✓ 2-approx for KNAPSACK → Greedy

r-approx for any $r > 1$ → DP2 $O(n^2 v_{max})$

n elements

W = knapsack capacity

w_i = occupancy of the i-th element in the input

v_i = its value

$\left.\right\} i \in [n]$

$S \subseteq [n]$ is feasible if $\sum_{i \in S} w_i \leq W$

(note that $w_i \leq W$, otherwise you discard before starting)

GREEDY approach

$\frac{v_i}{w_i}$ = "value" per unit

hp wlog. $\frac{v_1}{w_1} \geq \frac{v_2}{w_2} \geq \dots \geq \frac{v_n}{w_n}$

intuition take this first if possible

2-approx:

$S = \{\}$, $W' = W$ residual capacity in the knapsack

for $i = 1, 2, \dots, n$ **do**:

  if $w_i \leq W'$ **then** $S = S \cup \{i\}$, $W' = W' - w_i$

return S

Good heurists but it has a glitch

Glitch:

$$\left.\begin{array}{ll} v_1 = v_2 = \dots = v_{n-1} = 1, & v_n = W-1 \\ w_1 = w_2 = \dots = w_{n-1} = 1, & w_n = W \end{array}\right\} \Rightarrow \frac{v_i}{w_i} = 1 \text{ for } i < n$$

$$\frac{v_n}{w_n} < 1$$

$$S = \{1, \dots, n-1\} \Rightarrow \text{cost}(S) = \sum_{i \in S} v_i = n-1$$

$$S^* = \{n\} \Rightarrow OPT = \text{cost}(S^*) = W-1$$

approx ratio: $\dfrac{OPT}{\text{cost}(S)} = \dfrac{W-1}{n-1} \sim k$ arbitrarily large

W can be anything $\geq n-1$

$W = kn$ for any arbitrary $k > 1$

FIX: let m_G be the value returned by GREEDY

let $V_{max} = \max_{i \in [n]} V_i$

$\Big\}$ return $\max(m_G, V_{max})$

▶ This gives a 2-approximation

MAX $\Rightarrow$ we need to find $UB \geq OPT$

J = smallest element in $[n]$ that does not fit W' $\quad (J \geq 2)$

$$\underbrace{\sum_{i=1}^{J-1} w_i}_{\bar{w}_J} \leq W \quad \text{but} \quad \sum_{i=1}^{J} w_i > W$$

$$\bar{V}_J = \sum_{i=1}^{J-1} V_i$$

$$\underbrace{\sum_{i=1}^{J} V_i}_{UB = \bar{V}_J + V_J} \geq OPT$$

Since $OPT < \bar{V}_j + V_j$ we have two cases

1) $\bar{V}_j > V_j \Rightarrow OPT < 2\bar{V}_j \leq 2 m_G$ as S = solution in GREEDY satisfies $\{1, .., j-1\} \subseteq S$

2) $\bar{V}_j \leq V_j \Rightarrow OPT < 2 V_j \leq 2 V_{max}$ as $V_j \leq V_{max}$ by def.

Summing up: $OPT \leq 2(\max(m_G, V_{max}))$

$OPT \leq 2(\max(m_G, V_{max})) = 2 \cdot \text{cost}(S) \Rightarrow \dfrac{OPT}{\text{cost}(S)} \leq 2$

QED

FPTAS = fully poly-time approximate solution : you can get an r-approx for any $r > 1$
(PTAS)

$\boxed{O(n^3/\varepsilon) \text{ time}}$: $\forall \varepsilon : 0 < \varepsilon < 1$: $\frac{OPT}{cost(\tilde{S})} \leq \boxed{\frac{1}{1-\varepsilon}}$

i.e. $\boxed{cost(\tilde{S}) \geq (1-\varepsilon) \cdot OPT}$

any $r > 1$

DP2 = solve exactly an instance of knapsack in $O(n^2 V_{max})$ time

IDEA: scale the values $\tilde{v}_i = \left\lfloor \frac{v_i}{K} \right\rfloor$ for some suitable factor K to be fixed asap

(A) original: v_i, w_i, W

(B) scaled: $\tilde{v}_i, w_i, W$

$\Rightarrow$ S is feasible in (A) $\Longleftrightarrow$ S is feasible in (B) [no change in w_i, W]

Let's fix K so that $\tilde{v}_i$ are poly(n) as we will pay $O(n^2 \tilde{v}_{max})$

Good choice $\tilde{v}_i = \left\lfloor \frac{v_i}{K} \right\rfloor \leq \left\lfloor \frac{n}{\varepsilon} \right\rfloor \Rightarrow \frac{v_i}{K} \leq \frac{n}{\varepsilon} \Rightarrow K = \frac{V_{max} \cdot \varepsilon}{n}$

▶ Run DP2 on (B) in time $O(n^2 \tilde{v}_{max}) = O(n^3/\varepsilon)$

<u>Obs</u> we find the exact optimal solution $\tilde{S}$ for (B)

Q.: how good is $\tilde{S}$ in (A)? It's clearly feasible

What about $\displaystyle\sum_{i \in \tilde{S}} v_i \geq (1-\varepsilon) \sum_{i \in S^*} v_i$

▶ $$\sum_{i \in \tilde{S}} \tilde{v}_i \geq \sum_{i \in S^*} \tilde{v}_i$$

iii

Recall that $\tilde{v}_i = \left\lfloor \frac{v_i}{k} \right\rfloor \Rightarrow \frac{v_i}{k} - 1 \leq \tilde{v}_i \leq \frac{v_i}{k}$

i) $v_i \geq k\tilde{v}_i$ ii) $k\tilde{v}_i \geq v_i - k$ iii) $\sum_{i \in \tilde{S}} \tilde{v}_i \geq \sum_{i \in S^*} \tilde{v}_i$

$$\underset{\text{in } (A)}{\mathrm{cost}(\tilde{S})} = \sum_{i \in \tilde{S}} v_i \overset{i)}{\geq} \sum_{i \in \tilde{S}} k\tilde{v}_i \overset{iii)}{\geq} \sum_{i \in S^*} k\tilde{v}_i \overset{ii)}{\geq} \sum_{i \in S^*} (v_i - k) = \underbrace{\sum_{i \in S^*} v_i}_{OPT} - \sum_{i \in S^*} k \geq$$

$$\underset{|S^*| \leq n}{\geq} OPT - \underbrace{nk}_{\leq \varepsilon \cdot OPT} \geq OPT - \varepsilon\, OPT = (1-\varepsilon)\, OPT$$

$$kn = \left(\frac{\varepsilon \cdot V_{max}}{\cancel{n}}\right) \cdot \cancel{n} = \varepsilon V_{max} \underset{V_{max} \leq OPT}{\leq} \varepsilon\, OPT$$

QED

Summary on approximation algorithms seen in class:

- Greedy (be careful, very good in practice)
- Local search
- Dynamic programming
- Randomization

(missing tool: Linear Programming (LP))

PROBLEM	APPROX. FACTOR $r \geq 1$	LIMIT TO r	
Knapsack (max)	$(1-\varepsilon)^{-1}$ FPTAS $r = \frac{1}{1-\varepsilon}$	—	
VC (min vertex cover)	$r = 2$	$r \geq 1.36$	
SET COVER	$r = O(\lg n)$	$r = \Omega(\lg n)$	GREEDY
MAX INDEPENDENT SET	$r = O(n)$ $n = \lvert V \rvert$	$r = \Omega(n^{1-\varepsilon})$ for any $\varepsilon > 0$	
TSP	NP-hard	NO APPROX	

SET COVER:

universe U, collection $C \subseteq 2^U$

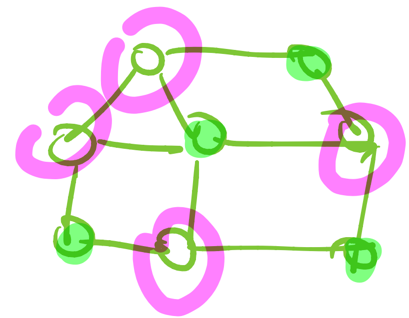
$C' \subseteq C$ is a set cover $= \bigcup_{S \in C'} = \bigcup_{S \in C}$

look for smallest $|C'|$

INDEPENDENT SET (IS)

$G = (V, E)$

$S \subseteq V$ is IS if $\forall u, v \in S : (u,v) \notin E$



S is IS $\Leftrightarrow$ $V - S$ is VC

MIN VC $\Leftrightarrow$ MAX IS