



# MODELAGEM NO GUROBI

PESQUISA OPERACIONAL ERE 2020/5

# PROBLEMA DO SAPATEIRO

Um sapateiro faz 6 sapatos por hora, se fizer somente sapatos, e 5 cintos por hora, se fizer somente cintos. Ele gasta 2 unidades de couro para fabricar 1 unidade de sapato e 1 unidade de couro para fabricar uma unidade de cinto. Sabendo-se que o total disponível de couro é de 6 unidades e que o lucro unitário por sapato é de 5 unidades monetárias e o do cinto é de 2 unidades monetárias, pede-se: o modelo do sistema de produção do sapateiro, se o objetivo é maximizar seu lucro por hora.

The screenshot shows a Jupyter Notebook window titled "PROBLEMA DO SAPATEIRO" with a subtitle "Last Checkpoint: uma hora atrás". The interface includes a menu bar (View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with icons for file operations, navigation, and execution. The code cell contains the following Python code:

```
import gurobipy as gp

from gurobipy import GRB

'''criando o modelo'''
m = gp.Model("sapateiro")

'''criando as variáveis'''
x = m.addVar(vtype=GRB.INTEGER, name="x")
y = m.addVar(vtype=GRB.INTEGER, name="y")

'''definindo o objetivo'''
m.setObjective(5*x+2*y, GRB.MAXIMIZE)

'''restrição: x + 2 y + 3 z <= 4'''
m.addConstr(10 * x + 12 * y <= 60, "c0")

'''retrição: x + y >= 1'''
m.addConstr(2 * x + y <= 6, "c1")

m.optimize()

for v in m.getVars():
    print('%s %g' % (v.varName, v.x))

print('Obj: %g' % m.objVal)
```

At the bottom of the notebook, it says "Using license file C:\Users\amand\gurobi.lic". On the right side of the notebook, there are navigation arrows and a pink square button.

## PROBLEMA DOS PRODUTOS P1 & P2

Certa empresa fabrica 2 produtos P1 e P2. O lucro por unidade de P1 é de 100 u.m. e o lucro unitário de P2 é de 150 u.m. A empresa necessita de 2 horas para fabricar uma unidade de P1 e 3 horas para fabricar uma unidade de P2. O tempo mensal disponível para essas atividades é de 120 horas. As demandas esperadas para os 2 produtos levaram a empresa a decidir que os montantes produzidos de P1 e P2 não devem ultrapassar 40 unidades de P1 e 30 unidades de P2 por mês. Construa o modelo do sistema de produção mensal com o objetivo de maximizar o lucro da empresa.

The screenshot shows a Jupyter Notebook window titled "PROBLEMA DOS PRODUTOS P1 & P2" with a "Last Checkpoint:" indicator. The interface includes a menu bar with "View", "Insert", "Cell", "Kernel", "Widgets", and "Help". Below the menu is a toolbar with icons for file operations, navigation, and execution. The main area contains Python code using the Gurobi library to define a linear programming model. The code includes comments in Portuguese explaining each step: creating the model, adding variables, setting the objective, and adding constraints. The model is named "Produtos P1 & P2". The objective is to maximize the profit, calculated as 100 times the quantity of P1 plus 150 times the quantity of P2. There are three constraints: a total time constraint (2x + 3y ≤ 120) and two individual production limits (x ≤ 40 and y ≤ 30). The code concludes by optimizing the model and printing the optimal values for the variables and the objective function.

```
import gurobipy as gp

from gurobipy import GRB

'''criando o modelo'''
m = gp.Model("Produtos P1 & P2")

'''criando as variáveis'''
x = m.addVar(vtype=GRB.INTEGER, name="x")
y = m.addVar(vtype=GRB.INTEGER, name="y")

'''adicionando o objetivo'''
m.setObjective(100*x+150*y, GRB.MAXIMIZE)

'''restrição: x + y >= 1'''
m.addConstr(2 * x + 3 * y <= 120, "c0")

'''restrição: x + y >= 1'''
m.addConstr(x <= 40, "c1")
m.addConstr(y <= 30, "c2")

m.optimize()

for v in m.getVars():
    print('%s %g' % (v.varName, v.x))

print('Obj: %g' % m.objVal)
```

# PROBLEMA DA REDE DE TV

Uma rede de televisão local tem o seguinte problema: foi descoberto que o programa "A" com 20 minutos de música e 1 minuto de propaganda chama a atenção de 30.000 telespectadores, enquanto o programa "B", com 10 minutos de música e 1 minuto de propaganda chama a atenção de 10.000 telespectadores. No decorrer de uma semana, o patrocinador insiste no uso de no mínimo, 5 minutos para sua propaganda e que não há verba para mais de 80 minutos de música.

Quantas vezes por semana cada programa deve ser levado ao ar para obter o número máximo de telespectadores? Construa o modelo do sistema.

The screenshot shows a Jupyter Notebook window titled "PROBLEMA REDE DE TV" with a subtitle "Last Checkpoint: uma hora atrás (au...". The interface includes a menu bar with "View", "Insert", "Cell", "Kernel", "Widgets", and "Help". Below the menu is a toolbar with icons for file operations, navigation, and execution. The main area displays Python code using the Gurobi library to formulate an optimization problem.

```
import gurobipy as gp

from gurobipy import GRB

'''criando o modelo'''

m = gp.Model("Rede de TV")

'''criando as variáveis'''
x = m.addVar(vtype=GRB.INTEGER, name="x")
y = m.addVar(vtype=GRB.INTEGER, name="y")

'''definindo o objetivo'''
m.setObjective(30000*x+10000*y, GRB.MAXIMIZE)

'''restrição: x + y >= 1'''
m.addConstr(x+y >= 5, "c0")

'''restrição:: x + y >= 1'''
m.addConstr(20 * x + 10 * y <= 80, "c0")


m.optimize()

for v in m.getVars():
    print('%s %g' % (v.varName, v.x))

print('Obj: %g' % m.objval)
```

# PROBLEMA DOS CINTOS DE COURO

Um empresa fabrica 2 modelos de cintos de couro. O modelo M1, de melhor qualidade, requer o dobro do tempo de fabricação em relação ao modelo M2. Se todos os cintos fossem do modelo M2, a empresa poderia produzir 1.000 unidades por dia. A disponibilidade de couro permite fabricar 800 cintos de ambos os modelos por dia. Os cintos empregam fivelas diferentes, cuja disponibilidade diária é de 400 para M1 e 700 para M2. Os lucros unitários são de \$ 4,00 para M1 e \$ 3,00 para M2. Qual o programa ótimo de produção que maximiza o lucro total diário da empresa? Construa o modelo do sistema descrito.

```
PROBLEMA CINTO DE COURO Last Checkpoint: uma hora :  
View Insert Cell Kernel Widgets Help  
  
  
import gurobipy as gp  
  
from gurobipy import GRB  
  
'''criando o modelo'''  
  
m = gp.Model("Cintos de Couro")  
  
'''criando as variáveis'''  
x = m.addVar(vtype=GRB.INTEGER, name="x")  
y = m.addVar(vtype=GRB.INTEGER, name="y")  
  
'''adicionando o objetivo'''  
m.setObjective(4*x+3*y, GRB.MAXIMIZE)  
  
'''restrição x + y >= 1'''  
m.addConstr(2 * x + y <= 1000, "c0")  
m.addConstr(x + y <= 800, "c1")  
  
''' restrição: x + y >= 1'''  
m.addConstr(x <= 400, "c2")  
m.addConstr(y <= 700, "c3")  
  
m.optimize()  
  
for v in m.getVars():  
    print('%s %g' % (v.varName, v.x))  
  
print('Obj: %g' % m.objval)
```

# PROBLEMA DA RACIONALIZAÇÃO

Uma empresa, após um processo de racionalização de produção, ficou com disponibilidade de 3 recursos produtivos, R1, R2 e R3. Um estudo sobre o uso desses recursos indicou a possibilidade de se fabricar 2 produtos P1 e P2. Levantando os custos e consultando o departamento de vendas sobre o preço de colocação no mercado, verificou-se que P1 daria um lucro de \$ 120,00 por unidade e P2, \$ 150,00 por unidade. O departamento de produção

forneceu a seguinte tabela de uso de recursos.

Produto	Recurso R1 por unidade	Recurso R2 por unidade	Recurso R3 por unidade
P1	2	3	5
P2	4	2	3
Disponibilidade de recursos por mês	100	90	120

Que produção mensal de P1 e P2 traz o maior lucro para a empresa? Construa o modelo do sistema.

## PROBLEMA RACIONALIZAÇÃO DE PRODUÇÃO

ViewInsertCellKernelWidgetsHelp

Run

Code

```
import gurobipy as gp

from gurobipy import GRB

'''criando o modelo'''

m = gp.Model("produtos")

'''criando as variáveis'''
x = m.addVar(vtype=GRB.INTEGER, name="x")
y = m.addVar(vtype=GRB.INTEGER, name="y")

'''definindo o objetivo'''
m.setObjective(120*x+150*y, GRB.MAXIMIZE)

'''retrição: x + y >= 1'''
m.addConstr(2 * x + 4 * y <= 100)
m.addConstr(3 * x + 2 * y <= 90)
m.addConstr(5 * x + 3 * y <= 120)

m.optimize()

for v in m.getVars():
    print('%s %g' % (v.varName, v.x))

print('Obj: %g' % m.objVal)
```

↑

↓