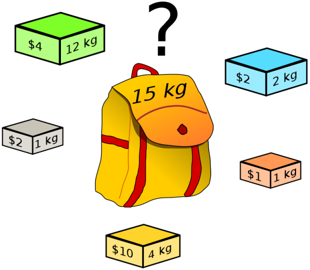
Problema da Mochila Binária

Fernando Gomes, Leonardo Holtz

Universidade Federal do Rio Grande do Sul



Sumário

Definição intuitiva

Dado um conjunto de itens, com cada item tendo um peso e um custo, qual a escolha de itens tal que a soma de seus pesos é menor que a capacidade de uma mochila a soma de seus custos é a maior possível?

Definição matemática: problema de otimização

Dados uma mochila com capacidade máxima W, um conjunto de n itens $y_1, y_2, ..., y_n$, cada um com um peso w_i e um valor v_i:

$$\text{maximizar} \sum_{i=1}^{n} v_i x_i$$

$$\text{sujeito a } \sum_{i=1}^{n} w_i x_i \leq W \text{ e } x_i \in \{0, 1\}$$

Neste caso, x_i representa o número de instâncias do item i dentro da mochila.

Problema de otimização: Exemplo

Dada uma mochila com capacidade máxima $W = 5$ e dado 4 itens $\{y_1, y_2, y_3, y_4\}$ com valores {50, 40, 30, 45} e com os pesos {4, 2, 1, 3}, qual a seleção de itens que possui a maior soma possível de valores com a soma de pesos menor ou igual a 5?

Definição matemática: problema de decisão

Dados uma mochila com capacidade máxima W, um valor mínimo V, um conjunto de n itens $y_1, y_2, ..., y_n$, cada um com um peso w_i e um valor v_i:

$$\text{maximizar} \sum_{i=1}^{n} v_i x_i \text{ t.q.}$$

$$\sum_{i=1}^{n} w_i x_i \leq W \text{ e } x_i \in \{0,1\}$$
$$\sum_{i=1}^{n} v_i x_i \geq V \text{ e } x_i \in \{0,1\} \text{ ?}$$

O problema de decisão se torna: "Existe uma seleção de x_i que satisfaça essa definição?"

Problema de decisão: Exemplo

Dada uma mochila com capacidade máxima $W = 5$ e dado 4 itens $\{y_1, y_2, y_3, y_4\}$ com valores {50, 40, 30, 45} e com os pesos {4 2 1 3}, existe uma seleção de itens que possui a soma de seus valores igual ou maior que 85 e a soma de pesos menor ou igual a 5?

Estudar na noite de anterior de uma prova:
Um estudante que tem uma prova no dia seguinte e tem um conjunto de capítulos para estudar, cada qual com uma probabilidade de cair na prova e um tempo necessário para estudo. Selecionar quais capítulos estudar antes da prova é análogo ao problema da mochila binária, e como veremos em breve, é *NP-completo*.
Outros exemplos:

- Gerenciadores de download
- Programas espaciais

Algoritmo de verificação

Para provar que PM[1] $\in$ NP, precisamos mostrar que existe um algoritmo capaz de verificar se um certificado do PM é correto em tempo polinomial.

[1] PM = Problema da mochila binária

Algoritmo em pseudocódigo

P: conjunto de pesos p_i
V: conjunto de valores v_i
VMIN: valor minímo para o problema de decisão
W: carga máxima da mochila
n: tamanho do problema
CERTIFICADO: seleção $x_1, ..., x_n$ $t.q.$ $x \in \{0, 1\}$

Pré-algoritmo

```
VerificaPM(P, V, VMIN, W, n, CERTIFICADO)
    ...
    returns SIM or NÃO
```

Algoritmo completo

```
VerificaPM(P, V, VMIN, W, n, CERTIFICADO)
   PesoTotal = 0
   ValorTotal = 0
   for i in 1:n
       PesoTotal += CERTIFICADO[i] * P[i]
       ValorTotal += CERTIFICADO[i] * V[i]
   endfor
   if PesoTotal <= W && ValorTotal > VMIN
       return True
   else
       return false
```

Análise de complexidade

- As atribuições das linhas 2 e 3 tem complexidade $\mathcal{O}(1)$.
- O laço *for* executa n vezes, e a complexidade das atribuições e multiplicações são constantes. Este laço é expresso pelo seguinte somatório:

$$\sum_{i=1}^{n} 2$$

  A resolução do somatório é dada por $2n$, logo, a complexidade total do laço é $\mathcal{O}(n)$.
- o retorno na linha 9 e 11 é constante.
- A complexidade total do algoritmo é, portanto, $\mathcal{O}(n)$.
- O problema da mochila (como problema de decisão) pode, então, ser verificado em tempo polinomial. Com isso concluímos que pertence ao conjunto NP.

Soma de subconjuntos

- Para demonstrar que PM $\in$ NP-difícil, utilizaremos uma redução do problema da soma de subconjuntos, que já sabemos que é NP-completo, para o problema da mochila.
- Note que estamos utilizando problemas de decisão, mas se demonstrarmos que eles pertencem a NP-difícil, é evidente que o problema de otimização associado também será NP-difícil (o problema de decisão é mais "fácil").

Soma de subconjuntos

Dado um conjunto de n números inteiros positivos $a_1, a_2, ..., a_n$, e um valor $t \geq 0$, existe um subconjunto destes números tal que:

$$\sum_{i=1}^{n} a_i x_i = t \text{ e } x_i \in \{0, 1\}$$

x_i indica se a_i faz parte do subconjunto (1) ou não (0)

O problema de decisão é similar ao anterior: "Existe uma seleção de a_i que satisfaça essa definição?"

Soma de subconjuntos: Exemplo

Considere o conjunto $A = \{1, 2, 3, 4, 5, 6\}$. Existe um subconjunto tal que a soma de seus elementos é igual a 10?
Sim! Um exemplo é o subconjunto $A_1 = \{5, 4, 1\}$. Apesar de existirem outros conjuntos que satisfaçam essa condição, é necessário a existência de pelo menos um para que a resposta para a pergunta seja sim.

Redução PM para SSC

A redução das instâncias do Problema da Mochila para a Soma de subconjuntos é trivial, já que devemos somar um subconjunto de valores a partir de todo um conjunto em ambos os casos.

Para todos os n elementos $a_1, a_2, ..., a_n$ fazemos a seguinte transformação:

$$w_i = a_i$$

$$v_i = a_i$$

Para a capacidade máxima W e o valor mínimo V, fazemos a seguinte transformação:

$$W = t$$

$$V = t$$

Redução PM para SSC

Com isso, conseguimos mapear as exatas mesmas respostas do problema de decisão da Soma de subconjuntos para o problema de decisão da Mochila.

$$\sum_{i=1}^{n} w_i x_i \leq W \text{ terá as mesmas respostas (sim ou não) que } \sum_{i=1}^{n} a_i x_i = t$$

$$\sum_{i=1}^{n} v_i x_i \geq V \text{ terá as mesmas respostas (sim ou não) que } \sum_{i=1}^{n} a_i x_i = t$$

Redução PM para SSC: Exemplo

Considerando o conjunto $a = \{2, 3, 5, 7, 10\}$ e um valor $t = 17$, a redução se dá da seguinte forma:

$$w_i = a_i \longrightarrow w = \{2, 3, 5, 7, 10\}$$

$$v_i = a_i \longrightarrow v = \{2, 3, 5, 7, 10\}$$

$$W = t \longrightarrow W = 17$$

$$V = t \longrightarrow V = 17$$

Redução PM para SSC: Exemplo

O problema da SSC, com o conjunto $a = \{2, 3, 5, 7, 10\}$ e um valor $t = 17$ possui solução. Uma solução válida é o subconjunto {2, 3, 5, 7}.
Após a redução, temos que este mesmo subconjunto é também uma solução válida para o Problema da Mochila:

Peso máximo da mochila: $2 + 3 + 5 + 7 \leq W = 17$

Valor mínimo da mochila: $2 + 3 + 5 + 7 \geq V = 17$

Algoritmo de redução

n: tamanho do problema
t: soma total formada por um subconjunto dos valores a_i
a: vetor com os valores a_i

Algoritmo de Redução

```
ReducaoSSCparaPM(n, t, a)
    W = t
    V = t
    for i in 1:n
        w[i] = a[i]
        v[i] = a[i]
    endfor
    return n, w, v, W, V
```

Análise da complexidade

- As atribuições das linhas 2 e 3 tem complexidade $\mathcal{O}(1)$.
- Nas linhas 4 a 7, o laço *for* executa n vezes com duas atribuições que são constantes. Isso é expresso pelo seguinte somatório

  $$\sum_{i=1}^{n} 2$$

  A resolução do somatório é dada por $2n$, logo, a complexidade total do laço é $\mathcal{O}(n)$.
- O retorno na linha 8 tem a complexidade $\mathcal{O}(1)$.
- A complexidade total do algoritmo é, portanto, $\mathcal{O}(n)$.
- Como a complexidade da redução está dentro do tempo polinomial, provamos que o Problema da Mochila $\in$ NP-difícil.

NP-completude do Problema da Mochila

- Foi provado que existe um algoritmo de verificação para o Problema da Mochila em tempo polinomial, logo, temos que o Problema da Mochila $\in$ NP.
- Foi provado também que existe uma redução em tempo polinomial do problema da Soma de Subconjuntos para o Problema da Mochila, logo, temos também que o Problema da Mochila $\in$ NP-difícil.
- Com isso, concluímos que o Problema da Mochila $\in$ NP-completo.