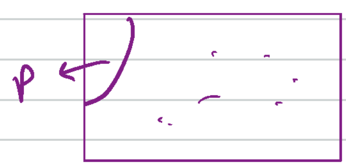
Lecture 23

CS 170

Sanjam Garg.

NP - complete problems still need a solution.

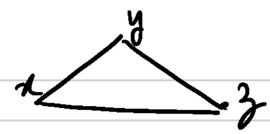


1) "Intelligent" exponential search. → Running time could be exponential
→ Practical instances ↳ run efficiently.

2) Approximation Algorithms. ⟶ polytime
↳ relationship with the optimal solution.

3) Heuristic → no guarantees on the runtime or the optimality of solution.

Clustering



① $\forall x,y \; d(x,y) \geq 0$
② $d(x,y) = 0$ iff $x = y$
③ $d(x,y) = d(y,x)$
④ TI: $d(x,y) + d(y,z) \geq d(x,z)$

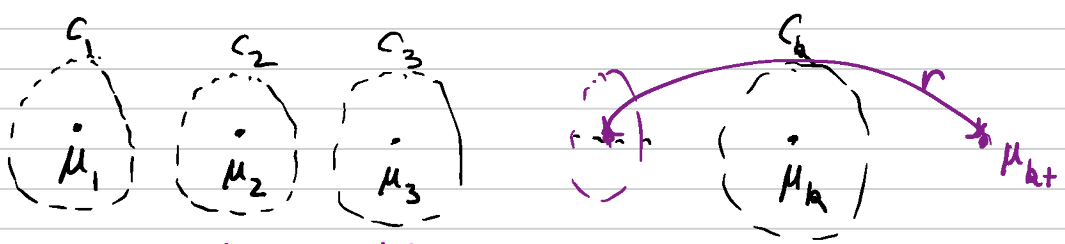
Input: Points $X = \{x_1 \dots x_n\}$ with distance metric $d(\cdot,\cdot)$; integer k

Output: k clusters $C_1 \dots C_k$ s.t. $\max_j \underbrace{\max_{x,y \in C_j} d(x,y)}$ is minimized

$k < n$

$d(x,y)$ is the diameter of C_j

Pick any point $\mu_1 \in X$ as the first cluster center
for $i = 2 \dots k$
    Let $\mu_i \in X$ be the point farthest from $\mu_1 \dots \mu_{i-1}$
Create k clusters: $C_i = \{$ all $x \in X$ closest to $\mu_i \}$



1) $\forall i \; \forall x \in C_i$ we have $d(x,\mu_i) \leq r$

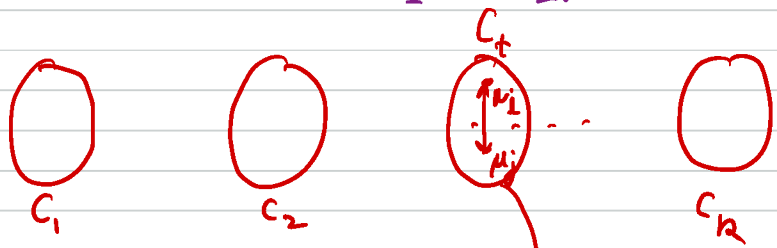
2) $\forall i,j \in \{1 \dots k+1\}$ $d(\mu_i, \mu_j) \geq r$

Lemma: $\forall i \quad \forall x, y \in C_i \quad d(x,y) \leq 2r$

Proof: $d_A = d(x,y) \leq d(x,\mu_i) + d(\mu_i, y)$ (TI)

$$\leq r + r$$

$$= 2r \quad \text{———} \quad (1)$$



$X = \{x_1, \ldots x_n\}$

$\mu_1 \ldots \mu_{k+1}$

$\exists t \in \{1 \ldots k\}$ s.t. μ_i & $\mu_j \in C_t$ $\quad i,j \in \{1 \ldots k+1\}$, $i \neq j$

$$d_{C_t} \geq d(\mu_i, \mu_j) \geq r$$

$$\Downarrow$$

$$d_{OPT} \geq r \quad \text{———} \quad (2)$$

From (1) & (2) $\rightarrow d_A \leq 2 d_{OPT}$

RC $\longrightarrow$ Travelling Salesman Problem (TSP)

Instance: $G = (V, E)$ | Instance: distances d_{ij}

Solution: Cycle S visiting each vertex exactly once. | Solution: a permutation $\tau : \{1 \ldots n\} \to \{1 \ldots n\}$ s.t.

$$d_{\tau(1)\tau(2)} + d_{\tau(2)\tau(3)} \ldots d_{\tau(n)\tau(1)} \leq n$$

$G \longrightarrow$ $d_{ij} = 1$ if $(i,j) \in E$

$d_{ij} = 1 + C$ if $(i,j) \notin E$ for some $C > 1$

if G had a RC $\Longrightarrow$ G' has a TSP solution of cost n

if G does not have a RC $\Longrightarrow$ G' has no TSP solution of cost $< n + C$

RC $\longrightarrow$ α-TSP

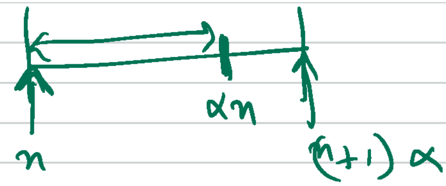
$$d_{\tau(1)\tau(2)} \ldots d_{\tau(n)\tau(1)} \leq \alpha n$$

$C = \alpha n$

if G has a RC $\Longrightarrow$ G' has a TSP solution of cost n

& if G does not have a RC $\Longrightarrow$ G' has no TSP solution of cost $n + n\alpha = n(1+\alpha)$

$\underbrace{1+1 \cdots 1}_{n \text{ times}} = n$



2- Travelling Salesman Problem (TSP)

Instance: distances d_{ij} $\rightarrow$ $\forall i,j,k$ $\quad d_{ij} + d_{jk} \geq d_{ik}$

Solution: a permutation $\tau : \{1 \dots n\} \rightarrow \{1 .. n\}$ s.t.

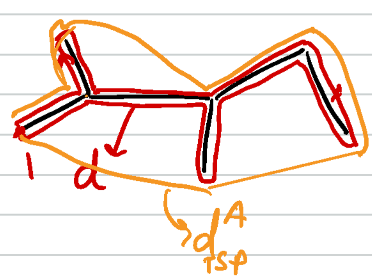
$$d_{\tau(1)\tau(2)} + d_{\tau(2)\tau(3)} \cdots d_{\tau(n)\tau(1)} \leq 2\, d_{OPT}$$

lemma: $\quad d_{MST} \leq d^{OPT}_{TSP}$

ⓐ MST can be a good starting point.

ⓑ $\quad d \leq 2\, d_{MST} \leq 2\, d^{OPT}_{TSP}$

ⓒ $\quad d^{A}_{TSP} \leq d \leq 2\, d^{OPT}_{TSP}$



Knapsack (without rep.)

2,12,317,456 $\qquad O(nW)$

Input: n items with weights $w_1, \dots, w_n$ & values $v_1, \dots, v_n$

Goal: Most valuable combination with total weight $\le W$

$\hookrightarrow O(nV)$

$0 < \epsilon < 1$ we will give an approx. algorithm s.t $K \ge (1-\epsilon)\underbrace{K^*}_{\text{optimal}}$

$\hookrightarrow$ poly $\left(n, \frac{1}{\epsilon}\right)$

$$\boxed{\begin{array}{l} \text{Discard any items with weight} > W \\ \text{let } v_{max} = \max_i v_i \\ \text{Rescale } \quad \hat{v}_i = \left\lfloor v_i \cdot \frac{n}{\epsilon v_{max}} \right\rfloor \\ \text{Run DP algorithm with values } \{\hat{v}_i\} \\ \text{Output the resulting choice of items.} \end{array}}$$

$\rightarrow n \times \frac{n}{\epsilon} \times n$

$= O\left(\frac{n^3}{\epsilon}\right)$

$\boxed{v_1 \dots v_n} \longrightarrow \boxed{\hat{v}_1 \dots \hat{v}_n}$

$\downarrow \qquad\qquad\qquad \downarrow$

$S \qquad\qquad\qquad \hat{S}$

$K^* \qquad\qquad\qquad \hat{K}$

K

1) $$\sum_{i \in S} \hat{v}_i = \sum_{i \in S} \left\lfloor \frac{v_i \, n}{\epsilon V_{max}} \right\rfloor \geq \sum_{i \in S} \left(\frac{v_i \, n}{\epsilon V_{max}} - 1 \right)$$

$$\geq \frac{\left(\sum_{i \in S} v_i \right) n}{\epsilon V_{max}} - |S|$$

$$\geq K^* \frac{n}{\epsilon V_{max}} - n$$

$$= \left(\frac{K^*}{\epsilon V_{max}} - 1 \right) n$$

$$\hat{K} \geq \left(\frac{K^*}{\epsilon V_{max}} - 1 \right) n$$

2) $$\underline{\sum_{i \in \hat{S}} v_i} \geq \sum_{i \in \hat{S}} \hat{v}_i \frac{\epsilon V_{max}}{n}$$

$$\geq \left(\sum_{i \in \hat{S}} \hat{v}_i \right) \frac{\epsilon V_{max}}{n} \geq \left(\frac{K^*}{\epsilon V_{max}} - 1 \right) \cancel{n} \times \frac{\epsilon V_{max}}{\cancel{n}}$$

$$= K^* - \epsilon V_{max}$$

$$\geq K^*(1 - \epsilon)$$