CS 602

Approximation Algorithm

Design algorithm that strictly runs in polynomial time ($n^{O(1)}$)
output is allowed to be a "provable" factor away from the optimal solution.

Maximization Problems

Ind Set → Set of vertices such that no two of them are connected

Variable $\alpha \geq 1$

α-apx if we output a solution that is $\left(\frac{1}{\alpha}\right)$ to the optimal solution

Minimization problems

Hamiltonian Cycle → Cycle that visits every vertex of G exactly once and returns back

α-apx if we output a solution that is at most α of the optimal solution

Polynomial-Time approximation solution

algorithm (of with some parameter $\varepsilon > 0$) for any input, output a sol^n within a factor $(1+\varepsilon)$ of the optimal solution. that runs in $n^{g(\frac{1}{\varepsilon})}$ for some comparable f^n.

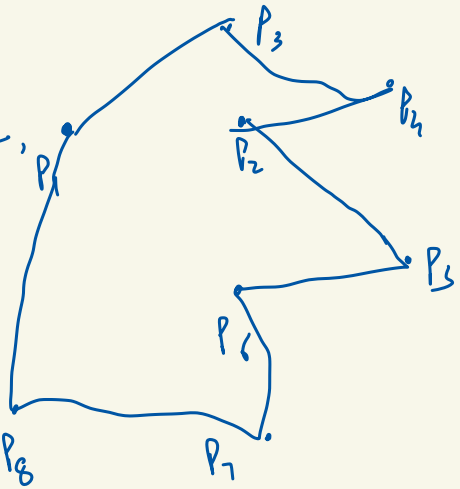
(Running time is polynomial in n, some ε)

Traveling Salesman Problem (TSP)

Given a list of cities ($P \subseteq \mathbb{R}^2$), and distances between each pair of cities, goal is to compute the shortest possible route that visits every city exactly once.

Decision Version

Given a length L, is it possible to find a solution of length L.



Graph:

A set of vertices, edges, weights $G = (V, E, w)$

Visit all the vertices without repetition (minimize the sum of edge weights)

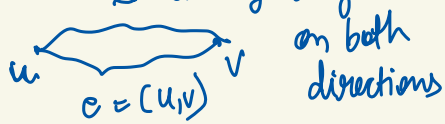
↓

Hamiltonian cycle problem

- Hamiltonial cycle is NP-complete (Richard Karl)
- No constant factor apx is possible

Symmetric

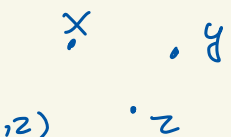
Same edge weights on both directions



Asymmetric

Metric Space:

- $d(u,v) \geq 0$
- $d(x,y) = d(y,x)$
- Triangle inequality $d(x,y) + d(y,z) \geq d(x,z)$



cost (S) = sum of the weights of the edges (union)

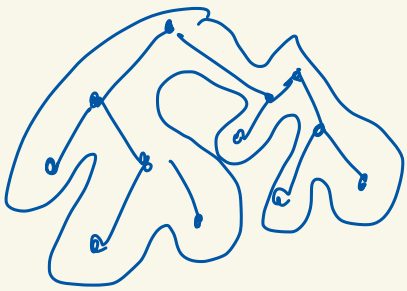
S = set of edges.

same as finding the min of Hamiltonian path
path cycle

Base Structure

— Min spanning Tree [kruskal]

Due to triangular inequality, we can remove duplicates and cost is reduced



Do the DFS traversal and delete duplicates

Analysis

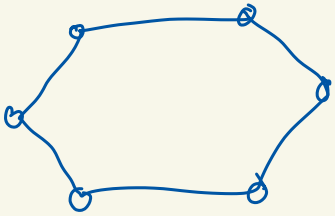
Every edge of the MST is traversed twice

cost (C) $\leq$ 2 × cost (MST)

↓
valid cycle

Cost (MST) $\leq$ cost (opt)

cost (C) $\leq$ 2 × cost (opt)



Q. Can we do better?

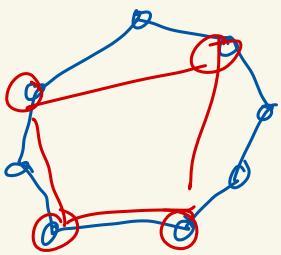
I/O = $G = (V, E)$ → OPT_G

(i) Take a subset $S \subseteq V$

Induced subgraph $G[S]$ → OPT_S

$OPT_S \leq OPT_G$

Triangle Inequality



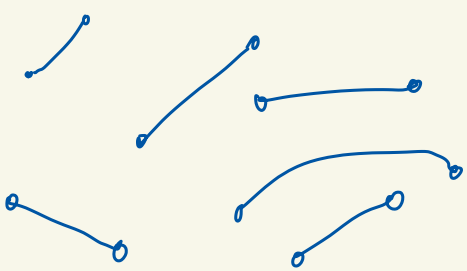
Property-2

Perfect Matching (can be computed in polynomial time)

Min cost PM

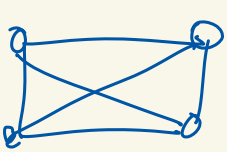
Perfect Matching with smallest cost

(Polynomial Time $\sim O(n^2)$)



Eulerian Tour (Circuit) → vertices can be repeated

each node has even degree then this is possible



$\sum d(v_i) = 2X[E]$

↑ every edge connected twice

Q How many odd degree vertices we have?

even

We will add another edge for perfect matching

Compute Eulerian circuit

↳ Vertex repetition is allowed, but we can delete the duplicates.

Analysis

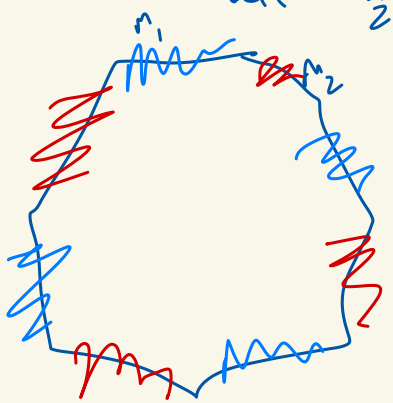
Cost (C) = Cost (MST) + Cost (Matching)

$\leq$ cost(optimal) $\leq \frac{1}{2}$ cost(opt)

↓

Cost (C') $\leq$ cost(C') $\leq \frac{3}{2}$ cost(optimal)

Two matching



Cost $(M_1) \leq$ Cost(opt)

Cost $(M_2) \leq$ cost(opt)

Cost $(M') \leq \frac{1}{2}$ cost(opt)

↓

Matching we choose because it is minimum

Metric TSP

- 2apx (MST doubling)
- 1.5 apx (Christofides Algorithm 1976)
  - ↓ $1.5-\varepsilon$ $\varepsilon = 10^{-30}$ (2021)

No apx is possible if the distances are arbitrary.

$G = (V, E, W)$
- Determine if there is a Hamiltonian cycle & some length t.

G' - complete graph TSP in G' of wt n.

Does there exist in PTAS for metric TSP
$(1+\varepsilon)$-apx
$n^{O(1)}$ time

No

Theorem: There can't be a PTAS $(220/219)$-apx, unless P = NP.

Restrict the metric
↓
Euclidean metric
A set of points in $\mathbb{R}^2$, with euclidean distances

$$d(x,y) = \|x-y\|_2$$

Find the shortest route that covers all the points.

The Traveling Salesman Problem

Cities = $\{1, 2, \ldots, n\}$

C ($n \times n$) matrix → cost of traveling between pairs of cities

↓ symmetric

↓ If we view this as undirected complete graph then the problem is Hamiltonian cycle problem.

↓ NP-complete

Approximation algorithm for the TSP can be used to solve the Hamiltonian cycle problem.

$G = (V, E)$

$c_{ij} = 1$ if $(i,j) \in E$

$n+2$ otherwise

↓

Input to TSP-algo

If Hamiltonian cycle then tour = n

otherwise $\geq (n+2) + (n-1)$

$= 2n+1$

↓

2-apx can increase the cost to $2n$ for hamiltonian cycle

← we can detect hamiltonian cycle

↓ Contradiction! (because HC is NP-complete)

Assumption: Restrict attention to metric space (metric TSP)

Algo(1): A spanning tree of a connected graph $G = (V, E)$ is a minimal subset of edges $F \subseteq E$ such that each pair of nodes in G is connected by a path using edges only in F.

minimum spanning tree : Total edge cost minimized.

* Cost (optimal tour of TSP) $\geq$ cost (MST)

↓

Take this tour and remove one edge

(We will get a spanning tree whose cost $\geq$ cost (MST))

Algo(1) = nearest addition algorithm → 2-apx algo

↓

$F = \{(i_2, j_2) \dots (i_n, j_n)\}$ → edges obtained

$$OPT \geq \sum_{l=2}^{n} c_{i_l j_l}$$

↓

Minimum spanning tree

Cost of the first two nodes (i_2, j_2)

↓

$2c_{i_2 j_2}$ (traversed two times)

j is inserted between (i, k)

increase

$$c_{ij} + \underbrace{c_{jk} - c_{ik}}_{\leq c_{ij}} \leq 2c_{ij}$$

$$\text{Cost (nearest-addition algo)} \leq 2 \sum_{l=2}^{n} c_{i_l j_l} \leq 2(OPT)$$

* Eulerian graph → Traversal of edges (each edge exactly once)

A graph is eulerian iff it is connected and each node has even degree

Algo(11) → Double Tree Algorithm

MST compute → replace each edge by two copies of itself

/

↓

resulting graph is Eulerian and has cost $\leq 2(OPT)$

Eulerian Traversal → sequence of edges (but vertices might repeat)

$i_0\ i_1 \dots i_k$ remove all but the first occurrence of each city in this sequence.

↳ Tour of each city once

two consecutive cities (i_l, i_m)

we have removed $i_{l+1} \dots i_{m-1}$

By triangle inequality, cost is decreased,

↳ In total cost is at most the total cost of all the edges in the Eulerian graph $\leq 2(OPT)$

double-tree = 2-apx algo.

Christofides Algorithm: MST compute

↳ O = set of odd-degree vertices

For a tree, sum of degrees = $2 \times |E|$ = even

↳ number of odd degree vertices = $|O|$ = even

$|O| = 2k$

↳ perfect matching $(i_1, i_2) \dots (i_{2k-1}, i_{2k})$

perfect-matching of minimum cost = $O(n^2)$

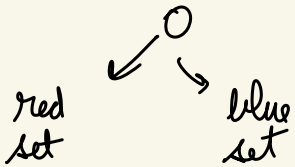
* Christofides = $\frac{3}{2}$ - apx algo

MST has cost $\leq$ OPT

tour on nodes of O has cost $\leq$ OPT

O ↓ subset of original graph

Consider the shortcut tour on the node set O. Colour edges red and blue



cost(red) + cost(blue) $\leq$ OPT

min (cost(red), cost(blue)) $\leq \frac{OPT}{2}$

$\leq$ Perfect matching cost

↳ Perfect matching + MST $\leq \frac{3}{2}$ OPT

* For any constant $\alpha < \frac{220}{219}$ no α-apx for the metric TSP.

Euclidean TSP

Given n points in $\mathbb{R}^2$ with euclidean distances i.e. $d(x,y) = \|x-y\|_2$ shortest tour that visits all points?

Euclidean TSP = NP-hard (do not know NP) length might be irrational

ε-nice instance

(1) Every point has integral coordinates in the interval $\left[0, O\left(\frac{n}{\varepsilon}\right)\right]^2$

(2) Any two different points have distances at least 4.

Consider the smallest bounding box around the points of the input instance. L = longer side of the box

$$L = \left\lceil \frac{8n}{\varepsilon} \right\rceil$$

* OPT_I = length of the optimum tour in I. We can transform I into an ε-nice instance I' such that $OPT_{I'} \leq (1+\varepsilon) OPT_I$

Proof: $I \to$ smallest bounding box $\to$ length longer = L

optimal tour $\geq 2L$ ← { two points opposite in the box have distance $\geq L$

Now to obtain $I' \to$ draw a fine grid with spacing $\frac{\varepsilon L}{2n}$ and map each point to the closest grid point.

$L = \left\lceil \frac{8n}{\varepsilon} \right\rceil = O\left(\frac{n}{\varepsilon}\right)$ $\quad \frac{\varepsilon L}{2n} \geq \frac{\varepsilon}{2n} \times \frac{8n}{\varepsilon} = 4$

$\Rightarrow I' = \varepsilon$-nice (integer coordinates and $d_{ij} \geq 4$)

By mapping points of I to the points in I', we moved each point at most by $\frac{\varepsilon L}{2n}$

↳ Edge changed by $\frac{\varepsilon L}{n}$ — n edges $\rightarrow \varepsilon L$

$L \leq opt$

$OPT_{I'} \leq OPT_I + \varepsilon L \leq (1+\varepsilon) OPT_I$

VC dimension

elements of $X \rightarrow$ points
elements of $R \rightarrow$ ranges

Range space $S = (X, R)$

- X: ground set (finite or infinite)
- R: family of subsets of X (finite or infinite)

x = finite subset of X

measure of a range $\quad \bar{m}(r) = \dfrac{|r \cap x|}{|x|}$

subset N (might be a multi-set) of x

estimate of the measure $\bar{m}(r)$ is $\quad \bar{s}(r) = \dfrac{|r \cap N|}{|N|}$

$Y \subseteq X \qquad R_{|Y} = \{r \cap Y \mid r \in R\}$

projection of R on Y. The range space S projected to Y is $S_{|Y} = (Y, R_{|Y})$

If $R_{|Y}$ contains all subsets of Y (if Y = finite, $|R_{|Y}| = 2^{|Y|}$) then Y is shattered by R

VC dimension ($\dim_{VC}(S)$) maximum cardinality of a shattered subset of X.

Interval → VC = 2
Disks → VC = 3
Convex sets → VC = ∞

Complement : range space $S = (X, R)$ $\quad \delta = \dim_{VC}(S)$

$\bar{S} = (X, \bar{R})$

$\bar{R} = \{ X \setminus r \mid r \in R \}$

If S shatters B, then for any $Z \subseteq B$, $(B \setminus Z) \in R_{|B}$

$Z = B \setminus (B \setminus Z) \in \bar{R}_{|B}$

$\Rightarrow$ $\bar{R}_{|B}$ contains all the subsets of B.

$\Rightarrow \bar{S}$ shatters B $\Rightarrow$ $\dim_{VC}(\bar{S}) = \dim_{VC}(S)$

* Let $P = \{p_1, \dots, p_{d+2}\}$ be a set of $d+2$ points in $\mathbb{R}^d$. There are real numbers $\beta_1 \dots \beta_{d+2}$ not all of them zero such that $\sum_i \beta_i p_i = 0$ and $\sum_i \beta_i = 0$

Proof: $q_i = (p_i, 1)$ $\quad q_1 \dots q_{d+2} \in \mathbb{R}^{d+1}$ are linearly dependent.

There are coefficients $\beta_1 \dots \beta_{d+2}$ not all of them zero such that $\sum_{i=1}^{d+2} \beta_i q_i = 0$ Considering only the first d-coordinates $\sum_{i=1}^{d+2} \beta_i p_i = 0$ $\quad (d+1)^{st}$ coordinate $\sum_{i=1}^{d+2} \beta_i = 0$

Radon's Thm: $P = \{p_1 \dots p_{d+2}\}$ in $\mathbb{R}^d$ Then, there exist two disjoint subsets C and D of P, such that $CH(C) \cap CH(D) \neq \emptyset$

$C \cup D = P$.

Proof: By previous thm, $\sum_i \beta_i p_i = 0$ and $\sum_i \beta_i = 0$

$\mu = \sum_{i=1}^{k} \beta_i = -\sum_{i=k+1}^{d+2} \beta_i$

$$\sum_{i=1}^{k} \beta_i p_i = -\sum_{i=k+1} \beta_i p_i$$

$V = \sum_{i=1}^{n} (\beta_i / H)\, p_i$ is a point in Convex hull $(p_1 \dots p_n)$

$V = \sum_{i=k+1}^{d+2} -(\beta_i / H) p_i \in CH(p_{k+1} \dots p_{d+2})$

V = intersection of the two convex hulls

* $P \subseteq \mathbb{R}^d$ = finite set s = point in $CH(P)$ h^+ = halfspace containing s. Then there exists a point of P contained inside h^+.

Proof: $h^+ = \{ t \in \mathbb{R}^d \mid \langle t, v \rangle \leq c \}$

$\sum_i \alpha_i = 1$ and $\sum_i \alpha_i p_i = s$

$\langle s, v \rangle \leq c \Rightarrow \left\langle \sum_{i=1}^{m} \alpha_i p_i, v \right\rangle \leq c \Rightarrow \beta = \sum_{i=1}^{m} \alpha_i \langle p_i, v \rangle \leq c$

$\beta_i = \langle p_i, v \rangle$ β is a weighted average of $\beta_1 \dots \beta_m$

$\Rightarrow$ there must be a β_i which is no larger than the average $\Rightarrow \beta_i \leq c \Rightarrow \langle p_i, v \rangle \leq c \Rightarrow p_i \in h^+$.

* Growth Function = $G_\delta(n) = \sum_{i=0}^{\delta} \binom{n}{i} \leq \sum_{i=0}^{\delta} \frac{n^i}{i!} \leq n^\delta$

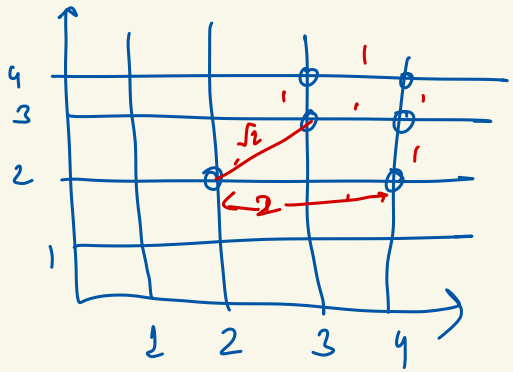
* Sauer's Lemma: If (X, R) is a range space of VC dimension δ with $|X| = n$ then $|R| \leq G_\delta(n)$

Proof: Holds for $n = 0$ and $\delta = 0$

$R_x = \{ r \setminus \{x\} \mid r \cup \{x\} \in R \text{ and } r \setminus \{x\} \in R \}$

$R \setminus x = \{ r \setminus \{x\} \mid r \in R \}$

$|R| = |R_x| + |R \setminus x| \leq G_{\delta-1}(n-1) + G_\delta(n-1) = G_\delta(n)$



Euclidean TSP is NP-hard but not known to be NP

Sum of square roots (SRS)

Given a set of positive integers $a_1, a_2, \ldots, a_k$ and t

decide $\sum_{i=1}^{k} a_i \leq t$

$\{a_1, \ldots a_k\}$ $\{b_1 \ldots b_k\}$

$\sum_{i=1}^{k} a_i \leq \sum_{i=1}^{k} b_i$

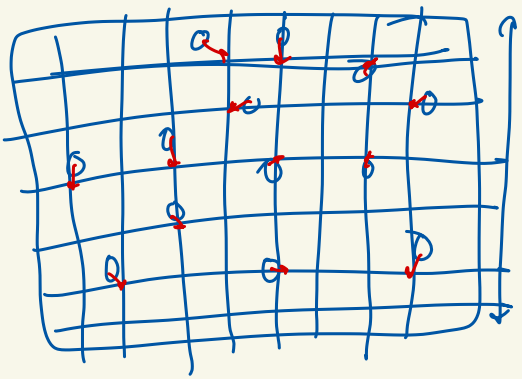
NP-hard not in NP

PTAS - polynomial time approximation scheme.
for euclidean TSP

→ Rounding the instance

→ Partitioning (Exploit the structure of the instance by breaking into nice instances)

→ Apply Dynamic programming



Snap each point to the closest grid point (To make the coordinates rational)

Given-ε

ε - "nice" instance

Defⁿ - An instance of Euclidean TSP is ε-nice if

1. Every point has integral co-ordinates in the interval $\left[0, O\left(\frac{n}{\varepsilon}\right)^2\right]$

2. Any two diff points have dist at least 4.

- Take a small bounding box (axis-parallel)

longer side — L.

s.t. rooted not origin

scale $t = \left\lceil \frac{8n}{\varepsilon} \right\rceil$ } I'

Lemma - I is i/p & OPT_I is optimal tour.

I' is ε-nice instance $OPT_{I'}$ is optimal tour

$OPT_{I'} \leq (1+\varepsilon) OPT_I$

OPT is at least $2L$

- Draw a fine grid with spacing $\frac{\varepsilon \times L}{2 \times n}$
- Map every pt to its closest grid point (multiple pts could be mapped to the same grid pt).
- All pts have integer coordinates

$$L = \left\lceil \frac{8n}{\varepsilon} \right\rceil \quad \text{or} \quad O\left(\frac{n}{\varepsilon}\right)$$

Grid spacing $\frac{\varepsilon L}{2n} \leq \frac{\varepsilon}{2n} \times \frac{8n}{\varepsilon} = 4$

→ Mapping each point in I has moved $\frac{\varepsilon L}{2n}$

Every edge in the sol^n changes by at most $\frac{\varepsilon L}{2n}$ n edges in OPT

Cost = $\varepsilon \times L$

↳ $OPT_I + \varepsilon \times L$

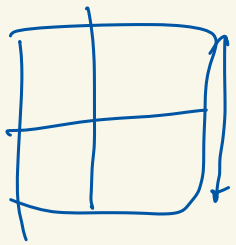
$OPT_I + \varepsilon \times OPT_I$

$OPT_{I'} \leq (1+\varepsilon) \times OPT_I$ / $L \leq OPT_I$

Partition the space

- Extend the bonding box to square with new side length L'

L' is the smallest power 2



* Recursively partition the box (square) into four equal sized squares until the side length is 1

(L' is power of 2)

- Each pt is separated
- one pt in each "non-empty" square

Partitioning terminate after $O(\log L')$ steps

Height of the quadtree

- $O(\log L')$
- $O\left(\lg\left(\frac{n}{\epsilon}\right)\right)$

Apply dynamic programming to the quad tree

Solve for each square that are leaves

Bottom-up combine

Portals

Limits the # of interactions

# of portals
- Accuracy improve
- Running Time

} Tradeoff

Select m = power of 2

$m \in \left[\frac{k}{\epsilon}, \frac{2k}{\epsilon}\right]$

For each square → put portals in corners

put (m-1) portals equally spaced

Portal-respecting tour (p-tour)

Defn: p-tour enters/exits through portals

- length of p-tour $\leq (1+\varepsilon) \times OPT$

Detours can add much more cost

Soln → Randomize

(i) Translate the grid by a random offset at most $\frac{L'}{2}$ in each coordinate

(ii) points remain grid points.

(iii) With high probability, the pts are nicely concentrated.

(iv) Higher levels in the partition (quad tree) have more portals → fine-grained tree

Defn
(a,b) dissection : origin of the grid is translated by $(-a,-b)$

Theorem : (a,b) picked up uniformly at random $\left\{0, \frac{L'}{2}\right\}$ with prob at least $\left(\frac{1}{2}\right)$

P-Tour such that cost (p-tour) $\leq (1+4\varepsilon) \times OPT$.

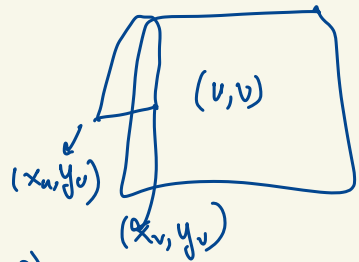
Proof : Extend non-p tour to a p-tour

For each vertical/horizontal line ℓ

$t(\ell)$ = # of times intersects ℓ

$$T_\ell = \sum_\ell t(\ell)$$

Claim : $T \le 2 \times opt$



- ℓ crosses $(x+1)$ vertical lines
- " " $(y+1)$ horizontal "

total contribution $(x+y+2)$

$\sqrt{x^2+y^2}$ → in OPT

$$\sqrt{2(a^2+b^2)} \ge a+b$$

$$\forall x,y \quad d(x,y) \ge 0$$

$$x+y+2 \le \sqrt{2(x^2+y^2)} + 2$$

$$\le \underbrace{2\sqrt{x^2+y^2}}_{OPT}$$

Bound - expected length of the detours

Detours might cross

$$|x_u - x_v| + |y_u - y_v| + 2$$

i of the quad-tree

$$\frac{L'}{2^i m}$$

If ℓ is in level i

$$\le \frac{L'}{2^i m}$$

Q: What is the prob that after random shift ℓ crosses ℓ at level-i

- l could be mapped to $\frac{l'}{2}$ many lines [translated by $(0, \frac{l'}{2})$]

- 2^{i-1} many lines of level i

$$\frac{2^{i-1}}{l'/2} = \frac{2^i}{l'}$$

Expected length $\sum_{i=1}^{k} \frac{2^i}{l'} \times \frac{l'}{2^i m} \leq \varepsilon$

By linearity of expectation $2\varepsilon \times OPT$

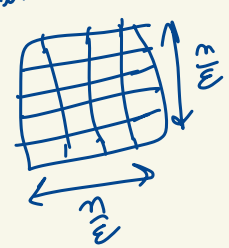
Markov Inequality

Pr (total length increase of detours $\geq 4\varepsilon\, OPT$)

$$\leq \frac{2\varepsilon\, OPT}{4\varepsilon\, OPT} = \frac{1}{2}$$

De-randomize

- fixed ε
- grid shifting by trying all possibilities



$O\left(\left(\frac{n}{\varepsilon}\right)^2\right)$

Final step (DP)

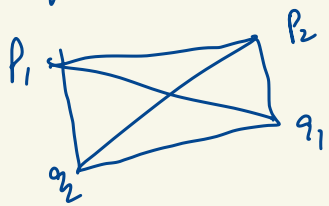
Given (a,b) dissection, get p-tour

Introduce state — square

— any set of possible ways of entering/exiting the squares

# of states — $1 + 4 + 4^2 + \dots + L'^2 = O\left(\frac{m^2}{\varepsilon^2}\right)$

Lemma: W.L.O.G. a p-tour is well-behaved 2-light



— 4 portals

— use one portal $\{0,1,2\}$

$m = O\left(\frac{k}{\varepsilon}\right) = O\left(\log \frac{L'}{\varepsilon}\right)$

n^{th} Catalan number $= \frac{1}{n+1}\binom{2n}{n} = O(2^{2n}) = O(2^{8m})$

Algorithm = try all paranthesis
- translate them into paths
- Discard anything that intersects

$m = O\left(\log \frac{n}{\varepsilon/\varepsilon}\right)$ # of entry exit $= O(n^{11\varepsilon})$

Computation of values

$A[(s_1,t_1) \dots (s_\ell,t_\ell)]$ — Compute the whole table.

Clustering

→ Learning, searching, data mining

→ Given data, find an interesting structure
→ Represented as points in $\mathbb{R}^d$

General metric space (X, d) where X is a set
$d : X * X \to [0, \infty)$

is a metric it satisfies —

(i) $x = y \to d_\mu(x, y) = 0$
(ii) $\forall x, y \quad d_\mu(x, y) = d_\mu(y, x)$
(iii) $\forall x, y, z \quad d_\mu(x, y) + d_\mu(y, z) \geq d_\mu(x, z)$

Assumption
(x, y) $d_\mu(x, y)$ in $O(1)$ time

Norm
↳ norm defines distances between pts

$p, q \in \mathbb{R}^d \qquad \|p - q\|_p = \left(\sum_{i=1}^{d} |p_i - q_i|^p \right)^{\frac{1}{p}}$ for $p \geq 1$

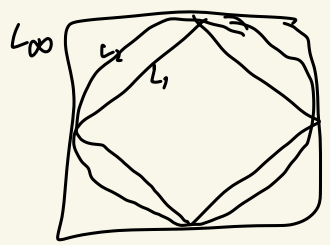
$p = 2$: Euclidean norm
$p = 1$: manhattan distance (L_1 norm)

L_∞ norm

$$\|p-q\|_\infty = \lim_{p\to\infty} \|p-q\|_p$$

$$\max_{i=1} |p_i - q_i|$$

Triangle inequality holds for L_∞ too, its call minkowski inequality



for any $p \in \mathbb{R}^d$

$\|p\|_p \le \|p\|_q$ if $p > q$

Lemma – for any $p \in \mathbb{R}^d$

$$\|p\|_1 / \sqrt{d} \le \|p\|_2$$

Proof: $p = (p_1 \dots p_d)$ $\quad p_i \ge 0 \ \forall i$

Const. α $\quad f(x) = x^2 + (a-x)^2$ minimized if $x = \frac{\alpha}{2}$

Let $\alpha = \|p\|_1 = \sum_{i=1}^{d} |p_i|$

By symmetry obs on $f(x)$ $\sum_{i=1}^{d} p_i^2$

$$\|p\|_2 \ge \sqrt{d\left(\frac{\alpha}{d}\right)^2} = \|p_i\|/\sqrt{d}$$

Metric space (X, d)

I/p : A set of points P, $|P| = n$

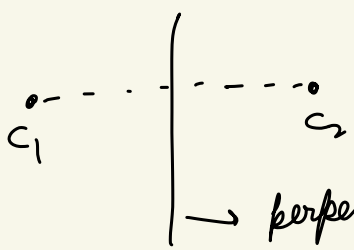
O/p : Find a clustering (set of centers) such that each pt is assigned to its nearest center

set of clusters C

$$\text{cluster}(c, \bar{C}) = \{ p \in P \mid d_M(p, \bar{C}) = d_M(p, c) \}$$

d_M : minimum across all the pts

Voronoi Partition



→ perpendicular bisector

now partitioned into two spaces

$$p_c = (d(p_1, c), d(p_2, c), \ldots, d(p_n, c))$$

i^{th} coordinate $d(p_i, c)$ dist to p_i to its closest centers

O/P = find a set of k-centers $C \subseteq P$ such that the maximum distance of a point in P to its closest center is minimized

Def^n: Given a set of k-centers C,

$$\|p\|_C\|_\infty = \max_{p \in P} d(p, C)$$

Find C, s.t $\|p_C\|_\infty$ is minimized

$$opt_\infty(P, k) = \min_{C \subseteq P} \|p'_C\|_\infty \quad K = \|C\|$$

- C_{opt}
- NP-hard
- Hard to approximate beyond 1.86
- 2-approx in the Euclidean space in $\mathbb{R}^2$

Greedy Algo

- Start by picking an arbitary pt. $\bar{c}_1$

$$C_1 = \{\bar{c}_1\}$$

- Compute the distances for each $p \in P$ from $\bar{c}_1$
- Take the pt with worst distance

$$(r_1 = \max_{p \in P} d_1\{p\})$$

say $\bar{c}_2$

$$C_2 = C_1 \cup \{\bar{c}_2\}$$

$$C_i = C_{i-1} \cup \{\bar{c}_i\}$$

$O(nk)$ → data from previous iterations

$O(n)$ space

For each pt $p \in P$, a single variable $d[p]$ with its current dist to the closest pt.

$$d[p] \leftarrow \min\left(d[p],\ d_M(p, \overline{c_i})\right)$$

Def^n : A ball of radius σ around a pt $p \in P$ is a set of pts in P with dist at most σ from p

$$b(p,\sigma) = \{q \in P \mid d_M(p,q) \le \sigma\}$$

Remark : k-center is essentially covering P with k-balls of minimum radius.

Thm : Greedy Algo computes a set K of k-center such that K is 2-apx $\|p_K\| \le 2 \times opt_\infty$

takes $O(n \times k)$ time.

Proof : Running Time ✓

Def^n $r_k = \|p_K\|_\infty$

Let $\overline{c_{k+1}}$ is the point ~~realising~~

$$r_k = \max_{p \in P} d(p,K)$$

$$C = K \cup \{\overline{c_{k+1}}\}$$

By the def^n of r_i:

$$r_1 \ge r_2 \ge \dots > r_k$$

$$i < j < k+1$$

$$d_M(\overline{c_i}\ \overline{c_j}) \ge d_M(\overline{c_j}, c_{j-1})$$

$$r_{j-1} \ge r_k$$

— the dist. between any pair of pts. in C is at least r_k

opt — covers P by using k balls

by triangle inequality any two points within such a ball are with a dist at most 2×opt.

↓

None of the balls contain two points from C ⊆ P

↓ contradiction!

Greedy permutation

Let this run till we exhaust all pts

$\langle P \rangle = \langle \bar{c}_1, \bar{c}_2, \ldots, \bar{c}_n \rangle$

↓

$\langle r_1, r_2 \ldots r_n \rangle$

Defⁿ : r-packing : A set $S \subseteq P$ for P

(i) covering property : all the pts in P are within dist of atmost r from S.

(ii) Separation property : $\forall p, q \quad d_M(p,q) \geq r$

r-packing gives compact representation

* Greedy permutation gives such a rep.

Thm: $\langle \bar{c}_1, \bar{c}_2, \ldots, \bar{c}_n \rangle$ $\quad \langle r_1, \ldots r_n \rangle$

for any i, we have $c_i = \langle \bar{c}_1 \ldots \bar{c}_i \rangle$ is an r_i - packing of P

Proof: By contradiction

$$r_{k-1} = d(\bar{c}_k, \bar{c}_{k-1}) \quad \forall \; k = 2, \ldots n$$

for $j < k \leq i \leq n$

$$d_\mu(\bar{c}_i, \bar{c}_k) = r_{k-1} \geq r_i$$

K- medians clustering

A set $P \subseteq X$ $|P| = n$, a parameter k. Find a set of k- points $C \subseteq P$ s.t. the sum of distances of the pts in P to its closest center is minimized.

Clustering price: $\|P_C\| = \sum_{p \in P} d(p, C)$

Objective f^n: $\text{opt}_p(P, k) = \min_{\substack{C \subseteq P \\ |C| = k}} \|P_C\|$

Optimal set of centres $- C_{opt}$.

Local search: move sol^n to sol^n in the space of candidate sol^n (the search space) by applying local changes

Continue until, end up on optimal or we exhaust the running time.

Notations:

A set $U = \{ P_c \mid C \in P^k \}$

$opt_\infty (P, k) = \min_{q \in U} \|q\|_\infty$ — k-center

$opt (P, k) = \min_{q \in U} \|q\|_1$ — k-median

1.86 Apx ✗

2 Apx ✓ (Greedy)

Claim: For any set P, $|P| = n$, k

$$opt_\infty (P, k) \leq opt_1 (P, k) \leq n \times opt_\infty (P, k)$$

Proof: $p \in \mathbb{R}^n$ $\quad \|p\|_\infty = \max_{i=1}^{n} |p_i| \leq \sum_{i=1}^{n} \|p_i\| = \|p\|_1$

$$\|p\|_1 \leq \sum_{i=1}^{n} |p_i| \leq \sum_{i=1}^{n} \max |p_i| \leq n \times \|p\|_\infty$$

C – set of centers $|C| = k$ realising $opt_1 (P, k)$ i.e.

$opt_1(P,k) = \|P_c\|_1$

$opt_\infty(P,k) \leq \|P_c\|_\infty$

$\leq \|P_c\|_1 = opt_1(P,k)$

Similarly, k realizing $opt_\infty(P,k)$

$opt_1(P,k) = \|P_c\|_1 \leq \|P_k\|_1$,

$\leq n \times \|P_k\|_\infty$

$= n \times opt_\infty(P,k)$

(2n-factor for medians)

2n-apx

use this as a first step for local search

L- is 2n-apx

Improve : parameter $0 < \delta < 1$

$\forall i \in [n]$ L_{curr}

Local search

- Set $L_{curr} \leftarrow L$
- At each iteration

We will check if the current sol^n L_{curr} can be improved by replacing one of the centers by one center from outside (non-centers)

Swap

$k \leftarrow (L_{curr} \setminus \{\bar{c}\})\ (\bar{o})$

if $\|P_k\|_1 \leq (1-\delta)\|P_{L_{curr}}\|$

- continue swap as long as it satisfies the constraint
- return L_{curr}

Running time: An iteration takes $O(nk)$ swaps
($n-k$ candidates to be swapped in k-candidates to be out)

Implementing swap (naively $O(nk)$)

overall $O((nk)^2)$

since

$$\frac{1}{1-\delta} \not\geq (1+\delta)$$

$$O\left((nk)^2 \log_{\frac{1}{1-\delta}} \frac{\|P_k\|_1}{opt_1}\right)$$

$$= O\left((nk)^2 \cdot \log_{1+\delta} 2n\right) = O\left((nk)^2 \log \frac{n}{\delta}\right)$$

k-means

Set $P \subseteq X$, k, find k pts $C \subseteq P$ $|C|=k$

$$\|P_C\|^2 = \sum_{p \in P} (d_C(p, C))^2$$

obj: s.t. $\|P_C\|_2^2$ is minimized

$$Opt_2(P,k) = \min_{C, |C|=k} ||P_C||_2^2$$

$O(1)$- factor for k-means as well

Thm: $0 < \varepsilon < 1$

$(2.5+\varepsilon)$ - approx

VC-dim

- A range space $(X, R) = S$

  X = ground set (finite / infinite)

  R = family of subsets of X.

Consider finite subset of X as the estimating ground set.

Defn (measure): fixed subset of X. For a range $r \in R$

measure for r

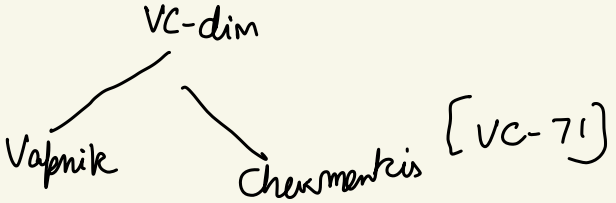
$$\overline{m(r)} = \frac{|r \cap X|}{|X|}$$

For a subset N (multi-set) of X, the estimate of the measure of $\overline{m(r)}$, for $r \in R$

$$\overline{s(r)} = \frac{|r \cap N|}{|N|}$$

Q. Can we get methods to generate N s.t.

$$\overline{s(r)} = \overline{m}(r) \quad \forall\, r \in \mathbb{R}$$



Defn: $S = (X, R)$ For $Y \subseteq X$

$$R_{|Y} = \{ r \cap Y \quad r \in R \}$$

be the projection of R on Y

The range space S_Y is projected to $S_{|Y} = (Y, R_{|Y})$

$2^{|Y|}$

If this is the cardinality then it is called shattered by R

Complement

$$S = (X, R) \qquad S = \dim_{VC}(S)$$

$$\overline{S} = (X, \overline{R}) \quad \text{where } \overline{R} := \{ X \setminus r \mid r \in R \}$$

Q. what is the VC-dim of $\overline{S}$?

A subset $B \subseteq X$, is shattered in $\overline{S}$ if it is shattered in S for any $Z \subseteq B$ $(B \setminus Z) \in R_{|B} \Rightarrow Z = B \setminus (B \setminus Z) \in \overline{R}_{|B}$

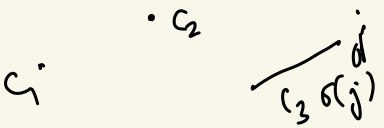
Local Search

- X be the set of arbitrary subset of k-centers (greedy solution of k-centers)

while true do:

(swap) if $i \in X$ and $i' \in F \setminus X$

$$\text{cost}(X - i + i') < \text{cost}(X)$$



$opt = X^*$ - optimal set of k-center X $\leftarrow X - i + i'$ otherwise break

Nearest

$i^* \in X^*$

$i \in X$

for each center chosen in X, nearest center in X^* minimum(i, i^*)

Inverse

For the centers in X^* inverse is the nearest map

Bijection

Claim: for any $j \in X$

$$d(\text{nearest}(\sigma^*(i)), j) \leq d_j + 2d_j^*$$

Half-spaces

Let $\mathcal{R}$ be the set of closed half spaces in $\mathbb{R}^d$

Claim: $P = \{p_1, \ldots, p_{d+2}\}$ set of points in $\mathbb{R}^d$

Real numbers $\beta_1, \beta_2, \ldots, \beta_{d+2}$ (not all are zero)

s.t. $\sum_i \beta_i p_i = 0$ & $\sum_i \beta_i = 0$.

Proof: $a_i = (p_i, 1)$ for $i = 1 \ldots d+2$

pts are linearly dependent. and these are coefficients $\beta_1, \ldots \beta_{d+2}$

$a_1, a_2, \ldots, a_{d+2} \in \mathbb{R}^{d+2}$ s.t. $\sum_{i=1}^{d+2} \beta_i p_i = 0$

- Considering first d-coordinates of these pts implies

$$\sum_{i=1}^{d+2} \beta_i p_i = 0$$

Similarly $(d+1)$ coordinates $\sum_{i=1}^{d+2} \beta_i = 0$

Radon's Thm: $P = \{p_1 \ldots p_{d+2}\}$ $\exists$ disjoint subsets C & D of P. $H(C) \cap H(D) = \phi$ then $C \cup D = P$

Shattering Dim:

Property: A range space (R) with VC-dim(S) means # of ranges grow polynomially on (n) (Generally this is exp^n)

Growth of f^n: $G_S(n) = \sum_{i=0}^{S} \binom{n}{i} \leq \sum_{i=0}^{S} \frac{n^i}{i!} \leq n^S$ for $S > 1$

Sauer's Lemma $S = (X, R)$ $VC(S) = S$ $|X| = n$ $|R| \leq G_S(n)$

Proof: $n = 0$ $S = 0$ → done!

$x \in X$

Contains x $R_x = \{ r \setminus \{x\} \mid r \cup \{x\} \in R \text{ and } r \setminus \{x\} \in R \}$

Does not contain x $R \setminus x = \{ r \setminus \{x\} \mid r \in R \}$

Observation: $|R_x| + |R \setminus x| = |R|$

Shatter function: $S = (X, R)$ shatter f^n

$\Pi_S(m)$ is the maximum # of sets that might be created by S, when restricted to the subsets of size m.

$$\Pi_S(m) = \max_{\substack{B \subset X \\ |B| = m}} |R_{|B|}|$$

Shattering dim: The smallest d such that $\Pi_S(m) = O(m^d) \quad \forall m$

Thm $S = (X, R)$ has shattering dim d, then the VC-dim is bounded by $O(d \log(d))$

Proof: $N \subseteq X$ be the largest subset of X. shattered by S and δ is the cardinality

$$2^\delta = |R_{|N|}| \leq \Pi_S(m)$$

$$\delta \leq \log c + d \log \delta \qquad (\delta \geq \max(2, 2\log c))$$

$$\Rightarrow \frac{\delta - \log c}{\log \delta} \leq d$$

$$\frac{\delta}{2 \log \delta} \leq d \Rightarrow \frac{\delta}{\log \delta} \leq O(1) \times d$$

$f(x) = \frac{x}{\log x} \rightarrow$ non-increasing $x \geq c$

$f(x) \geq c \quad \forall x > 1$

$c \geq \sqrt{e}$ if $f(x) \leq x$ then $x \leq c \log x$

ε-net and ε-sampling

$S = (X, R)$ $\quad$ x is a finite subset of X

$0 \le \varepsilon \le 1$

Informally, ε-sampling captures R, upto some ε-error

a subset $c \subseteq x$ is an ε-sample for x if for any range $r \in R$

$$|\overline{m}(r) - \xi(r)| \le \varepsilon$$

$\overline{m}(r)$: measure, $\xi(r)$: estimate

Thm: (ε-sample, VC-1) — There is a +ve constant c s.t. if (X, R) is any range space with VC dim δ.

$x \subseteq X$ finite subset of X and $\forall \varepsilon, \phi > 0$

$\exists$ a random subset $C \subseteq X$ of cardinality $s = \frac{c}{\varepsilon^2}\left(\delta \log \frac{\delta}{\varepsilon} + \log \frac{1}{\phi}\right)$

(with probability $= \phi$)

ε-net: A set $N \subseteq X$ is an ε-net for x if for any range space $r \in R$ if $\overline{m}(r) \ge \varepsilon$ $(|r \cap x| \ge \varepsilon \cdot |x|)$ then r contains at least one pt. of N (ie. $r \cap N \neq \phi$)

(Intuitively: hit all heavy subsets)

ε-net Thm (HW#7)

$S = (X, R)$ has VC dim $(S) \leq \delta$)

x is a finite subset of X.

Suppose $0 \leq \varepsilon \leq 1$ & $\phi < 1$

- N a set obtained by random independent draws.
- $m \geq \max\left(\frac{4}{\varepsilon} \log \frac{4}{\phi}, \frac{8\delta}{\varepsilon} \log \frac{16}{\varepsilon} \right)$

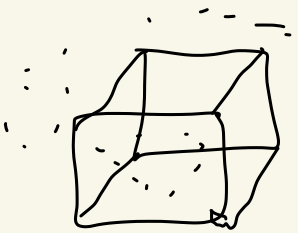
Then, N is a ε-net with prob $(1-\phi)$.

* Remark: Both of the thms hold for spaces with shattering dim δ. $\left(O\left(\frac{1}{\varepsilon} \log \frac{1}{\phi} + \frac{\delta}{\varepsilon} \log \frac{\delta}{\varepsilon} \right) \right)$

Range Searching

$P \in \mathbb{R}^d$ we have a database

Given a hyper rectangle, we want to report the points that lie inside



Allow 1 % error,

ε-sample (thm) says there is a subset of const. size (which depends on ε)

Use this to perform an estimation.

Rectangle has bounded VC-dim

Random sample with probability $(1-\phi)$

Learning Concepts

Assume we know a f^n that returns 1 if inside, 0 otherwise

Query Oracle

There is a distribution D defined over the space. We pick points from D.

Growth function $G_d(n) = \sum_{i=0}^{d} \binom{n}{i} \leq \sum_{i=0}^{d} \frac{n^i}{i!}$

$\leq n^d$

Sauer's Lemma $S=(X,R)$

Suppose $VCdim(S) \leq d$

$|R| \leq G_d(n) \leq \sum_{i=0}^{d} \binom{n}{i}$

$|R| = \Pi_F(n)$

Proof: By induction on n, d

$n=d=1$ holds

Assume that it holds for $n-1$ & d

and as well as for $n-1$ & $d-1$

We prove for n & d

define $f^n : \sum_{i=0}^{d} \binom{n}{i} = h(n,d)$

Our induction hypothesis is for F with $VC\text{-}dim \leq d$

$\Pi_F(n) \leq d$

$$\binom{n}{i} = \binom{n-1}{i} + \binom{n-1}{i-1}$$

$$h(n,d) = h(n-1,d) + h(n-1,d-1)$$

recurrence

Now let's fix a class F

$VC\text{-dim}(F) = d$ and a set

$X_1 = \{x_1, \dots x_m\} \subseteq X$

$f_1 = F_{1X}$

$f_2 = F_{2X}$

$$F_3 = \{ f_{1X_2} \mid f \in F \ \& \ f' \in F \text{ s.t. } \forall x \in X_2 \ f'(x) = f(x) \ \& \ f'(x_1) = -f(x_1) \}$$

$VC\text{-dim}(F')$

$\leq VC\text{-dim}(F) \leq d$

$|F_1| = |F_2| + |F_3|$

$\leq d \quad \leq d-1$

Induction hypothesis

$$\begin{cases} |F_2| \leq h(n-1,d) \\ |F_3| \leq h(n-1,d-1) \end{cases}$$

$$|F_1| \leq h(n-1,d) + h(n-1,d-1)$$

$$\leq h(n,d)$$

Ex. Let F be s.t. $VC\text{-dim}(F) \leq d$ for $n \geq d$

$$\Pi_F(n) \leq \left(\frac{ne}{d}\right)^d$$

Set-cover / Hitting set (piercing)

U = universe of elements
X = set of subsets

$S = (U, X)$ - Set system

choose a subset $X' \subseteq X$ which is a cover

— NP hard

Greedy approximation — (1) sort all the sets based on cardinality

(2) choose the set with max cardinality.

log factor

↳ ∃ a lower bound shows that we can't get better than log factor.

Wish: for "nice" set families, can we beat greedy (log factor)

$S = (X, R)$

X → set of elements

R → set of ranges

Goal: choose a subset $R' \subseteq R$ that cover X.

class of objects → has bounded VC-dim.

ε-net: Sample that "hits" all the heavy sets ($\geq \varepsilon n$)

A set $N \subseteq X$ is an ε-net for a finite subset x if for any range $r \in R$, $m(r) = \frac{m(x)}{|x|} \geq \varepsilon$ then r contains at least one pt.

Construction of ε-net — choose a random sample if it is large enough its ε-net

- small hitting set " (which is also a net)

ε-net thm

We can get a subset N by m ind. draws for a (uniformly chosen) finite subset x.

$$N \geq \max\left\{ \frac{4}{\varepsilon} \log \frac{4}{\phi}, \frac{8d}{\varepsilon} \log \frac{16}{\varepsilon} \right\}$$

with prob $> 1 - \phi$.

Suppose the shattering dim is d.

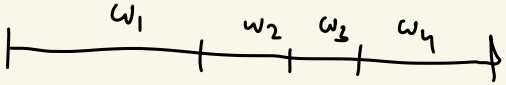
$$\text{Sample size} \geq O\left(\frac{d}{\varepsilon} \log \frac{d}{\varepsilon}\right)$$

Weighted net: suppose the elements are wt, ($w: x \to \mathbb{R}^+$)

r-subset

$$w(r) = \sum_{j \in r} w(j)$$

Goal: all the γ's with wt. $\geq \varepsilon \times w$



Algorithm for set cover

(1) Repeatedly select an ε-net (for some ε)

$S = (X, R)$ dual $- S^* = (X^*, R^*)$

shattering dim of S^*

Size of the net $= O\left(\frac{\delta^*}{\varepsilon} \log \frac{\delta^*}{\varepsilon}\right)$

Verify if it is a net. If not – discard.

↳ if it is a net check if it is a set cover

If yes – done

Let $R_p = \{\gamma \in R \mid p \in \gamma\}$ all ranges that contain p.

Double the weight of the elements in R_p

Observation: every time we double we are increasing not more than $(1+\varepsilon)$ multiplicative function

$$w_i \leq (1+\varepsilon)^i w_0$$

$i \rightarrow$ iteration

Q. what is the min wt. of elements ($k = |opt|$) in opt?

$$k \times 2^{i/k} \leq w_i = (1+\varepsilon)^i w_0 = (1+\varepsilon)^i \times m \leq e^{\varepsilon i} \times m$$

$$k \times 2^{i/k} \leq w_i$$

$$i = k \times q$$

$$k \times 2^q \leq e^{\varepsilon i} \times m$$

$$\log k + q \leq \log m + \varepsilon i$$

$$q(1 - \varepsilon k) \leq \log m - \log k = \log\left(\frac{m}{k}\right)$$

Suppose, we take $\varepsilon = \frac{1}{2k}$

$$\Rightarrow q \leq O\left(\log \frac{m}{k}\right)$$

# of iteration $O\left(k \log \frac{m}{k}\right)$

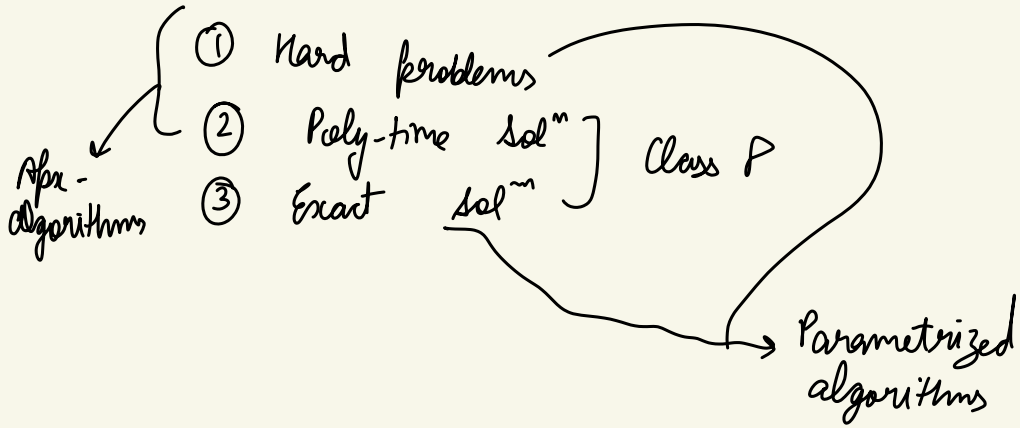
Size of our cover $O(S^* k \log S^* \times k)$ $k = opt.$ cover

Q. How to choose ε?

ε is independent on k

Suppose $\varepsilon = \frac{1}{4k}$ instead of $\frac{1}{2k}$ Guess the value of ε $\frac{1}{k_i}$

$O\left(k_i \log \frac{m}{k_i}\right)$ iterations



Idea: Aim is to get exact algo
But we want to isolate exp^n terms (parameters)
⇒ obtain very fast sol^n when the parameter small.
(Hope: parameters are small in practice).

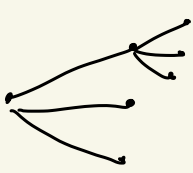
parameters — non-negative integer $k(x)$ (Comes with prob i/p)
- denote by k
- Not necessarily efficiently computable.

Parametrized Problem

problem + parameter (k)
(w.r.t. k)

Goal: poly. complexity on n
Exp^n complexity on k

Example:



I/p: $G = (V, E)$ $k \in \mathbb{N}$
O/p: Does there exist a k-size vertex cover

output a set ($\leq k$) st. $\forall e \in E$
$\exists v \in S$
which has e

Brute force solution

(1) Try all $\binom{n}{k} + \binom{n}{k-1} + \dots + \binom{n}{0}$

All sets of k vertices

— Test valid VC takes $O(E)$ time

— Total = $O(V^k E)$ poly for fixed k.

slow for large n and reasonable k.

Branching (Bounded search tree technique)

→ Pick an arbitrary edge $e \in E$
(u, v)

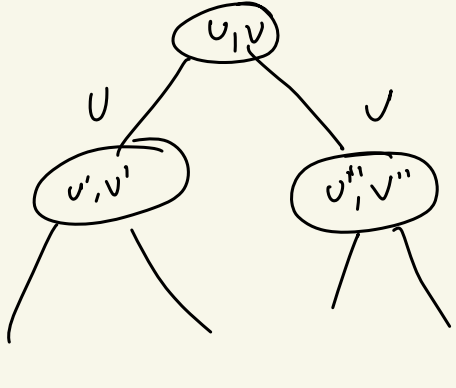
→ Know either $u \in S$ or $v \in S$ ~~or~~ $\{u, v\} \in S$

Guess — Try both

① Add u to S
(delete u & N(u) from S)
recurse $k' = k - 1$

② same for v.

— Return or of the outcomes



depth $\leq k$

Bottom-up

$O(2^k \times V)$

$k \leq \log V \rightarrow$ good

Fixed parameter - tractable (FPT)

If $\exists$ an algo with running time $\leq f(k) \times n^{O(1)}$

$f: \mathbb{N} \to \mathbb{N}$ — parameter

$n^{O(1)}$ — polynomial

Q: why $f(k) \times n^{O(1)}$ and not $f(k) + n^{O(1)}$?

Thm: $f(k) \times n^c \iff f(k) + n^{c'}$

Proof: $\Rightarrow$ if $n \geq f(k)$

$f(k) \times n^c \leq f(k)^{c+1}$

if $f(k) \leq n$

$f(k) \times n^c \leq n^{c+1}$

$\therefore f(k) \times n^c \leq \max \{ f(k)^{c+1}, n^{c+1} \}$

$\leq f(k)^{c+1} + n^{c+1}$

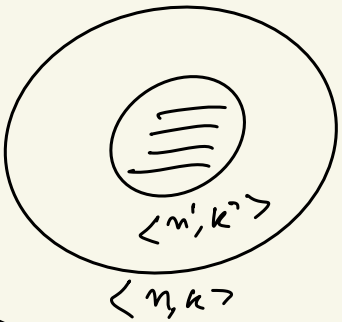
$\Leftarrow$ Trivial, assuming $f(k)$ & $n^{c'} \geq 1$

Kernelization

Simplifying self-reduction (poly-time reduction)

i/p — $\langle n, k \rangle$

converts it into $\langle n', k' \rangle$



How small? $|n'| \leq f(k)$

Equivalent — $Ans((n,k)) = Ans((n',k'))$

Thm: FPT $\iff$ kernelization

Proof: ($\Leftarrow$)

kernelization $\Rightarrow$ $n' \leq f(k)$

run any finite $g(n')$

$\Rightarrow$ $n^{O(1)} + g(f(k))$ time $\rightarrow$ FPT

($\Rightarrow$)

k is known in advance

A runs in $f(k) > n^c$

if $n \leq f(k)$ is kernelized

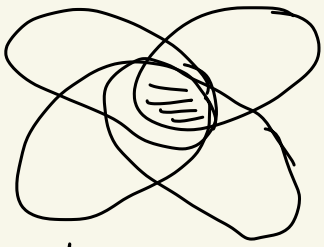
if $f(k) \leq n$

run A $\rightarrow$ $f(k) \times n^c \leq n^{c+1}$

o/p , $O(1)$ size

Sunflower Lemma (Erdos Rado Sons)

- Classical result from 1960
- Apply in kernelization



- k petals
- A core Y

→ Collection of sets $S_1, \ldots, S_k$ s.t. $S_i \cap S_j = Y$ $\forall i \neq j$

(* None of them can be empty)

Petals $\forall i$ $S_i \setminus Y$

Lemma F - family of sets (no duplication) over a universe U s.t. each set has cardinality exactly d.

- If $|F| > d! (k-1)^d$ then F contains k petals

- Poly-time algorithm to compute this
  - w.r.t → $|F|, |U|, k$

Proof: for $d=1$, singletons

Suppose $d > 1$

Let $G = \{S_1, S_2 \ldots S_\ell\} \subseteq F$ be inclusion-wise maximal family of pairwise disjoint sets in F

- If $\ell \geq k$ then G is a sunflower already with at least k petals
- Assume $\ell < k$

$S = \bigcup_{i=1}^{l} S_i$ Then $|S| \geq d \times (k-1)$

Since G is maximal, every set $A \subseteq F$ intersects at least one set from G. $A \cap S \neq \phi$.

There is an element $v \in U$ which is contained in atleast $\frac{|F|}{|S|}$ sets.

$$\frac{|F|}{|S|} > \frac{d!\,(k-1)^d}{d(k-1)} = \underbrace{(d-1)!\,(k-1)^{d-1}}_{\text{Sets for } F}$$

- Take all sets of F containing this element u.
Construct F' of sets union cardinality $(d-1)$ by removing v.

$$|F| > d!\,(k-1)^d$$

By induction hypothesis

F' contains a sunflower $\{S_1' \dots S_k'\}$ with k-petals

$$\{\{S_1' \cup u\} \dots \{S_k' \cup v\}\}$$

Poly-Time Algorithm

(1) Greedily select maximal sets. If size is at least k done. Else find u and return.

Erdos-Rado - FIGO

Each set in F has cardinality d

If $|F| > d!(k-1)^d$ then there is a sunflower.

$(\log k)^d \longrightarrow$ recent bound.

d-Hitting set

(Application of sunflower Lemma)

Input: Family of sets A over U. each set has cardinality at most d.

a non-negative integer k

Output: whether there is a subset $H \subseteq U$ of size at most k.

such that H contains 1 element of each of sets of A.

Proof: If A contains a sunflower say $S = \{S_1 \dots S_{k+1}\}$ of $\#(k+1)$ then every hitting set H of A of cardinality at most k intersects its core Y.

Reduction rule: (U, A, k)

Return (U', A', k') $A' = (A \setminus S) \cup \{x\}$

and $U' = U \setminus_{x \in A'} x$

If # of sets are larger than $d! \times k^d$ find a sunflower — Apply reconsider kernel size $O(d!k^d)$

Kernelization

A data reduction rule for a parametrized problem Q is a function $\phi : \Sigma^* \times \mathbb{N} \to \Sigma^* \times \mathbb{N}$ that maps an instance (I,k) of Q to an equivalent instance (I',k') of Q such that ϕ is computable in time polynomial in $|I|$ and k.

$$\text{Size}_A(k) = \sup \{ |I'| + k' : (I',k') = A(I,k), I \in \Sigma^* \}$$

* A kernelization algorithm for a parametrized problem Q is an algorithm A that, given an instance (I,k) of Q works in polynomial time and returns an equivalent instance (I',k') of Q. Moreover, we require that $\text{size}_A(k) \leq g(k)$ for some computable function $g : \mathbb{N} \to \mathbb{N}$.

* If a parametrized problem Q is FPT then it admits a kernelization algorithm

Proof: $Q = FPT \Rightarrow \exists A$ $(I,k) \in Q$ in time $f(k)|I|^c$

(I,k) algo runs A on (I,k) for at most $|I|^{c+1}$ steps

If it terminates with an answer, use that for yes/no.

If A does not terminate within $|I|^{c+1}$ steps, then return (I,k) itself

$$f(k) \cdot |I|^c > |I|^{c+1} \Rightarrow |I| < f(k)$$

$$|I| + k \leq \underbrace{f(k) + k}_{\text{computable}} \quad \text{(kernel size)}$$

Sunflower Lemma

A sunflower with k petals and a core Y is a collection of sets $S_1 \dots S_k$ such that $S_i \cap S_j = Y$ for all $i \neq j$; the sets $S_i \setminus Y$ are petals and we require none of them to be empty (Y can be empty).

* Let A be a family of sets (without duplicates) over a universe U, such that each set in A has cardinality exactly d. If $|A| > d!(k-1)^d$, then A contains a sunflower with k petals and such a sunflower can be computed in time polynomial in $|A|$, $|U|$ and k.

For $d=1$, family of singletons, statement holds

$d \geq 2$ A = family of sets of cardinality at most d over a universe U such that $|A| > d!(k-1)^d$.

$G = \{S_1 \dots S_\ell\} \subseteq A$ be an inclusion-wise maximal family of pairwise disjoint sets in A.

If $\ell \geq k$ then G is a sunflower with at least k petals.

G is maximal, every set $A \in \mathcal{A}$ intersects at least one set from G i.e. $A \cap S \neq \emptyset$.

$$S = \bigcup_{i=1}^{\ell} S_i \qquad |S| \leq d(k-1)$$

There is an element $v \in V$ contained in at least

$$\frac{|\mathcal{A}|}{|S|} > \frac{d!\,(k-1)^d}{d(k-1)} = (d-1)(k-1)^{d-1}$$

sets from $\mathcal{A}$. We take all sets of $\mathcal{A}$ containing such an element v, and construct a family $\mathcal{A}'$ of sets of cardinality $d-1$ by removing from each set the element v. Because $|\mathcal{A}'| > (d-1)!\,(k-1)^{d-1}$, by the induction hypothesis $\mathcal{A}'$ contains a sunflower $\{S_1', \ldots, S_k'\}$ with k-petals. Then $\{S_1' \cup \{v\}, \ldots, S_k' \cup \{v\}\}$ = sunflower with k-petals.

d-hitting set

Given a family $\mathcal{A}$ of sets over a universe U, where each set in the family has cardinality at most d, and a positive integer k. The objective is to decide whether there is a subset $H \subseteq U$ of size at most k.

such that H contains at least one element from each set in A.

* d-Hitting sets admits a kernel with at most $d! k^d$ sets and at most $d! k^d \cdot d^2$ elements.

Let (U, A, k) be an instance of d-hitting set and assume that A contains a sunflower $S = \{S_1 \cdots S_{k+1}\}$ of cardinality $k+1$ with core Y. Then return (U', A', k) where $A' = (A \setminus S) \cup \{Y\}$ is obtained from A by deleting all sets $\{S_1 \cdots S_{k+1}\}$ and by adding a new set Y and $U' = \bigcup_{X \in A'} X$.

Additional Notes

* Voronoi Partitions: set of centers C, every point of P assigned to nearest neighbour in C

$$\Pi(C, \bar{C}) = \{ p \in P \mid d(p, \bar{C}) \leq d(p, C) \}$$

* Greedy clustering algorithm: arbitary point $\bar{c}_1$ into C_1
  for every point $p \in P$ compute $d_1[p]$ from $\bar{c}_1$
  Pick point $\bar{c}_2$ with highest distance from $\bar{c}_1$
  Add this to the set of centers and denote this expanded set of centers as C_2.

  overall algorithm = $O(nk)$

  → This algorithm is 2-apx.

Proof: Case-1 Every clusters of C_{opt} contains exactly one point of k.

$p \in P$

$\bar{c}$ = center p belongs in C_{opt}

$\bar{k}$ = center of k that is in $\Pi(C_{opt}, \bar{C})$

$d(p, \bar{c}) = d(p, C_{opt}) \leq r_\infty^{opt}(P, k)$

$d(\bar{k}, \bar{C}) = d(\bar{k}, C_{opt}) \leq r_\infty^{opt}$

$d(p, \bar{k}) \leq d(p, \bar{c}) + d(\bar{c}, \bar{k}) \leq 2 r_\infty^{opt}$

Case-1: Two centers $\bar{k}$ and $\bar{v}$ of k both in $\Pi(C_{opt}, \bar{C})$

v was added later

$$r_\infty^{k}(P) \leq r_\infty^{C_{i-1}}(P) = d(\bar{v}, C_{i-1})$$
$$\leq d(\bar{v}, \bar{k})$$
$$\leq d(\bar{v}, \bar{c}) + d(\bar{c}, \bar{k})$$
$$\leq 2 r_\infty^{opt}$$

* A set $S \subseteq P$ is a r-net for P if the following two properties hold: (i) Covering property = All the points of P are in distance at most r from the points of S.
(ii) Separation property = for any pair of points $p, q \in S$
$d(p,q) \geq r$.

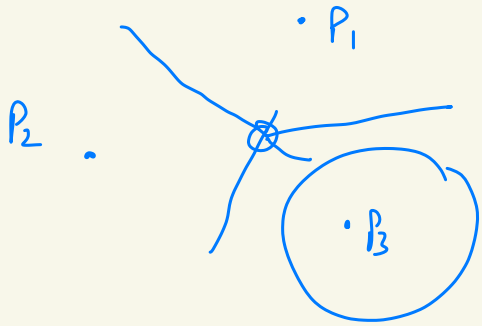
* Let P be a set of n-points in a finite metric space, and let its greedy permutation be $\langle \overline{c_1}, \ldots, \overline{c_n} \rangle$ with the associated sequence of radiuses $\langle \overline{r_1} \ldots \overline{r_n} \rangle$ for any i, $C_i = \langle \overline{c_1} \ldots \overline{c_i} \rangle$ is a r_i-net of P.

* $0 < p < q$ $\quad ||x||_p \geq ||x||_q$

$||x||_1 \leq \sqrt{n}\ ||x||_2$ and $||x||_2$

Nearest Neighbour Search

- Let P be a set of n points in $\mathbb{R}^d$ pre process the pts in P.
- s.t. given a query pt. q, we can determine efficiently the closest pt to q in P.



worst possible running time $= O(n^2)$

Voronoi diagram

$O(n \log n)$ (through a complex algo)

- Voronoi Diagram $O(n \log n)$
- Apply point location query roughly takes $n^{\lceil d/2 \rceil}$ time.

Approximate NN

- Specify a parameter $\varepsilon > 0$

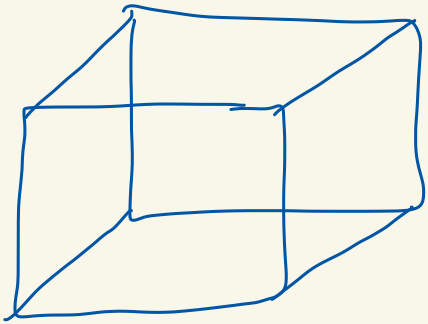
Build a data structure that ans. $(1+\varepsilon)$-Apx NN (ANN)

Input: P in $\mathbb{R}^d$: for a query pt. q

$nn(q) = nn(q, P)$ closest point of q in P

$d(q, P)$ = distance of q to the closest point in P

$\|q - s\| \leq (1+\varepsilon)\, d(q, P) \rightarrow$ Find s.



Spread $= \ell$ is bounded

Unit hypercube in $\mathbb{R}^d$

$\mathcal{T}$ is a quadtree of the space

$\text{diam}(P) = \Omega(1)$

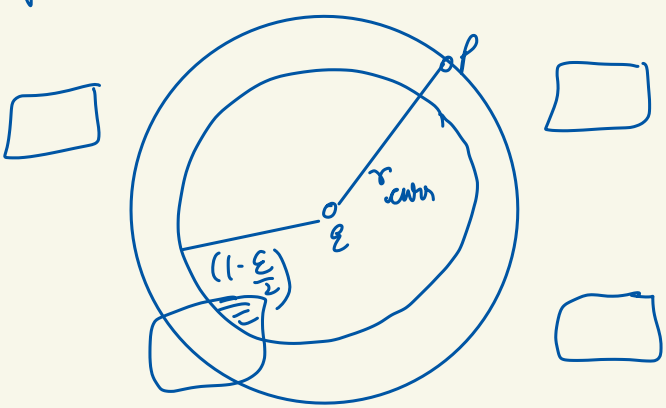
Assume for each (internal) node u in $\mathcal{T}$

$\exists$ a representative (repu) is one of pts of P stored that rooted at u.

Idea: 1. Maintain a set of nodes of $\mathcal{T}$ that might contain ANN of q.

2. Each node has a representative (compute the distance to the query q)

3. At each (1), the search gets refined by replacing a node by its children.



If $\|q - \text{rep}_w\| - \text{diam}(\square_q) > (1-\frac{\varepsilon}{2})\, r_{curr}$ $\Rightarrow$ Abort

If not then keep doing

Algo:

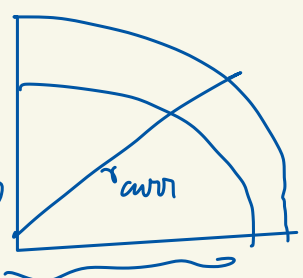
Let $A_0 = \{root(T)\}$ $r_{curr} = ||q - rep(root(T))||$

- In the i^{th} iteration
  - for $i>0$ Alg expands the nodes of A_{i-1} to get to A_i
  - (if cond. not satisfied) add w to A_i
  - Continue until all elements of A_{i-1} are considered
  - stop when A_i is empty.

Correctness:

Alg adds a node w to A_i only if P_w is not contain pts which are close to q than current best.

- Say w is the last node inspected by Alg s.t. $nn(q) \in P_w$



if it throws away:

$$||q - nn(q)|| \geq ||q - rep_w|| - ||rep_w - nn(q)||$$

$$\geq ||q - rep_w|| - diam(\square_w)$$

$$\geq (1-\tfrac{\varepsilon}{2})\, r_{curr}$$

so, $||q - nn(q)|| / (1-\frac{\varepsilon}{2}) \geq r_{curr}$

$(1-\frac{\varepsilon}{2}) \leq (1+\varepsilon)$ $\quad r_{curr} \leq (1+\varepsilon) \times d(q,P)$

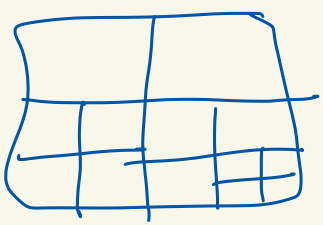
Running Time : Algorithm visits T level by level

As long as level's grid cells are bigger than ANN (# nodes visited so far)

running time $= O\left(\varepsilon^{-1} + \log\left(\frac{1}{w}\right)\right)$

↓ Prop. to spread.

Compressed Quadtree



Alg handled nodes level by level

(keep a heap of integers in the range of $O(\log \phi(P))^1 \ldots \log(\phi(P))$

$O\left(\varepsilon^{-d} + \log\left(\frac{\text{diam}(\tau)}{w}\right)\right)$

General unbounded case:

(a) Get rough apx

(b) Apply previous arguments

Embedding finite Metric spaces into Normed spaces (Metric Embedding)

Recap : Metric space is a pair (X, d), X is a set $\rho: X \times X \to [0, \infty)$ is a metric if the following happens :

(I) $\rho(x,y) = 0$ if $x = y$

(II) $\rho(x,y) = \rho(y,x)$

(III) $\forall x, y$ $\rho(x,y) + \rho(y,z) \geq \rho(x,z)$ (Triangle inequality)

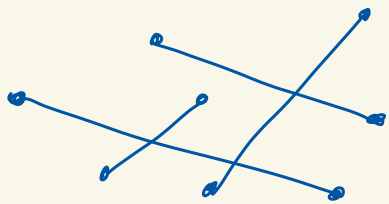
A metric on n points $(n \times n)$ matrix $\binom{n}{2}$

Applications

Microbiology — X is a collection of bacterial strains.

"Nice" Representation

P in $\mathbb{R}^2$



assign $\forall x \in X$ to a point $f(x)$ in $\mathbb{R}^2$ s.t.,

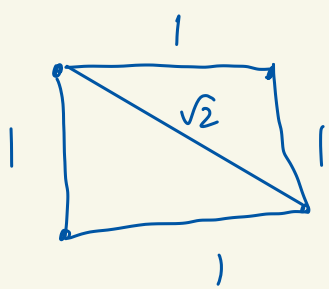
$$f(x,y) \approx \| f(x) - f(y) \|_2$$

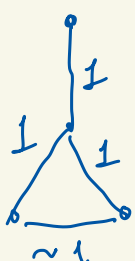
Identify clusters, isolated entries and so on.

Geometry helps to get faster algo.

Too good to be true.

$X = \{ p_1, p_2, p_3, p_4 \}$ → isometric space





Isometric Embedding

A mapping $f: X \to Y$

$X \to$ metric space with a metric ρ

$Y \to$ metric space with a metric σ.

* is isometric if it preserves distances

$$\sigma(f(x), f(y)) = \rho(x, y) \quad \forall\ x, y \in X$$

Typically small "error" (distribution) in embedding is ok

D embedding —

A mapping $f : X \to Y$ is called a D-embedding for $D > 1$ (real numbers) if $\exists\ r > 0$ s.t. $\forall\ x, y \in X$

$$r \times \rho(x, y) \leq \sigma(f(x), f(y)) \leq D \times r \times \rho(x, y)$$

The infimum of numbers D s.t. f is a D-embedding — distortion

If X is a Euclidean space (in a normed space generally). We can scale up/down.

Mapping with bounded D, are called bi-lipschitz

Distortion can be defined in terms of lipschitz constants of f and inverse of f (f^{-1})

$$\|f\|_{lip} = \sup \left\{ \sigma(f(x), f(y)) \Big/ \rho(x, y) \right\}$$
$$\forall\ x, y \in X,\ x \neq y$$

distortion on f

$$\|f\|_{lip} \times \|f^{-1}\|_{lip}$$

ℓ_p - norm pt $x \in \mathbb{R}^d$ $p \in [1, \infty)$

$$\|x\|_p = \left(\sum_{i=1}^{d} |x_i|^p \right)^{1/p}$$

Q: Does $\exists$ some euclidean space into which a given metric space embeds isometrically

$x = (x_1 \dots)$ of reals $\|x\|_p < \infty$

$$\|x\|_p = \left(\sum_{i=1}^{d} |x_i|^p \right)^{1/p}$$

Remark - ℓ_2 is a separable hilbert space
↳ has countable basis.

Known:

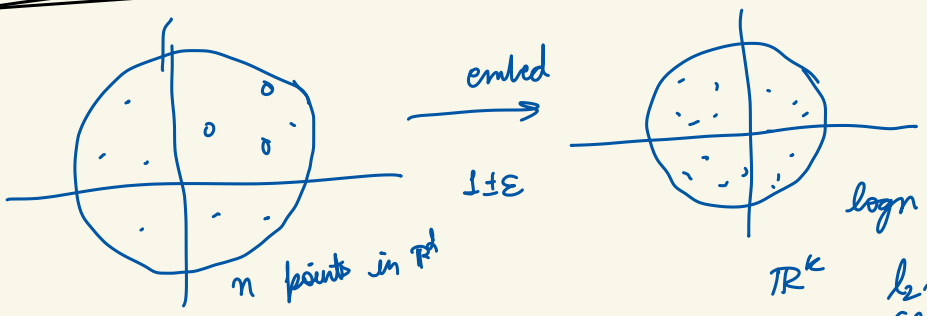
n dim. regular simplex has no isometric embedding in $\mathbb{R}^k$, $k < n$

Johnson - Lindenstrauss Flattening Lemma

X be a n pt. set in $\mathbb{R}^d$ $\varepsilon \in (0,1]$ $\exists$ a $(1+\varepsilon)$ embedding of X into ℓ_2^k $k \approx O(\log n)$

Recap:
- Metric Embedding
- Isometric embedding
- Ex: isometric can't be preserved even for simple cases
- D-embedding

Johnson - Lindenstrauss Lemma

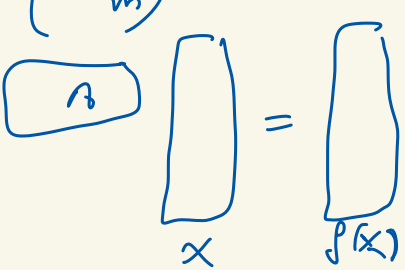


lemma: given any $\varepsilon \in (0,1]$ and a set $X \subset \mathbb{R}^d$ with $|X| = n$
$\exists$ a linear map $f: \mathbb{R}^d \to \mathbb{R}^m$ with $m = O\left(\frac{\log n}{\varepsilon^2}\right)$ that extends (X, d_2) to $(\mathbb{R}^m, d_2)$ with distortion $(1 \pm \varepsilon)$

Idea: Use randomness to define the map f
$A \in \mathbb{R}^{m \times d}$ be a random matrix with entries $N\left(0, \frac{1}{m}\right)$
chosen i.i.d from Gaussian.

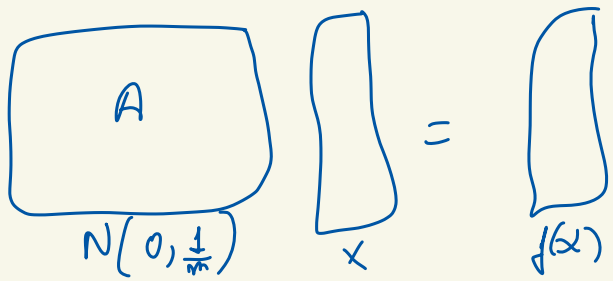
$$f(x) = Ax \quad \forall x$$

Hope - this exists with high probability
Goal - Identify such a f^m



Geometry: If the rows of A were orthogonal then $\exists$ orthogonal projection onto the row span of A.

Pick a random subspace given by the row span of A.
Project X onto that subspace.



Proof: multiplying A preserves the ℓ_2 distances

Fix $x, y \in \mathbb{R}^d$ show with high probability P

$$\underbrace{\|A(x-y)\|_2}_{\|f(x)-f(y)\|_2} = (1 \pm \varepsilon)\|x-y\|_2$$

Then we will take the union bound of all $x, y \in X$ $\leq n^2$ pairs.

Fact: Gaussian distribution = spherically symmetric

$A(x-y)$ is sort of scaled version of the first column

Equivalent to show that $\|y\|_2 = 1 \pm \varepsilon$

Chernoff Style Bound

We have a set of ind. gaussian random variables

$$z_1, z_2, \ldots, z_m \sim \mathcal{N}(0,1)$$

$$\mathbb{P}\left[\sum_i z_i^2 > (1+\varepsilon)m\right] \leq \underbrace{e^{-m\varepsilon^2/8}}_{\text{Small}}$$

Let's assume this happens.

$\|y\|_2 = 1 \pm \varepsilon$ with high p

Y is a random vector with entries $\sim N\left(0, \frac{1}{m}\right)$

$$\mathbb{P}\left[\|y\|_2 \geq 1+\varepsilon\right] = \mathbb{P}\left[\|y\|_2^2 > (1+\varepsilon)^2\right]$$
$$\leq \mathbb{P}\left[\|y\|_2^2 > 1+\varepsilon\right)$$

def. of l_2 - norm + eqn. 1

$$\mathbb{P}\left[\sum_i z_i > m(1+\varepsilon)\right] \leq e^{-m\varepsilon^2/8}$$

$$\mathbb{P}\left[\|Ax(x-y)\|_2 \geq (1\pm\varepsilon)\|x-y\|_2 \leq e^{-m\varepsilon^2/8}\right.$$

Union bound $\forall\, x, y \in X \approx O(n^2)$

$$\mathbb{P}\left(\exists x,y \in X \text{ s.t. } \|Ax(x-y)\|_2 \geq ((1+\varepsilon)\|x-y\|_2\right.$$
$$\leq n^2 \times e^{-m\varepsilon^2/8}$$

Q: what is m? $O\left(\frac{\log n}{\varepsilon^2}\right) \approx \frac{1}{\text{poly}(n)}$

Fast JL transformation

- matrix multiplication is slow
- Via FFT (fast fourier transformation) we can speed up.

Motivation

Bring any metric to L_2

Bourgain's Embedding

Transformation from other metric to Euclidean.

[Bourgain 1985) [Linial, London, Rabinovich 1995]

Given any metric space (X,d) with $|X| = n$

$\exists$ an embedding of (X,d) into $(\mathbb{R}^k, d_1)$ where

$k \simeq O(\log^2 n)$

Distortion $\sim O(\log n)$

d_1 — L_1 metric dist.

works for any L_p for $p \geq 1$

Remark: Distortion is tight because some metrics require $\Omega(\log n)$. Sometimes, it's possible to improve the dim.

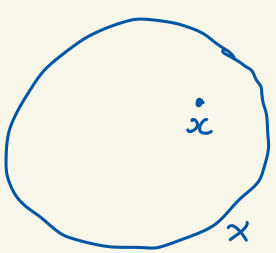
Algorithm:

- For $i = 1, 2, \ldots, \log n$ → const. (will decide later)
- For $j = 1, 2, \ldots, c \times \log n$
- Choose a set $S_{i,j} \subseteq X$ at random such that every $x \in X$ is contained in $S_{i,j}$ with probability 2^{-i} (independently)

Embedding (def^n):

$$\forall x \in X,\ f(x) = \Big(d(x, S_{1,1})\ d(x, S_{1,2}) \ldots\ d(x, S_{\log n, c\log n}) \Big)$$

d minimum distance between x and any point of the set.



Q: Why this is good?

(i) Non-expansive

(ii) Non-shrinking

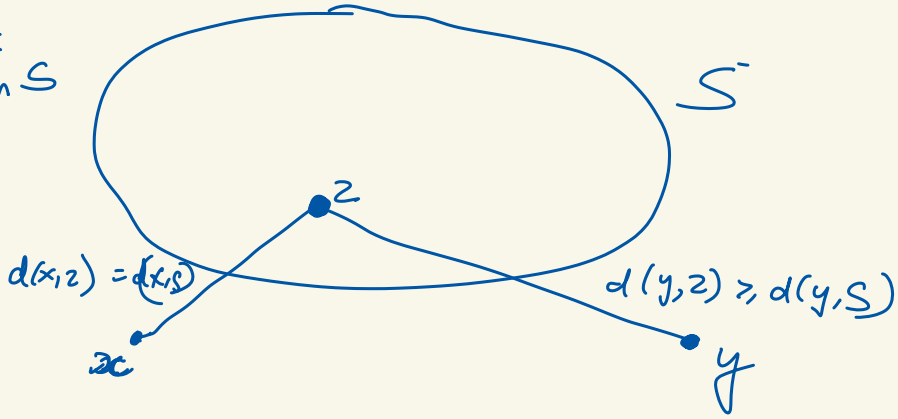
Non-expansive

This map is non-expansive

$\forall x, y \in X$

$$\|f(x) - f(y)\| \leq d(x,y)$$

z is the closest point to x in S



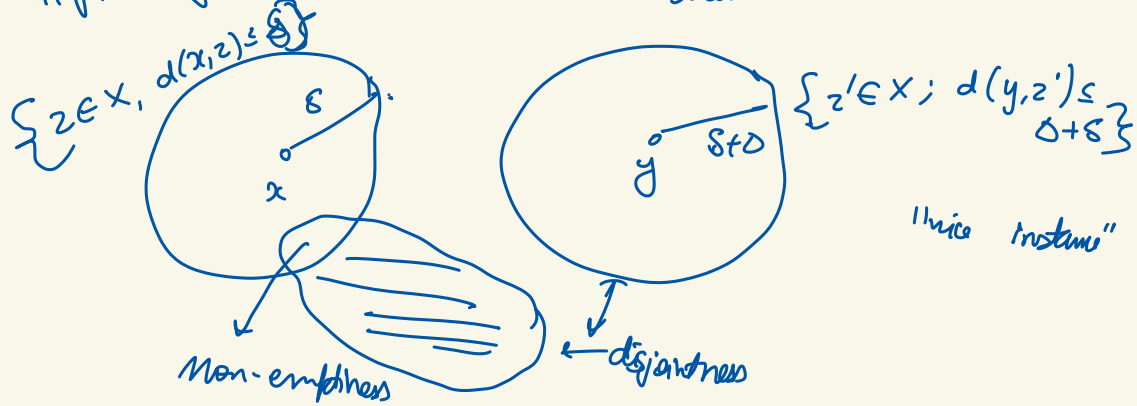
$$f(x) - f(y) = d(y,S) - d(z,S)$$
$$\leq d(y,z) - d(x,z)$$
$$\leq d(x,y)$$

(≤ : Triangle Inequality)

Non-shrinking

$$\|f(x) - f(y)\| \geq \Delta$$

bounding distortion above and below



$d(y,S) - d(x,S) \geq (\delta+\Delta) - \delta = \Delta$

Idea: Pick a lot of random S and our hope is that "nice instance" happens enough number of times. (inner loop) — this happens if S has enough intensity.

— Good choice of S depends on $d(x,y)$ (outer loop) it varies the size of S

Proof: $\underbrace{k\, d(x,y)}_{\log n} \underset{\text{ineq-2}}{\leq} \|f(x)-f(y)\| \underset{\text{ineq-1}}{\leq} k \times d(x,y)$

ineq-1 $\|f(x)-f(y)\|_1 \leq k \times d(x,y)$

enough to show $|d(x,S) - d(y,S)| \leq d(x,y)$ for every S.

$x \longmapsto d(x,S)$ is non-expanding

$$\|f(x)-f(y)\|_1 = \sum_{i,j} |d(x,S_{ij}) - d(y,S_{ij})| \leq k \times d(x,y)$$

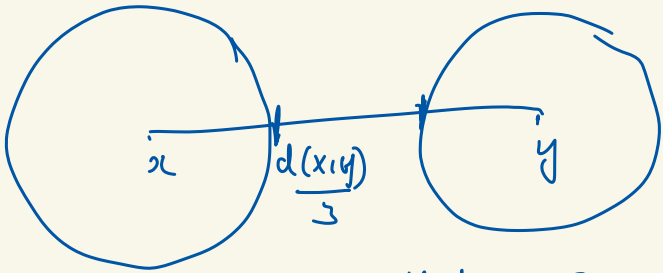
ineq-2 $\underbrace{k \times d(x,y)}_{\log n} \leq \|f(x)-f(y)\|_1$

choose bunch of different deltas (δ's) & "nice" thing holds with good prob.

Fix $x, y = 0$

choose $0 = \delta_0 < \delta_1 < \delta_2 < \dots < \delta_t$

s.t. δ_i δ_i is the smallest δ $B(x,\delta_i)$ & $B(x,\delta_i)$ contain $> 2^i$ pts.



Stop at the largest k so that $\delta_t < \frac{d(x,y)}{3}$

$$\delta_{t+1} = \frac{d(x,y)}{3}$$

nice instance holds $|d(x,S) - d(y,S)| \geq \delta_{t+1} - \delta_t$

Claim: It is very likely that this situation happens.

By defn of δ_i $\quad > 2^i$

$$\mathbb{P}[x \in S_{i,j}] = 2^{-i}$$

$\mathbb{P}\{$nice event happens$\}$ $= \mathbb{P}\{$non-empty intersection happens$\} \times \mathbb{P}[$disjointness happens$]$

$$\geq \left(1 - \left(1 - \frac{1}{2^i}\right)^{2^i}\right) \qquad \geq \left(1 - \frac{1}{2^i}\right)^{2^{i+1}}$$

$$\geq \left(1 - \frac{1}{e}\right) \times \left(\frac{1}{2}\right)^2$$

$$\geq \frac{1}{2^5}$$

Claim: Fix i

$$\mathbb{P}\Big[\geq 2^{-6} \text{ fraction of } S_{i,j} \text{ have nice event} \Big]$$

$$1 - \mathbb{P}\Big[\sum_{j=1}^{c\log n} \mathbb{1}\{\text{nice event}\} \leq \frac{1}{2} \times \frac{c\log n}{2^5} \Big)$$

Invoke Chernoff bound

$$\geq 1 - \exp\left(\frac{-c\log n}{8 \times 2^5}\right) \geq 1 - \frac{1}{n^3}$$ if we choose $c \geq 3 \times 8$

Putting these together

By a union bound $\approx n^2$ pair and $c\log n$ choices of j with high prob. $\forall i, j$

$$r_{ij} \geq \frac{c \times \log n}{2^6}$$

S_{ij} have

$$|d(x, S_{ij}) - d(y, S_{ij})| \geq \delta_{i+1} - \delta_i$$

$$\|f(x) - f(y)\| = \sum_{ij} |d(x, S_{ij}) - d(y, S_{ij})|$$

$$\geq \sum_{ij} \left(\frac{c\log n}{2^6}\right)(\delta_{i+1} - \delta_i)$$

$$= \frac{c\log n}{2^6} \times \delta_{t+1} = \frac{c\log n}{3 \times 2^6} \times d(x,y)$$

$k = c\log n$

$$\geq \frac{k \times d(x,y)}{3 \times 2^6 \times \log n}$$

non-shrinking

Approximate Nearest Neighbour Search in Low Dimension

* Voronoi Diagram $= O(n^{\lceil d/2 \rceil})$
* $(1+\varepsilon)$ - approximate neighbours if $\|q-s\| \leq (1+\varepsilon) d(q,P)$ $s \in P$

Alternatively $\forall\, t \in P \quad \|q-s\| \leq (1+\varepsilon)\, \|q-t\|$

* Spread of a point set P = $\phi(P)$ = ratio of diameter and the distance of the closest pair of P.

* Bounded spread case

Algorithm: $A_0 = \{root(T)\}$ $\quad r_{curr} = \|q - rep_{root(T)}\|$

ith iteration → $i > 0$ $\quad A_{i-1} \to A_i$

$v \in A_{i-1} \Rightarrow C_v$ = set of children of v in T (quadtree of P)

$w \in C_v$ $\quad r_{curr} \leftarrow min(r_{curr}, \|q - rep_w\|)$

Algorithm checks if $\|q - rep_w\| - diam(\square_w) < \left(1 - \frac{\varepsilon}{2}\right) r_{curr}$ if so add w to A_i

Correctness: $\|q - nn(q)\| \geq \|q - rep_w\| - \|rep_w - nn(q)\|$

$\geq \|q - rep_w\| - diam(\square_w)$

$\geq \left(1 - \frac{\varepsilon}{2}\right) r_{curr}$

$\|q - nn(q)\| \Big/ \left(1 - \frac{\varepsilon}{2}\right) \geq r_{curr}$ $\quad \frac{1}{1 - \frac{\varepsilon}{2}} \leq 1 + \varepsilon$

* Let P = set of n points contained inside the unit hypercube in $\mathbb{R}^d$ and let T = quadtree of P, where $\text{diam}(P) = \Omega(1)$.
q = query point, $\varepsilon > 0$ = parameter. $(1+\varepsilon)$-ANN to q can be computed in $O\left(\varepsilon^{-d} + \log\left(\frac{1}{\bar{w}}\right)\right)$ time $\bar{w} = d(q, P)$

Proof: If $w \in T$ considered by the algorithm and $\text{diam}(\square w) < \left(\frac{\varepsilon}{4}\right)\bar{w}$ then

$$\begin{aligned} \|q - \text{rep}_w\| - \text{diam}(\square w) &\geq \|q - \text{rep}_w\| - \left(\tfrac{\varepsilon}{4}\right)\bar{w} \\ &\geq r_{\text{out}} - \left(\tfrac{\varepsilon}{4}\right) r_{\text{out}} \\ &\geq \left(1 - \tfrac{\varepsilon}{4}\right) r_{\text{out}} \end{aligned}$$

$\Rightarrow$ no nodes of depth $\geq h = \left\lceil -\lg\left(\frac{\bar{w}\varepsilon}{4}\right)\right\rceil$ are considered

The number of relevant cells (i.e. cells that the algorithm dequeues and do not get immediately thrown out) = number of grid cells of $G_{2^{-i-1}}$ that intersect a box of sidelength $2l_i$ centered at q:

$$n_i = \left(2\left\lceil \frac{l_i}{2^{-i-1}} \right\rceil\right)^d \leq \left(1 + (2^i \bar{w})^d\right)$$

$$(a+b)^d \leq \left(2\max(a,b)\right)^d \leq 2^d\left(a^d + b^d\right)$$

$$\sum_{i=0}^{h} n_i = O\left(\log \frac{1}{\bar{w}} + \frac{1}{\varepsilon^d}\right)$$

Dynamic Inputs

- Online Algorithms
- Dynamic Algorithms
- Streaming Algorithms

Online Algorithms

I - instance
σ - ordering

when σ_i arrives, we have to make the decision instantly

$\langle \sigma_1, \sigma_2, \ldots, \sigma_n \rangle$

Decisions are irrevocable

measure the optimality

Competitive ratio = $\sup \frac{Alg(\sigma)}{OPT(\sigma)}$

Dynamic

$\longrightarrow$ Eg: B.S.T (Binary Search Tree)

I - instance

$\sigma = \langle \sigma_1 \ \ldots \ \sigma_n \rangle$

insertion and deletion both are allowed.

Goal: Build data structures which can handle insertion + deletion

optimize: Update complexity
Query complexity
optimality

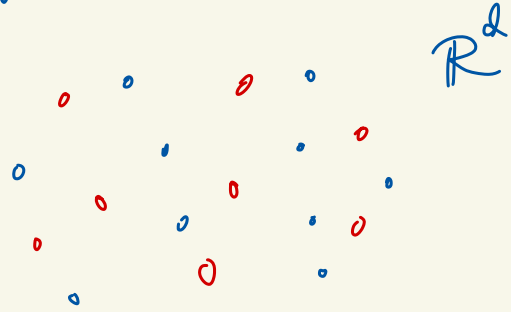
Streaming

I - instance
σ - sequence

optimize: space complexity
& Query & Time

Data Reduction (core set)

A small subset of the input which captures all interesting features



Coreset for directional width

$\exists P - |P| = n$ pts in $\mathbb{R}^d$; for any $d \in \mathbb{N}$

Projecting P onto the direction $v \in \mathbb{R}^d$ in the set

$\rightarrow$ P onto a line l through the origin

Formally, for a vectors $v \in \mathbb{R}^d$ $\quad v \neq 0$

$$\bar{w}(v, P) = \max_{p \in P} \langle v, p \rangle - \min_{p \in P} \langle v, p \rangle$$

Properties: (a) Its translation invariant for any vectors $s \in \mathbb{R}^d$

$$\bar{w}(v, s+P) = \bar{w}(v, P)$$

(b) Direction width scales linearly

(c) for any set $P \subseteq \mathbb{R}^d$

$$\bar{w}(v, P) = \bar{w}(v, \underbrace{CH(P)}_{\text{Convex Hull}})$$

(d) For any $Q \subseteq P$ and any vectors v

$$\bar{w}(v, Q) \leq \bar{w}(v, P)$$

Coreset – $S \subseteq P$ is ε-coreset for directional width if

$$\forall u \in S^{(d-1)}, \quad \bar{w}(u, S) \geq (1-\varepsilon)\, \bar{w}(u, P)$$

Sphere of directions in $\mathbb{R}^d$

Sketch: $X \subseteq Y \subseteq P \subseteq \mathbb{R}^d$

Y is an ε-coreset (for directional width) of P and X is an ε'-coreset of Y. Then X is a $(\varepsilon+\varepsilon')$ coreset of P.

Merge $X \subseteq P \subseteq \mathbb{R}^d$ and $X' \subseteq P' \subseteq \mathbb{R}^d$

Then $(X \cup X')$ is also a coreset of $(P \cup P')$

*

I/P → [ALG] → O/P

whole I/P is known (numbered) in advance

1. Online Algorithms
2. Dynamic Algorithms
3. Streaming Algorithms

Comp. ratio

Update and Query complexity with approx. loss

Space complexity with approx. loss

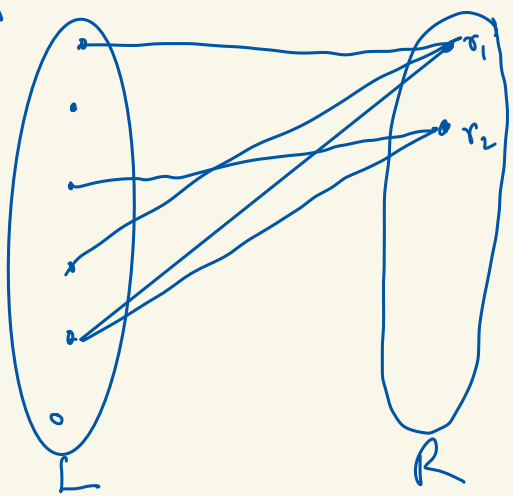
Sketch

- Core set (sketch for compression of large i/ps)

Online Algo.

I - instance

σ - ordering $\langle \sigma_1, \ldots, \sigma_n \rangle$ At step i, σ_i $i \in [n]$



I/P : L is known in advance

R comes one by one

a vertex $v \in R$ arrives with all incident edges

* A vertex $u \in R$ can be matched only at the time of its arrival.

Goal = maintain the maximum matching

Greedy Algo (Deterministic)

When a vertex $v \in R$ arrives look at the neighbourhood (S_v) look at the unmatched neighbours & match with an arbitrary vertex.

Thm (LB) : It is not possible to get better than $\frac{1}{2}$ for any deterministic Alg.

Thm : We can get $\frac{1}{2}$ by using Greedy.

Proof : via weak LP- duality.

$L \cup R = V$

E

Primal: $\max \sum_{e \in E} x_e$ (edge prices)

sub to $\sum_{e \in S(v)} x_e \leq 1$ for all $v \in (L \cup R)$

$x_e \geq 0$ for all $e \in E$

Dual: $\min \sum_{v \in L \cup R} p_v$ (vertex prices)

sub to. $p_u + p_v \geq 1 \quad \forall \ (u,v) \in E$

$p_v \geq 0 \quad \forall \ u \in L \cup R$

for every vertex $v \in L \cup R$

$q_v = \frac{1}{2}$ if greedy matches v

$= 0$ otherwise

* Comparing with the feasible solution in the dual setting.

When the algorithm adds (u,v) to its matching $q_u, q_v, -\frac{1}{2}$

$|M| = \sum_{v \in L \cup R} q_v$

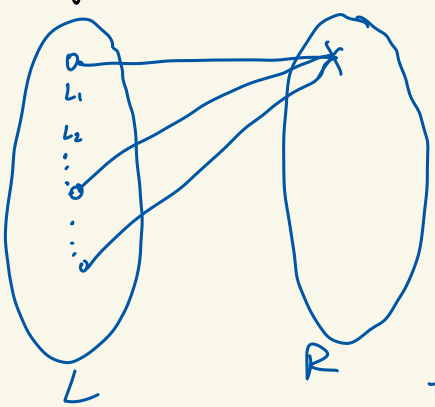
M - matching that greedy algorithm computes

Obs: Finally $(L \cup R, E)$ at least one of q_u, q_v is $\frac{1}{2}$ if not both

Implies if we scale up q by a factor 2 $\quad p = 2q$

$|M| = \frac{1}{2} \sum_{v \in L \cup R} p_v \geq \frac{1}{2} \times OPT$

Online Bipartite Matching



I/P : left side of the bipartite matching

O/P : maximum matching maintained

Comp. ratio $\sup_{\sigma} \frac{ALG(\sigma)}{OPT(\sigma)}$

Thm: There exist no deterministic online alg. with comp ratio better than $\frac{1}{2}$.

Thm: This is tight (eg: greedy algo)

Random matching

- whenever a vertex (r_i) appears in R (for any $i \in (n)$)
- Pick a random unmatched vertex (say l_j) match.

Exp. size $\frac{n}{2} + O(\log n)$

Matrix $B_{ij} = 1$ if $i = j$ or if $\frac{n}{2} \le j \le n$, $1 \le i \le \frac{n}{2}$

$= 0$ otherwise

Ranking

Initialization: Pick a random permutation (say, π) of the left side.

Matching phase: when a vertex arrives (r_k) match r_k to vertex ($l_j \in L$) with highest rank.

Lower Bound claim: Thm: It is not possible to get better than $n\left(1 - \frac{1}{e}\right) + O(n)$

T — $n \times n$ — complete upper-triangular matrix

Assume that columns of T arrive in the order $\langle n, n-1, \ldots 1 \rangle = \sigma$

k^{th} column arrival — $(n-k+1)$

Defⁿ: $T - n \times n$

With every permutation π on $\{1, \ldots, n\}$ associated with $\langle T, \pi \rangle$. Let P be the uniform prob. distribution over $n!$ instances.

Lemma :- A deterministic algorithm "greedy" then exp. size of the matching which A computes $\langle T, \pi \rangle$ is same as the exp size of the matching that "random" computes.

Lemma :- The performance of some online algorithm (Alg) is upper bounded by exp. size what random achieves on $\langle T, \pi \rangle$

Proof :- For Alg A for $\langle T, \pi \rangle$ as wet "random" if we have k available rows at some time t. Then there are equally likely to be the k rows for the rows for the first $n-t+1$ rows of T.

② for each k, the prob. that there are k eligible options. at time t is same for random as it is for A.

Proof: Let $\mathbb{E}[R, \langle T, \pi \rangle]$ - exp size of matching by some randomized Alg.

$$\mathbb{E}[Alg] \min_{\pi} \{ \mathbb{E}[R, \langle T, \pi \rangle] \} \leq \max_{A} \{ \mathbb{E}[A(P)] \}$$

↓

Yao's min-max principle

Yao's Lemma

For any randomized Algo ∃ a prob. distribution over the I/P for Alg. s.t. the exp cost of Alg on its worst case is as large as the best deterministic Alg on a random i/p from this same distribution.

Lemma:- Expected size of matching by a random alg. over T is $n\left(1-\frac{1}{e}\right) + o(n)$

Proof: $x(t)$, $y(t)$ → random variables — # of columns remaining and # of rows still available at t

$$\Delta(x) = x(t+1) - x(t)$$
$$\Delta(y) = y(t+1) - y(t)$$
$$E[\Delta y] = -1 - \frac{y(t)}{x(t)} \cdot \frac{y(t)-1}{y(t)}$$
$$= -1 - \frac{y(t)-1}{x(t)}$$
$$\frac{E[\Delta y]}{E[\Delta x]} = 1 + \frac{y(t)-1}{x(t)}$$

Karp's thm: As the prob. tending to 1 as n tends to ∞

$$\frac{dy}{dx} = 1 + \frac{y-1}{x}$$
$$y = 1 + x\left(\frac{n-1}{n} - \log\frac{x}{n}\right)$$

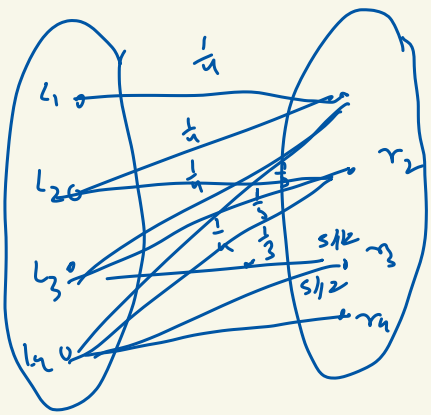
Waterfilling Algorithm

Each vertex $v \in L$ as a containes of capacity 1.

→ when a vertex $w \in R$ arrives, it pours water of unit 1.

Pour in the current lowest containes

– keep doing until equal



Dynamic Optimality

Binary Search Trees (BST)

– AVL

– Red-Black

operations → O(log n)

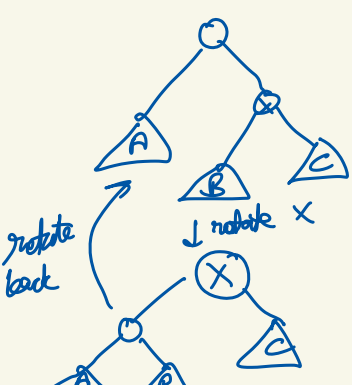
Q: Is this the best complexity that we can hope for?

BST-model (pointer-machine model)

Data is stored as keys in BST.

Operations: left, right, parent

– rotation



search(x)

- start at root
- must visit node with key x.

$\exists$ LB can't beat $\log n$ (Adversarial)

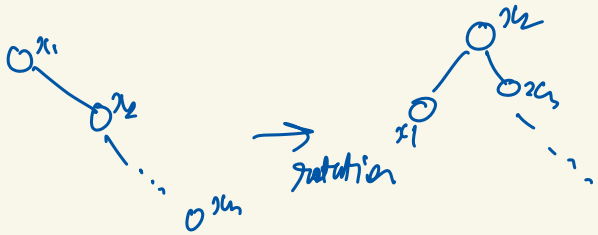
Q) Is this $\log n$ always required

- Depends on the sequence (σ) that we have.

Keys - $\{1, 2, \ldots, n\}$

Sequence σ - $\langle x_1 < x_2 < x_3 \ldots < x_n \rangle$

Maintain a data structure s.t. amortized is $O(1)$



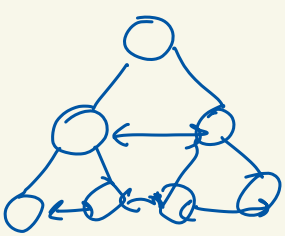
Sequential access property

Stronger Version:

Dynamic finger property

$|x_i - x_{i-1}| = k$

$\Rightarrow O(\log k)$ operations



Static OPT / Entropy Bound

k appears p_k fraction of time

$$O\left(\sum_{i=1}^{n} p_i \log \frac{1}{p_i}\right)$$

amortized per operation

Working set property (roughly output sensitive)

If t_i distinct keys have been accessed so far

$O(\log t_i)$/operation

Unified property

t_{ij} distinct keys accessed in $x_i, \ldots, x_j$ then x_j costs

$$O\left(\log \min_{i} \left[(x_i - x_j) + t_{ij} + 2\right]\right)$$

Generalizes dynamic finger property

Conjecture – $O(1)$ complexity

Total cost of access keys $= O(OPT)$

What is OPT? (offline) minimum over all BST Algo over the access seq. X.

Q: Does $\exists$ online BST which $O(1)$ to the offline OPT.

$O(\log n)$ → trivial

$O(\log \log n)$ → Best known

Splay tree, Greedy Conj.

Primal Dual Method

Linear programming

$A \rightarrow m \times n$ matrix

$x \rightarrow n$ vectors

$b \rightarrow m$ vectors

$w \rightarrow n$-vectors

$Ax \geq b$

$x \geq 0$

$\min(w^T x)$

$y^T A x \geq y^T b$

$y \rightarrow m$-vectors

multiply i^{th} inequality by y_i

$y_i \geq 0$

add up all the inequalities

suppose this holds

$y^T A \leq w^T$

$w^T x \geq y^T A x$

$\geq y^T A b$

for any solution x to primal and y for the dual

$\max(y^T b)$

$y^T A \leq w^T \iff A^T y \leq w$

$y \geq 0$

→ dual of the linear program

Suppose (x^*, y^*) are solutions to the primal and dual that satisfy following conditions -

(i) if $x_i^* > 0$ the dual constraint corresponding to x_i is satisfied exactly (inequality is an equality).

→ i^{th} row of A^T

(ii) if $y_i^* > 0$ the primal constraint corresponding to j^{th} row in A is tight (an equality).

→ Complimentary slackness

If these conditions are satisfied then the cost of x^* and y^* are the same and both are optimal solutions for respective LPs.

$$w^T x^* = \sum_{i=1}^{n} w_i x_i^* = \sum_{i=1}^{n} \left(\sum_{j=1}^{m} a_{ij} y_j^*\right) x_i^* = \sum_{i, x_i^* > 0} \left(\sum_{j=1}^{m} a_{ij} y_j^*\right) x_i^*$$

$x_i^* > 0$ because the corresponding inequality is tight

$$= \sum_{i, x_i^* > 0} \left(\sum_{j, y_j^* > 0} a_{ij} y_j^*\right) x_i^*$$

$$= \sum_{j, y_j^* > 0} \left(\sum_{i, x_i^* > 0} a_{ij} x_i^*\right) y_j^*$$

$$= \sum_{j, y_j^* > 0} b_j y_j^*$$

second condition of complimentary slackness

$$= y^{*T} b$$

if $x_i^* > 0$ then the i^{th} row $A^T y = w^T$, $y \geq 0$ is an equality

Integer Linear Programming

Linear programs with the additional restriction that the variables are required to take integer values

↳ NP Hard

* Vertex Cover in a graph

Given an undirected graph with positive weights assigned to vertices, find a min-wt subset of vertices such that every edge has at least one endpoint in the subset.

$x_i \rightarrow$ for each vertex i

$x_i = 0$ means vertex i is not in the subset.

$x_i = 1$ means it is in the subset.

wt. of subset $= \sum_{i=1}^{n} w_i x_i$

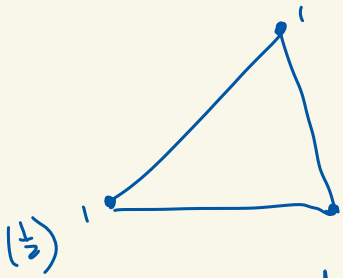
ILP formulation

$$\min \sum_{i=1}^{n} w_i x_i$$

for every edge ij $\quad x_i + x_j \geq 1$

for all vertices i $\quad x_i \geq 0$

LP-relaxation allow fractional values for x_i



optimal LP solution but not a vertex cover.

any dual LP solution $\leq$ max dual LP solution $=$ min LP solution $\leq$ min integer solution $\leq$ any integer solution

dual vertex cover

$$\min \sum w_i x_i$$

$$x_i + x_j \geq 1 \quad \text{each edge}$$

$$x_i \geq 0$$

y_{ij} for each edge ij

$y_{ij} \geq 0$

for every vertex i

$$\sum_{\text{all edges } ij \text{ incident with } i} y_{ij} \leq w_i$$

Approximate complimentary slackness

if $x_i^* > 0$ then corresponding inequality is tight

if $y_i^* > 0$ then corresponding primal inequality is approximately tight.

instead of requiring equality, we allow LHS to be some constant c times RHS.

even this could be approximate

initially start with all dual variables = 0 and $x_i = 0$

→ Pick any edge and start increasing its dual value.

↳ do this till the dual inequality corresponding to at least one of the endpoints becomes tight.

↳ Select the endpoint for which the inequality becomes tight. (include in vertex cover) and delete all edges incident with it.

→ Repeat until all edges are deleted.

* a vertex is selected only when the corresponding dual inequality is tight.

primal inequality is approximately satisfied with a constant factor of 2.

because for any edge ij $1 \leq x_i + x_j \leq 2$

Cost of integer solution $\leq 2 \times$ cost of dual LP solution

Set Cover

Let U and a collection of subsets of U

$\{S_1 \dots S_m\}$

each set S_i has wt w_i

Find min-wt collection $C' \subseteq C$ such that every element in U is contained in some subset in C'.

$\{e_1 \dots e_n\}$

$\{S_i$

↳ edges incident with vertex i (v_i)

Vertex cover = special case of set cover

Assume that every element belongs to some set in C

$x_i \rightarrow$ set S_i 0 if S_i is not selected
1 if S_i is selected.

$\min (\sum w_i x_i)$

for every element a_i in U, $\sum_{\text{such that } a \in S_i} x_i \geq 1$

(one set has to be selected).

If every element in U belongs to at most k sets
k-approximation algorithm.

Shortest Path with positive costs on edges

think of this as a set cover problem

directed graphs → two vertices s and t
+ve integer wts to the edges
min-cost path from s to t.

set covers
$\{x_1 \ldots x_n\}$ and a collection of subsets of U → their union is U → with cost assigned to each subset
min it collection of subsets whose union is U.

For edge e_i we have a variable e_i $(\min w_i e_i)$

for every subset S of vertices that includes s but not t.

$$\sum_{\text{all edges } e_i \text{ that join a vertex in } S \text{ to a vertex not in } S} e_i \geq 1$$

2^{n-1} possible constraints

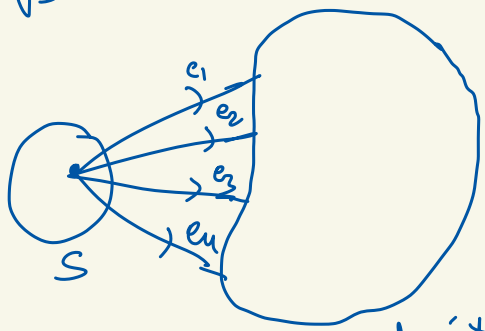
y_S → increasing its value → till inequality for some edge becomes tight

⇓

the min wt edge leaving S

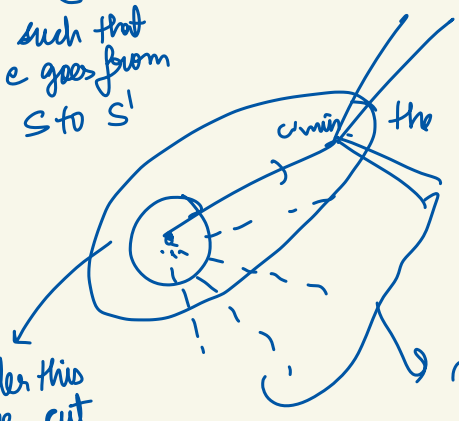
↓

we can include it in the set cover.



for an edge e the dual constraint

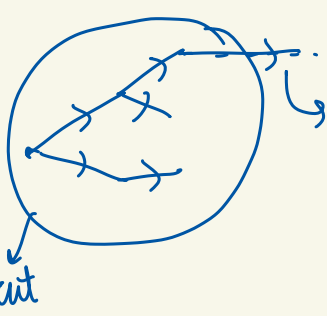
$$\sum_{S \text{ such that } e \text{ goes from } S \text{ to } S'} y_S \leq w_e$$



the weights of other edges are reduced by c_{min}

consider this as the cut

now we will check these edges in the next iteration



→ get a set covers as soon as t gets added to the tree

delete all edges not on the path from s to t.

Finally when only edge in a path are kept, for every cut with a non-zero dual value, exactly one edge in the cut will be included in the solution.

If a dual variable is not zero then the corresponding primal inequality is tight.

Undirected graph with positive weights assigned to vertices. Find a min wt. subset of vertices whose removal leaves only isolated vertices as edges.

Vertex cover → find a subset of vertices whose removal only leaves isolated vertices.

set of all edges V {edges incident with v} → Vertex covers

set of all paths of length 2 (3 vertices)

{set of all paths of length 2 that contain v}

V

x_v

3- approximation

min $w_v x_v$

For every path $v_1 v_2 v_3$ of length 2

$x_{v_1} + x_{v_2} + x_{v_3} \geq 1$

K_n (all weights 1)

optimal integer solution is $n-2$

$x_v = \frac{1}{3}$

satisfies all inequalities, it is a fractional solution whose cost is $\frac{n}{3}$.

$\frac{n}{3} \approx \frac{1}{3}$ (Cost of OPT - integer solution)

Integrality gap of the LP ratio between $\frac{\text{optimal integer solution}}{\text{optimal fractional solution}}$

↓

cannot hope to prove a better bound than this for a primal dual problem.

for any subset S of 3 or more vertices that induces a connected graph

$$\sum_{v \in S} d_S(v) x_v \geq |E_S| - \frac{|S| - \tau(S)}{2}$$

$d_S(v)$ → no. of neighbours of v in S

where $\tau(S)$ denotes the min number of vertices from S to be removed to leave only isolated vertices.

number of subgraphs may be exponential

initially we have a connected graph with at least 3 vertices.

dual inequality of a vertex $\sum_{\text{all such sets containing } v} d_S(v)\, y_S \leq w_v$

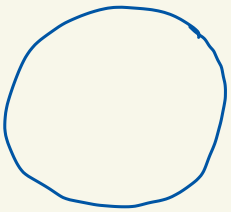
Initially increase y_S for the set S of all vertices in G.

till it becomes tight for some vertex

↓

this one for which $\frac{w_v}{d(v)}$ is minimum.

$y_v \uparrow v$



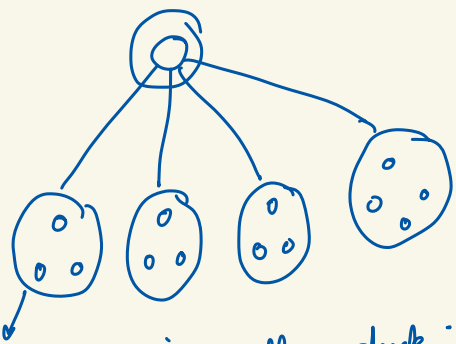
Select the vertex with minimum value $\frac{w}{\text{degree}} = \delta$

$y_v = \delta$

reduce the weights of all other vertices u by $\delta \cdot d(u)$

new-wt of v = original-wt of u $- \frac{w(v)}{deg(v)} \times deg(u)$

delete the selected vertex, all edges incident with it and recursively work with all connected components with at least 3 vertices.

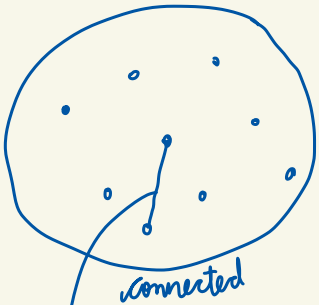


After the recursive call, check if initially selected vertex v required i.e. whether deleting it still gives a valid sol^n and if no, delete it.

Whenever a vertex is selected $\Rightarrow$ dual inequality is tight

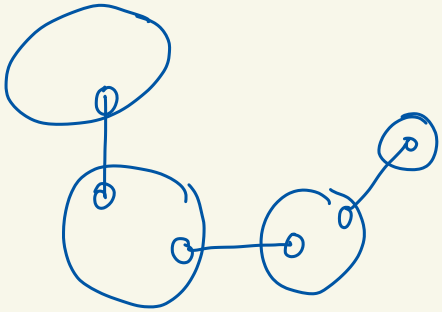
For any minimal subset with the required property in the graph $G[S]$ where y_S is not zero, satisfies the primal inequality within a factor of 2.

$$\sum_{x \in X} d(x) \leq 2|E| - |S| + \tau(S)$$



$S = V$

Adds 2 to both LHS and RHS, hence we can reduce the graph. $|E| \geq |S| - 1$ graph is connected



Connected

k edges between vertices in X such that removing any one disconnects the graph then

$\tau(S) \geq k+1$

This is satisfied for any minimal subset X and connected graph G with at least 3-vertices.

* If we are allowed to leave complete subgraphs, then 3-apx can be found.