

Programação Linear Inteira (Mista)

12 de agosto de 2019

Programação Linear Inteira (Mista)

Nos exemplos vistos até agora, nossos modelos foram todos de Programação Linear. As variáveis poderiam assumir valores reais, desde que satisfizessem as restrições impostas.

Em alguns casos, precisamos impor que as variáveis assumam apenas valores inteiros. Neste caso temos um problema de Programação Linear Inteira. Se apenas algumas variáveis precisarem ser inteiras, temos um problema de Programação Linear Inteira Mista.

Na primeira aula, vimos o seguinte exemplo de Problema da Mistura para montar uma ração para aves.

Exemplo: Problema da mistura - Ração

- Queremos saber quais as quantidades ideais de cada ingrediente para elaborar uma ração.
- As necessidades nutricionais devem ser atendidas e o custo total dos ingredientes deve ser o menor possível.
- Temos os ingredientes e seus custos.
- Para fazer uma ração para aves, é necessário uma certa quantidade nutrientes: vitamina A (V_A), vitamina B (V_B) e proteína (P).

Exemplo: Problema da mistura - Ração

- Deseja-se preparar uma ração que contenha no mínimo 7 unidades de V_A , 9 unidades de V_B e 1 unidade de P .

Nutrientes	Ingredientes		Qtde Mínima
	Milho (M)	F. Osso (F)	
Vitamina A (V_A)	2	3	7
Vitamina B (V_B)	3	2	9
Proteína (P)	1	0	1
Custos (R\$/kg)	65	30	

- Como misturar (as quantidades) os ingredientes de modo a atender as necessidades nutricionais e produzir uma ração de menor custo possível?

Exemplo: Problema da mistura - O que decidir?

Precisamos decidir as quantidades dos ingredientes presentes na mistura.

Assim, definimos as seguintes variáveis de decisão:

- x_M = quantidade de milho adicionado a mistura (kg).
- x_F = quantidade de farinha de osso adicionado a mistura (kg).

Exemplo: Modelagem

O modelo matemático pode ser dado por

$$\text{minimizar } f(x_M, x_F) = 65x_M + 30x_F$$

$$\text{sujeita a } 2x_M + 3x_F \geq 7,$$

$$3x_M + 2x_F \geq 9,$$

$$1x_M + 0x_F \geq 1,$$

$$x_M \geq 0, x_F \geq 0.$$

Exemplo: Programação Linear Inteira

Suponha agora que os ingredientes pudessem ser comprados e usados apenas em sacos, não mais em quilos.

Neste caso, as variáveis deveriam assumir apenas valores inteiros.

Exemplo: Programação Linear Inteira

O modelo passa a ser:

$$\text{minimizar } f(x_M, x_F) = 65x_M + 30x_F$$

$$\text{sujeita a } 2x_M + 3x_F \geq 7,$$

$$3x_M + 2x_F \geq 9,$$

$$1x_M + 0x_F \geq 1,$$

$$x_M \geq 0, x_F \geq 0.$$

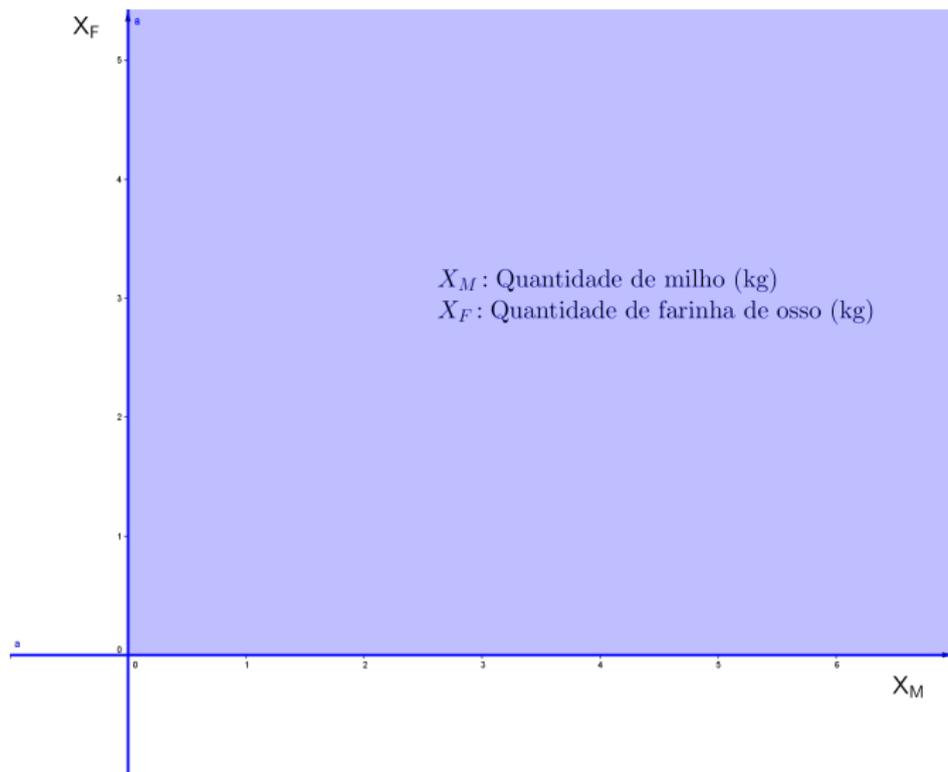
$$x_M, x_F \in \mathbb{Z}.$$

Exemplo: Programação Linear Inteira

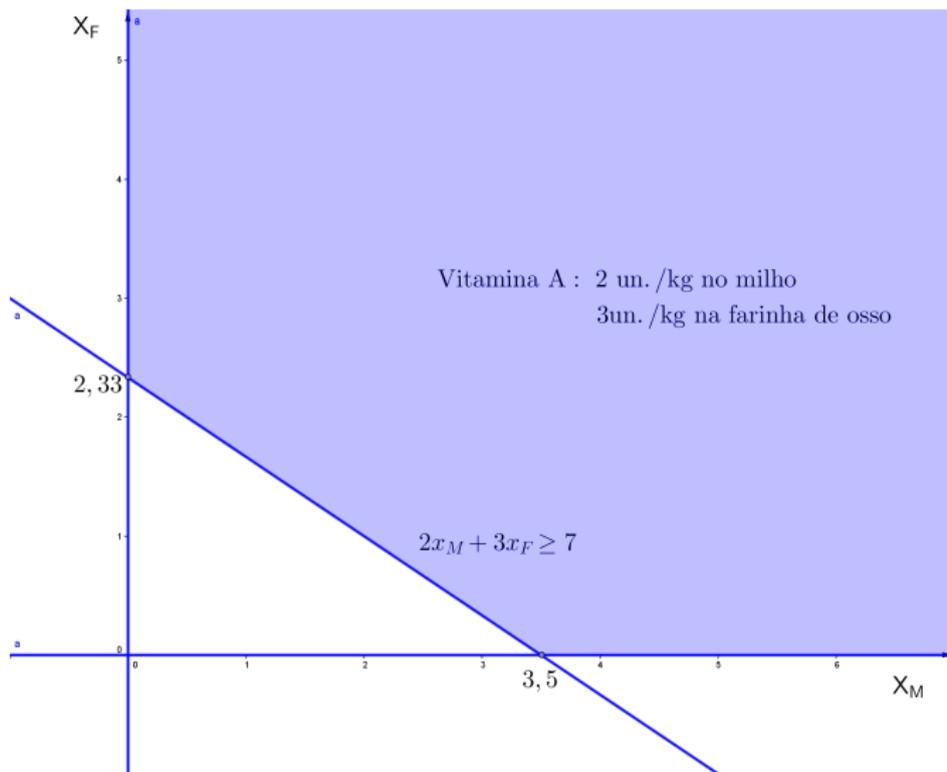
Neste caso particular, será que esta restrição adicional é mesmo necessária?

Vamos ver o que aconteceu quando buscamos a solução graficamente para o problema da primeira aula, que não tinha a restrição de que as variáveis x_M e x_F assumissem apenas valores inteiros.

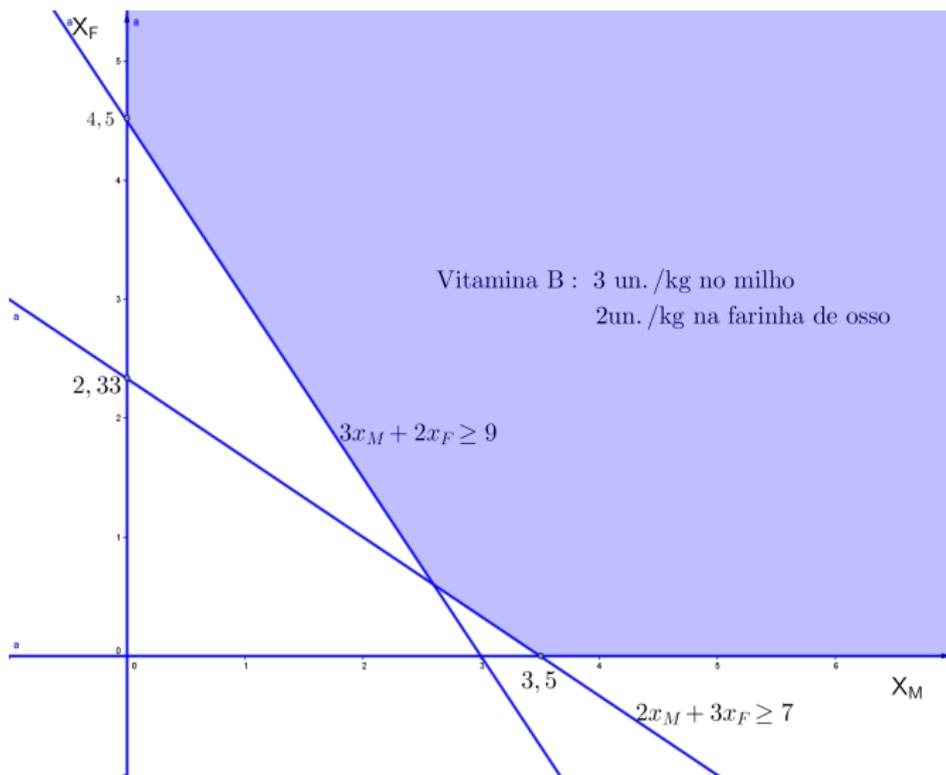
Exemplo: Problema da mistura - Ração



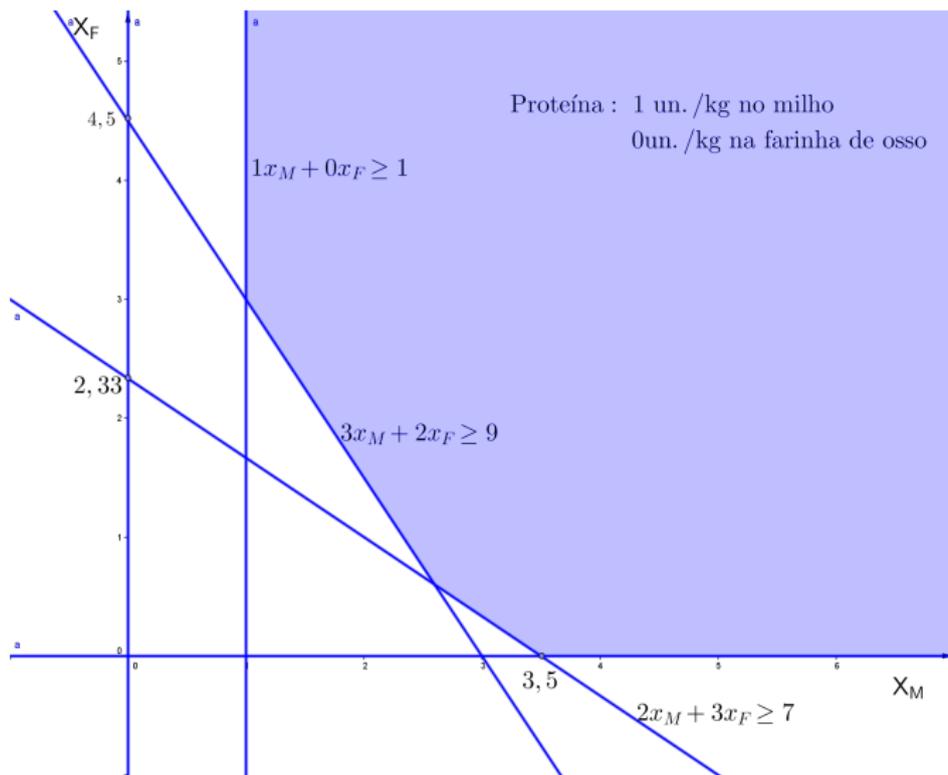
Exemplo: Problema da mistura - Ração



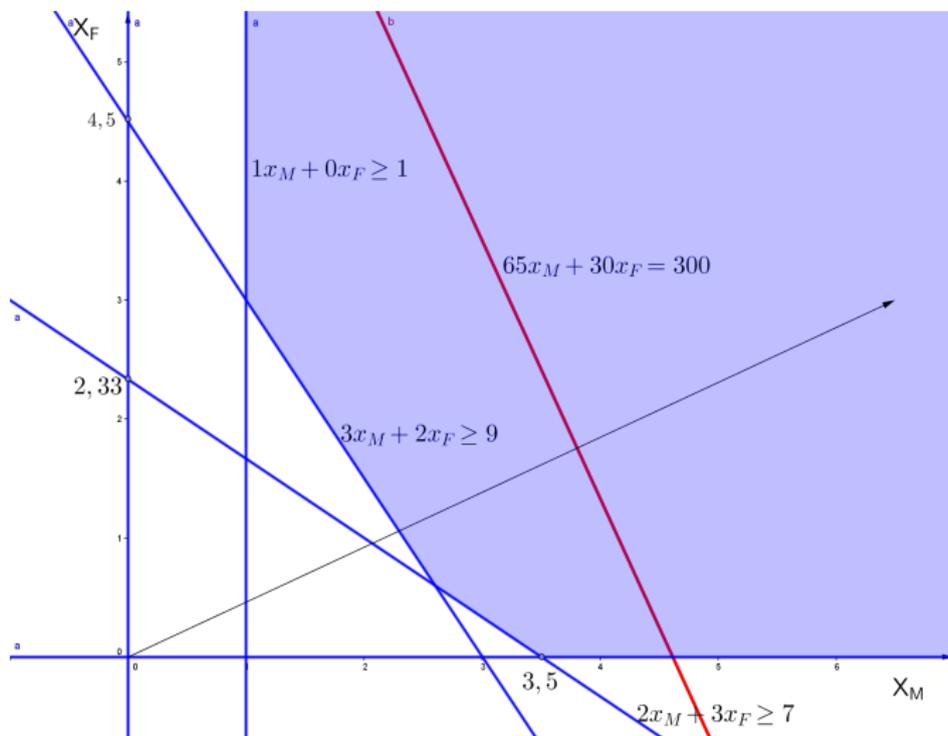
Exemplo: Problema da mistura - Ração



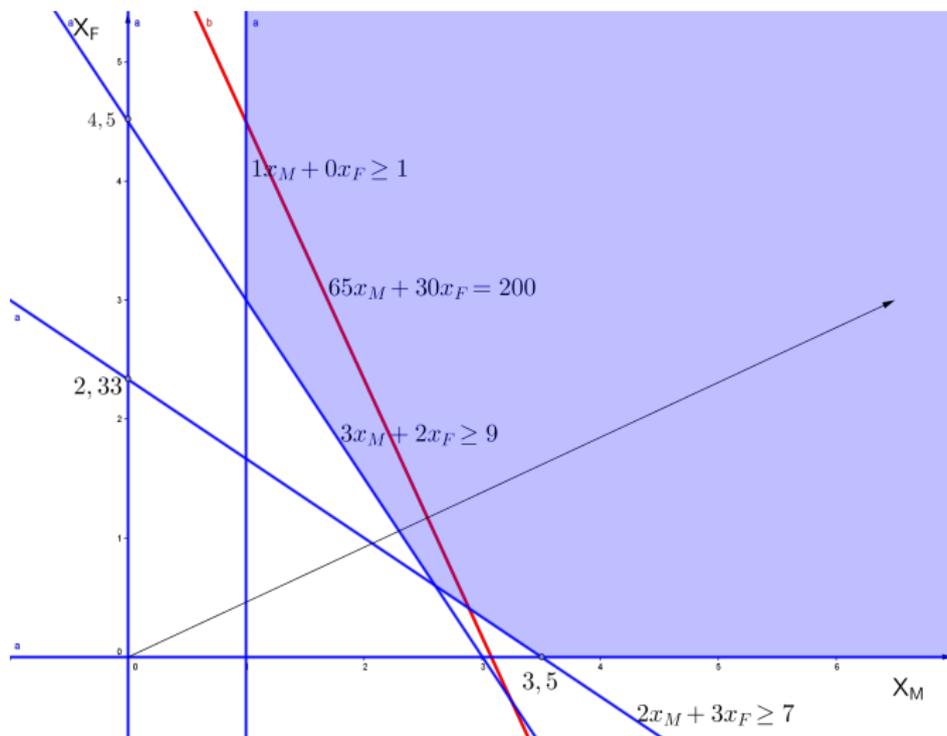
Exemplo: Problema da mistura - Ração



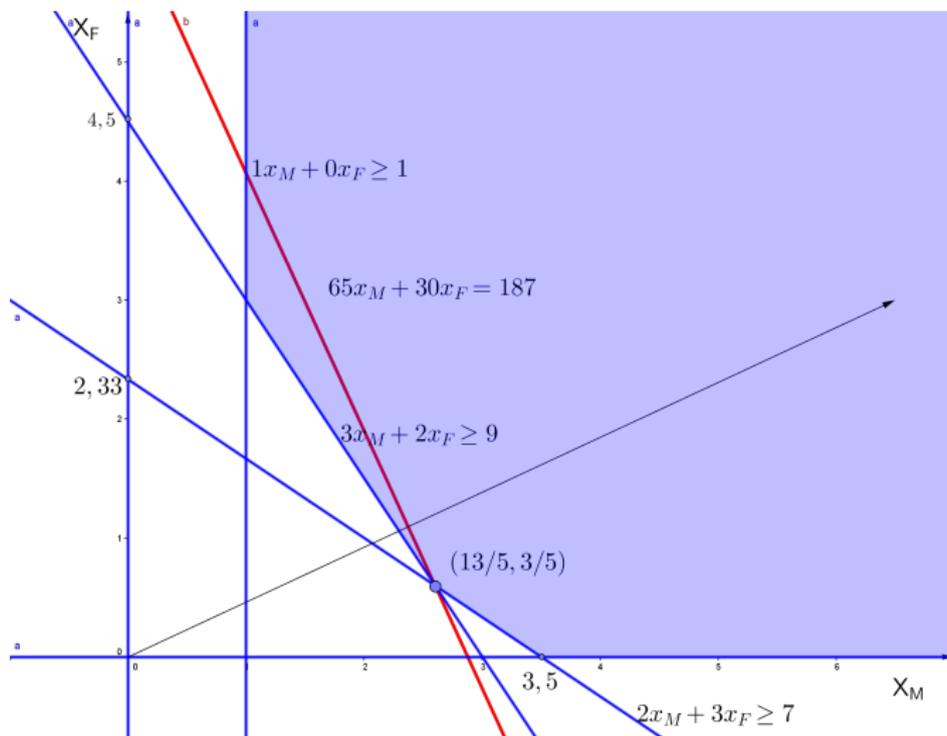
Exemplo: Problema da mistura - Ração



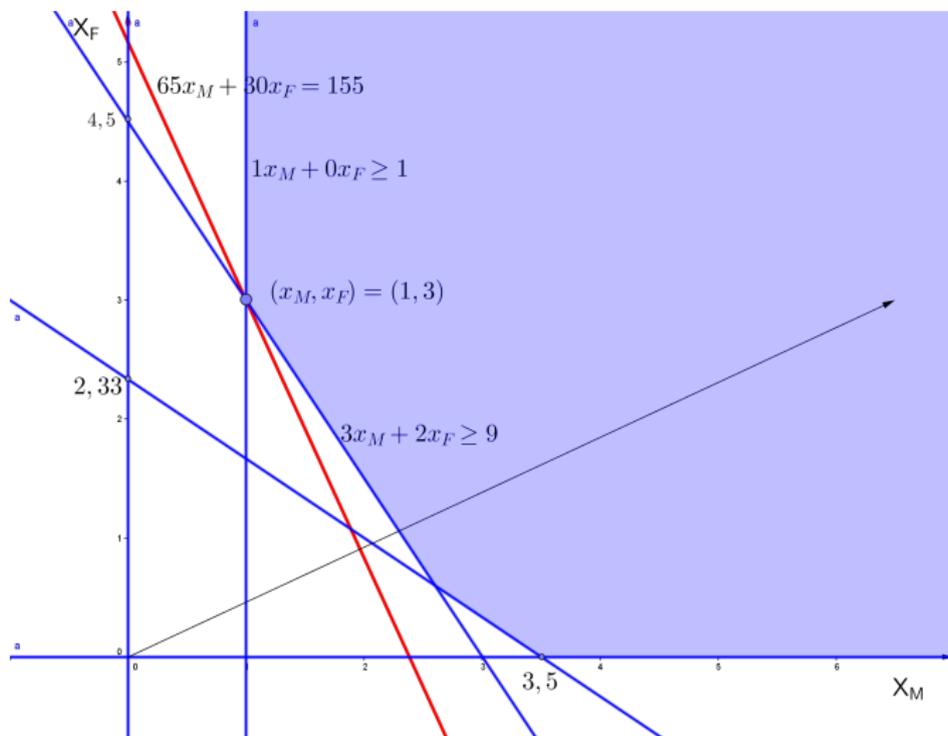
Exemplo: Problema da mistura - Ração



Problema da mistura - Ração

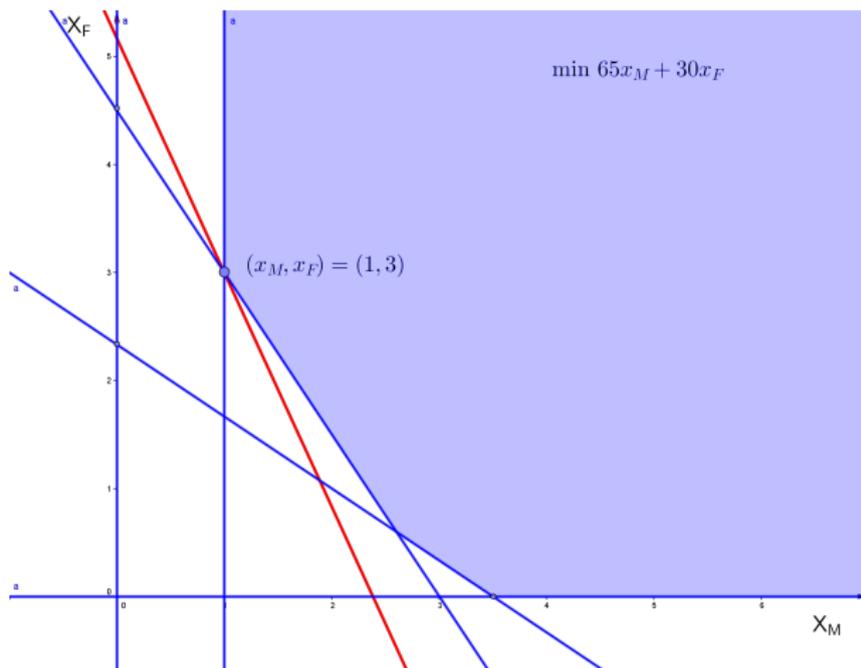


Exemplo: Problema da mistura - Ração



Exemplo: Problema da mistura - Ração

O custo mínimo da mistura é dado por $f^* = 155$. Para atingir este valor, devemos misturar 1kg de milho e 3kg de farinha de osso.



Exemplo: Programação Linear Inteira

Neste caso particular, como a solução do problema original já tem valores inteiros para as variáveis, não precisamos impor a restrição de que as variáveis x_F e x_M sejam inteiras.

Claro que este nem sempre é o caso.

Pense no que aconteceria se a função objetivo neste exemplo fosse outra.

Em alguns casos, como vimos, precisamos impor que algumas variáveis do modelo sejam inteiras pela própria natureza do problema que está sendo modelado.

Mas, em alguns casos, podemos usar variáveis inteiras como uma ferramenta de modelagem.

A seguir, veremos alguns exemplos do uso de variáveis binárias (que podem assumir somente valores 0 ou 1) para representar algumas situações importantes.

As variáveis binárias são muito usadas para representar eventos que podem ou não ocorrer.

Nestes casos, costumamos usar uma variável x que assume os seguintes valores:

$$x = \begin{cases} 1, & \text{se o evento ocorre,} \\ 0, & \text{caso contrário.} \end{cases}$$

Exemplo: Problema da Mochila

Suponha que você tenha uma mochila, com capacidade de peso c , e um conjunto de itens $1, 2, 3, \dots, n$, cada um com um peso p_i e um valor v_i . Você deseja colocar na mochila o maior valor em itens, de forma que a capacidade seja respeitada.

Para modelar este problema, vamos definir as variáveis

$$x_i = \begin{cases} 1, & \text{se o item } i \text{ vai para a mochila,} \\ 0, & \text{caso contrário,} \end{cases}$$

para $i = 1, \dots, n$.

Exemplo: Problema da Mochila

O modelo fica:

$$\text{maximizar } v_1x_1 + v_2x_2 + \dots + v_nx_n$$

$$\text{sujeita a } p_1x_1 + p_2x_2 + \dots + p_nx_n \leq c,$$

$$x_1 \in \{0, 1\},$$

$$\vdots$$

$$x_n \in \{0, 1\}.$$

Representação de condições “se-então”

Uma forma de modelar uma condição do tipo “se-então” é usando uma variável binária. Um caso muito frequente de uso deste tipo de condição é quando temos um custo fixo.

Por exemplo,

- se temos um custo para ligar uma máquina e, depois de ligada, o custo de produção de um produto usando a máquina é proporcional à quantidade produzida;
- se um salário de vendedor é fixo e, a cada venda, há um acréscimo proporcional ao valor vendido;
- se vamos modelar o preço de uma corrida como a de táxi, que tem um custo inicial fixo e é acrescentado um custo proporcional à quilometragem andada.

Representação de condições “se-então”

Essas funções de custo fixo podem ser definidas da seguinte forma:

$$K(x) = \begin{cases} s + cx, & \text{se } x > 0, \\ 0, & \text{se } x = 0, \end{cases}$$

para uma constante s e uma constante c .

Note que esta função não é linear, mas ela é muito parecida com uma.

Representação de condições “se-então”

Podemos usar uma variável binária para representar esta função de custo fixo.

Definimos uma variável y tal que $y = 1$ se $x > 0$ e $y = 0$ caso contrário (ou seja, se $x = 0$).

Desta forma, a função passa a ser

$$K(x) = sy + cx.$$

Note que, como estamos tratando de custo, $s > 0$.

Representação de condições “se-então”

Agora precisamos impor uma restrição que garanta que sempre que $x > 0$, y irá assumir o valor 1; e sempre que $x = 0$, y irá assumir o valor 0.

Note que, usando a restrição

$$x \leq My,$$

para um valor de M muito grande, temos que sempre que $x > 0$, y deve ser 1.

Quando $x = 0$, y poderia ter valor ou 0 ou 1. Mas, como estamos minimizando o custo na função objetivo e $s > 0$, isso faz com que, na solução, $y = 0$.

Representação de condições “se-então”

Outro exemplo de condição do tipo “se-então” é o seguinte.

Suponha que 4 itens (1,2,3 e 4) podem ser produzidos em uma máquina denotada por k . Mas, se o item 1 é produzido em k , então os itens 2, 3 e 4 não podem ser produzidos nesta máquina.

Para modelar este caso, podemos definir as variáveis binárias

$$x_{ik} = \begin{cases} 1, & \text{se o item } i \text{ é produzido na máquina } k, \\ 0, & \text{caso contrário.} \end{cases}$$

Representação de condições “se-então”

Neste caso, precisamos garantir que se $x_{1k} = 1$, então $x_{2k} = x_{3k} = x_{4k} = 0$.

Definimos então a variável binária

$$y = \begin{cases} 1, & \text{se o item 2, 3 ou 4 foi produzido na máquina } k, \\ 0, & \text{caso contrário.} \end{cases}$$

Definimos, então, as seguintes restrições:

$$x_{2k} + x_{3k} + x_{4k} \leq 3y,$$

$$x_{1k} \leq 3(1 - y).$$

Quando colocamos restrições em um modelo, queremos que todas sejam satisfeitas. No entanto, em alguns casos, queremos que ou uma ou outra restrição seja satisfeita (e não as duas ao mesmo tempo).

Para representar este tipo de situação, também podemos usar uma variável binária.

Digamos que temos 2 restrições

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \leq b_1, \quad (1)$$

$$a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n \leq b_2, \quad (2)$$

e queremos que exatamente uma delas seja satisfeita.

Representação de restrições disjuntivas

Vamos definir uma variável y tal que

$$y = \begin{cases} 1, & \text{se a restrição (1) é satisfeita,} \\ 0, & \text{caso contrário.} \end{cases}$$

Trocando as restrições (1) e (2) por

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \leq b_1 + M(1 - y),$$

$$a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n \leq b_2 + My,$$

para M suficientemente grande, temos que uma, e somente uma, das restrições é satisfeita.

Note que, com as ferramentas apresentadas, podemos representar diversas relações lógicas.

Considere o seguinte exemplo: suponha que existam 5 investimentos financeiros e defina as variáveis binárias x_i como

$$x_i = \begin{cases} 1, & \text{se o investimento } i \text{ é selecionado,} \\ 0, & \text{caso contrário.} \end{cases}$$

- Se queremos que no máximo 3 investimentos sejam selecionados, usamos a restrição

$$x_1 + x_2 + x_3 + x_4 + x_5 \leq 3.$$

- Se queremos que ou o investimento 1 ou o investimento 2 seja selecionado, usamos

$$x_1 + x_2 \leq 1.$$

Representação de outras relações lógicas

- Se queremos que, se o investimento 1 é selecionado, o investimento 2 também seja, usamos

$$x_1 \leq x_2.$$

- Se queremos que, se o investimento 2, 3 ou 4 é selecionado, o investimento 1 também seja, usamos

$$x_2 + x_3 + x_4 \leq 3x_1.$$

Ou, alternativamente,

$$x_2 \leq x_1, \quad x_3 \leq x_1, \quad x_4 \leq x_1.$$

Representação de valores discretos

Em alguns casos, queremos que as variáveis assumam um conjunto específico de valores.

Por exemplo, suponha que uma variável x possa assumir somente os valores $\{2; 3; 7; -3; 2, 5\}$.

Uma maneira de representar isso em um modelo é definir variáveis binárias y_1, y_2, y_3, y_4 e y_5 e impor as restrições

$$x = 2y_1 + 3y_2 + 7y_3 - 3y_4 + 2.5y_5,$$

$$y_1 + y_2 + y_3 + y_4 + y_5 = 1.$$