Parametric search

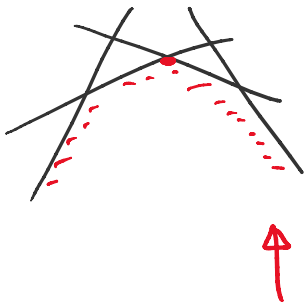
disadvantages
- extra log
- need parallelization
- complicated

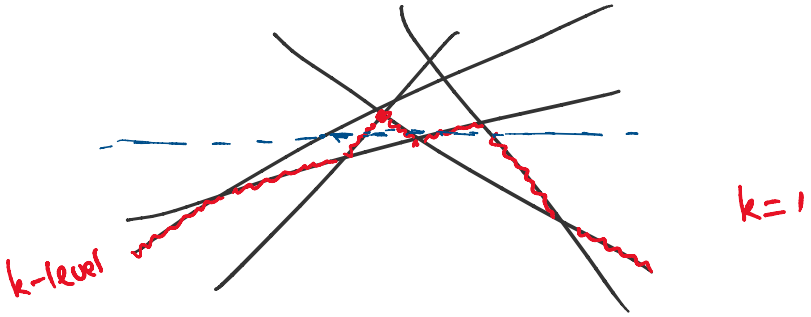
⇒ simpler alternative?

Ex 2D linear program (LP) with k violations

Given n lower halfplanes in 2D, & k,

find highest pt inside all but k halfplanes



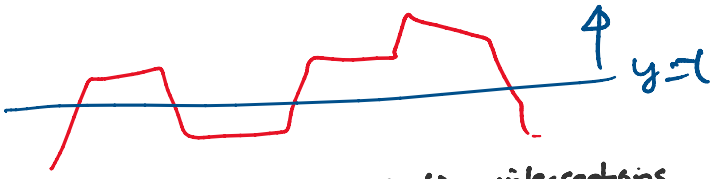


equiv: find highest pt on the k-level

k=0: standard LP: O(n) time

Decision Problem:

given t, decide whether k-level intersects y=t

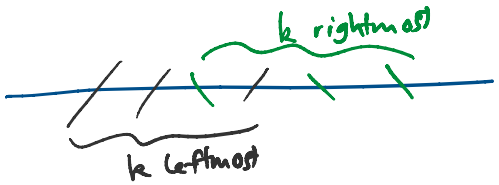


$T_D = O(n \log n)$ by sorting intersections along y=t & linear scan

(more carefully: $T_D = O(n + k \log n)$)

[illegible]

(more carefully: $T_D = O(n + k \log n)$)



by parametric search, $T = O(n)$, $\tau_D = O(\log n)$ $\Rightarrow$ $O(n \log^3 n)$ time

(alternative: by binary search using slope selection)

rand. search: $O(n \log n)$ ...

Warm-Up Exercise

find min of ℓ numbers $a_1, \ldots, a_\ell$

randomly permute $a_1, \ldots, a_\ell$
min $\leftarrow \infty$
for $i = 1, \ldots, \ell$
  if $a_i <$ min
    min $= a_i$ (*)

$O(\ell)$ time

how many times (*) is done?

worst case: ℓ

expected?

backwards analysis:

$\Pr[$ in last iteration, (*) is done $]$

$= \Pr[\ a_\ell = \min(a_1, \ldots, a_\ell)\ ]$

(a_ℓ: rand.; $\min(a_1, \ldots, a_\ell)$: fixed)

$= \frac{1}{\ell}$.

$\Rightarrow$ expected total $= \frac{1}{\ell} + \frac{1}{\ell - 1} + \ldots + 1$

$\leq \ln \ell + 1 = O(\log \ell)$

Now, imagine $a_1, \ldots, a_\ell$ are unknown:

- decision oracle: given i, t, is $a_i < t$? in T_D time
- evaluation oracle: given i, compute a_i in T_E time ($T_D \ll T_E$)

goal: compute $\min(a_1, \ldots, a_\ell)$

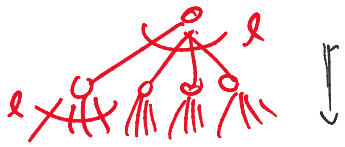
naive soln: $O(T_E \ell)$ time

rand. soln: $O(T_D \ell + T_E \log \ell)$ expected time

A Recursive Rand. Technique: (C.'98)

Lemma Let f be a problem with

- decision oracle: given input I of size n, value t, decide whether $f(I) < t$? in $T_D(n)$ time
- decomposition scheme: given input I of size n, can form smaller inputs $I_1, \ldots, I_\ell$ of size $\frac{n}{b}$ s.t. $f(I) = \min\{f(I_1), \ldots, f(I_\ell)\}$ for some consts ℓ, $b > 1$



(in $\ll T_D(n)$ time)

Then can evaluate $f(I)$ in $O(T_D(n))$ expected time

assuming $T_D(n) \gg n^\varepsilon$

(more precisely, $T_D(n)/n^\varepsilon$ is increas.)

Pf: Let $T(n)$ be expected time to evaluate $f(I)$ for $|I| = n$.

$$\Rightarrow T(n) \le O\left(T_D(n)\cdot \ell\right) + T\left(\frac{n}{b}\right)\cdot(\ln \ell + 1)$$

$$\Rightarrow \boxed{T(n) \le (\ln \ell + 1)\, T\left(\frac{n}{b}\right) + O\left(T_D(n)\right)}$$

Master Thm $\Longrightarrow$ $T(n) = O\left(T_D(n)\right)$ if $T_D(n) >> n^{\log_b(\ln \ell + 1) + \varepsilon}$ (†)

What if (†) not true?

repeat k times for large const k

$\ell \to \ell^k$

$b \to b^k$

$$\log_{b^k}(\ln \ell^k + 1) = \frac{\log(k \ln \ell + 1)}{k \log b} \to 0 \text{ as } k \to \infty$$

□

Ex1: closest pair of a set P of n pts in $\mathbb{R}^d$

divide P into $P_1, \ldots, P_r$ of size $\frac{n}{r}$ for const r

$\Rightarrow f(P) = \min_{i,j \in \{1,\ldots,r\}} f(P_i \cup P_j)$

$\Rightarrow \binom{r}{2}$ subproblems of size $\le \frac{2n}{r}$

$\ell = \binom{r}{2}$, $b = r/2$.

e.g. $r=3$, $\ell = 3$, $b = 3/2$.

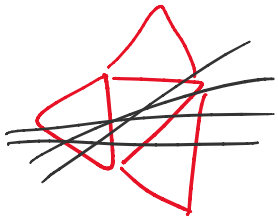
$O(n \log n) \Rightarrow$ closest pair

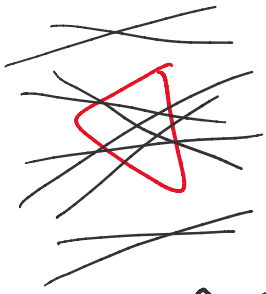
e.g. 1-3, x - - -

$T_D(n) = O(n \log n)$ by grid $\Rightarrow$ closest pair in $O(n \log n)$ time

Ex2 2D LP with k violations

Apply Cutting Lemma to n lines





inside Δ, k-level of H $\equiv (k - x_\Delta)$-level of H_Δ

$O(r^2)$ subproblems of size $\frac{n}{r}$ for r const

for each cell Δ,
let H_Δ = halfplanes whose lines intersect Δ
$(|H_\Delta| \le \frac{n}{r})$

x_Δ = # lines strictly below Δ

$$\Rightarrow f(H, k) = \max_\Delta f(H_\Delta, k - x_\Delta)$$

$l = O(r^2)$, $b = r$

$\Rightarrow T_D(n) = O(n \log n) \Rightarrow$ $O(n \log n)$ expected time.

alternatively:

$$T_D(n) = O(n + k \log n) \Rightarrow O(n + k^{1-\varepsilon} n^\varepsilon \log n)$$
$$= O(n + k^{1-\varepsilon} n^\varepsilon \log n)$$

expected time

(open: $O(n + k \log n)$?)