Questão 4 - Prova de otimização

Luiz Augusto Dembicki Fernandes, GRR20202416

25 de outubro de 2023

A)

i) Para maximizar o lucro precisamos levar em conta os custos de cada carvão e o preço de venda assim temos:

$$lucro\ com\ c = 2110R\$ \cdot (x + y) - (x \cdot 2000R\$ + y \cdot 1500R\$)$$

Onde x e y são quantidades de carvão A e B em toneladas. Como queremos trabalhar com um problema de minimização, a função objetivo se torna então:

$$-lucro\ com\ c = -2110R\$ \cdot (x + y) + (x \cdot 2000R\$ + y \cdot 1500R\$)$$

As restrições especificadas:

$$92\% \cdot x + 81\% \cdot y \leqslant 88\%(x + y)$$

$$1\% \cdot x + 2\% \cdot y \leqslant 1.5\%(x + y)$$

$$7\% \cdot x + 14\% \cdot y <= 10\%(x + y)$$

$$x + y \geqslant 26\%$$

ii) Foi utilizado python em ambiente jupyter notebook, com o método simplex modificado(HiGHS),da biblioteca Scipy, já que se trata de um problema de programação linear. Tal método atua por operações nas matrizes, que geometricamente o problema se torna encontrar um vértice qual seja o ponto ótimo, por meio de operações lineares é possível mover de vértice em vértice.

Foram criados dois casos, um em caso de toneladas forem variáveis continuas resultando em A: 16.54545 ton; B: 9.45455 ton; Lucro: 7587.27273 R$; E para variáveis discretas, neste caso inteiras, A: 17 ton; B: 9 ton; Lucro: 7360 R$ Segue o código fonte:

```
from scipy.optimize import linprog

# a)
# Digitos GRR
```

```
X = 6
Y = 4
Z = 1

PC = 2100 + (Z * 10) # Preco de venda de C por tonelada em reais
# Funcao objetivo
Fobj = (- PC + 2000, - PC + 1500) # Custos e preco de venda
# Inequacoes
# lado direito
LD = [[-0.04, 0.07], # inequacao 1 Carbono -4% * x + 7% * y <= 0
      [-0.005, 0.005], # inequacao 2 Enxofre -0.5% * x + 0.5% * y <= 0
      [-0.03, Y * 1e-2], # inequacao 3 cinzas -3% * x + Y% * y <= 0
      [1, 1]] # inequacao 4 demanda - x - y <= - 20 - X
# lado esquerdo
LE = [0, # inequacao 1 Carbono -4% * x + 7% * y <= 0
      0, # inequacao 2 Enxofre -0.5% * x + 0.5% * y <= 0
      0, # inequacao 3 cinzas -3% * x + Y% * y <= 0
      20 + X] # inequacao 4 demanda  - x - y <= - 20 - X
# Optimizacao por Simplex ++
opt = linprog(c = Fobj, A_ub = LD, b_ub = LE)
print(f"A:{opt.x[0]:.5f} B:{opt.x[1]:.5f}, Lucro: {opt.fun:.5f}")
opt = linprog(c = Fobj, A_ub = LD, b_ub = LE, integrality = 1)
print(f"A:{opt.x[0]} B:{opt.x[1]}, Lucro: {opt.fun}")
```

B) Tal qual anterior a função objetivo é dependente do custo de compra e de venda:

$$A + B + C - (C1 + C2 + C3 + C4)$$

como queremos em termos de minimização:

$$F_{obj} = -(A + B + C) + C1 + C2 + C3 + C4$$

E teremos as seguintes restrições, advindas dos blends requeridos, da conservação da massa e da quantidade máxima de barris :

$$C1 \leqslant 15\%A$$

$$C2 \geqslant 40\%A$$

$$C3 \leqslant 50\%A$$

$$C1 \leqslant 10\%B$$

$$C2 \geqslant 10\%B$$

$$C1 \leqslant 20\%C$$

$$C1 + C2 + C3 + C4 = A + B + C$$

$$C1 \leqslant 3000$$

$$C2 \leqslant 2000$$

$$C3 \leqslant 4000$$

$$C4 \leqslant 1000$$

Também foi assumido que só é possível adquirir e comercializar um barril inteiro:

$$\{C1, C2, C3, C4, A, B, C\} \in \mathbb{Z}$$

Com isso foi também utilizado um método simplex modificado, dessa vez com o pacote pulp, obtendo C1 = 121 barris, C2 = 2000 barris, C3 = 2500 barris, C4 = 1000 barris, lucro total = 6935.5 $ / dia

Segue o código fonte:

```python
import pulp
# b)
# problema
lucro = pulp.LpProblem("lucro", pulp.LpMinimize)
# Variaveis
C1 = pulp.LpVariable("C1", 0, 3000, pulp.LpInteger)
C2 = pulp.LpVariable("C2", 0, 2000, pulp.LpInteger)
C3 = pulp.LpVariable("C3", 0, 4000, pulp.LpInteger)
C4 = pulp.LpVariable("C4", 0, 1000, pulp.LpInteger)
A = pulp.LpVariable("A", 0, cat = pulp.LpInteger)
B = pulp.LpVariable("B", 0, cat = pulp.LpInteger)
C = pulp.LpVariable("C", 0, cat = pulp.LpInteger)
# equacoes e inequacoes
lucro += 13 * C1 + 15.3 * C2 + 14.6 * C3 + 14.9 * C4 - (16.2 * A + 15.75 * B
# restricoes
lucro += ( C1 - 15 * 1e-2 * A <= 0 , "Blend C1 em A")
lucro += ( - C2  + 40 * 1e-2 * A <= 0 , "Blend C2 em A")
lucro += ( C3 - 50 * 1e-2 * A <= 0 , "Blend C3 em A")
lucro += ( C1 - 10 * B <= 0 , "Blend C1 em B")
lucro += ( - C2 + 10 * 1e-2 * B <= 0 , "Blend C2 em B")
lucro += ( C1 - 20 * 1e-2 * C <= 0 , "Blend C1 em C")
lucro += (C1 + C2 + C3 + C4 - (A + B + C) == 0, "conservacao de massa")

lucro.writeLP("Questao4b")
lucro.solve()
print("Status:", pulp.LpStatus[lucro.status])
for v in lucro.variables():
    print(v.name, "=", v.varValue)
print("lucro total = ", - pulp.value(lucro.objective))
```