
****DESCARGAMOS LA BBIBLOTECA Y EL SOLVER****

```
!pip install gurobipy
from gurobipy import Model, GRB
import pandas as pd
```

****Ejercicios****

Una empresa produce dos productos: A y B. Cada unidad de A requiere 2 horas de trabajo y 3 kg de material. Cada unidad de B requiere 4 horas de trabajo y 2 kg de material.

La empresa tiene disponibles 100 horas de trabajo y 90 kg de material.

Las ganancias son de 30 USD por cada unidad de A y 50 USD por cada unidad de B.

¿Cuántas unidades de A y B debe producir para maximizar la ganancia?

Crear modelo

```
model = Model("Producción_simple")
```

Variables

```

A = model.addVar(name="A", lb=0)
B = model.addVar(name="B", lb=0)

# Función objetivo: maximizar la ganancia
model.setObjective(30 * A + 50 * B, GRB.MAXIMIZE)

# Restricciones
model.addConstr(2 * A + 4 * B <= 100, name="Trabajo")
model.addConstr(3 * A + 2 * B <= 90, name="Material")

# Resolver
model.optimize()

# Resultados
if model.status == GRB.OPTIMAL:
    print(f"\nUnidades a producir:")
    print(f"A: {A.x}")
    print(f"B: {B.x}")
    print(f"Ganancia total: ${model.objVal:.2f}")

**Versión con análisis de sensibilidad**

# Modelo
model = Model("Producción_sensibilidad")

# Variables

```

```
A = model.addVar(name="A", lb=0)
```

```
B = model.addVar(name="B", lb=0)
```

```
# Objetivo
```

```
model.setObjective(30 * A + 50 * B, GRB.MAXIMIZE)
```

```
# Restricciones
```

```
model.addConstr(2 * A + 4 * B <= 100, name="Trabajo")
```

```
model.addConstr(3 * A + 2 * B <= 90, name="Material")
```

```
# Resolver
```

```
model.optimize()
```

```
# Resultados
```

```
if model.status == GRB.OPTIMAL:
```

```
    print(f"\nUnidades a producir:")
```

```
    print(f"A: {A.x}")
```

```
    print(f"B: {B.x}")
```

```
    print(f"Ganancia total: ${model.objVal:.2f}")
```

```
# Precios sombra
```

```
print("\nPrecios sombra (dual):")
```

```
for c in model.getConstrs():
```

```
    print(f"{c.ConstrName}: {c.pi:.2f}")
```

```
# Intervalos de sensibilidad
```

```
print("\nIntervalos de sensibilidad (obj):")  
  
for v in model.getVars():  
    print(f"{v.VarName}: [{v.SAObjLow}, {v.SAObjUp}]")
```

Supón que producir una unidad de A cuesta 30 USD y una de B cuesta 50 USD.

Necesitamos producir al menos 10 unidades de A y al menos 5 unidades de B, cumpliendo además con las limitaciones de recursos disponibles:

- * 100 horas de trabajo

- * 90 kg de material

Cada unidad de A requiere 2 h de trabajo y 3 kg de material.

Cada unidad de B requiere 4 h de trabajo y 2 kg de material.

¿Cuántas unidades de cada producto debe producir la empresa para cubrir la demanda al menor costo posible?

```
model = Model("Minimización_simple")
```

```
# Variables
```

```
A = model.addVar(name="A", lb=0)
```

```
B = model.addVar(name="B", lb=0)
```

```
# Objetivo: minimizar el costo total
```

```
model.setObjective(30 * A + 50 * B, GRB.MINIMIZE)
```

```
# Restricciones de recursos
```

```
model.addConstr(2 * A + 4 * B <= 100, name="Trabajo")
```

```
model.addConstr(3 * A + 2 * B <= 90, name="Material")
```

```
# Restricciones de demanda mínima
```

```
model.addConstr(A >= 10, name="Demanda_A")
```

```
model.addConstr(B >= 5, name="Demanda_B")
```

```
# Resolver
```

```
model.optimize()
```

```
# Resultados
```

```
if model.status == GRB.OPTIMAL:
```

```
    print(f"\nUnidades a producir:")
```

```
    print(f"A: {A.x}")
```

```
    print(f"B: {B.x}")
```

```
    print(f"Costo total: ${model.objVal:.2f}")
```

```
**Versión con análisis de sensibilidad**
```

```
# Crear modelo
```

```
model = Model("Minimización_sensibilidad")
```

Variables

A = model.addVar(name="A", lb=0)

B = model.addVar(name="B", lb=0)

Objetivo

model.setObjective(30 * A + 50 * B, GRB.MINIMIZE)

Restricciones

model.addConstr(2 * A + 4 * B <= 100, name="Trabajo")

model.addConstr(3 * A + 2 * B <= 90, name="Material")

model.addConstr(A >= 10, name="Demanda_A")

model.addConstr(B >= 5, name="Demanda_B")

Resolver

model.optimize()

Resultados

if model.status == GRB.OPTIMAL:

print(f"\nUnidades a producir:")

print(f"A: {A.x}")

print(f"B: {B.x}")

print(f"Costo total: \${model.objVal:.2f}")

Precios sombra

print("\nPrecios sombra:")

for c in model.getConstrs():

```
print(f"{c.ConstrName}: {c.pi:.2f}")
```

```
# Intervalos de sensibilidad
```

```
print("\nIntervalos de sensibilidad (costos):")
```

```
for v in model.getVars():
```

```
    print(f"{v.VarName}: [{v.SAObjLow}, {v.SAObjUp}]")
```

Una empresa fabrica 3 productos: P1, P2 y P3.

Cada uno genera una utilidad diferente y consume cierta cantidad de recursos (horas de máquina y kg de material).

Se tienen limitaciones en la disponibilidad de estos recursos.

¿Cuántas unidades de cada producto deben producirse para maximizar las utilidades totales?

```
# Asegúrese de subir el archivo 'datos.xlsx' en tu entorno de Colab
```

```
archivo = 'datos.xlsx'
```

```
productos_df = pd.read_excel(archivo, sheet_name="Productos")
```

```
recursos_df = pd.read_excel(archivo, sheet_name="Recursos")
```

```
consumo_df = pd.read_excel(archivo, sheet_name="Consumo")
```

```
productos = list(productos_df['Producto'])
```

```
recursos = list(recursos_df['Recurso'])
```

```

utilidad = dict(zip(productos_df['Producto'], productos_df['Utilidad']))
disponibilidad = dict(zip(recursos_df['Recurso'], recursos_df['Disponibilidad']))
consumo = {
    r: dict(zip(consumo_df['Producto'], consumo_df[r])) for r in recursos
}

model = Model("Maximizar_Utilidades")

X = model.addVars(productos, name="Producción", lb=0)

model.setObjective(sum(utilidad[p] * X[p] for p in productos), GRB.MAXIMIZE)

for r in recursos:
    model.addConstr(sum(consumo[r][p] * X[p] for p in productos) <= disponibilidad[r],
name=f"Restr_{r}")

model.optimize()

if model.status == GRB.OPTIMAL:
    print("\nProducción óptima:")
    for p in productos:
        print(f"{p}: {X[p].x:.2f} unidades")
    print(f"\nUtilidad total: ${model.objVal:.2f}")
else:
    print("No se encontró solución óptima.")

```