Quiz on Thursday - last 2 lectures on NP completeness

# CSE525 Lec21
# Knapsack Approx.

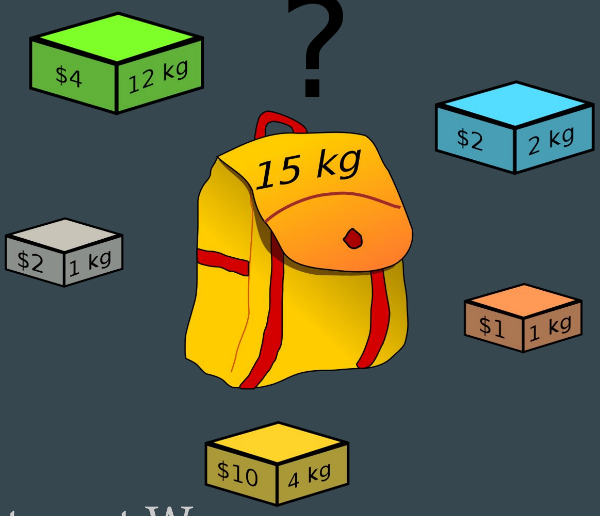
Debajyoti Bera (M21)

0-1 Knapsack - NP complete

# Knapsack

Fractional Knapsack - polynomial time

choose any items $K\%$ $(K \in [0, 100])$




**Instance:**

- n items, v[i] : value of item i, w[i] = weight of item i
- Total capacity: W

$|V_1| + |V_2| + \cdots + |V_n|$
$+ |W_1| + \cdots + |W_n|$
$+ |W| \; \log_2 W$

**Question:** Largest value by choosing items with total weight at most W

DP algorithm

**Fractional Knapsack:** Any fraction of any item can be chosen (value scaled accordingly).

$- O(nW)$ - not poly $(|W|)$

$W = 2^{\lg W}$

$- O(n \sum V_i)$

all values are integers

**0-1 Knapsack:** Any item can be either not chosen or chosen completely.

Fractional Kn

**Greedy algorithm (not covered in course):** Select items in decreasing order of value/wt as long as weight limit is not violated. Optimal for fractional knapsack.

# Approximating 0-1 Knapsack

Fractional solution: take item 5 100%, " 17 ", " 3 ", " 9 ", 6 55%

Approx: take item 5 100%, " 17 ", " 3 ", "

```
def modified-greedy-knapsack(X):
    v-max = largest value in v[...]
    Vg = IntegerGreedy(X) // greedy-knapsack soln. except fraction
    return max { v-max, Vg}
```

that fits in Knapsack → poly time

Running time?

$2 * \max\{v\text{-max}, Vg\} \geq OPT$

Relate Vmax, Vg, OPT

How to show that Modified Greedy Algorithm is 2-rel-approx ?

**Q:** Show that v-max + Vg >= OPT.

optimal for 0-1 Knapsack

→ related to fractional Kn.

Counter example

| a | b | c | d | |
|---|---|---|---|---|
| ~~100~~ 10 13 | 9 | 8 | 7 | : values |
| 2 | 1 | 1 | 1 | : wts |
| 5 | 9 | 8 | 7 | : val/wt |

W = 4

Vg :- { b, c, d } val = 24

opt :- { a, b, c } val = 30

approx :- max(24, 13) = 24

Use the fact that greedy-knapsack gives optimal solution for fractional knapsack.

| Item | J1 | J2 | J3 | J4 | J5 | ... | ... | |
|---|---|---|---|---|---|---|---|---|
| Fract. Greedy | 1 | 1 | 1 | 1 | 0.7 | | | OptFr = ? OPTFRAC |
| Integer Greedy | 1 | 1 | 1 | 1 | | | | Vg = ? = Vg + fraction of some value |

≤ Vmax

OPT FRAC

opt for fractional knapsack ≥ opt for 0-1 Knapsack OPT

# Approximating Knapsack

$\max\{a,b\} \geq a$

$\max\{a,b\} \geq b$

$2 * \max\{a,b\} \geq a+b$

$\therefore\ 2 * \max\{V_g, V_{max}\} \geq OPT$

**Q:** Show that v-max + Vg >= OPT.

Use the fact that greedy-knapsack gives optimal solution for fractional knapsack.

2

Theorem: 0-1 Knapsack has a _??_-relative approximation algorithm.

| Item | J1 | J2 | J3 | J4 | J5 | … | … | |
|---|---|---|---|---|---|---|---|---|
| Fract. Greedy | 1 | 1 | 1 | 1 | 0.7 | | | OptFr = ? |
| Integer Greedy | 1 | 1 | 1 | 1 | | | | Vg = ? |

# Arbitrary approximation of integer 0-1 Knapsack

Any approximation ratio $r = 2, 3, 1.5, 1.05, 1.01, 1.001$

all values are integral

$O(n \sum_{i=1}^{n} v_i)$

- 0-1 knapsack : DP based algorithm
  - If all values are integers, then algorithm runs time depends on values
- solX = optimal set of items of X
- optX = maximum value for X = v[a] + v[b] + ...
- Xs: Scale down all values in X
- solXs = optimal set of items of Xs

Fix s s.t.

1. Running time?
2. **We show optX >= apX >= optX/r**

Approx $\geq \frac{OPT}{r}$

def algo_approx_knapsack(X, r): // r : desired ratio

0. remove any item with weight > W
1. compute (integer) s based on X and r
   values scaling factor
2. scale X by s to create new integer instance Xs
3. solXs = solve 0-1 knapsack on Xs
   using $O(n \sum v_i')$ DP
4. return apX = sum of values of solXs-items using values from X

$\to O(n \sum \lfloor v_i/s \rfloor)$

set of items

running time is polynomial in X

$n \sum \lfloor \frac{v_i}{s} \rfloor \leq n \cdot n \cdot \frac{V_{max}}{s}$

$\leq n^2 \cdot \frac{V_{max} \cdot nr}{V_{max}(r-1)}$

$= n^3 \cdot \frac{r}{r-1}$

# Example

| Objects | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| values ~~Profit~~ | 10 | 5 | 15 | 7 | 6 | 18 | 3 |
| Weight | 2 | 3 | 5 | 7 | 1 | 4 | 1 |
| s=3 | 3 | 1 | 5 | 2 | 2 | 6 | 1 |

~~m = 7~~

~~n = 15~~

W


SolXs = $\{1, 2, 3, 5, 6\}$

ofX = $10 + 5 + 15 + 6 + 18 = 54$

$$VS\text{-}OPT = \left\lfloor \frac{15}{4} \right\rfloor + \left\lfloor \frac{23}{4} \right\rfloor + \left\lfloor \frac{40}{4} \right\rfloor + \left\lfloor \frac{6}{4} \right\rfloor$$

$$= 3 + 5 + 10 + 1$$

$$4 * (3+5+10+1) \leq OPT \leq 4 * ((3+1) + (5+1) + (10+1) + (1+1)) = 4*(3+5+10+1) + 4 * |OPT| \leq n$$

| Items | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Wt <= | W |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| w | 3 | 12 | ... | … | … | … | … | … | 5 | 2 | | |
| v | 24 | 43 | ... | … | … | … | … | … | 16 | 7 | | |
| vs | 4 | 8 | … | … | … | … | … | … | 3 | 1 | s | = 5 |
| vs-sol | 4 | 8 | … | … | … | | | opt. for | scaled values | → | vs-opt | =? |
| vs-re scale | 20 | 40 | … | … | … | | | | | | vs-opt x s | =? |
| apprx | 24 | 43 | … | … | … | | | | | | approx | 24 + 43 + … =? |
| opt | | | | | | | | | 16 | 7 | opt = 16+7 + … | |

Show: OPT <= APPROX + n*s

Q: Show vs-sol 4 + 8 + … >= 3 + 1 + … (sum of scaled vals of opt solution)

Q: Relate APPROX and vs-sol 4 + 8 + …

Q: Relate OPT and vs-opt 3 + 1 + …

$V'_1 + V'_2 + \dots + V'_5$ (vs-sol) $\geq V'_5 + V'_6 + V'_7 + V'_8 + V'_9 + V'_{10}$ (vs-opt)

vs-sol ≥ vs-opt

$V_1 + \dots + V_5 \leq V_5 + \dots + V_{10}$

APPROX ≥ s * vs-sol

OPT ≥ s * vs-opt

$\boxed{\text{vs-opt} + ns \geq \text{OPT}}$

$s(V'_5 + \dots + V'_{10} + 1) \geq V_5 + V_6 + \dots + V_{10}$

$V_5 + V_6 + \dots + V_{10} \geq s * (V'_5 + V'_6 + \dots + V'_{10})$

| Items | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Wt <= | W |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| w | 3 | 12 | ... | … | … | … | … | … | 5 | 2 | | |
| v | 24 | 43 | ... | … | … | … | … | … | 16 | 7 | | |
| vs | 4 | 8 | … | … | … | … | … | … | 3 | 1 | s | = 5 |
| vs-sol | 4 | 8 | … | … | … | | | | | | vs-opt | =? |
| vs-re scale | 20 | 40 | … | … | … | | | | | | vs-opt x s | =? |
| apprx | 24 | 43 | … | … | … | | | | | | approx | =? |
| opt | | | | | | | | | 15 | 7 | | |

Choose s = v-max * (r-1)/nr.

**Q:** Show that APPROX >= OPT/r.

Ex. what s gives ?

$OPT \geq V_{max}$

$APPROX + ns \geq OPT$

$APPROX \geq OPT - ns = OPT - V_{max} \times \frac{r-1}{r}$

$\geq OPT - OPT \times \frac{r-1}{r} = \frac{OPT}{r}$