MAC0315 - EP2: Fase 2 do método Simplex

Felipe Anez de Toledo Blassioli

May 18, 2015

1 Iteração do Simplex

Foi feita a implementação ingênua da fase do método 2 simplex de acordo com a página 90 do livro. Sinalizei com comentários as passagens de cada iteração, que em linhas gerais consistem em:

1. Cálculo do vetor de custos reduzidos.
2. Cálculo da direção viável.
3. Cálculo do θ^*.(o máximo que podemos andar na direção viável)
4. Atulização da solução viável básica atual e da base correspondente.

2 Exemplos

2.1 Com solução ótima

Queremos resolver o problema:

$$\begin{aligned} \min \quad & -x_1 - x_2 && (1)\\ \text{s.t.} \quad & 2x_1 + x_2 \le 4 && (2)\\ & x_1 + 3x_2 \le 8 && (3)\\ & x_1, x_2 \ge 0 && (4) \end{aligned}$$

Na forma padrao:

$$\begin{aligned} \min \quad & -x_1 - x_2 - 0s_1 - 0s_2 && (5)\\ \text{s.t.} \quad & 2x_1 + x_2 + s_1 = 4 && (6)\\ & x_1 + 3x_2 + s_2 = 8 && (7)\\ & x_1, x_2, s_1, s_2 \ge 0 && (8) \end{aligned}$$

Temos que $m = 2$ e $n = 4$.
Para esse problema é fácil escolher um conjunto L.I de colunas de A:

$$A = \begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & 3 & 0 & 1 \end{bmatrix}$$

Já que as duas últimas colunas A_3 e A_4 formam a base canônica do $\mathbb{R}^2$. Tendo em mãos a base, achamos nossa S.V.B resolvendo o sistema de equações:

```
base = [3,4];
x = zeros(n , 1);
x(base) = A(:,base) \ b;
```

```
Simplex: Fase 2

Iterando 0
3 4.000000
4 8.000000

Valor da funcao objetivo : 0.000000
```

Temos que calcular o custo reduzido nos índices das variáveis não-básicas (já que o custo reduzido das variáveis básicas é zero). Isso é feito do seguinte modo:

$$\bar{c_j} = c_j - c_j^T B^{-1} A_j$$

Em Octave:

```
tmp = setdiff( 1:n , B_ind );
reduced_costs = c(tmp) - c(B_ind)*inv(B)*A(:,tmp);
```

Na primeira iteração temos como resultado:

```
Custos reduzidos
1 -1.000000
2 -1.000000

Entra na base : 1
```

Temos que todos os custos reduzidos são negativos, fossem TODOS não-negativos, a S.V.B atual seria uma solução ótima, como não é o caso, escolhemos um dos índices (j) e achamos a j-ésima direção viável $u = B^-1A_j$. O custo da função objetivo decresce ao longo dessa direção. É importante ressaltar que a suposição de que não existem S.V.B degeneradas garante que as direções básicas calculadas são viáveis.

Em Octave:

```
find( reduced_costs < 0 );

j = tmp( find( reduced_costs < 0 )(1) );
u = inv(B) * A(:,j);
```

```
Direcao
3 2.000000
4 1.000000
```

Ao se mover ao longo da direção **d** a variável x_j fica positiva e todas as outras variáveis não básicas permanecem em zero. A esse crescimento que nos referimos ao dizer que x_j (ou A_j) entra na base.

Como andar na j-ésima direção básica reduz o custo da função objetivo, queremos saber qual o máximo que podemos andar nesta direção. Para isso calculamos $\theta^* = min \frac{X_{B(i)}}{u_i}$ para $\{i = 1, .., m | u_i > 0\}$.

Em Octave:

```
idx = find( u > 0 );
[theta, l] = min( x(B_ind(idx)) ./ u(idx) );

Theta*
2.000000
```

Agora que temos o θ^*, resta-nos atualizar nossa S.V.B corrente e a base correspondente e continuar as iterações do algoritmo. A atualização consiste em trocar $A_B(l)$ por A_j e a S.V.B atual tem componente j igual a θ^* e as componentes básicas subtraídas do quanto andamos: $x_B(i) - \theta * u_i$.

Em Octave:

```
A( :, B_ind(l) ) = A( :, j );
x(B_ind) = x(B_ind) - theta*u;
x(j) = theta;

Sai da base: 3
```

Na segunda iteração ocorre a mesma coisa que na primeira. A única diferença é que temos uma componente positiva dos custos reduzidos.

```
Iterando 1
1 2.000000
4 6.000000

Valor da funcao objetivo : -2.000000


Custos reduzidos
2 -0.500000
3 1.000000

Entra na base : 2

Direcao
1 0.500000
4 2.500000

Theta*
2.400000

Sai da base: 4
```

Na terceira iteração temos que todas as componentes do custo reduzido são maiores do que zero e, portanto temos uma solução ótima.

```
Iterando 2
1 0.800000
2 2.400000

Valor da funcao objetivo : -3.200000
```

```
Custos reduzidos
3 1.000000
4 1.000000
ind = 0
v =

   0.80000
   2.40000
   0.00000
   0.00000


Solucao otima encontrada com custo : -3.200000
1 0.800000
2 2.400000
3 0.000000
4 0.000000
```

2.2 Ilimitado

No problema abaixo, é evidente que quanto mais x_1 e x_2 crescem, menor fica a função objetivo.

$$\begin{aligned} \min \quad & -x_1 - x_2 && (9)\\ \text{s.t.} \quad & x_1 - x_2 = 1 && (10)\\ & x_1, x_2 \geq 0 && (11)\\ & && (12) \end{aligned}$$

De fato, o custo ótimo é $-\infty$, já que nenhuma componente da j-ésima direção viável é positiva:

Simplex: Fase 2
Iterando 0 1 1.000000
Valor da funcao objetivo : -1.000000
Custos reduzidos 2 -2.000000
Entra na base : 2
Direcao 1 -1.000000 ind = -1 v =
-1 0
Custo otimo menos infinito
Direcao viavel : 1 -1.000000 2 0.000000

2.3 Verificação dos problemas usando Octave

Utilizando a rotina glpk do Octave, obtém-se os mesmos resultados:

```
A = [2,1,1,0;1,3,0,1]
b = [4;8]
c = [-1, -1, 0, 0]
[xopt,zmx]=glpk(c,A,b)
```

```
A = [1,-1]
b = [1]
c = -[1, 1]
[xopt,zmx]=glpk(c,A,b)

xopt =

    0.80000
    2.40000
    0.00000
    0.00000

zmx = -3.2000

xopt =

     NA
     NA

zmx = NA
```