11 April 2025

Linear Programming & Knapsack

Plan

* Analysis of Min Vertex Cover
* Announcements
* Knapsack

Minimum Weighted Vertex Cover

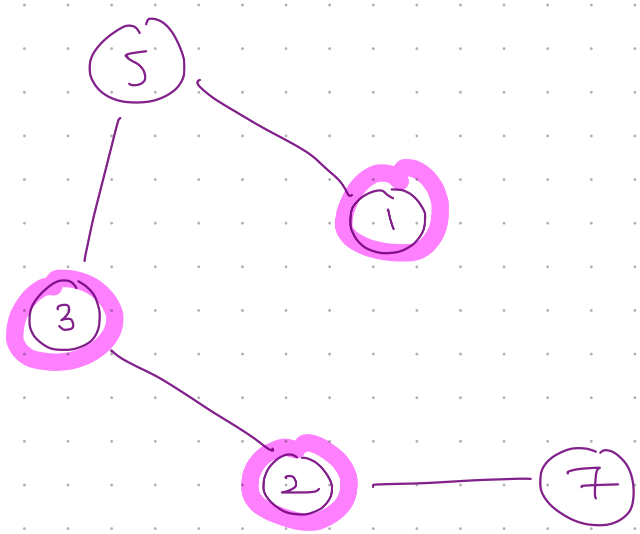
Given: Undirected Graph $G = (V, E)$

w/ vertex weights $W = \{w_v\}_{v \in V}$

$w_v \geq 0$

Find: minimum weight

Vertex Cover $C \subseteq V$ $\sum_{v \in C} w_v$.



Approximate Minimum Weighted Vertex Cover

Given: Undirected Graph $G=(V,E)$

w/ vertex weights $W = \{w_v\}_{v \in V}$

$w_v \geq 0$

Find: approximately minimum weight

Vertex Cover $C \subseteq V$

$$W(C) = \sum_{v \in C} w_v .$$

$$W^*(G) = \min_{C^* \subseteq V} W(C^*)$$

Approximation Ratio of Algorithm A

(for minimization problem)

$$\boxed{r_A = \max_G \frac{W(A(G))}{W^*(G)}}$$

Integer Linear Programming (NP-Hard)

variables $x_1, x_2, \ldots, x_n$

linear inequalities

$$\langle a_i, x \rangle = \sum_{j=1}^{n} a_{ij} \cdot x_j \leq b_i \qquad i = 1 \ldots m$$

integer constraints

$$x_j \in \mathbb{Z}$$

Linear optimization:

Find $$\min_{\vec{x}} \langle c, x \rangle = \sum_{j=1}^{n} c_j \cdot x_j$$

s.t. constraints,

Linear Programming (Poly-time solvable!)

variables $x_1, x_2, ---, x_n$ $\quad x_j \in \mathbb{R}$

linear inequalities

$$\langle a_i, x \rangle = \sum_{j=1}^{n} a_{ij} \cdot x_j \leq b_i \qquad i=1 \dots m$$

Linear optimization:

Find $\min_{\vec{x}} \langle c, x \rangle = \sum_{j=1}^{n} c_j \cdot x_j$

s.t. constraints,

Linear Programming Relaxation.

Min Wt. Vertex Cover ILP

Variables $\{ x_v : v \in V \}$

Constraints

$$x_v \in \{0,1\} \quad \forall v \in V.$$

$$x_u + x_v \geq 1 \quad \forall (u,v) \in E.$$

Objective

$$\min_{\vec{x}} \sum_{j=1}^{n} w_j \cdot x_j$$

Linear Programming Relaxation.

Min Wt. Vertex Cover ~~I~~LP

Variables $\{ x_v : v \in V \}$

Constraints

$\cancel{x_v \in \{0,1\} \quad \forall v \in V.}$ $\quad 0 \le x_v \le 1 \quad \forall v \in V.$

$$x_u + x_v \ge 1 \quad \forall (u,v) \in E.$$

Objective

$$\min_{\vec{x}} \sum_{j=1}^{n} w_j \cdot x_j$$

Approximate VC

On input $G = (V, E)$. Construct & solve LP

Variables $\{ x_v : v \in V \}$

Linear Inequalities

$$0 \leq x_v \leq 1 \quad \forall v \in V.$$

$$x_u + x_v \geq 1 \quad \forall (u,v) \in E.$$

Objective

$$\min_{\vec{x}} \sum_{j=1}^{n} w_j \cdot x_j$$

Approximate VC

On input $G = (V, E)$. Construct & solve LP

Variables $\{ x_v : v \in V \}$

Linear Inequalities

$$0 \le x_v \le 1 \quad \forall v \in V.$$
$$x_u + x_v \ge 1 \quad \forall (u,v) \in E.$$

Objective

$$\min_{\vec{x}} \sum_{j=1}^{n} w_j \cdot x_j$$

Then, **ROUND** solution

$C = \emptyset$

For $v \in V$.

  if $x_v \ge 1/2$, $C \leftarrow C \cup \{v\}$

Return C

<u>Theorem</u>: Approximate VC returns a 2-approximate VC.

Theorem: Approximate VC returns a 2-approximate VC.

Claim C is a vertex cover.

Claim $w(C) \leq 2 \cdot w^*(G)$

Theorem: Approximate VC returns a 2-approximate VC.

Claim C is a vertex cover.

* For all edges $(u,v) \in E$ $x_u + x_v \geq 1$

$\Longrightarrow$ at least one of $x_u, x_v \geq 1/2$

$\Longrightarrow$ at least one of $u, v \in C$.

Claim $w(C) \leq 2 \cdot w^*(G)$

Theorem: Approximate VC returns a 2-approximate VC.

Claim C is a vertex cover.

* For all edges $(u,v) \in E$ $\quad x_u + x_v \geq 1$

$\Rightarrow$ at least one of $x_u, x_v \geq 1/2$

$\Rightarrow$ at least one of $u, v \in C$.

Claim $W(C) \leq 2 \cdot W^*(G)$

$$W(C) = \sum w_j \cdot \mathbb{1}[x_j \geq 1/2]$$

$$\leq 2 \cdot \sum w_j \cdot x_j$$

Theorem: Approximate VC returns a 2-approximate VC.

Claim C is a vertex cover.

* For all edges $(u,v) \in E$ $\quad x_u + x_v \geq 1$

$\Rightarrow$ at least one of $x_u, x_v \geq 1/2$

$\Rightarrow$ at least one of $u, v \in C$.

Claim $W(C) \leq 2 \cdot W^*(G)$

$$W(C) = \sum w_j \cdot \mathbb{1}[x_j \geq 1/2]$$

$$\leq 2 \cdot \sum w_j \cdot x_j$$

$$= 2 \cdot \mathrm{OPT}(\text{VC-LP})$$

Theorem: Approximate VC returns a 2-approximate VC.

Claim C is a vertex cover.

* For all edges $(u,v) \in E$ $\quad x_u + x_v \geq 1$

$\implies$ at least one of $x_u, x_v \geq 1/2$

$\implies$ at least one of $u, v \in C$.

Claim $w(C) \leq 2 \cdot w^*(G)$

$$
\begin{aligned}
w(C) &= \sum w_j \cdot \mathbb{1}\left[x_j \geq 1/2\right] \\
&\leq 2 \cdot \sum w_j \cdot x_j \\
&= 2 \cdot \mathrm{OPT}(\text{VC-LP}) \\
&\leq 2 \cdot \mathrm{OPT}(\text{VC-ILP}) = 2 \cdot \left[\text{Min Wt. VC}\right]
\end{aligned}
$$

Announcements

* HW 7 Out Now

* No recitation this weekend

Knapsack

Given: collection of n items $\mathcal{U}$

* w_i = weight of i^{th} item
* v_i = value of i^{th} item

total capacity C

Find: Subset of items $S \subseteq \mathcal{U}$

maximizing value $\sum_{i \in S} v_i$

s.t. weight limit $\sum_{i \in S} w_i \leq C$.

5	7	4	2	11	3
4	3	10	6	15	4
A	B	C	D	E	F

$C = 16$

5	7	4	2	11	3
4	3	10	6	15	4
A	B	C	D	E	F

$C = 16$

$S = \{E\}$

$V_s = 11$

$W_s = 15$

5	7	4	2	11	3
4	3	10	6	15	4
A	B	C	D	E	F

$C = 16$

$S = \{E\}$ $\quad V_s = 11$, $W_s = 15$

$S = \{A, B, F\}$ $\quad V_s = 15$, $W_s = 11$

Complexity of Knapsack

Theorem. Knapsack is NP-Hard.

Complexity of Knapsack

Theorem. Knapsack is NP-Hard.

Theorem. There exists an algorithm for Knapsack that runs in time

$$O(n^2 \cdot v^*)$$

where $v^* = \max_i v_i$

Complexity of Knapsack

Theorem. Knapsack is NP-Hard.

Theorem. There exists an algorithm for Knapsack that runs in time

$$O(n^2 \cdot v^*)$$

for integer values where $v^* = \max_i v_i$

Knapsack is Weakly NP-Hard.

* Reduction from SAT to Knapsack uses exponentially large values v_i!

Approximating Weakly NP-Hard Problems

Idea: Reduce Knapsack to Knapsack

Approximating Weakly NP-Hard Problems

Idea Reduce Knapsack to $\widetilde{\text{Knapsack}}$

large exact values → small approximate values

Approximating Weakly NP-Hard Problems

Idea Reduce Knapsack to $\widetilde{\text{Knapsack}}$

large exact values

small approximate values

Approach Divide values by scaling factor M

$$\left\{ \begin{array}{cccc} v_1 & v_2 & \cdots & v_n \\ w_1 & w_2 & \cdots & w_n \\ & & C & \end{array} \right\} \longrightarrow \left\{ \begin{array}{cccc} v_1/M & v_2/M & \cdots & v_n/M \\ w_1 & w_2 & \cdots & w_n \\ & & C & \end{array} \right\}$$

ApproxKnapsack $(\{v_i\}, \{w_i\}, C)$

* Scale values $\tilde{v}_i \leftarrow \left\lceil \frac{v_i}{M} \right\rceil$ for $i = 1 \ldots m$

* $S \leftarrow$ ExactKnapsack $(\{\tilde{v}_i\}, \{w_i\}, C)$

* Return S.

Approx Knapsack $(\{v_i\}, \{w_i\}, C)$

* Scale values $\tilde{v}_i \leftarrow \left\lceil \frac{v_i}{M} \right\rceil$ for $i = 1 \ldots n$

* $S \leftarrow$ Exact Knapsack $(\{\tilde{v}_i\}, \{w_i\}, C)$

* Return S.

Claim. S is a feasible Knapsack solution

Approx Knapsack ($\{v_i\}$, $\{w_i\}$, C)

- Scale values $\tilde{v}_i \leftarrow \left\lceil \frac{v_i}{M} \right\rceil$ for $i = 1 \ldots n$
- $S \leftarrow$ Exact Knapsack ($\{\tilde{v}_i\}$, $\{w_i\}$, C)
- Return S.

<u>Claim.</u> S is a feasible Knapsack solution

<u>Claim.</u> Let S^* be the optimal Knapsack solution.

$$\sum_{i \in S^*} v_i \leq \sum_{i \in S} v_i + n \cdot M$$

Approx Knapsack ($\{v_i\}$, $\{w_i\}$, C)

* Scale values $\tilde{v}_i \leftarrow \left\lceil \frac{v_i}{M} \right\rceil$ for $i = 1 \dots m$

* $S \leftarrow$ Exact Knapsack ($\{\tilde{v}_i\}$, $\{w_i\}$, C)

* Return S.

<u>Claim</u>. S is a feasible Knapsack solution

<u>Claim</u>. Let S^* be the optimal Knapsack solution.

$$\sum_{i \in S^*} v_i \leq M \cdot \sum_{i \in S^*} \left\lceil \frac{v_i}{M} \right\rceil$$

Approx Knapsack $(\{v_i\}, \{w_i\}, C)$

* Scale values $\tilde{v}_i \leftarrow \left\lceil \frac{v_i}{M} \right\rceil$ for $i = 1 \dots n$

* $S \leftarrow$ Exact Knapsack $(\{\tilde{v}_i\}, \{w_i\}, C)$

* Return S.

Claim. S is a feasible Knapsack solution

Claim. Let S^* be the optimal Knapsack solution.

$$\sum_{i \in S^*} v_i \leq M \cdot \sum_{i \in S^*} \left\lceil \frac{v_i}{M} \right\rceil \leq M \sum_{i \in S} \left\lceil \frac{v_i}{M} \right\rceil$$

By Exact Knapsack

Approx Knapsack $(\{v_i\}, \{w_i\}, C)$

* Scale values $\tilde{v}_i \leftarrow \left\lceil \frac{v_i}{M} \right\rceil$ for $i = 1 \dots n$

* $S \leftarrow$ Exact Knapsack $(\{\tilde{v}_i\}, \{w_i\}, C)$

* Return S.

Claim. S is a feasible Knapsack solution

Claim. Let S^* be the optimal Knapsack solution.

$$\sum_{i \in S^*} v_i \leq M \cdot \sum_{i \in S^*} \left\lceil \frac{v_i}{M} \right\rceil \leq M \sum_{i \in S} \left\lceil \frac{v_i}{M} \right\rceil \leq \sum_{i \in S} v_i + n \cdot M$$

By Exact Knapsack

Approximation Ratio

$$\frac{\sum_{i \in S} v_i}{\sum_{i \in S^*} v_i} \geq \frac{\sum_{i \in S^*} v_i - n \cdot M}{\sum_{i \in S^*} v_i}$$

Running Time

Approximation Ratio

$$\frac{\sum_{i \in S} v_i}{\sum_{i \in S^*} v_i} \geq \frac{\sum_{i \in S^*} v_i - n \cdot M}{\sum_{i \in S^*} v_i}$$

$$\geq 1 - \frac{n \cdot M}{\max_i v_i}$$

(assumes $w_i \leq C$)

Running Time

Approximation Ratio

$$\frac{\sum_{i \in S} v_i}{\sum_{i \in S^*} v_i} \geq \frac{\sum_{i \in S^*} v_i - n \cdot M}{\sum_{i \in S^*} v_i}$$

$$\geq 1 - \frac{n \cdot M}{\max_i v_i} \qquad (\text{assumes } w_i \leq C)$$

$$= 1 - \varepsilon \qquad \text{for } M \leq \frac{\varepsilon \cdot \max_i v_i}{n}$$

Running Time

Approximation Ratio

$$\frac{\sum_{i \in S} v_i}{\sum_{i \in S^*} v_i} \geq \frac{\sum_{i \in S^*} v_i - n \cdot M}{\sum_{i \in S^*} v_i}$$

$$\geq 1 - \frac{n \cdot M}{\max_i v_i} \qquad \text{(assumes } w_i \leq C \text{)}$$

$$= 1 - \varepsilon \qquad \text{for } M = \frac{\varepsilon \cdot \max_i v_i}{n}$$

Running Time

$$O\left(n^2 \cdot \max_i \left\lceil \frac{v_i}{M} \right\rceil\right)$$

Approximation Ratio

$$\frac{\sum_{i \in S} v_i}{\sum_{i \in S^*} v_i} \geq \frac{\sum_{i \in S^*} v_i - n \cdot M}{\sum_{i \in S^*} v_i}$$

$$\geq 1 - \frac{n \cdot M}{\max_i v_i} \qquad \text{(assumes } w_i \leq C \text{)}$$

$$= 1 - \varepsilon \qquad \text{for } M = \frac{\varepsilon \cdot \max_i v_i}{n}$$

Running Time

$$O\left(n^2 \cdot \max_i \left\lceil \frac{v_i}{M} \right\rceil\right) \leq O\left(n^2 \cdot \left(\frac{\cancel{\max_i v_i} \quad n}{\cancel{\max_i v_i} \cdot \varepsilon}\right)\right)$$

$$= O(n^3/\varepsilon)$$

Theorem. Knapsack has a Fully-Polynomial-Time Approximation Scheme (FPTAS)

↳ For any $\varepsilon > 0$, there exists an algorithm that $(1-\varepsilon)$-approximates Knapsack running in $O(n^3/\varepsilon)$ time

Arbitrarily good approximation in polynomial time!