Florence Alyssa Sakuma Shibata, Shayenne da Luz Moura

Método Simplex
Fase 2

São Paulo

2015

Sumário

Introdução

O método simplex é baseado em encontrar uma solução viável ótima para um problema de programação linear e realiza esta busca movendo-se de uma solução viável básica para outra, percorrendo os lados do poliedro que define a região viável, sempre numa direção onde o custo se reduz. Enfim, uma solução viável básica é alcançada quando nenhuma das direções viáveis reduzem o custo; então a solução viável básica é ótima e o algoritmo termina.

1 Descrição do algoritmo

O método simplex utilizado como base para o desenvolvimento do algoritmo está descrito em Bertsimas e Tsitsiklis (1997, pág. 90-91).

Dadas uma matriz $A \in \Re^{m \times n}$, uma solução viável básica $x \in \Re^n$, $b \in \Re^m$, o vetor de custos $c \in \Re^n$, o algoritmo realiza os seguintes passos:

1 Gera a matriz básica B associada a x.

2 Calcula o vetor de custos reduzidos para toda variável não básica. Se nenhuma componente é negativa, então a solução viável básica atual é ótima, acabou.

3 Caso contrário, armazena o índice da variável cujo custo reduzido foi menor. Calcula $u = B^{-1}A_j$. Se nenhum componente de u é positivo, então o custo ótimo é $-\infty$, acabou;

4 Caso contrário, toma $\theta^* = min_{(i=1,\ldots,m|u_i>0)}\{\frac{x_{B(i)}}{u_i}\}$. Seja l o índice onde o mínimo foi encontrado. Forma uma nova base substituindo $A_{B(l)}$ por A_j. Sendo y é a nova solução viável básica, os valores das novas variáveis básicas são $y_j = \theta^*$, $y_{B(i)} - \theta^* u_i$, $i \neq l$. Volta ao passo 1.

É necessário que o problema possua pelo menos uma solução viável básica e que todas as soluções viáveis básicas sejam não degeneradas.

2 Funcionamento do algoritmo

A seguir está a descrição do funcionamento do algoritmo quando o problema de programação linear dado possui pelo menos uma solução ótima ou quando o custo ótimo é ilimitado.

Em alguns exemplos não são dados os vetores b, uma vez que dada a solução viável básica x e não é necessário calcular a partir de b.

2.1 Solução ótima

Existem duas possibilidades para este caso:

- A solução ótima é dada;
- A solução ótima é encontrada ao percorrer as direções viáveis que reduzem o custo.

Ao aplicar o algoritmo com os dados descritos a seguir, pode-se verificar seu funcionamento.

```
> A = [1 0 1 0 0; 0 1 0 1 0; 3 2 0 0 1];
> x = [0;0;4;6;18];
> b = [4;16;18];
> m = 3;
> n = 5;
> c = [3;5;0;0;0];

> [ind, v] = simplex(A, b, c, m, n, x);
#=====================================
           Iteracao: 0
#=====================================

Variaveis basicas:
3:  4.000000
4:  6.000000
5:  18.000000

Valor funcao objetivo: 0.000000
```

O algoritmo calcula quais são as váriáveis básicas a partir da solução x dada e o custo associado a essa solução c^Tx.

```
Custos reduzidos:
1: 3.000000
2: 5.000000
```

Como o vetor de custos reduzidos das variáveis não básicas é positivo não existe direção viável que reduza o custo. Logo, o custo ótimo é alcançado em x. Assim, x é solução ótima.

```
Solucao otima com custo 0.000000:
1  0.000000
2  0.000000
3  4.000000
4  6.000000
5  18.000000
```

O algoritmo então devolve o valor da solução ótima que é igual a solução viável básica x dada.

O problema de programação linear descrito a seguir também possui uma solução viável básica ótima que é encontrada após algumas iterações.

```
> A = [1 2 3 0 1 0 0 0 ;-1 2 6 0 0 1 0 0; 0 4 9 0 0 0 1 0; 0 0 3 4 0 0 0 1];
> c = [0;0;0;0;1;1;1;1];
> x = [0; 0; 0; 0; 3; 2; 5; 1];
> m = 4;
> n = 8;

> [ind, v] = simplex(A, b, c, m, n, x);
#====================================
           Iteracao: 0
#====================================
Variaveis basicas:
5:  3.000000
6:  2.000000
7:  5.000000
8:  1.000000
```

```
Valor funcao objetivo: 11.000000

Custos reduzidos:
1: 0.000000
2: -8.000000
3: -21.000000
4: -4.000000
```

O algoritmo calcula quais são as váriáveis básicas da solução viável básica x, o custo associado a ela e os custos reduzidos das variáveis não básicas. Quando encontra um valor negativo sabe-se que existe uma solução viável básica, diferente de x, cujo custo associado é menor. A direção de menor custo associado é escolhida, neste caso, a variável 3.

```
Entra na base: 3

Direcao
5: 3.000000
6: 6.000000
7: 9.000000
8: 3.000000

Theta*: 0.333333

Sai da base: 6
```

É calculada a direção viável que reduz o custo, além do θ máximo que se pode andar para estar sobre uma nova solução viável básica. Como nessa direção o valor da função objetivo é menor que $-\infty$ existe uma solução viável básica associada. A variável encontrada que possuir menor custo reduzido entra na base e apenas uma das variáveis básicas torna-se zero, pois as soluções são todas não degeneradas. Neste caso, aquela que torna o θ máximo, a variável 6. Uma nova iteração se inicia com a solução viável básica associada a nova base.

```
#=====================================
           Iteracao: 1
#=====================================
Variaveis basicas:
5:  2.000000
3:  0.333333
```

```
7:  2.000000
8:  0.000000

Valor funcao objetivo: 4.000000

Custos reduzidos:
1: -3.500000
2: -1.000000
6: 3.500000
4: -4.000000

Entra na base: 4

Direcao
5: 0.000000
3: 0.000000
7: 0.000000
8: 4.000000

Theta*: 0.000000

Sai da base: 8
```

Esta iteração realiza os mesmos passos da iteração anterior, encontrando a direção de menor custo reduzido, colocando na base a variável 4. O θ máximo que se pode andar para estar sobre uma nova solução viável básica é menor que $-\infty$ e a variável 8 sai da base. Uma nova iteração se inicia com a solução viável básica associada a nova base.

```
#=====================================
           Iteracao: 2
#=====================================
Variaveis basicas:
5:  2.000000
3:  0.333333
7:  2.000000
4:  0.000000

Valor funcao objetivo: 4.000000

Custos reduzidos:
```

```
1: -3.000000
2: -2.000000
6: 3.000000
8: 1.000000

Entra na base: 1

Direcao
5: 1.500000
3: -0.166667
7: 1.500000
4: 0.125000

Theta*: 0.000000

Sai da base: 4
```

Idêntica a iteração anterior, encontra a direção de menor custo reduzido, colocando na base a variável 1. O θ máximo que se pode andar para estar sobre uma nova solução viável básica é menor que $-\infty$ e a variável 4 sai da base. Uma nova iteração se inicia com a solução viável básica associada a nova base.

```
#=====================================
           Iteracao: 3
#=====================================
Variaveis basicas:
5:  2.000000
3:  0.333333
7:  2.000000
1:  0.000000

Valor funcao objetivo: 4.000000

Custos reduzidos:
4: 24.000000
2: -8.000000
6: 0.000000
8: 7.000000

Entra na base: 2
```

```
Direcao
5: 4.000000
3: 0.000000
7: 4.000000
1: -2.000000

Theta*: 0.500000

Sai da base: 5
```

Esta iteração realiza os mesmos passos da iteração anterior, encontrando a direção de menor custo reduzido, colocando na base a variável 2. O θ máximo que se pode andar para estar sobre uma nova solução viável básica é menor que $-\infty$ e a variável 5 sai da base. Uma nova iteração se inicia com a solução viável básica associada a nova base.

```
#=====================================
           Iteracao: 4
#=====================================
Variaveis basicas:
2:  0.500000
3:  0.333333
7:  0.000000
1:  1.000000

Valor funcao objetivo: 0.000000

Custos reduzidos:
4: 0.000000
5: 2.000000
6: 2.000000
8: 1.000000

Solucao otima com custo 0.000000:
1  1.000000
2  0.500000
3  0.333333
4  0.000000
5  0.000000
6  0.000000
```

```
7  0.000000
8  0.000000
```

Esta possui o vetor de custos reduzidos positivo. Logo não existe direção viável que reduza o custo. O algoritmo devolve a solução viável básica em que está como solução ótima, pois esta possui menor custo associado.

2.2 Custo ótimo ilimitado

Como exemplo do funcionamento do algoritmo para um problema de programação linear que contém solução ilimitada segue os dados de entrada.

```
> A = [1 -1 1 0; 2 -1 0 1];
> c = [-2; -1; 0; 0];
> m = 2;
> n = 4;
> x = [0;0;10;40];
```

A execução do algoritmo devolve as iterações a seguir.

```
> [ind, v ]= simplex(A, b, c, m, n, x);
#====================================
           Iteracao: 0
#====================================
Variaveis basicas:
3:  10.000000
4:  40.000000

Valor funcao objetivo: 0.000000

Custos reduzidos:
1: -2.000000
2: -1.000000

Entra na base: 1

Direcao
3: 1.000000
4: 2.000000
```

```
Theta*: 10.000000

Sai da base: 3
```

Esta iteração encontra a direção de menor custo reduzido, neste caso, colocando na base a variável 1. O θ máximo que se pode andar para estar sobre uma nova solução viável básica é menor que $-\infty$ e a variável 3 sai da base. Uma nova iteração se inicia com a solução viável básica associada a nova base.

```
#=====================================
           Iteracao: 1
#=====================================
Variaveis basicas:
1:  10.000000
4:  20.000000

Valor funcao objetivo: -20.000000

Custos reduzidos:
3: 2.000000
2: -3.000000

Entra na base: 2

Direcao
1: -1.000000
4: 1.000000

Theta*: 20.000000

Sai da base: 4
```

Assim como a iteração anterior, encontra a direção de menor custo reduzido, colocando na base a variável 2. O θ máximo que se pode andar para estar sobre uma nova solução viável básica é menor que $-\infty$ e a variável 4 sai da base. Uma nova iteração se inicia com a solução viável básica associada a nova base.

```
#=====================================
           Iteracao: 2
```

```
#=====================================
Variaveis basicas:
1:  30.000000
2:  20.000000

Valor funcao objetivo: -80.000000

Custos reduzidos:
3: -4.000000
4: 3.000000
```

Ao andar na direção da variável básica j cujo custo reduzido é o menor negativo encontra-se o vetor $u = B^{-1}A_j$ com elementos não positivos, isso quer dizer que a direção encontrada leva a solução ilimitada, com custo ótimo $-\infty$.

```
O valor da funcao objetivo vai para -Inf

Direcao
1: -1.000000
2: -2.000000
0: 0.000000
0: 0.000000
```

O algoritmo devolve então a direção viável que possui solução ilimitada.

Conclusão

O algoritmo simplex implementado para resolver problemas de programação linear com soluções viáveis básicas não degeneradas e com pelo menos uma ótima é correto.

Mantendo as hipóteses em todos os casos, obtêm-se os resultados esperados em problemas cuja solução ótima é dada, quando o custo ótimo é ilimitado, ou seja necessário encontrar uma direção viável cujo custo seja menor, encontrando a solução ótima.

Referências

BERTSIMAS, D.; TSITSIKLIS, J. *Introduction to Linear Optimization*. [S.l.]: Athena Scientific, 1997. (Athena Scientific series in optimization and neural computation). ISBN 9781886529199. Citado na página 3.