ナップサック問題で学ぶ動的計画法

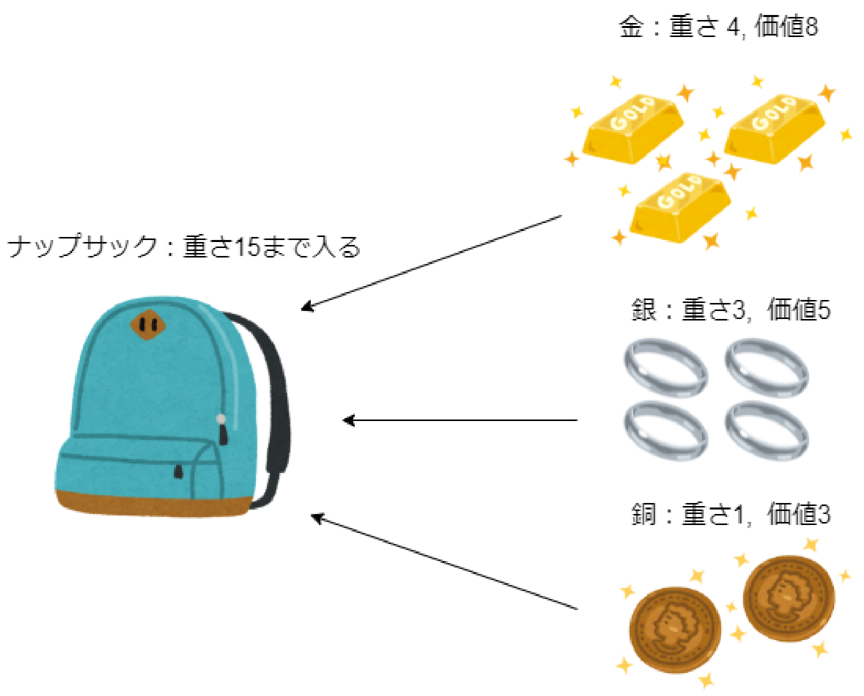
2022/06/04

by @shimamoto

目次

1. ナップサック問題
2. 動的計画法

1. ナップザック問題



ナップサックに入る範囲で価値の総和の最大値は？

愚直に調べると計算量は$\mathcal{O}(2^n)$

→　もっと効率的に調べられないか？ → **動的計画法**

2. 動的計画法

動的計画法のポイント

1. 全体問題を部分問題に分割化
2. 途中の計算結果を保持

3. ナップサック問題での動的計画法

問題の定式化

ナップザック問題 品物テーブル									
N	1	2	3	4	5	6	7	8	9
Item	銅1	銅2	銀1	銀2	銀3	銀4	金1	金2	金3
w[i]	1	1	3	3	3	3	4	4	4
v[i]	3	3	5	5	5	5	8	8	8

N = 9 個の品物があり、i 番目の重さは w[i], 価値は v[i] である。このとき、ナップサックの上限重量 W = 15 を超えない範囲で価値の総和の最大値を求める

3. ナップサック問題での動的計画法

1. 全体問題を部分問題に分割化

i 個までの部分問題に分けて、上限重量がjでの価値の総和の最大値を考える。

このとき、

i 番目の品物を選ぶ場合と選ばない場合の両方のパターンがある。

また、j = i - 1 個の部分問題の上限重量 + v[j]

3. ナップサック問題での動的計画法

1. 全体問題を部分問題に分割化

例 i = 1, w = 1 のとき

銅1を選ぶか選ばないか

- 選ぶ　→ + 3,
- 選ばない　→ + 0

このときは銅1を選ぶパターンを選択する

ナップザック問題 品物テーブル

N	1	2	3	4	5	6	7	8	9
Item	銅 1	銅 2	銀 1	銀 2	銀 3	銀 4	金 1	金 2	金 3
w[i]	1	1	3	3	3	3	4	4	4
v[i]	3	3	5	5	5	5	8	8	8

3. ナップサック問題での動的計画法

2. 途中の計算結果を保持

i 個までの部分問題に分けて、上限重量がjの価値の総和の最大値を保持

例：i = 1, w = 1のとき

dp[1][1] = 3, dp: 価値の総和の最大値を保持する配列

このdpは次の計算に使用する

3. ナップサック問題での動的計画法

例：i = 2, w = 2のとき

銅2を選ぶか選ばないか

- 選ぶ　→ + dp[1][1] + 3 = 6
- 選ばない　→ + dp[1][1] = 3

dp[2][2] = 6

ナップザック問題 品物テーブル

N	1	2	3	4	5	6	7	8	9
Item	銅 1	銅 2	銀 1	銀 2	銀 3	銀 4	金 1	金 2	金 3
w[i]	1	1	3	3	3	3	4	4	4
v[i]	3	3	5	5	5	5	8	8	8

3. ナップサック問題での動的計画法

例：i = 3, w = 3のとき

銅2を選ぶか選ばないか

- 選ぶ　→ + dp[2][0] + 3 = 3 ,
- 選ばない　→ + dp[2][2] = 6

dp[3][3] = 6

ナップザック問題 品物テーブル

N	1	2	3	4	5	6	7	8	9
Item	銅1	銅2	銀1	銀2	銀3	銀4	金1	金2	金3
w[i]	1	1	3	3	3	3	4	4	4
v[i]	3	3	5	5	5	5	8	8	8

ex

- 数学的には漸化式を利用した方法といえる

品物を選ぶ場合

$$\mathrm{dp}_{\mathrm{i+1,w}} = \mathrm{Max}(\mathrm{dp}_{\mathrm{i+1,w}}, \mathrm{dp}_{\mathrm{i,w-w_i}} + \mathrm{v_i})$$

品物を選ばない場合

$$\mathrm{dp}_{\mathrm{i+1,w}} = \mathrm{Max}(\mathrm{dp}_{\mathrm{i+1,w}}, \mathrm{dp}_{\mathrm{i,w}})$$

dp : 途中総重量wにおける価値を保存しているテーブル
$\mathrm{w_i}$: ある品物の重量
$\mathrm{v_i}$: ある品物の価値