Modelagem de Problemas de Otimização

Alexandre Salles da Cunha

UNIVERSIDADE FEDERAL DE MINAS GERAIS

UFMG - ICEx
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

Algumas referências bibliográficas

1. W.E. Hart, J-P Watson, D.L. Woodruff. Pyomo: modeling and solving mathematical programs in Python, *Mathematical Programming Computation*, 3:219–260, 2011.
2. W.E. Hart, C.D. Laird, J-P Watson e outros. *Pyomo - Optimization Modeling in Python*. 2a. Edição, Springer, 2012.
3. On-line Pyomo Project: `http://www.pyomo.org/`
4. H. P. Williams. *Model Building in Mathematical Programming*, 4a. Edição, John Wiley & Sons, 1999.

Objetivos

1. Apresentar a abordagem de *Otimização*.
2. Discutir como formular um problema *gerencial* ou *de decisão* como um problema de Otimização.
3. Introduzir o `Pyomo`, como ferramenta de modelagem de problemas de Otimização.

Pyomo

Acrônimo para Python Optimization Modeling Objects. `Pyomo` é uma AML (Algebraic Modeling Language) que estende `Python` para incluir objetos de modelagem de problemas de otimização.

- Os modelos de otimização são modelos abstratos que descrevem as relações entre os elementos de um sistema, tais como desigualdades, equações, dependências lógicas, etc.
- Os modelos são essencialmente uma representação (possivelmente simplificada) de como os elementos se relacionam. Devem ser independentes dos dados observados, no sentido de que sua formulação não deve ser instanciada. Separação entre instância e problema.
- Ao formalizar a natureza lógica destas relações por meio de modelos de otimização e ao resolver estes modelos para conjuntos de dados específicos, podemos auxiliar a tomada de decisões gerenciais mais adequadas.

Processo Integrado de Decisão

1. Compreensão do funcionamento dos elementos do sistema.
2. Às vezes, isso pode ocorrer com o auxílio da disciplina de Ciência dos Dados, por meio de *insights* sobre as interdependências dos dados produzidos pela operação de um sistema existente.
3. Melhor compreensão do problema gerencial que precisa ser resolvido.
4. Modelagem matemática do sistema como um problema de otimização, representando as interdependências entre elementos do sistema (ou dos dados coletados), e como estes afetam o objetivo gerencial.
5. Resolução do Modelo (via pacote de otimização).
6. Análise de resultados e iteração do processo.

Dados do problema

- Conjunto de pontos de oferta de um produto: N
- Conjunto de pontos de demanda do produto: M
- Disponibilidade do produto em $n \in N : o_n$
- Demanda do produto em $m \in M : r_m$
- Distância entre $n \in N$ e $m \in M$: d_{nm}
- Custo de transporte (por km): ct

O que se deseja ?

Enviar o produto dos pontos de oferta para os pontos de demanda, satisfazendo as demandas estabelecidas, ao mínimo custo de transporte.

1. Preparação: entendimento do problema gerencial, tratamento dos dados.
2. Modelagem: tradução do problema gerencial no problema de Otimização, segundo a premissa de dissociação da instância do problema e do artefato abstrato que é o modelo.
3. Codificação: O modelo matemático deve ser representado computacionalmente (escolha de um ambiente de modelagem, `Pyomo`+`Python`, por exemplo).
4. Instanciação e resolução do problema de otimização: escolha do algoritmo/ferramenta computacional capaz de resolver o problema.
5. Eventualmente pode ser necessário o desenvolvimento de um algoritmo específico para a resolução do problema.
6. Análise dos resultados.

Produto a ser obtido

Construção de um objeto, Modelo do problema de Otimização, composto por:

1. Parâmetros (custos fixos, variáveis, traduzem relações identificadas no problema)
2. Variáveis de decisão (aquilo que se deseja decidir)
3. A função objetivo, que deve ser maximizada ou minimizada.
4. As restrições a serem respeitadas pelas variáveis escolhidas.

Metodologia

Com base na descrição do problema gerencial:

1. Identificamos os parâmetros do problema.
2. Identificamos aquilo que se deseja decidir. A cada um destes atributos a decidir, associamos uma variável de decisão. As variáveis devem ser capazes de traduzir as restrições operacionais e o objetivo a ser alcançado.
3. Formulamos a função objetivo e as restrições, a partir da escolha de variáveis.
4. Identificamos o tipo de algoritmo apropriado à resolução do modelo obtido.
5. Revisamos o modelo: pode ser necessário repetir o processo para melhorar o modelo, ou para se obter um modelo com características diferentes (linear, por exemplo).

Decisões a serem tomadas: quanto enviar de cada ponto de oferta a cada ponto de demanda.

Variáveis de decisão escolhidas

Para cada par (oferta,demanda), $(n, m) \in N \times M$, definimos uma variável x_{nm} que representa a quantidade do produto que é enviada de n para m.

Atendimento da demanda A soma das quantidades enviadas de todos pontos de oferta para cada ponto de demanda deve pelo menos igualar a demanda daquele ponto.

$$\sum_{n \in N} x_{nm} \geq r_m, \quad m \in M$$

Capacidade de oferta Para todo ponto de oferta, a soma das quantidades que envia para os pontos de demanda não pode exceder sua disponibilidade/capacidade.

$$\sum_{m \in M} x_{nm} \leq o_n, \quad n \in N$$

Não negatividade das variáveis de decisão

$$x_{nm} \geq 0, \quad n \in N, m \in M$$

Custo Total de Transporte dever ser mínimo

Definindo $c_{nm} = (ct)d_{nm}$ como o custo de enviar uma unidade de mercadoria de n para m, a contribuição da variável x_{nm} para o custo é $c_{nm}x_{nm}$.
Logo, o custo total, a ser minimizado é:

$$\sum_{n \in N} \sum_{m \in M} c_{nm}x_{nm}$$

Problema de Programação Linear

$$\min \sum_{n\in N} \sum_{m\in M} c_{nm} x_{nm}$$
$$\sum_{n\in N} x_{nm} \geq r_m, \quad m \in M$$
$$\sum_{m\in M} x_{nm} \leq o_n, \quad n \in N$$
$$x_{nm} \geq 0, \quad n \in N, m \in M$$

- Algoritmos de solução: Especialização do Método Simplex, Algoritmo do Transporte.
- Problema de Programação Linear é polinomialmente solúvel.
- Outras classes de problemas de otimização podem ser bastante mais difícies de serem resolvidos.

`https://pyomo.readthedocs.io/en/`

Suporta a modelagem de problemas de otimização e análise de soluções em ambiente Python. O conjunto de funcionalidades permite:

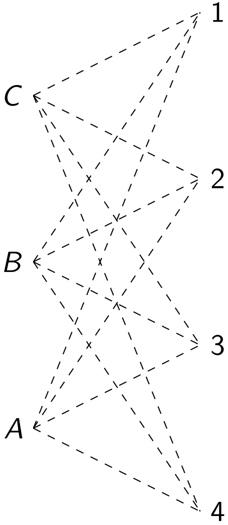
1. Criar o modelo: classes `ConcreteModel()` ou `AbstractModel()`
2. Instanciação: antes de criar o modelo (no caso Concreto) ou ao exportar o modelo (no caso Abstrato)
3. Exportar o modelo instanciado para um algoritmo de otimização definido pelo usuário
4. Comandar a sua resolução
5. Recuperar e analisar a solução obtida.

O `Pyomo` não incopora os algoritmos de otimização propriamente.

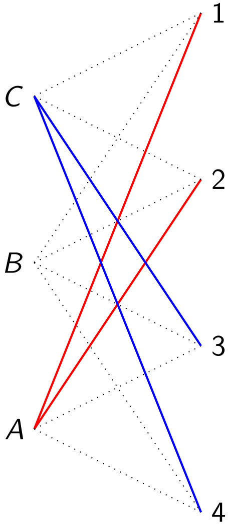
Versão não capacitada

- Uma empresa precisa decidir onde instalar armazéns de forma a atender seus clientes.
- Após levantamento da área comercial, foi identificado um conjunto N de possíveis localizações para estes potenciais armazéns.
- Os clientes da empresa foram geograficamente agrupados e representados por um conjunto de clientes agregados M.
- Caso o armazém $i \in N$ seja aberto, um custo fixo f_i é incorrido. O custo de atender 100% da demanda do cliente $j \in M$ pelo armazém $i \in N$ é d_{ij}.
- Deseja-se definir onde instalar os armazéns e como atender cada cliente, considerando os armazéns que foram abertos.

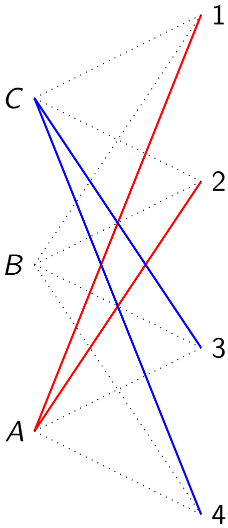
Grafo bipartido

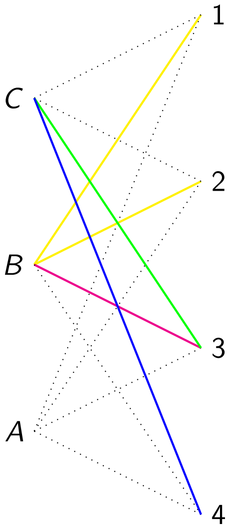


Possível solução ao custo $f_C + f_A + d_{C,3} + d_{C,4} + d_{A,1} + d_{A,2}$.



Exemplo: Há múltiplas formas de garantir atendimento dos clientes





Decisões a serem tomadas: quais armazéns abrir e, a partir destes, como atender os clientes.

Quais armazéns abrir?

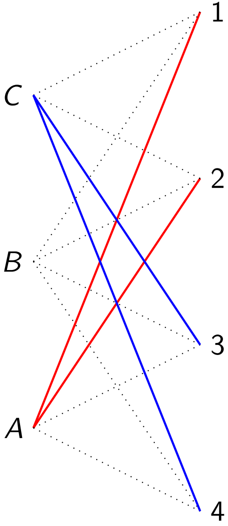
Para cada armazém $i \in N$, definimos uma variável $y_i \in \{0, 1\}$. No nosso modelo, a variável y_i assumirá valor 1 caso o armazém i seja aberto. Caso contrário, $y_i = 0$.

Como atender a demanda dos clientes ?

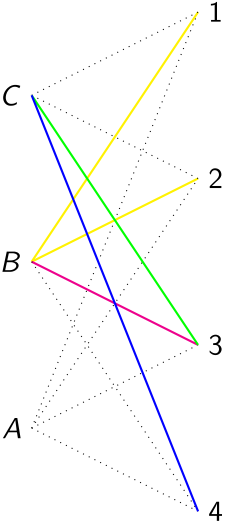
Para cada par (armazém,cliente), (i, j), $i \in N, j \in M$, definimos uma variável x_{ij} que representa o percentual da demanda do cliente j que será atendida pelo armazém i, caso este seja aberto.

Exemplo: Atribuição de variáveis em soluções viáveis

$y_C = y_A = x_{C,3} = x_{C,4} = x_{A,1} = x_{A,2} = 1.$



$y_C = y_B = 1,$
$x_{B,1} = x_{B,2} = x_{C,4} = 1,$
$x_{B,3} = x_{C,3} = 0.5$



Condições a serem respeitadas:

- Aspecto 1: Armazém fechado não atende cliente algum.
- Aspecto 2: A demanda total do cliente deve ser atendida.
- Aspecto 3: Natureza das variáveis de decisão (discretas, faixa de valores possíveis).

Aspecto 1

$$x_{ij} \leq y_i, \quad i \in N, j \in M$$

Aspecto 2

$$\sum_{i \in N} x_{ij} = 1, \quad j \in M$$

Aspecto 3

$$x_{ij} \geq 0, \quad i \in N, j \in M$$
$$y_i \in \{0, 1\}, \quad i \in N.$$

Custo total da operação:

1. Custo fixo, depende de quais armazéns foram abertos: f_i para os $i \in N$ abertos, isto é, tais que $y_i = 1$.
2. Custo variável, depende da demanda e de quais armazéns são empregados para atender os clientes: para algum armazém $i \in N$ aberto, e algum cliente $j \in M$, $d_{ij}x_{ij}$.

Senso da otimização

Custo total deve ser minimizado.

Função Objetivo

$$\min \quad \sum_{i \in N} f_i y_i + \sum_{i \in N} \sum_{j \in M} d_{ij} x_{ij}$$

$$
\begin{aligned}
\min \quad & \sum_{i \in N} f_i y_i + \sum_{i \in N} \sum_{j \in M} d_{ij} x_{ij} \\
& x_{ij} \leq y_i && i \in N, j \in M \\
& \sum_{i \in N} x_{ij} = 1 && j \in M \\
& x_{ij} \geq 0 && i \in N, j \in M \\
& y_i \in \{0, 1\} && i \in N
\end{aligned}
$$

1. Linear: a contribuição de cada variável para a função objetivo e para as restrições é um termo que varia linearmente com a variável.
2. Inteiro-misto: há variáveis ($y_i \in \{0, 1\}$) que só podem assumir valores discretos e variáveis ($x_{ij} \in [0, 1]$) que podem assumir valores reais, em um intervalo contínuo de valores.
3. Algoritmos de solução: Branch-and-bound, Decomposição de Benders,...

Alguns outros tipos de modelos de problemas de otimização

1. Puramente lineares.
2. Não lineares.
3. Binários quadráticos.
4. Não lineares inteiros, inteiros-mistos.
5. ...

O tipo de modelo define o tipo de algoritmo a ser empregado. No caso do modelo que obtivemos (linear+variáveis discretas) para o problema de localização de armazéns, pode-se empregar alguma variante de algoritmo *Branch-and-bound*.

Dados do problema

- Conjunto de pontos de oferta de um produto: N
- Conjunto de pontos de demanda do produto: M
- Disponibilidade do produto em $n \in N : o_n$
- Demanda do produto em $m \in M : r_m$
- Grafo direcionado: (V, A)
  - Vértices: $V = N \cup M$
  - Arcos: $A \subseteq V \times V$ (sem loop)
- Matriz simétrica de distâncias: $\{d_{ij} : i, j \in V\}$
- Custo de transporte (por $km.m^3$): ct
- Custo de construção (por km construído): cc

- Diferentemente do Problema do Transporte, neste problema a rede de estradas não é disponível. Precisa ser construída.
- Custos: fixos (construção da estrada) e variáveis (fluxo do produto nos trechos construídos).
- Duas decisões que precisam ser tomadas de forma integrada: quais arcos construir e como enviar o produto dos pontos de oferta e demanda.
- Necessidade de uso de variáveis binárias que indicam se um arco $(i, j) \in A$ será construído ou não.

Variáveis de decisão

- $\mathbf{x} = \{x_{ij} \in \{0,1\} : (i,j) \in A\}$.
  A variável binária x_{ij} deve valer 1 caso o arco (i,j) seja construído. Caso contrário, $x_{ij} = 0$.
- $\mathbf{f} = \{f_{ij} \geq 0 : (i,j) \in A\}$.
  A variável de fluxo f_{ij} indica quanto do produto é transportado pelo arco $(i,j) \in A$.

Restrições de balanço de fluxo

Indicam conservação de massa em cada vértice do grafo: soma dos fluxos de entrada é igual à soma dos fluxos de saída.

$$\sum_{j \in V | j \neq i} f_{ji} + o_i = \sum_{j \in V | j \neq i} f_{ij}, \ \ i \in N$$

$$\sum_{j \in V | j \neq i} f_{ji} = \sum_{j \in V | j \neq i} f_{ij} + r_i, \ \ i \in M$$

Acoplamento entre variáveis de fluxo **f** e topológicas **x**

Condições lógicas a representar:

- $f_{ij} > 0 \rightarrow x_{ij} = 1$.
- $x_{ij} = 0 \rightarrow f_{ij} = 0$.

Definimos um valor M suficientemente grande ($M = \sum_{i \in N} o_i$).

$$f_{ij} \leq M x_{ij} \text{ para todo } (i,j) \in A$$

Função objetivo

Soma de duas parcelas:

- Custo de construção dos arcos: $\sum_{(i,j)\in A}(cc)d_{ij}x_{ij}$
- Custo de movimentação do produto: $\sum_{(i,j)\in A}(ct)d_{ij}f_{ij}$

$$\min \sum_{(i,j)\in A}(cc)d_{ij}x_{ij} + \sum_{(i,j)\in A}(ct)d_{ij}f_{ij}$$

Modelo de Programação Inteira Mista

$$\min \sum_{(i,j)\in A} (cc)d_{ij}x_{ij} + \sum_{(i,j)\in A} (ct)d_{ij}f_{ij}$$

$$\sum_{j\in V|j\neq i} f_{ji} + o_i = \sum_{j\in V|j\neq i} f_{ij}, \;\; i \in N$$

$$\sum_{j\in V|j\neq i} f_{ji} = \sum_{j\in V|j\neq i} f_{ij} + r_i, \;\; i \in M$$

$$f_{ij} \leq Mx_{ij} : (i,j) \in A$$

$$f_{ij} \geq 0 : (i,j) \in A$$

$$x_{ij} \in \mathbb{B} : (i,j) \in A$$

Tabela: Distâncias entre pares de localidades (em km). As distâncias são simétricas.

	O1	O2	O3	O4	D1	D2	D3	D4	D5	D6	D7
O1	-	150	150	132	143	159	126	166	106	120	181
O2	-	-	123	84	123	124	92	147	120	108	110
O3	-	-	-	108	101	113	100	138	75	115	106
O4	-	-	-	-	93	123	82	97	68	72	90
D1	-	-	-	-	-	115	107	123	102	95	110
D2	-	-	-	-	-	-	132	146	112	93	119
D3	-	-	-	-	-	-	-	113	85	105	99
D4	-	-	-	-	-	-	-	-	104	108	132
D5	-	-	-	-	-	-	-	-	-	105	104
D6	-	-	-	-	-	-	-	-	-	-	113
D7	-	-	-	-	-	-	-	-	-	-	-

- $ct = R\$0.2$ por km por m^3
- $cc = R\$7.50$ por km.

Tabela: Oferta (em m^3)

O1	O2	O3	O4	Total
660	301	271	99	1331

Tabela: Demanda (em m^3).

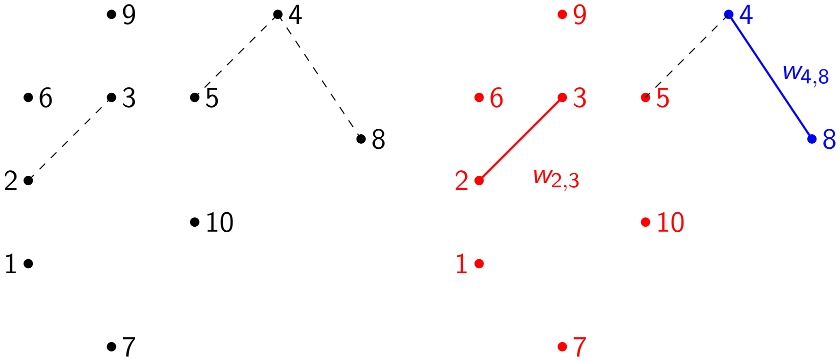
D1	D2	D3	D4	D5	D6	D7	Total
247	394	265	105	90	85	145	1331

Problema gerencial

Uma empresa deseja identificar dois grupos de funcionários de forma que cada grupo reuna funcionários que sejam o mais harmônicos possível. Cada um dos dois conjuntos de funcionários deve ter aproximadamente o mesmo tamanho e todo funcionário deve ser atribuído a algum conjunto. Os funcionários são representados pelo conjunto V. Para cada par $i, j \in V$ de funcionários distintos, há um valor (ou peso) $w_{ij} \in [-1, 1]$ que indica o quão profissionalmente harmônicos i e j são. Valores de w_{ij} próxiimos a 1 indicam que i e j são mais harmônicos e próximos de -1 indicam que não são harmônicos.

Como podemos formular o problema de agrupamento como um problema de otimização ?

Ilustração de um agrupamento não equilibrado



O que precisamos definir ?

Quais funcionários são atribuídos ao mesmo conjunto, sendo dois o número de conjuntos que se deseja encontrar.

Seja $C \subset V$ um dos conjuntos de funcionários e $\overline{C} = V \setminus C$, seu complemento em C.

- $y_i \in \{0, 1\}$ para todo funcionário $i \in V$.
- Se $i \in C$, $y_i = 1$. Caso contrário, $i \in \overline{C}$, $y_i = 0$.

O agrupamento deve atender qual objetivo ?

Os conjuntos devem ser o mais harmônicos possível.

Como traduzir matematicamente este conceito ?

Dado o par $C, \overline{C}$, definimos a qualidade do agrupamento pela função que mede a soma dos pesos dos pares de funcionários que estão no mesmo conjunto do agrupamento:

$$q(C, \overline{C}) = \sum_{i,j \in C, j>i} w_{ij} + \sum_{i,j \in \overline{C}, j>i} w_{ij}$$

Problema de agrupamento: Função objetivo

$$\max \sum_{i \in V} \sum_{j > i} w_{ij}(y_i y_j + (1 - y_i)(1 - y_j))$$

y_i	y_j	$w_{ij} y_i y_j$	$w_{ij}(1 - y_i)(1 - y_j)$
1	1	w_{ij}	0
0	0	0	w_{ij}
0	1	0	0
1	0	0	0

Análise da função objetivo e das variáveis

Quando dois funcionários i, j estão no mesmo conjunto, independentemente de ser C ou $\overline{C}$, um dos termos $y_i y_j$ ou $(1 - y_i)(1 - y_j)$ assumirá valor 1. Por outro lado, quando estão em conjuntos diferentes, ambos assumem valor nulo.

Os conjuntos devem ter aproximadamente o mesmo tamanho

Vamos impor que o número de funcionários alocados a C deve ser a metade de $|V|$.

$$\sum_{i \in V} y_i = \left\lfloor \frac{|V|}{2} \right\rfloor$$

Cada variável só assume valores 0 ou 1. Nenhum outro valor é admissível.

Alternativas para modelar esta restrição:

- Explicitamente impondo os valores admissíveis: $y_i \in \{0, 1\}$ para todo $i \in V$
- Impondo restrições não lineares equivalentes: $y_i(1 - y_i) = 0$ para todo $i \in V$.

Problema de Otimização Não Linear Inteiro

$$\max \sum_{i \in V} \sum_{j > i} (y_i y_j + (1 - y_i)(1 - y_j)) w_{ij}$$

$$\sum_{i \in V} y_i = \left\lfloor \frac{|V|}{2} \right\rfloor$$

$$y_i \in \{0, 1\} \qquad i \in V$$

Problema de Otimização Não Linear

$$\max \sum_{i \in V} \sum_{j > i} (y_i y_j + (1 - y_i)(1 - y_j)) w_{ij}$$

$$\sum_{i \in V} y_i = \left\lfloor \frac{|V|}{2} \right\rfloor$$

$$y_i (1 - y_i) = 0 \qquad i \in V$$

Produto $y_i y_j$ aparece na função objetivo

$$\max \sum_{i \in V} \sum_{j > i} (y_i y_j + (1 - y_i)(1 - y_j)) w_{ij}$$

Variáveis auxiliares

- Vamos utilizar variáveis auxiliares $u_{ij} \in \{0, 1\} : i, j \in V, j > i$ para representar o produto $y_i y_j$ para qualquer par $i, j : j > i$.
- Temos que garantir que $u_{ij} = 0$ quando algum dos valores y_i ou y_j é zero.
- Temos que garantir que $u_{ij} = 1$ quando ambos os valores y_i e y_j são um.

Variável auxiliar u_{ij} é usada para substituir o produto $y_i y_j$ na função objetivo.

Restrições adicionais garantem que $u_{ij} = y_i y_j$ quando $y_i, y_j \in \{0, 1\}$

- Para garantir que $u_{ij} = 0$ quando $y_i y_j = 0$, impomos:

$$u_{ij} \leq y_i \qquad i, j \in V, j > i$$

$$u_{ij} \leq y_j \qquad i, j \in V, j > i$$

- Para garantir que $u_{ij} = 1$ quando $y_i = y_j = 1$, impomos:

$$u_{ij} \geq y_i + y_j - 1 \qquad i, j \in V, j > i$$

Linearizando a função objetivo:

$$
\begin{aligned}
&\sum_{i \in V} \sum_{j > i} (y_i y_j + (1 - y_i)(1 - y_j)) w_{ij} \\
&= \sum_{i \in V} \sum_{j > i} (2 y_i y_j - y_i - y_j + 1) w_{ij} \\
&= \sum_{i \in V} \sum_{j > i} 2 w_{ij} u_{ij} - \sum_{i \in V} (\sum_{j > i} w_{ij}) y_i + \sum_{i \in V} \sum_{j > i} w_{ij}
\end{aligned}
$$

Problema Linear Inteiro

$$\max \sum_{i \in V} \sum_{j > i} 2 w_{ij} u_{ij} - \sum_{i \in V} (\sum_{j > i} w_{ij}) y_i + \sum_{i \in V} \sum_{j > i} w_{ij}$$

$$\sum_{i \in V} y_i = \left\lfloor \frac{|V|}{2} \right\rfloor$$

$$\begin{aligned}
y_i &\in \{0,1\} && i \in V \\
u_{ij} &\in \{0,1\} && i,j \in V, j > i \\
u_{ij} &\leq y_i && i,j \in V, j > i \\
u_{ij} &\leq y_j && i,j \in V, j > i \\
u_{ij} &\geq y_j + y_i - 1 && i,j \in V, j > i
\end{aligned}$$

Problema gerencial

- Suponha que um sistema seja caracterizado por um de m estados, mutuamente excludentes. Por exemplo, considere uma corrida de cavalos. Neste sistema, o estado corresponde ao cavalo vencedor da corrida.
- Um leiloeiro organiza um leilão do tipo *parimutuel*: os contratos são firmados entre as partes (entre a casa e cada aposta) apenas depois que a casa recolher todas as propostas de apostas.
- Cada aposta corresponde a subconjuntos dos m possíveis cavalos (estados). A casa se compromete a pagar 1 unidade monetária caso o cavalo vencedor seja um dos selecionados na aposta. Caso nenhum dos cavalos selecionados na aposta seja o vencedor, nada é pago.

Dados do problema

- Existem n apostadores que submetem ordens, propostas de apostas para a casa. Só após receber todas as ordens, o leiloeiro decide quais serão firmadas. Vamos assumir que cada apostador apresente uma única ordem.
- Cada ordem $j = 1, \dots, n$ é definida por:
  - $\mathbf{a}_j = (a_{1j}, a_{2j}, \cdots, a_{mj}) \in \{0,1\}^m$: um vetor m dimensional de 0s e 1s que indica quais cavalos foram selecionados naquela ordem;
  - o preço máximo π_j que se dispõe a pagar pela ordem j;
  - e q_j, o número máximo de ordens nestes termos que deseja firmar.
- Após receber todas as n ordens, o leiloreiro deve decidir quais e quantas ordens firmar.

Como formular este problema de decisão ?

O que se deseja decidir ?

- Quantas ordens de cada um dos n tipos devem ser firmadas.
- Variável de decisão: $\mathbf{x} = \{x_j \in \mathbb{Z}_+ : j = 1, \ldots, n\}$ indicando a quantidade de cada uma das n ordens que o leiloeiro irá emitir.

Restrições do problema (triviais)

- O número de ordens j não dever exceder q_j:
  $x_j \leq q_j, \;\; j = 1, \ldots, n.$
- Integralidade e não-negatividade das variáveis:
  $x_j \in \mathbb{Z}_+, \;\; j = 1, \ldots, n$

Qual o sentido da decisão (sentido de otimização) ?

- Dentre as *n* ordens colocadas, escolher quais e qual quantidade delas, com o objetivo de maximizar o lucro do leiloreiro.
- O lucro é dado pela diferença entre o valor recebido pelas ordens que firmar e o que paga para as ordens firmadas, que contemplarem o vencedor. Este é um dado desconhecido.

Como lidar com esta incerteza ?

Uma possibilidade de tratamento da incerteza, que será adotada no modelo que propomos, é decidir pelo caso mais conservador, isto é, o pior caso para o tomador de decisão (leiloeiro).

Representando a receita a ser recebida

- Quantidade recebida pelo leiloeiro pela $j-$ ésima ordem: $\pi_j x_j$.
- Quantidade total recebida, referente às n ordens: $\sum_{j=1}^{n} \pi_j x_j$

Representando o custo incorrido (no pior caso)

- Se $i \in \{1, \ldots, m\}$ for o índice do cavalo vencedor, o leiloeiro deve pagar 1 para cada ordem que tiver escolhido o cavalo i, isto é:

$$\sum_{j=1}^{n} a_{ij} x_j = \sum_{j: a_{ij}=1} x_j$$

- O pior dentre os m possíveis casos é:

$$\max_{i=1,\ldots,m} \left\{ \sum_{j=1}^{n} a_{ij} x_j \right\}$$

Senso de otimização

- Defina
$$f(\mathbf{x}) := \sum_{j=1}^{n} \pi_j x_j - \max_{i=1,\ldots,m} \left\{ \sum_{j=1}^{n} a_{ij} x_j \right\}$$
- Função objetivo do problema: $\max f(x)$.

Tratando a não linearidade de $f(\mathbf{x})$

- O termo $\max_{i=1,\ldots,m}\{\sum_{j=1}^{n} a_{ij}x_j\}$ torna $f(\mathbf{x})$ não linear.
- Para linearizar, introduzimos uma variável adicional $s \in \mathbb{R}$, para recupera o valor do máximo.
- Redefinimos a função objetivo a ser maximizada por $v(\mathbf{x}, s) := \sum_{j=1}^{n} \pi_j x_j - s$.
- Inserimos m restrições adicionais: $\sum_{j=1}^{n} a_{ij}x_j \leq s : i = 1, \ldots, m$

$$\begin{aligned} \max \sum_{j=1}^{n} \pi_j x_j - s & & \\ \sum_{j=1}^{n} a_{ij} x_j \leq s & & i = 1, \ldots, m \\ x_j \leq q_j & & j = 1, \ldots, n \\ x_j \in \mathbb{Z}_+ & & j = 1, \ldots, n \end{aligned}$$

Observação sobre a integralidade

Neste problema em particular, não é necessário explicitamente impor a restrição de integralidade.

Descrição do problema gerencial

A Union-Airways é uma empresa de transporte aéreo que está ampliando o número de vôos em seu aeroporto. Para tanto, precisa contratar mais agentes para atendimento ao público. Deseja-se saber qual é o número de profissionais que deve ser contratado. A gerência da empresa reconhece a necessidade de controle de custos, mas ao mesmo tempo, deseja proporcionar um atendimento com requisitos mínimos de qualidade aos clientes. Assim sendo, para vários períodos do dia, a empresa conhece o número mínimo de agentes que devem estar em atividade, para que este serviço de mínima qualidade seja oferecido. Os funcionários da empresa são contratados para trabalhar em turnos específicos, com remunerações específicas para cada turno.

Dados específicos do problema da Union-Airways

Sobre os turnos de trabalho

	Horário coberto		Custo por funcionário
	de	até	R$
Turno 1	6:00	14:00	170,00
Turno 2	8:00	16:00	160,00
Turno 3	12:00	20:00	175,00
Turno 4	16:00	24:00	180,00
Turno 5	22:00	06:00	195,00

Sobre a qualidade de serviço

	Horário da janela		Número mínimo de funcionários
	de	até	
Janela 1	6:00	8:00	48
Janela 2	8:00	10:00	79
Janela 3	10:00	12:00	65
Janela 4	12:00	14:00	87
Janela 5	14:00	16:00	64
Janela 6	16:00	18:00	73
Janela 7	18:00	20:00	82
Janela 8	20:00	22:00	43
Janela 9	22:00	06:00	52

Modelagem do problema e não da instância do problema

Princípio da dissociação entre problema e instância

Nunca é exagaerado lembrar...... Embora no caso particular da Union-Airways, os dados do problema sejam disponíveis, vamos formular o problema geral de planejamento de mão de obra e não uma instância do mesmo.

Definindo os parâmetros necessários para formular o problema

- $J = \{1, \ldots, n\}$: conjunto de turnos de trabalho. O turno de trabalho $j \in J$ é caracterizado pelo seu horário de início s_j e de fim e_j.
- $I = \{1, \ldots, m\}$: conjunto de janelas de tempo consideradas. Uma janela de tempo é um período do dia, para a qual há uma necessidade específica de número de profissionais em atendimento. A janela de tempo $i \in I$ é caracterizada por seu horário de início e fim, respectivamente s_i, e_i.
- Matriz binária de cobertura $\mathbf{A} = (a_{ij} \in \{0, 1\})_{i \in I, j \in J}$:
  - $a_{ij} = 1$ se o turno j cobre integralmente a janela i, isto é, se $[s_i, e_i] \subseteq [s_j, e_j]$
  - $a_{ij} = 0$, em caso contrário.
- c_j : custo unitário de um profissional atuando no turno $j \in J$.
- b_i : número mínimo de profissionais trabalhando na janela i.

Matrix A para o caso da Union-Airways

	$6 \to 14$	$8 \to 16$	$12 \to 20$	$16 \to 24$	$22 \to 06$
$6 \to 8$	1	0	0	0	0
$8 \to 10$	1	1	0	0	0
$10 \to 12$	1	1	0	0	0
$12 \to 14$	1	1	1	0	0
$14 \to 16$	0	1	1	0	0
$16 \to 18$	0	0	1	1	0
$18 \to 20$	0	0	1	1	0
$20 \to 22$	0	0	0	1	0
$22 \to 06$	0	0	0	0	1

O que se deseja conhecer

O número mínimo de profissionais em atuação em cada turno.

Variáveis de decisão

Vetor de variáveis $\mathbf{x} = \{x_j \in \mathbb{Z}_+ : j \in J\}$ para definir estas quantidades.

Sentido de otimização

O custo de mão de obra deve ser o mínimo possível.

Custo

- Constribuição do turno $j \in J$ para o custo total: $c_j x_j$.
- Função objetivo a ser minimizada, referente ao custo de todos os turnos:

$$\min \sum_{j \in J} c_j x_j$$

Restrições estruturais do problema

- A qualidade mínima de atendimento deve ser respeitada.
- Respeitar a qualidade mínima significa dispor, em cada janela de tempo $i \in I$, de um número total de profissionais que iguale ou exceda b_i:

$$\sum_{j \in J} a_{ij} x_j \geq b_i : i \in I$$

Restrições de integralidade e não negatividade

$$x_j \in \mathbb{Z}_+ : j \in J.$$

A instância da Union-Airways deste problema

$$
\begin{aligned}
\min \quad & 170x_1 + 160x_2 + 175x_3 + 180x_4 + 195x_5 \\
& x_1 \geq 48 \\
& x_1 + x_2 \geq 79 \\
& x_1 + x_2 \geq 65 \\
& x_1 + x_2 + x_3 \geq 87 \\
& x_2 + x_3 \geq 64 \\
& x_3 + x_4 \geq 73 \\
& x_3 + x_4 \geq 82 \\
& x_4 \geq 43 \\
& x_5 \geq 52 \\
& x_i \in \mathbb{Z}_+, \quad i = 1, \dots, 5
\end{aligned}
$$

Algumas destas restrições são desnecessárias ? Sim, tanto para o problema em variáveis inteiras, quanto para a sua relaxação linear.

Uma pequena refinaria mistura cinco tipos de gasolinas para produzir dois tipos de combustíveis, normal e premium. A disponibilidade diária em número de barris de cada gasolina básica, bem como suas taxas de performance e custo por barril, são indicados na tabela a seguir. O contrato da refinaria impõe que sejam produzidos no mínimo 8000 barris/dia de combustível premium. Os preços de venda, por barril, do combustível normal e premium são respectivamente de 2,85 US$ e 3,75 US$.

O combustível normal deve ter taxa de performance de pelo menos 85, enquanto que a taxa de performance do premium deve ser de no mínimo 95. A performance de um combustível é proporcional à razão das gasolinas básicas utilizadas para sua produção. Por exemplo, uma mistura com 50% de gasolina tipo 1 e 50% de gasolina tipo 2 terá taxa de performance de 75.

Dados das gasolinas que podem ser misturadas

Gasolina básica Tipo	Taxa de Performance	Barris por dia	Custo US$ por barril
1	70	2000	0,80
2	80	4000	0,90
3	85	4000	0,95
4	90	5000	1,15
5	99	5000	2,00

Dados do problema:

- I : conjunto dos tipos de combustíveis a serem produzidos.
- J : conjunto dos tipos de gasolina básica, de entrada.
- $\{p_i : i \in I\}$: taxa de performance requerida para cada combustível a ser produzido.
- $\{l_i : i \in I\}$: preço de venda de cada combustível, por barril.
- $\{d_i : i \in I\}$: produção mínima exigida para cada combustível, em barris/dia.
- $\{g_j : j \in J\}$: taxa de performance observada de cada tipo de gasolina básica.
- $\{c_j : j \in J\}$: custo por barril de cada tipo de gasolina.
- $\{o_j : j \in J\}$: disponibilidade máxima de cada tipo de gasolina, em barris/dia.

- $\mathbf{x} = \{x_i : i \in I\}$.
  x_i denota o volume (em barris/dia) a ser produzido do combustível $i \in I$.

- $\mathbf{y} = \{y_{ij} : i \in I, j \in J\}$.
  y_{ij} denota o volume (em barris/dia) da gasolina básica $j \in J$ empregada para produção do combustível $i \in I$.

Função objetivo

Deseja-se maximizar a margem de contribuição, isto é, a diferença entre a receita obtida pela venda dos combustíveis e o custo das gasolinas básicas empregadas na mistura:

$$\max \sum_{i \in I} l_i x_i - \sum_{i \in I} \sum_{j \in J} c_j y_{ij}$$

- Balanço de massa:

$$x_i = \sum_{j \in J} y_{ij}, \; i \in I$$

- A taxa de performance de cada combustível deve ser atendida:

$$\sum_{j \in J} g_j y_{ij} \geq p_i x_i, \; i \in I$$

- A demanda mínima de cada combustível deve ser produzida

$$x_i \geq d_i, \; i \in I$$

- Consumo de gasolina básica é limitada à sua disponibilidade:

$$\sum_{i \in I} y_{ij} \leq o_j, \; j \in J$$

- Não negatividade: $x_i \geq 0, \; i \in I$