Métodos de Apoio à Decisão

Algoritmo do Simplex

João Pedro Pedroso

2021/2022

- Aulas passadas:
  - Formulação em programação matemática
    1. variáveis
    2. restrições
    3. objetivo
  - Exemplos em otimização linear
- Hoje:
  - Algoritmo do simplex para programação linear

Resolução gráfica

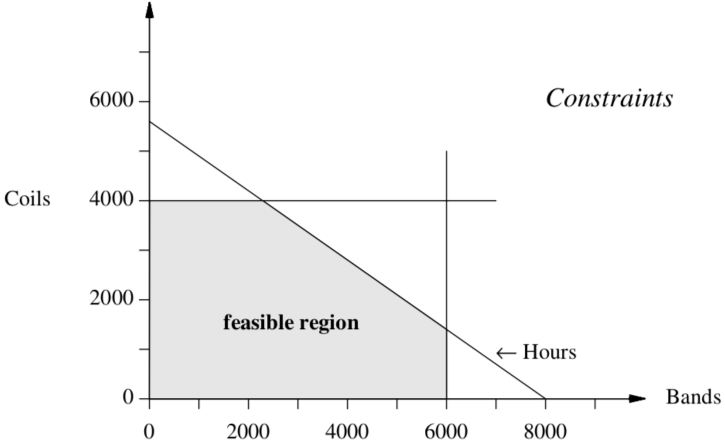
$$\begin{aligned}
\text{maximize} \quad & 25x_B + 30x_C \\
\text{subject to} \quad & x_B/200 + x_C/140 \leq 40 \\
& 0 \leq x_B \leq 6000 \\
& 0 \leq x_C \leq 4000
\end{aligned}$$

Modelo AMPL

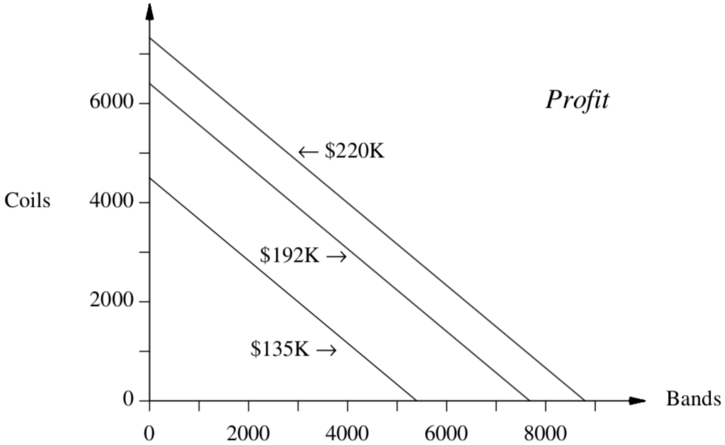
Modelo

```
var xb;
var xc;

maximize z: 25*xb + 30*xc;

subject to
hours: xb/200 + xc/140 <= 40;

capB: 0 <= xb <= 6000;
capC: 0 <= xc <= 4000;
```

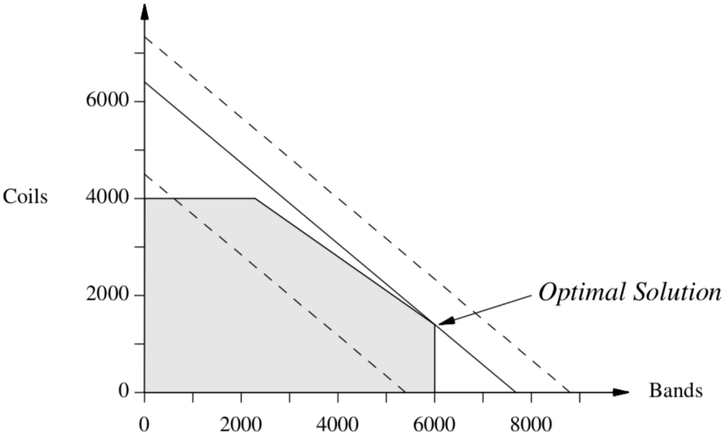
Visualização gráfica: região admissível



Visualização gráfica: linhas de isolucro



Visualização gráfica: ótimo



Algoritmo do Simplex

Algoritmos do século XX

Computing in Science and Engineering, volume 2, no. 1, 2000: 10 algorithms with the greatest influence on the development and practice of science and engineering in the 20th century:

- Metropolis Algorithm for Monte Carlo
- **Simplex Method for Linear Programming**
- Krylov Subspace Iteration Methods
- The Decompositional Approach to Matrix Computations
- The Fortran Optimizing Compiler
- QR Algorithm for Computing Eigenvalues
- Quicksort Algorithm for Sorting
- Fast Fourier Transform
- Integer Relation Detection
- Fast Multipole Method

(sem ordem particular)

Método do simplex

- Desenvolvido para a resolução de problemas de otimização lineares
- Proposto por George Dantzig (década '40, sec.XX)
- Previamente: método da programação linear, Leonid Kantorovich em 1939 (um método não "computacional" tinha sido proposto por Fourier)
- Hoje em dia: muitas aplicações, incluindo em *otimização inteira*

Algoritmos para programação linear

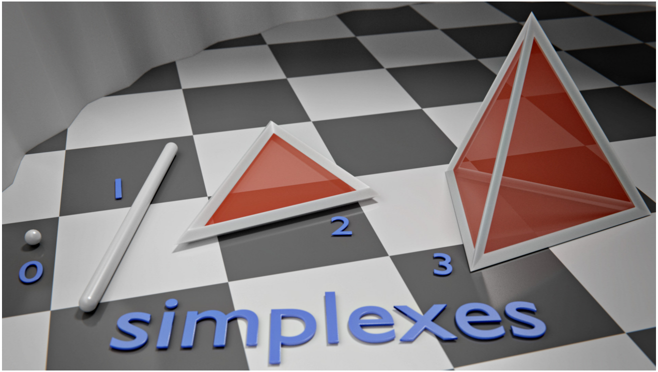
- Vimos como resolver problemas com duas variáveis graficamente
- Para problemas com mais de duas variáveis, é necessário utilizar um algoritmo
- Na prática, o algoritmo mais utilizado é o do *simplex*
  - permite a resolução de problemas com muitos milhares de variáveis e restrições
  - funciona através da análise e movimentos em pontos extremos (vértices) da região admissível
  - há problemas particulares em que não é eficiente: pode demorar tempo exponencial em termos do tamanho do problema, a encontrar a solução
  - para problemas "pequenos" ($<<$ 1000000 variáveis/restrições), geralmente é mais rápido do que algoritmos "eficientes" (e é o mais utilizado)

Noções preliminares: conjuntos convexos, pontos extremos

- Um conjunto S é convexo se para qualquer par de pontos do conjunto, o segmento de reta que os une está completamente contido em S
- A região admissível de qualquer problema linear é um conjunto convexo
- Um ponto P diz-se um ponto extremo de um conjunto S se para qualquer segmento de reta que esteja completamente contido em S e que contenha P, se verifica que P é um ponto extremo desse segmento de reta

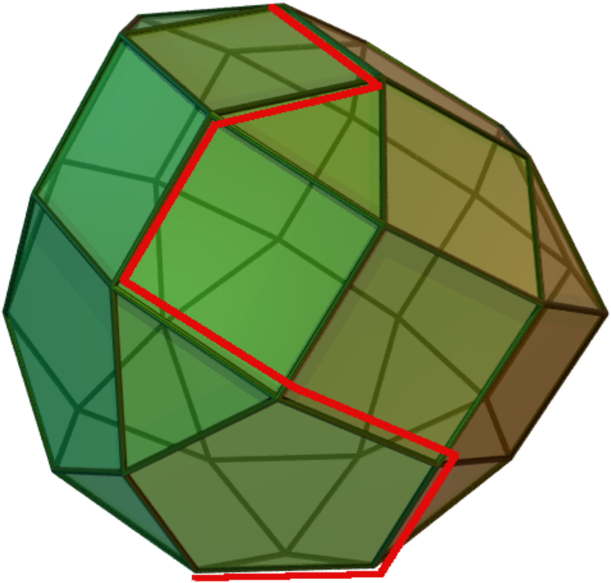
Noções preliminares: poliedro, polítopo

- Em espaços de dimensão superior a dois:
  - ao conjunto de pontos que satisfazem uma desigualdade linear chama-se um *semi-espaço*
- A intersecção de semi-espaços é chamada um *poliedro*
- Um poliedro limitado é um polítopo
- Num espaço de dimensão n, um *polítopo com* $n+1$ *vértices* é um simplex



Hjhornbeck, https://commons.wikimedia.org/w/index.php?curid=57828604

Método do simplex: visualização



Forma standard de um problema linear

- Na forma standard de um problema linear:
  1. todas as restrições são equações
  2. todas as variáveis são não-negativas
- Preliminar para a utilização do algoritmo do simplex
- Permite fazer uma análise da solução obtida

Exemplo

$$
\begin{array}{lll}
\text{maximizar} & z = 4x_1 + 3x_2 & \\
\text{sujeito a} & x_1 + x_2 & \leq 40 \\
 & 2x_1 + x_2 & \leq 60 \\
 & x_1, x_2 & \geq 0
\end{array}
$$

- como colocá-lo na forma standard?

Redução à forma standard: *variáveis de desvio*

- variável de desvio: quantidade de recurso correspondente a uma restrição que não é utilizada
  - exemplo: $s_1 = 40 - x_1 - x_2 \Rightarrow x_1 + x_2 + s_1 = 40$
  - o mesmo para a segunda restrição
- as restrições são satisfeitas sse $s_i \geq 0, \forall i$

$$\begin{aligned} \text{maximizar} \quad & z = 4x_1 + 3x_2 \\ \text{sujeito a} \quad & x_1 + x_2 \leq 40 \\ & 2x_1 + x_2 \leq 60 \\ & x_1, x_2 \geq 0 \end{aligned}$$

$$\begin{aligned} \text{maximizar} \quad & z = 4x_1 + 3x_2 \\ \text{sujeito a} \quad & x_1 + x_2 + s_1 = 40 \\ & 2x_1 + x_2 + s_2 = 60 \\ & x_1, x_2, s_1, s_2 \geq 0 \end{aligned}$$

Forma standard

$$
\begin{array}{lrrrrr}
\max / \min z = & c_1 x_1 + & c_2 x_2 + & \ldots + & c_n x_n & \\
\text{sujeito a} & a_{11} x_1 + & a_{12} x_2 + & \ldots + & a_{1n} x_n = & b_1 \\
 & a_{21} x_1 + & a_{22} x_2 + & \ldots + & a_{2n} x_n = & b_2 \\
 & \ldots & & & & \\
 & a_{m1} x_1 + & a_{m2} x_2 + & \ldots + & a_{mn} x_n = & b_m \\
 & x_1, & x_2, & \ldots, & x_n \geq & 0
\end{array}
$$

- todas as restrições são equações (i.e., igualdades)
- todas as variáveis são não negativas
  - se no problema original $x_i \leq 0$
    $\rightarrow$ substituir por $-y_i, y_i \geq 0$
  - se no problema original x_i é livre (não tem restrição de sinal)
    $\rightarrow$ substituir por $y_i^+ - y_i^-$, $\quad y_i^+, y_i^- \geq 0$

Variáveis básicas e não básicas

- Considere-se o sistema anterior $Ax = b$, com m equações lineares e n variáveis
- uma *solução básica* é obtida fazendo
  - $n - m$ variáveis iguais a 0
  - resolvendo o sistema para as restantes variáveis, que são chamadas as *variáveis básicas*
- Exemplo: determinar todas as soluções básicas para o sistema

$$\begin{array}{rcrcrcr} x_1 & + & x_2 & & & = & 3 \\ & - & x_2 & + & x_3 & = & -1 \end{array}$$

Soluções admissíveis

- Def: a uma solução básica do problema na forma standard em que todas as variáveis são não-negativas chama-se *solução básica admissível*
- Teorema 1: A região admissível de qualquer problema de programação linear (PL) é um conjunto convexo. Se o PL tem uma solução única, deverá haver um ponto extremo da região admissível que é ótimo.
- Teorema 2: Para qualquer PL, há um único ponto extremo da região admissível correspondendo a cada solução básica admissível. Há pelo menos uma solução básica admissível correspondendo a cada ponto extremo da região admissível
- Def: *soluções básicas admissíveis adjacentes*: para um PL com m restrições, duas soluções básicas dizem-se *adjacentes* se os seus conjuntos de variáveis básicas têm $m - 1$ variáveis em comum.

Definições prévias

- Problema linear na **forma standard**:
  1. todas as restrições são equações
  2. todas as variáveis são não-negativas:
- Sistema de equações na **forma canónica**: cada equação tem uma variável com
  1. coeficiente 1 nessa equação
  2. coeficiente 0 em todas as outras equações
- Forma **linha zero** da função objetivo:

$$z - c_1x_1 - c_2x_2 - \ldots - c_nx_n = 0$$

Algoritmo do simplex: descrição geral

1. Converter o problema à forma *standard*.
2. Determinar uma solução básica admissível (SBA).
3. Verificar se a SBA é ótima; se sim, STOP.
4. Se não, passar para outra SBA, adjacente à anterior mas com um melhor objetivo, utilizando operações algébricas elementares.
5. Começar uma nova iteração (passo 3).

Algoritmo do simplex para problemas de maximização

1. Converter o problema à forma *standard*.
2. Determinar uma solução básica admissível (SBA):
   - todas as restrições $\leq$
   - termos do lado direito todos positivos,
   - então variável de desvio $s_i \longrightarrow$ variável da base para a linha i
   - caso contrário: utilizar outra estratégia.
3. Variáveis não-básicas têm todas coeficientes ≥ 0 na linha 0?
   - se sim, então a solução é ótima.
   - se não: escolher a variável que tem o coeficiente mais negativo para entrar na base (heurística).
4. Passar de uma SBA para outra adjacente mas com um melhor objetivo:

   4.1 Determinar o valor máximo da variável que entra na base tal que todas as variáveis da base se mantenham não negativas.

   4.2 Por o sistema na forma canónica:
      - variável que entra: coeficiente 1 na linha limitante; a variável da base associada a essa linha sai da base
      - eliminar a variável que entra na base de todas as outras linhas.
5. Começar uma nova iteração, a partir do passo 3.

Exemplo

Uma companhia de mobiliário fabrica secretárias, mesas, e cadeiras. O fabrico de cada tipo de móvel requer madeira e dois tipos de trabalho especializado: acabamentos e carpintaria. A quantidade de cada destes recursos necessárias para o fabrico de cada móvel são as seguintes:

Recurso	Secretárias	Mesas	Cadeiras
madeira	8 tábuas	6 tábuas	1 tábuas
acabamentos	4 horas	2 horas	1.5 horas
carpintaria	2 horas	1.5 horas	0.5 horas

Dispõe-se de 48 tábuas, 20 horas de acabamentos, e 8 horas de carpintaria. O preço de venda é de 60 euros para secretárias, 30 euros para mesas, e 20 euros para cadeiras. Admite-se que as vendas de secretárias e de cadeiras são ilimitadas, mas que não se consegue vender mais de 5 mesas.
Como todos os recursos foram já comprados, pretende-se estabelecer o plano de produção que maximiza a receita.

Formulação

$$
\begin{aligned}
\text{maximizar } z = & 60x_1 + 30x_2 + 20x_3 \\
\text{sujeito a } & 8x_1 + 6x_2 + x_3 && \leq 48 \\
& 4x_1 + 2x_2 + 1.5x_3 && \leq 20 \\
& 2x_1 + 1.5x_2 + 0.5x_3 && \leq 8 \\
& x_2 && \leq 5 \\
& x_1, x_2, x_3 && \geq 0
\end{aligned}
$$

Forma standard (introdução de variáveis de desvio):

$$
\begin{array}{rlllllllll}
\text{maximizar } z = & 60x_1 & +30x_2 & +20x_3 & & & & & & \\
\text{sujeito a } & 8x_1 & +6x_2 & +x_3 & +s_1 & & & & = & 48 \\
& 4x_1 & +2x_2 & +1.5x_3 & & +s_2 & & & = & 20 \\
& 2x_1 & +1.5x_2 & +0.5x_3 & & & +s_3 & & = & 8 \\
& & x_2 & & & & & +s_4 & = & 5 \\
& x_1 & ,x_2 & ,x_3, & s_1, & s_2, & s_3, & s_4 & \geq & 0
\end{array}
$$

Resolução pelo algoritmo do simplex

- Por inspeção verifica-se que se $x_1 = x_2 = x_3 = 0$ o sistema fica forma canónica.
- $VNB_1 = \{x_1, x_2, x_3\}$
- $VB_1 = \{z, s_1, s_2, s_3, s_4\}$.
- $x_1 = x_2 = x_3 = 0 \quad \Rightarrow \quad z = 0, s_1 = 48, s_2 = 20, s_3 = 8, s_4 = 5.$

Iteração 1

"Quadro do simplex" correspondente à formulação anterior:

	z	x_1	x_2	x_3	s_1	s_2	s_3	s_4	rhs	vb
*linha*0	1	-60	-30	-20	0	0	0	0	0	$z = 0$
*linha*1	0	8	6	1	1	0	0	0	48	$s_1 = 48$
*linha*2	0	4	2	1.5	0	1	0	0	20	$s_2 = 20$
*linha*3	0	2	1.5	0.5	0	0	1	0	8	$s_3 = 8$
*linha*4	0	0	1	0	0	0	0	1	5	$s_4 = 5$

- rhs = *right hand side*, termo do lado direito (termo independente);
- vb = valor da variável da base associada à linha.

Notas:

1. Linha zero: $z = 0 + 60x_1 + 30x_2 + 20x_3 \Rightarrow$ se se aumentar o valor de x_1, x_2, x_3 aumenta-se z (a solução básica atual não é ótima).
2. Escolhemos a variável com maior coeficiente nessa equação para entrar na base, e passamos à iteração seguinte.

Iteração 2

- Aumentando o valor de x_1 aumentamos o valor do objetivo;
- Mas não podemos aumentar x_1 indefinidamente...
- Para a solução se manter admissível: todas as variáveis ≥ 0.
- Nesta solução, vemos que:

linha 1:	$s_1 = 48 - 8x_1 \longrightarrow x_1 \leq 6$		para manter $s_1 \geq 0$
linha 2:	$s_2 = 20 - 4x_1 \longrightarrow x_1 \leq 5$		para manter $s_2 \geq 0$
linha 3:	$s_3 = 8 - 2x_1 \longrightarrow x_1 \leq 4$		para manter $s_3 \geq 0$
linha 4:	$s_4 = 5$		(independente de x_1)

- x_1 máximo é 4, e a linha limitante é a linha 3 $\Rightarrow$ s_3 sai da base;
- Colocando o sistema na forma canónica:

  $VB_2 = \{z, s_1, s_2, x_1, s_4\}$, $VNB_2 = \{s_3, x_2, x_3\}$

	z	x_1	x_2	x_3	s_1	s_2	s_3	s_4	rhs	vb
*linha*0	1	0	15	-5	0	0	30	0	240	$z = 240$
*linha*1	0	0	0	-1	1	0	-4	0	16	$s_1 = 16$
*linha*2	0	0	-1	0.5	0	1	-2	0	4	$s_2 = 4$
*linha*3	0	1	0.75	0.25	0	0	0.5	0	4	$x_1 = 4$
*linha*4	0	0	1	0	0	0	0	1	5	$s_4 = 5$

Iteração 3

- Aumentando x_3 aumentamos o valor do objetivo; essa variável vai entrar na base.

linha 1:	$s_1 = 16 + x_3$	x_3 não restringido
linha 2:	$s_2 = 4 - 0.5x_3 \rightarrow x_3 \leq 8$	para manter $s_2 \geq 0$
linha 3:	$x_1 = 4 - 0.25x_3 \rightarrow x_3 \leq 16$	para manter $x_1 \geq 0$
linha 4:	$s_4 = 5$	(independente de x_1)

- Linha limitante: linha 2 $\Rightarrow$ s_2 sai da base.
  $VB_3 = \{z, s_1, x_3, x_1, s_4\}$, $VNB_3 = \{s_3, x_2, s_2\}$

	z	x_1	x_2	x_3	s_1	s_2	s_3	s_4	rhs	vb
*linha*0	1	0	5	0	0	10	10	0	280	$z = 280$
*linha*1	0	0	-2	0	1	2	-8	0	24	$s_1 = 24$
*linha*2	0	0	-2	1	0	2	-4	0	8	$x_3 = 8$
*linha*3	0	1	1.25	0	0	-0.5	1.5	0	2	$x_1 = 2$
*linha*4	0	0	1	0	0	0	0	1	5	$s_4 = 5$

- Linha zero: $z = 280 - 5x_2 - 10s_2 - 10s_3$
- Aumentando o valor de qualquer variável que não está na base, o valor de z irá piorar; portanto, a solução atual é ótima.
- Plano de produção ótimo: 2 secretárias, 0 mesas, e 8 cadeiras

Custos reduzidos

- Coeficiente das variáveis de decisão na linha 0 (no quadro ótimo): **custo reduzido**.
- Custo reduzido de uma variável (não básica) no quadro ótimo:
  - quantidade em que o objetivo diminuiria se se aumentasse o valor da variável em uma unidade, em relação à solução ótima *(maximização)*.
- Válido se não implicar alterações no conjunto das variáveis da base (se todas as variáveis básicas continuarem não negativas).
- Neste exemplo: o custo reduzido de x_2 é 5.
  - Se se produzisse uma mesa, em vez da produção ótima de zero, a receita diminuiria em 5 euros.
  - *(Verificar que se $x_2 = 1$ todas as variáveis básicas se mantêm positivas.)*

Variáveis de desvio

- Valor de uma variável de desvio na solução ótima: quantidade de recurso que não é utilizada, na restrição correspondente.
- No exemplo, na solução ótima:
  - todas as horas de carpintaria e acabamentos são utilizadas ($s_1 = s_3 = 0$; as restrições correspondentes são ativas)
  - há 24 tábuas que não são utilizadas ($s_1 = 24$)
  - existe procura para 5 mesas adicionais ($s_4 = 5$)

Apoio na Internet

Para programas interativos com o algoritmo do simplex, ver as páginas:

- `http://www.phpsimplex.com/simplex/simplex.htm?l=en`
- `https://vanderbei.princeton.edu/JAVA/pivot/simple.html`
- `https://www.mathstools.com/section/main/simplex_online_calculator`

Noções estudadas

- Forma standard de um problema linear
- Forma canónica de um problema linear.
- Variáveis de desvio
- Soluções admissíveis
- Variáveis básicas e não básicas
- Algoritmo do simplex para programação linear
- Custos reduzidos

Próxima aula

- Algoritmo do simplex: método do *Big M*
- Análise de sensibilidade: visualização em problemas com duas variáveis.