

Laboratorio 2 Modelado Simulación y Optimización

Ejercicio 1:

Modelo Matemático

Conjuntos

- O : Conjunto de ciudades de origen. $O = \{\text{Bogotá, Medellín}\}$.
- D : Conjunto de ciudades de destino. $D = \{\text{Cali, Barranquilla, Pasto, Tunja, Chía, Manizales}\}$.

Parámetros

- c_{od} : Costo de transporte por tonelada desde la ciudad de origen o a la ciudad de destino d . Ejemplo: $c_{\text{Bogotá, Barranquilla}} = 2.5$.
- d_d : Demanda de productos en toneladas en la ciudad de destino d . Ejemplo: $d_{\text{Cali}} = 125$.
- s_o : Capacidad de oferta de productos en toneladas en la ciudad de origen o . Ejemplo: $s_{\text{Bogotá}} = 550$.

Variables de Decisión

- x_{od} : Cantidad de toneladas transportadas desde la ciudad de origen o a la ciudad de destino d .

Función Objetivo

Minimizar el costo total de transporte:

$$\text{Minimizar } \sum_{o \in O} \sum_{d \in D} c_{od} \cdot x_{od}$$

Restricciones

- **Restricción de Oferta:** La cantidad total enviada desde una ciudad de origen no puede exceder la oferta disponible en esa ciudad:

$$\sum_{d \in D} x_{od} \leq s_o \quad \forall o \in O$$

- **Restricción de Demanda:** La cantidad total recibida en cada ciudad de destino debe satisfacer exactamente su demanda:

$$\sum_{o \in O} x_{od} = d_d \quad \forall d \in D$$

- **No Negatividad:** Las toneladas transportadas deben ser no negativas:

$$x_{od} \geq 0 \quad \forall o \in O, d \in D$$

Resultados de la implementación en Pyomo:

```
Model unknown

Variables:
  x : Size=12, Index=0+0
    Key : Lower : Value : Upper : Fixed : Stale : Domain
    ('Bogota', 'Barranquilla') : 0 : 175.0 : None : False : False : NonNegativeReals
    ('Bogota', 'Cali') : 0 : 0.0 : None : False : False : NonNegativeReals
    ('Bogota', 'Chia') : 0 : 150.0 : None : False : False : NonNegativeReals
    ('Bogota', 'Manizales') : 0 : 0.0 : None : False : False : NonNegativeReals
    ('Bogota', 'Pasto') : 0 : 225.0 : None : False : False : NonNegativeReals
    ('Bogota', 'Tunja') : 0 : 0.0 : None : False : False : NonNegativeReals
    ('Medellin', 'Barranquilla') : 0 : 0.0 : None : False : False : NonNegativeReals
    ('Medellin', 'Cali') : 0 : 125.0 : None : False : False : NonNegativeReals
    ('Medellin', 'Chia') : 0 : 75.0 : None : False : False : NonNegativeReals
    ('Medellin', 'Manizales') : 0 : 200.0 : None : False : False : NonNegativeReals
    ('Medellin', 'Pasto') : 0 : 0.0 : None : False : False : NonNegativeReals
    ('Medellin', 'Tunja') : 0 : 250.0 : None : False : False : NonNegativeReals

Objectives:
  obj : Size=1, Index=None, Active=True
    Key : Active : Value
    None : True : 1715.0

Constraints:
  supply_con : Size=2
    Key : Lower : Body : Upper
    Bogota : None : 550.0 : 550.0
    Medellin : None : 650.0 : 700.0
  demand_con : Size=6
    Key : Lower : Body : Upper
    Barranquilla : 175.0 : 175.0 : 175.0
    Cali : 125.0 : 125.0 : 125.0
    Chia : 225.0 : 225.0 : 225.0
    Manizales : 200.0 : 200.0 : 200.0
    Pasto : 225.0 : 225.0 : 225.0
    Tunja : 250.0 : 250.0 : 250.0
Toneladas de Bogota a Cali: 0.0
Toneladas de Bogota a Barranquilla: 175.0
Toneladas de Bogota a Pasto: 225.0
Toneladas de Bogota a Tunja: 0.0
Toneladas de Bogota a Chia: 150.0
Toneladas de Bogota a Manizales: 0.0
Toneladas de Medellin a Cali: 125.0
Toneladas de Medellin a Barranquilla: 0.0
Toneladas de Medellin a Pasto: 0.0
Toneladas de Medellin a Tunja: 250.0
Toneladas de Medellin a Chia: 75.0
Toneladas de Medellin a Manizales: 200.0
```

```
Dual Values (Shadow Prices):
DUALS for constraint: supply_con
supply_con[Bogota]: Dual Value = -0.2
supply_con[Medellin]: Dual Value = 0.0
DUALS for constraint: demand_con
demand_con[Cali]: Dual Value = 2.5
demand_con[Barranquilla]: Dual Value = 2.7
demand_con[Pasto]: Dual Value = 1.8
demand_con[Tunja]: Dual Value = 1.0
demand_con[Chia]: Dual Value = 1.0
demand_con[Manizales]: Dual Value = 0.8
```

Análisis de sensibilidad a partir de los valores duales

Restricciones de Oferta (Duales):

- **Bogotá:** Valor Dual = -0.2

- Este valor dual negativo sugiere que aumentar la oferta en Bogotá en una unidad (tonelada) reduciría el costo total de transporte en 0.2 USD).

- Para Bogotá podría ser beneficioso aumentar la oferta en Bogotá si es posible, ya que puede ayudar a reducir ligeramente los costos totales.

- **Medellín:** Valor Dual = 0.0

- Un valor dual de cero indica que la restricción de oferta de Medellín no es vinculante. Aumentar o disminuir la oferta aquí no afectará el costo total de transporte.

- Para Medellín hay una ventaja inmediata en aumentar la oferta en Medellín, ya que no reducirá los costos.

Restricciones de Demanda (Duales):

- **Cali:** Valor Dual = 2.5

- Un valor dual de 2.5 significa que satisfacer una tonelada adicional de demanda en Cali aumentaría el costo total de transporte en 2.5 USD.

- Satisfacer demanda adicional en Cali es relativamente costoso, por lo que cumplir con cualquier demanda extra aquí incrementará significativamente los costos.

- **Barranquilla:** Valor Dual = 2.7

- Un valor dual alto indica que satisfacer demanda adicional en Barranquilla aumentaría el costo total de transporte en 2.7 USD por tonelada.

- En Barranquilla es aún más costoso satisfacer demanda adicional en Barranquilla en comparación con Cali, por lo que limitar la demanda excesiva aquí es clave para mantener bajos los costos de transporte.

- **Pasto:** Valor Dual = 1.8

- La demanda adicional en Pasto aumentará el costo total de transporte en **1.8 USD por tonelada.

- Aunque sigue siendo costoso, cumplir con demanda adicional en Pasto es más barato que en Cali o Barranquilla.

- **Tunja:** Valor Dual = 1.0

- Satisfacer más demanda en Tunja costará 1.0 USD por tonelada.

- Tunja es relativamente más barato para satisfacer demanda extra en comparación con otras ciudades.

- **Chía:** Valor Dual = 1.0

- Satisfacer más demanda en Chía también costará 1.0 USD por tonelada.

- Al igual que Tunja, Chía ofrece un costo relativamente bajo para satisfacer demanda adicional.

- **Manizales:** Valor Dual = 0.8

- El valor dual más bajo en la demanda es para Manizales. Satisfacer más demanda aquí aumentará el costo total en solo 0.8 USD por tonelada.

- Si hay necesidad de aumentar la demanda en algún lugar, Manizales es la ciudad más rentable para suministrar más.

Ajuste de Oferta: Mover 50 Toneladas de Medellín a Bogotá

Nuevas Ofertas:

Bogotá: 600 toneladas

Medellín: 650 toneladas

Nuevo Plan de Envíos:

Desde Bogotá:

Barranquilla: 175 toneladas

Pasto: 225 toneladas

Tunja: 200 toneladas

Chía: 0 toneladas

Manizales: 0 toneladas

Total enviado: 600 toneladas

Desde Medellín:

Cali: 125 toneladas

Tunja: 50 toneladas

Chía: 225 toneladas

Manizales: 200 toneladas

Total enviado: 600 toneladas

Análisis Comparativo

Mayor Oferta en Bogotá:

Permite que Bogotá suministre más a Tunja, donde su costo es ligeramente más alto que el de Medellín (1.4 USD/ton frente a 1.0 USD/ton).

Sin embargo, la oferta reducida de Medellín significa que no puede satisfacer toda la demanda en Tunja, por lo que Bogotá compensa a pesar del mayor costo.

El costo total puede aumentar ligeramente porque se envían más toneladas desde Bogotá a Tunja a un costo más alto en comparación con Medellín.

¿Es Beneficioso el Cambio en la Oferta?

No necesariamente. El aumento en la oferta de Bogotá lleva a envíos más costosos a Tunja.

Medellín tiene un costo más bajo para enviar a Tunja (1.0 USD/ton), por lo que es más económico que Medellín satisfaga la mayor parte de la demanda en Tunja.

Conclusión

Cambio de Oferta (Mover 50 toneladas a Bogotá): Puede no reducir los costos debido a mayores costos de envío desde Bogotá a ciertos destinos. No es ideal hacer ese cambio.

Ejercicio 2:

Modelo matemático:

1 Conjuntos

- E : Conjunto de equipos, $E = \{1, 2, \dots, k\}$, donde k es el número de equipos.
- L : Conjunto de localidades, $L = \{0, 1, 2, \dots, n-1\}$, donde n es el número de localidades.
- L_o : Localidad de origen, denotada por 0.

2 Parámetros

- d_{ij} : Distancia entre la localidad i y la localidad j , donde $i, j \in L$.

3 Variables de Decisión

- x_{eij} : Variable binaria que toma el valor de 1 si el equipo $e \in E$ viaja de la localidad $i \in L$ a la localidad $j \in L$, y 0 en caso contrario.
- u_{ei} : Variable de enteros no negativos utilizada para la eliminación de subcircuitos.

4 Función Objetivo

Minimizar la distancia total recorrida por los equipos:

$$\min \sum_{e \in E} \sum_{i \in L} \sum_{j \in L} d_{ij} \cdot x_{eij}$$

5 Restricciones

- **Restricción de salida del origen:** Cada equipo debe salir de la localidad de origen exactamente una vez.

$$\sum_{j \in L, j \neq 0} x_{e0j} = 1 \quad \forall e \in E$$

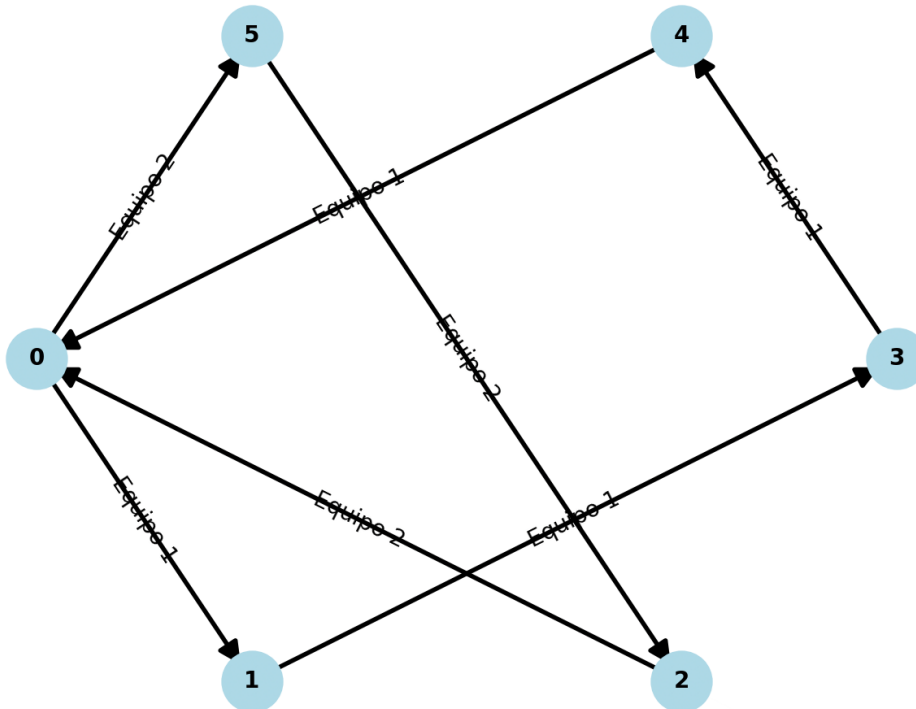
Resultados:

```

(2, 3, 3) : 0 : 0.0 : 1 : False : False : Binary
u : Size=12, Index=equipos*localidades
Key : Lower : Value : Upper : Fixed : Stale : Domain
(1, 0) : 0 : 0 : None : False : True : NonNegativeIntegers
(1, 1) : 0 : 0.0 : None : False : False : NonNegativeIntegers
(1, 2) : 0 : 0.0 : None : False : False : NonNegativeIntegers
(1, 3) : 0 : 1.0 : None : False : False : NonNegativeIntegers
(1, 4) : 0 : 2.0 : None : False : False : NonNegativeIntegers
(1, 5) : 0 : 0.0 : None : False : False : NonNegativeIntegers
(2, 0) : 0 : 0 : None : False : True : NonNegativeIntegers
(2, 1) : 0 : 0.0 : None : False : False : NonNegativeIntegers
(2, 2) : 0 : 1.0 : None : False : False : NonNegativeIntegers
(2, 3) : 0 : 0.0 : None : False : False : NonNegativeIntegers
(2, 4) : 0 : 0.0 : None : False : False : NonNegativeIntegers
(2, 5) : 0 : 0.0 : None : False : False : NonNegativeIntegers

Objectives:
obj : Size=1, Index=None, Active=True
Key : Active : Value
None : True : 9.0

Constraints:
salida_origen : Size=2
Key : Lower : Body : Upper
1 : 1.0 : 1.0 : 1.0
2 : 1.0 : 1.0 : 1.0
regreso_origen : Size=2
Key : Lower : Body : Upper
1 : 1.0 : 1.0 : 1.0
2 : 1.0 : 1.0 : 1.0
visitar_localidad_una_vez : Size=5
Key : Lower : Body : Upper
1 : 1.0 : 1.0 : 1.0
2 : 1.0 : 1.0 : 1.0
3 : 1.0 : 1.0 : 1.0
4 : 1.0 : 1.0 : 1.0
5 : 1.0 : 1.0 : 1.0
  
```



Ejercicio 3:

Modelo matemático:

1 Conjuntos

- L : Conjunto de localidades, $L = \{L_1, L_2, \dots, L_n\}$.
- S : Conjunto de tipos de sensores, $S = \{S_1, S_2, S_3\}$.

2 Parámetros

- C_l : Costo de instalación en la localidad $l \in L$.
- E_s : Consumo de energía del sensor $s \in S$.
- C_{sl} : Costo de comunicación entre el sensor $s \in S$ y la localidad $l \in L$.
- Cov_{sl} : Parámetro que indica si el sensor $s \in S$ cubre la localidad $l \in L$.
- $Adj_{ll'}$: Parámetro que indica si la localidad $l \in L$ es adyacente a la localidad $l' \in L$.

3 Variables de Decisión

- x_{sl} : Variable binaria que toma el valor 1 si el sensor $s \in S$ se instala en la localidad $l \in L$, y 0 en caso contrario.

4 Función Objetivo

Minimizar el costo total de instalación, consumo de energía y comunicación:

$$\min \sum_{s \in S} \sum_{l \in L} (C_l \cdot x_{sl} + E_s \cdot x_{sl} + C_{sl} \cdot x_{sl})$$

5 Restricciones

- **Restricción de cobertura:** Cada localidad debe estar cubierta por al menos un sensor o por un sensor en una localidad adyacente.

$$\sum_{s \in S} \left(Cov_{sl} \cdot x_{sl} + \sum_{l' \in L} Cov_{sl'} \cdot Adj_{ll'} \cdot x_{sl'} \right) \geq 1 \quad \forall l \in L$$

- **Restricción de necesidad específica:** Si una localidad necesita un sensor específico, este debe estar instalado en la localidad o en una localidad adyacente.

$$\sum_{l' \in L} Adj_{ll'} \cdot x_{sl'} + x_{sl} \geq 1 \quad \forall s \in S, \forall l \in L$$

Donde $Cov_{sl} = 1$ indica que la necesidad del sensor s en la localidad l es obligatoria.

Resultados sin modificar:

```
Variables:
x : Size=36, Index={S1, S2, S3}*{L1, L2, L3, L4, L5, L6, L7, L8, L9, L10, L11, L12}
Key : Lower : Value : Upper : Fixed : Stale : Domain
('S1', 'L1') : 0 : 0.0 : 1 : False : False : Binary
('S1', 'L10') : 0 : 0.0 : 1 : False : False : Binary
('S1', 'L11') : 0 : 1.0 : 1 : False : False : Binary
('S1', 'L12') : 0 : 0.0 : 1 : False : False : Binary
('S1', 'L2') : 0 : 1.0 : 1 : False : False : Binary
('S1', 'L3') : 0 : 0.0 : 1 : False : False : Binary
('S1', 'L4') : 0 : 0.0 : 1 : False : False : Binary
('S1', 'L5') : 0 : 0.0 : 1 : False : False : Binary
('S1', 'L6') : 0 : 0.0 : 1 : False : False : Binary
('S1', 'L7') : 0 : 0.0 : 1 : False : False : Binary
('S1', 'L8') : 0 : 1.0 : 1 : False : False : Binary
('S1', 'L9') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L1') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L10') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L11') : 0 : 1.0 : 1 : False : False : Binary
('S2', 'L12') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L2') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L3') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L4') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L5') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L6') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L7') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L8') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L9') : 0 : 0.0 : 1 : False : False : Binary
('S3', 'L1') : 0 : 0.0 : 1 : False : False : Binary
('S3', 'L10') : 0 : 0.0 : 1 : False : False : Binary
('S3', 'L11') : 0 : 0.0 : 1 : False : False : Binary
('S3', 'L12') : 0 : 0.0 : 1 : False : False : Binary
('S3', 'L2') : 0 : 1.0 : 1 : False : False : Binary
('S3', 'L3') : 0 : 0.0 : 1 : False : False : Binary
('S3', 'L4') : 0 : 0.0 : 1 : False : False : Binary
('S3', 'L5') : 0 : 0.0 : 1 : False : False : Binary
('S3', 'L6') : 0 : 0.0 : 1 : False : False : Binary
('S3', 'L7') : 0 : 0.0 : 1 : False : False : Binary
('S3', 'L8') : 0 : 1.0 : 1 : False : False : Binary
('S3', 'L9') : 0 : 0.0 : 1 : False : False : Binary

Objectives:
obj : Size=1, Index=None, Active=True
Key : Active : Value
None : True : 741.0
```

Resultados:

```
Sensor S1 se instala en ubicación L2
Sensor S1 se instala en ubicación L8
Sensor S1 se instala en ubicación L11
Sensor S2 se instala en ubicación L11
Sensor S3 se instala en ubicación L2
Sensor S3 se instala en ubicación L8
```

El modelo de optimización para la red de sensores minimiza los costos totales de instalación, consumo de energía y comunicación, al mismo tiempo que garantiza una cobertura eficiente de las zonas críticas. Los sensores se han instalado en ubicaciones estratégicas (L2, L8 y L11), aprovechando la adyacencia para cubrir áreas adicionales sin necesidad de incrementar el número de dispositivos. Con un costo total de 741 unidades, la solución equilibra de manera efectiva los recursos limitados, asegurando que se cumplan las restricciones de necesidad en localidades clave y maximizando la eficiencia de la red.

Resultados con la modificación:

```
Variables:
x : Size=36, Index={S1, S2, S3}*{L1, L2, L3, L4, L5, L6, L7, L8, L9, L10, L11, L12}
Key : Lower : Value : Upper : Fixed : Stale : Domain
('S1', 'L1') : 0 : 0.0 : 1 : False : False : Binary
('S1', 'L10') : 0 : 0.0 : 1 : False : False : Binary
('S1', 'L11') : 0 : 1.0 : 1 : False : False : Binary
('S1', 'L12') : 0 : 0.0 : 1 : False : False : Binary
('S1', 'L2') : 0 : 1.0 : 1 : False : False : Binary
('S1', 'L3') : 0 : 0.0 : 1 : False : False : Binary
('S1', 'L4') : 0 : 0.0 : 1 : False : False : Binary
('S1', 'L5') : 0 : 0.0 : 1 : False : False : Binary
('S1', 'L6') : 0 : 0.0 : 1 : False : False : Binary
('S1', 'L7') : 0 : 0.0 : 1 : False : False : Binary
('S1', 'L8') : 0 : 1.0 : 1 : False : False : Binary
('S1', 'L9') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L1') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L10') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L11') : 0 : 1.0 : 1 : False : False : Binary
('S2', 'L12') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L2') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L3') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L4') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L5') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L6') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L7') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L8') : 0 : 0.0 : 1 : False : False : Binary
('S2', 'L9') : 0 : 0.0 : 1 : False : False : Binary
('S3', 'L1') : 0 : 0.0 : 1 : False : False : Binary
('S3', 'L10') : 0 : 0.0 : 1 : False : False : Binary
('S3', 'L11') : 0 : 0.0 : 1 : False : False : Binary
('S3', 'L12') : 0 : 0.0 : 1 : False : False : Binary
('S3', 'L2') : 0 : 1.0 : 1 : False : False : Binary
('S3', 'L3') : 0 : 0.0 : 1 : False : False : Binary
('S3', 'L4') : 0 : 0.0 : 1 : False : False : Binary
('S3', 'L5') : 0 : 0.0 : 1 : False : False : Binary
('S3', 'L6') : 0 : 0.0 : 1 : False : False : Binary
('S3', 'L7') : 0 : 0.0 : 1 : False : False : Binary
('S3', 'L8') : 0 : 1.0 : 1 : False : False : Binary
('S3', 'L9') : 0 : 0.0 : 1 : False : False : Binary

Objectives:
obj : Size=1, Index=None, Active=True
Key : Active : Value
None : True : 700.0
```

Resultados:
 Sensor S1 se instala en ubicación L2
 Sensor S1 se instala en ubicación L8
 Sensor S1 se instala en ubicación L11
 Sensor S2 se instala en ubicación L11
 Sensor S3 se instala en ubicación L2
 Sensor S3 se instala en ubicación L8

Al modificar la función objetivo para enfocarse únicamente en los costos de instalación, el valor de la función objetivo se redujo de 741 a 700, lo que representa una disminución en los costos operativos. Sin embargo, las ubicaciones seleccionadas para los sensores permanecieron sin cambios: los sensores S1, S2 y S3 continúan instalados en L2, L8 y L11. Esto nos dice que estas ubicaciones son estratégicas para garantizar una cobertura eficiente.