$ (d is large)

We want to sparsify :

$f(v_1), f(v_2), \dots, f(v_n) \in \mathbb{R}^m$

We want pairwise distance to be kept.

$$(1-\varepsilon) \| f(v_i) - f(v_j) \|_2 \leq \| v_i - v_j \|_2 \leq (1+\varepsilon) \| f(v_i) - f(v_j) \|_2$$

Theorem (Johnson–Linderstrass): ← abbriev. JL

For $m > \frac{\log n}{\varepsilon^2}$,

$$(1-\varepsilon)\,\| f(v_i) - f(v_j) \|_2 \le \| v_i - v_j \|_2 \le (1+\varepsilon)\,\| f(v_i) - f(v_j) \|_2$$

can be achieved with positive probability ($d \to \infty$; $n \to \infty$)

The way we compute f is just a projection.

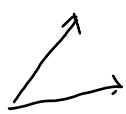
E.g. Compute all pairwise distances.

$O(n^2 d)$

JL & then compute $O(\log n \, d \cdot n + n^2 \log n)$ ← for given ε

Attempt 1: Choose $\log n / \varepsilon^2$ coordinates at random for each v.

Attempt 2:



sample directions $\{g_i\}$

compress it as $v_i \cdot g_j$

Algorithm: $A = \frac{1}{\sqrt{m}} \begin{bmatrix} \cdots \end{bmatrix}_{m \times d}$ take $f(\cdot) = A \times (\cdot)$

$\mathcal{N}(0,1)$ for each element

Definition (ε, δ) JL Property $\}$ $A \sim$ a distribution on $\mathbb{R}^{m \times d}$

For any vector $v \in \mathbb{R}^d$

$$(1-\varepsilon)\,\|Av\| \le \|v\| \le (1+\varepsilon)\,\|Av\| \quad \text{w.p. } 1-\delta$$

Theorem If $m > \frac{\log(1/\delta)}{\varepsilon^2}$ and entries of A are sampled from $\mathcal{N}(0, \frac{1}{m})$ then this distribution satisfies (ε, δ) JL property.

J–L Theorem can be proved using union bound and this theorem.

↑ over C_n^2 vertices

Proof $\quad g \cdot v = \mathcal{N}(0, \|v\|_2^2)$

$$A v = \frac{1}{\sqrt{m}} \begin{bmatrix} \mathcal{N}(0,1) \\ \vdots \\ \mathcal{N}(0,1) \end{bmatrix} \qquad \|v\|_2 = 1$$

$$\|Av\|_2 \approx \|v\|_2 = 1 \qquad \mathbb{E}[\|Av\|_2^2] = 1$$

$$\mathbb{P}\left\{\sum_i G_i^2 > (1+\varepsilon)m\right\} \le \exp(-\varepsilon^2 m/\delta)$$

Cannot apply Chernoff, G_i^2 can be larger than 1

$$\mathbb{P}\left\{\sum_i G_i^2 < (1-\varepsilon)m\right\} \le \exp(-\varepsilon^2 m/\delta)$$

2024/09/20 Subspace Embedding

Let U be $\mathbb{R}^d$, interested in a subspace of k dimensions spanned by $v_1, v_2, \cdots, v_k \in U$.

Theorem : If we project each $v_i \in V$ w/ $A_{m \times d}$ then for all $v \in$ subspace

$$\|v\|(1-\varepsilon) \le \|Av\| \le \|v\|(1+\varepsilon) \quad \text{when} \quad m > \frac{k\log(1/\varepsilon) + \log(1/\delta)}{\varepsilon^2}$$

w.p. $1-\delta$

ε-net $\quad \forall v \in$ subspace, there is a vector u in the ε-net.

2024/09/20 Expansion of Min Distribution on Unif(0,1)

$X_1, \cdots, X_n \overset{i.i.d.}{\sim} \text{Unif}(0,1)$

Smarter : Compute the probability of $Y \sim \text{Unif}(0,1)$ smaller than all X_i s.

$\mathbb{E}[X_i | X_i] = \mathbb{P}_r\{Y < X_i | X_i\}$

$$\mathbb{E}[\min\{X_1, \cdots, X_n\}] = \mathbb{E}[\,\mathbb{E}[X_i \mid \forall j : X_j \ge X_i]\,] = \frac{\int_0^1 (1-x)^{n-1} - (1-x)^n \, d(1-x)}{\int_0^1 (1-x)^{n-1}\, d(1-x)}$$

$$= \frac{\int_0^1 \int_{x_1}^1 \cdots \int_{x_1}^1 x_1 \cdot \cancel{p_{X_1,\cdots,X_i,\cdots,X_n}(x_1,\cdots,x_n)}\, dx_2 \cdots dx_n\, dx_1}{\int_0^1 \int_{x_1}^1 \cdots \int_{x_1}^1 \cancel{p_{X_1,\cdots,X_i,\cdots,X_n}(x_1,\cdots,x_n)}\, dx_2 \cdots dx_n\, dx_1} = \frac{1}{n+1}$$

2024/09/21 VQ for K-V Cache Compression

For now the reduced problem is:

$$f_{k,v}(q) = \mathrm{softmax}(qk^T)v$$

$v \in \mathbb{R}^{N\times d}$

$k \in \mathbb{R}^{N\times 2d}$ where $k = \frac{1}{\sqrt{2}}\begin{pmatrix} \cdots\cdots \\ v_j R_j \quad v_j R_j^T \\ \cdots\cdots \end{pmatrix}$ — j th line (so $\|k\|=1$)

$q \in \mathbb{R}^{2d}$

$$\mathrm{softmax}\begin{pmatrix} \cdots\cdots \\ a_{i1}\ a_{i2} \cdots a_{iN} \\ \cdots\cdots \end{pmatrix} = \mathrm{softmax}\begin{pmatrix} \cdots & \cdots & \cdots \\ \exp(a_{i1})/Z_i & \cdots & \exp(a_{iN})/Z_i \\ \cdots & \cdots & \cdots \end{pmatrix}$$

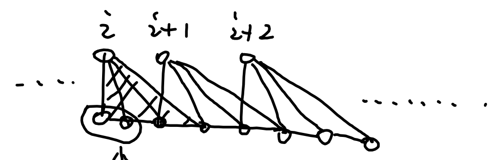
$$\sum_j \exp(a_{ij}) := Z_i$$

R_j is a rotation matrix: $\begin{pmatrix} \sin j\theta_1 & -\cos j\theta_1 & & & \\ \cos j\theta_1 & \sin j\theta_1 & & & \\ & & \ddots & & \\ & & & \sin j\theta_d & -\cos j\theta_d \\ & & & \cos j\theta_d & \sin j\theta_d \end{pmatrix}$

Idea: for points near the unit sphere, use a dynamic version of ε-net

2024/09/21 Upper Bound for Streaming ECC

Asymptotic performance



$ki+m$ packets $\rightsquigarrow$ $i+\frac{m}{k}$ kinds

θ loss rate $\qquad \theta^k$ loss rate

If you want to first decode i, then the first two packets must be decoded before afterward packets.

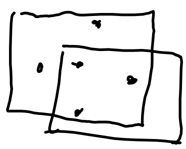
The missing rate is θ^2 for each packet, which is exponential to the # of steps when we move a ball.

2024/09/22 Geometric Set Covering

Setting : $X \subseteq \mathbb{R}^d \leftarrow$ pts

$\mathcal{R} \subseteq 2^{\mathbb{R}^d} \leftarrow$ usually something like rectangles

Select a minimal volume/number of sets from $\mathcal{R}$ to cover the points.



How to relate to gaussian approximation ?

How ε-net helps ?

2024/09/22 Matching in Graph Theory

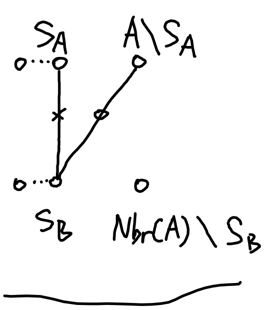
A matching $\mathcal{M}$ in G is a set of pairwise non-adjacent edges.

one vertex has at most one edge

Maximum : the largest possible number of edges

Maximal : not a subset of other matchings

Alterating Path : one matching and one unmatched



Suppose $\mathcal{M}$ is the maximum matching :

$S_A := \{ a : (a, b) \in \mathcal{M} \}$ $S_B := \{ b : (a, b) \in \mathcal{M} \}$

A vertex $x \in A \setminus S_A$ cannot be a neighbour of unmatched node, so it all its neighbors are matched.

Maybe add some exercise on combinatorics ?

Solution to problem 1