Simulación y Optimización

Programación lineal
Parte II

Enrique E. Tarifa, Facultad de Ingeniería, UNJu

Mapa curricular de la materia

- Simulación
- Optimización

Mapa curricular de la materia

Simulación

Optimización

Mapa curricular de Optimización

- Definiciones
- Modelo de optimización
- Programación no lineal
- Programación lineal
- Análisis de sensibilidad

Mapa curricular de Optimización

- Definiciones
- Modelo de optimización
- Programación no lineal
- Programación lineal
- Análisis de sensibilidad

Mapa curricular de programación lineal

1. Programación lineal
2. Problemas LP clásicos
3. Problema de mezcla
4. Problema de mezcla con composición
5. Problema de transporte
6. Casos del problema de transporte
7. Problema de asignación
8. Casos del problema de asignación

Programación matemática

Programación matemática

$$\underset{x_j}{\text{Max}}\, FO\left(x_1, x_2, \ldots, x_n\right) \quad j = 1, 2, \ldots, n$$

s. a:

$$h_i\left(x_1, x_2, \ldots, x_n\right) = 0 \quad i = 1, 2, \ldots, m$$

$$g_k\left(x_1, x_2, \ldots, x_n\right) \leq 0 \quad k = 1, 2, \ldots, o$$

$$GL = n - m$$

Clasificación de modelos

- Programación Lineal (LP, *Linear Programming*)
- Programación No Lineal (NLP, *Nonlinear Programning*)
- Programación Lineal Entera (ILP, *Integer Linear Programming*)
- Programación No Lineal Entera (INLP, *Integer Nonlinear Programming*)
- Programación Lineal Entera Mixta (MILP, *Mixed Integer Linear Programming*)
- Programación No Lineal Entera Mixta (MINLP, *Mixed Integer Nonlinear Programming*)

$$FO\left(x_1, x_2, \ldots, x_n\right) = a_1 x_1 + a_2 x_2 + \ldots + a_n x_n$$

Programación Lineal

Programación lineal

$$\underset{x_j}{\text{Max}} \quad c_1 x_1 + c_2 x_2 + \ldots + c_j x_j + \ldots + c_n x_n$$

s. a:

$$a_{11} x_1 + a_{12} x_2 + \ldots + a_{1j} x_j + \ldots + a_{1n} x_n \le b_1$$

$$a_{21} x_1 + a_{22} x_2 + \ldots + a_{2j} x_j + \ldots + a_{2n} x_n \le b_2$$

$$\ldots$$

$$a_{i1} x_1 + a_{i2} x_2 + \ldots + a_{ij} x_j + \ldots + a_{in} x_{in} \le b_i$$

$$\ldots$$

$$a_{m1} x_1 + a_{m2} x_2 + \ldots + a_{mj} x_j + \ldots + a_{mn} x_{mn} \le b_m$$

$$x_j \ge 0 \quad j = 1, 2, \ldots, n$$

$$b_i \ge 0 \quad i = 1, 2, \ldots, m$$

$$GL = n$$

Programación lineal

$$x = \begin{pmatrix} x_1 \\ x_2 \\ \cdots \\ x_n \end{pmatrix} \qquad c = \begin{pmatrix} c_1 \\ c_2 \\ \cdots \\ c_n \end{pmatrix} \qquad b = \begin{pmatrix} b_1 \\ b_2 \\ \cdots \\ b_m \end{pmatrix} \qquad A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix}$$

Programación lineal

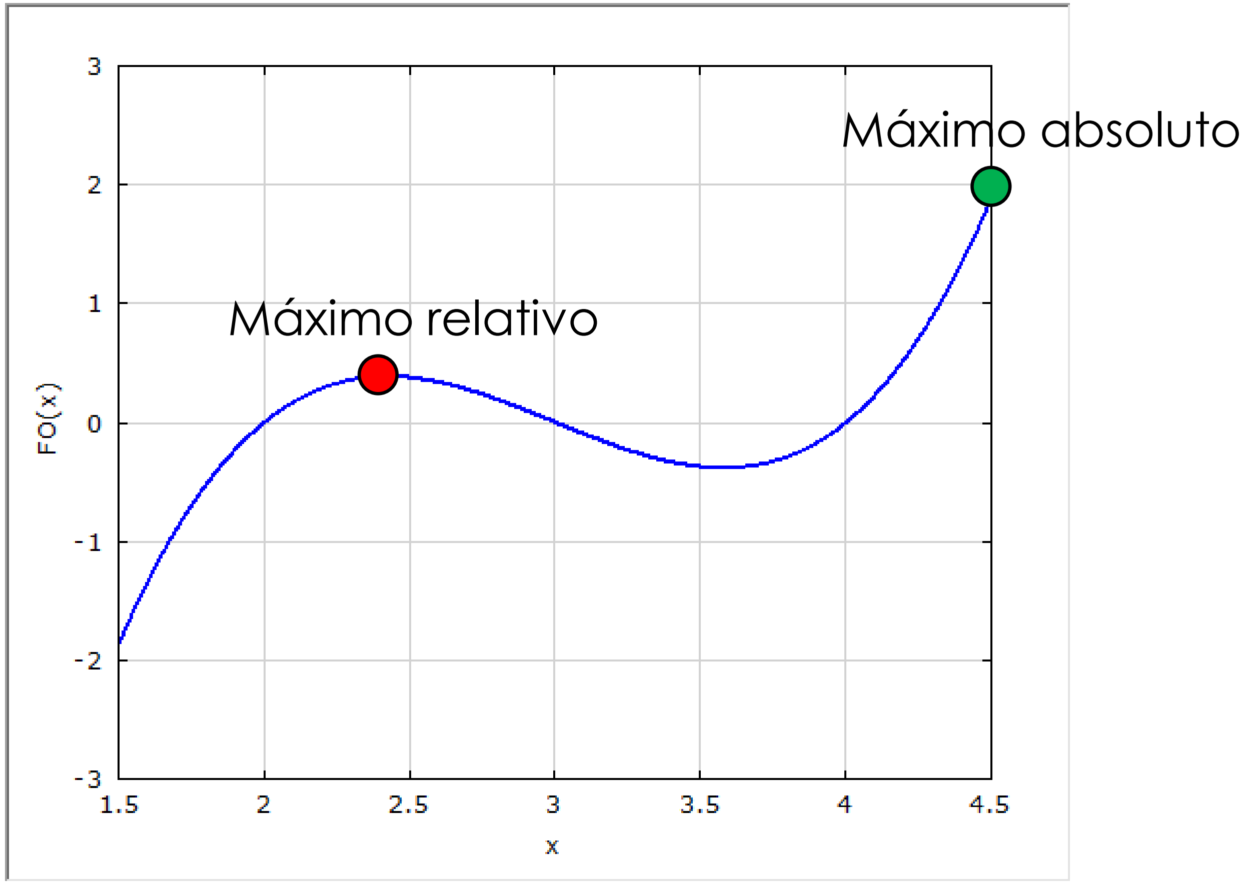
$$\underset{x}{\text{Max}}\; c^{\mathrm{T}} x$$

s. a:

$$A x \leq b$$

$$x \geq 0 \quad b \geq 0$$

Modelo no lineal



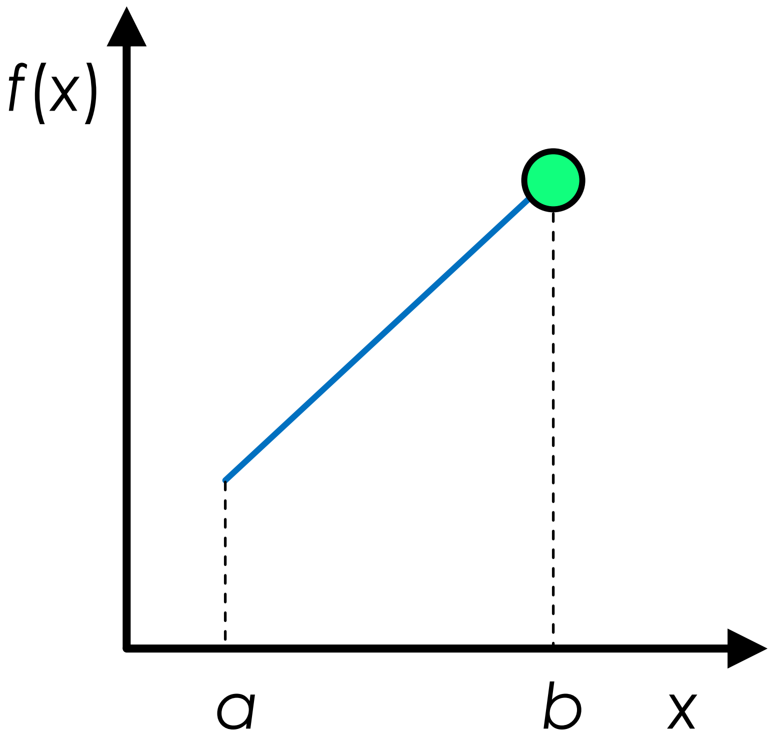
Programación lineal

$$\underset{x}{\text{Max}}\ f(x)$$

s. a:

$$x \geq a$$

$$x \leq b$$



- El óptimo es global.
- Es único.
- Está en un extremo.

Problemas LP clásicos

Problemas LP clásicos

- Problema de Producción: Determina los niveles de actividades que compiten por recursos con el fin de maximizar los beneficios.
- Problema de Mezcla: Determina la combinación óptima de ingredientes en una mezcla para cumplir con ciertos requerimientos al menor costo posible.

Problemas LP clásicos

- Problema de Transporte: Optimiza las rutas de distribución de mercancías desde varios orígenes a varios destinos para minimizar los costos de envío.
- Problema de Asignación: Distribuye eficientemente recursos limitados (personal, maquinarias, presupuestos) para maximizar los beneficios.

Problema de mezcla

Problema de mezcla

Se desea producir una cantidad q (kg) de un producto mezclando n ingredientes considerando m propiedades de la mezcla. La cantidad a utilizar de cada ingrediente j es x_j (kg). El costo unitario del ingrediente j es c_j ($/kg). A su vez, el contenido específico de la propiedad i en el ingrediente j es w_{ij}. El contenido específico w_i^{f} de la propiedad i en la mezcla debe respetar especificaciones de contenido mínimo $w_i^{\min}$ y máximo $w_i^{\max}$. El problema consiste en determinar los valores de x_j que minimicen el costo.

Elementos del modelo

- Índices:
  - Para propiedades: $i = 1, 2, \ \dots, m.$
  - Para ingredientes: $j = 1, 2, \ \dots, n.$

Elementos del modelo

- Parámetros (en total, $3 + n + m\,n + 2\,m$):
  - m: Tipos de propiedades.
  - n: Tipos de ingredientes.
  - q: Cantidad de producto (kg).
  - c_j: Costo unitario del ingrediente j (\$/kg).
  - w_{ij}: Contenido específico de la propiedad i en el ingrediente j.
  - $w_i^{\min}$: Cota mínima de la propiedad i en el producto.
  - $w_i^{\max}$: Cota máxima de la propiedad i en el producto.

Elementos del modelo

- Variables de decisión (en total, $m + n$):
  - w_i^{f}: Contenido específico de la propiedad i en el producto.
  - x_j: Cantidad del ingrediente j en el producto (kg).

Elementos del modelo

- Función objetivo:
  - Costo (\$): $\sum_{j=1}^{n} c_j \, x_j$
- Restricciones (en total, $1 + 3\,m$ y las de no negatividad):
  - Balance de materia (kg): $\sum_{j=1}^{n} x_j = q$
  - Balance de propiedad: $\sum_{j=1}^{n} w_{ij} \; x_j = w_i^{\mathrm{f}} \, q$
  - Requerimientos de mínimos: $w_i^{\mathrm{f}} \geq w_i^{\min}$
  - Requerimientos de máximos: $w_i^{\mathrm{f}} \leq w_i^{\max}$

Problema de mezcla

$$\operatorname*{Min}_{w_i^{\mathrm{f}},x_j} \sum_{j=1}^{n} c_j\, x_j$$

s. a:

$$\sum_{j=1}^{n} x_j = q$$

$$\sum_{j=1}^{n} w_{ij}\, x_j = w_i^{\mathrm{f}}\, q \qquad i = 1,2,\ldots,m$$

$$w_i^{\mathrm{f}} \geq w_i^{\min} \qquad i = 1,2,\ldots,m$$

$$w_i^{\mathrm{f}} \leq w_i^{\max} \qquad i = 1,2,\ldots,m$$

$$w_i^{\mathrm{f}} \geq 0 \qquad i = 1,2,\ldots,m$$

$$x_j \geq 0 \qquad j = 1,2,\ldots,n$$

Eliminando w_i^{f} $\Longrightarrow$

$$\operatorname*{Min}_{x_j} \sum_{j=1}^{n} c_j\, x_j$$

s. a:

$$\sum_{j=1}^{n} x_j = q$$

$$\sum_{j=1}^{n} w_{ij}\, x_j \geq w_i^{\min}\, q \qquad i = 1,2,\ldots,m$$

$$\sum_{j=1}^{n} w_{ij}\, x_j \leq w_i^{\max}\, q \qquad i = 1,2,\ldots,m$$

$$x_j \geq 0 \qquad j = 1,2,\ldots,n$$

$$GL = n - 1$$

Problema de mezcla con composición

Problema de mezcla con composición

Se desea producir un producto mezclando n ingredientes considerando m propiedades de la mezcla. La fracción peso de cada ingrediente j es x_j. El costo unitario del ingrediente j es c_j ($/kg). A su vez, el contenido específico de la propiedad i en el ingrediente j es w_{ij}. El contenido específico w_i^{f} de la propiedad i en la mezcla debe respetar especificaciones de contenido mínimo $w_i^{\min}$ y máximo $w_i^{\max}$. El problema consiste en determinar los valores de x_j que minimicen el costo.

Es el modelo anterior con $q = 1$.

Elementos del modelo

- Índices:
  - Para propiedades: $i = 1, 2, \ldots, m$.
  - Para ingredientes: $j = 1, 2, \ldots, n$.

Elementos del modelo

- Parámetros (en total, $2 + n + m\,n + 2\,m$):
  - m: Tipos de propiedades.
  - n: Tipos de ingredientes.
  - c_j: Costo unitario del ingrediente j (\$/kg).
  - w_{ij}: Contenido específico de la propiedad i en el ingrediente j.
  - $w_i^{\min}$: Cota mínima de la propiedad i en el producto.
  - $w_i^{\max}$: Cota máxima de la propiedad i en el producto.

Elementos del modelo

- Variables de decisión (en total, $m + n$):
  - w_i^{f}: Contenido específico de la propiedad i en el producto.
  - x_j: Fracción peso del ingrediente j en el producto.

Elementos del modelo

- Función objetivo:
  - Costo (\$/kg): $\sum_{j=1}^{n} c_j \, x_j$
- Restricciones (en total, $1 + 3\,m$ y las de no negatividad):
  - Balance de materia: $\sum_{j=1}^{n} x_j = 1$
  - Balance de propiedad: $\sum_{j=1}^{n} w_{ij} \; x_j = w_i^{\mathrm{f}}$
  - Requerimientos de mínimos: $w_i^{\mathrm{f}} \geq w_i^{\min}$
  - Requerimientos de máximos: $w_i^{\mathrm{f}} \leq w_i^{\max}$

Problema de mezcla

$$\underset{w_i^{\mathrm{f}},x_j}{\mathrm{Min}} \sum_{j=1}^{n} c_j\, x_j$$

s. a:

$$\sum_{j=1}^{n} x_j = 1$$

$$\sum_{j=1}^{n} w_{ij}\, x_j = w_i^{\mathrm{f}} \qquad i = 1, 2, \ldots, m$$

$$w_i^{\mathrm{f}} \geq w_i^{\min} \qquad i = 1, 2, \ldots, m$$

$$w_i^{\mathrm{f}} \leq w_i^{\max} \qquad i = 1, 2, \ldots, m$$

$$w_i^{\mathrm{f}} \geq 0 \qquad i = 1, 2, \ldots, m$$

$$x_j \geq 0 \qquad j = 1, 2, \ldots, n$$

Eliminando w_i^f →

$$\underset{x_j}{\mathrm{Min}} \sum_{j=1}^{n} c_j\, x_j$$

s. a:

$$\sum_{j=1}^{n} x_j = 1$$

$$\sum_{j=1}^{n} w_{ij}\, x_j \geq w_i^{\min} \qquad i = 1, 2, \ldots, m$$

$$\sum_{j=1}^{n} w_{ij}\, x_j \leq w_i^{\max} \qquad i = 1, 2, \ldots, m$$

$$x_j \geq 0 \qquad j = 1, 2, \ldots, n$$

$$GL = n - 1$$

Alimento balanceado

Alimento balanceado

En una planta productora de alimento balanceado, la formulación de un tipo de alimento requiere dos ingredientes activos y un soporte para dar volumen. El costo del soporte es despreciable. El producto final debe reunir los requerimientos solicitados en la Tabla 1. El contenido nutricional de los ingredientes y sus costos se especifican en la Tabla 2.

Alimento balanceado

Tabla 1: Contenido específico mínimo de nutrientes en el producto (g/kg)

Nutriente	A	B	C	D
Cantidad	90	50	20	2

Tabla 2: Concentración de nutrientes (g/kg) y costo ($/kg) de los ingredientes.

	A	B	C	D	Costo
Ingrediente 1	100	80	40	10	0.4
Ingrediente 2	200	150	20	0	0.6
Soporte	0	0	0	0	0

Elementos del modelo

- Parámetros:
  - $m = 4$, nutrientes
  - n = 3, ingredientes
  - $w_i^{\min}$: Cota mínima del nutriente i en el producto (Tabla 1).
  - w_{ij}: Contenido específico del nutriente i en el ingrediente j (Tabla 2),
  - c_j: Costo unitario del ingrediente j (Tabla 2).

Elementos del modelo

- Variables de decisión:
  - x_1: Fracción peso del ingrediente 1 en el producto.
  - x_2: Fracción peso del ingrediente 2 en el producto.
  - x_3: Fracción peso del soporte en el producto.

Elementos del modelo

- Función objetivo:
  - Costo ($/kg): $0.4\,x_1 + 0.6\,x_2$
- Restricciones:
  - Balance de materia: $x_1 + x_2 + x_3 = 1$
  - Concentración de nutriente A (g/kg): $100\,x_1 + 200\,x_2 \geq 90$
  - Concentración de nutriente B (g/kg): $80\,x_1 + 150\,x_2 \geq 50$
  - Concentración de nutriente C (g/kg): $40\,x_1 + 20\,x_2 \geq 20$
  - Concentración de nutriente D (g/kg): $10\,x_1 \geq 2$

Alimento balanceado

$$\underset{x_j}{\text{Min}} \sum_{j=1}^{n} c_j \, x_j$$

s. a:

$$\sum_{j=1}^{n} x_j = 1$$

$$\sum_{j=1}^{n} w_{ij} \, x_j \geq w_i^{\min} \qquad \forall i$$

$$x_j \geq 0 \qquad \forall j$$

$$\underset{x_1, x_2, x_3}{\text{Min}} \; 0.4 \, x_1 + 0.6 \, x_2$$

s.a:

$$x_1 + x_2 + x_3 = 1$$

$$100 \, x_1 + 200 \, x_2 \geq 90$$

$$80 \, x_1 + 150 \, x_2 \geq 50$$

$$40 \, x_1 + 20 \, x_2 \geq 20$$

$$10 \, x_1 \geq 2$$

x_1, x_2, x_3 no negativas

Alimento balanceado.lg4

Modelo en LINGO

Lingo Model - Alimento balanceado

```
! Alimento balanceado;
[FO] Min = 0.4*x1+0.6*x2;

[RM] x1+x2+x3 = 1;
[RA] 100*x1+200*x2 >= 90;
[RB] 80*x1+150*x2 >= 50;
[RC] 40*x1+20*x2 >= 20;
[RD] 10*x1 >= 2;
```

Resultados en LINGO

Variable	Value	Reduced Cost
X1	0.3666667	0.000000
X2	0.2666667	0.000000
X3	0.3666667	0.000000

Row	Slack or Surplus	Dual Price
FO	0.3066667	-1.000000
RM	0.000000	0.000000
RA	0.000000	-0.2666667E-02
RB	19.33333	0.000000
RC	0.000000	-0.3333333E-02
RD	1.666667	0.000000

Problema de transporte

Problema de transporte

Se debe distribuir un bien desde *m* centros de abastecimiento, llamados orígenes, a *n* centros de recepción, llamados destinos. El origen *i* dispone de s_i unidades para distribuir, y el destino *j* tiene una demanda de d_j unidades que debe ser satisfecha por los orígenes. El costo para enviar una unidad desde el origen *i* al destino *j* es c_{ij}. Se desea determinar la cantidad de unidades x_{ij} que se deben enviar desde cada origen *i* a cada destino *j* para que el costo de transporte sea mínimo.

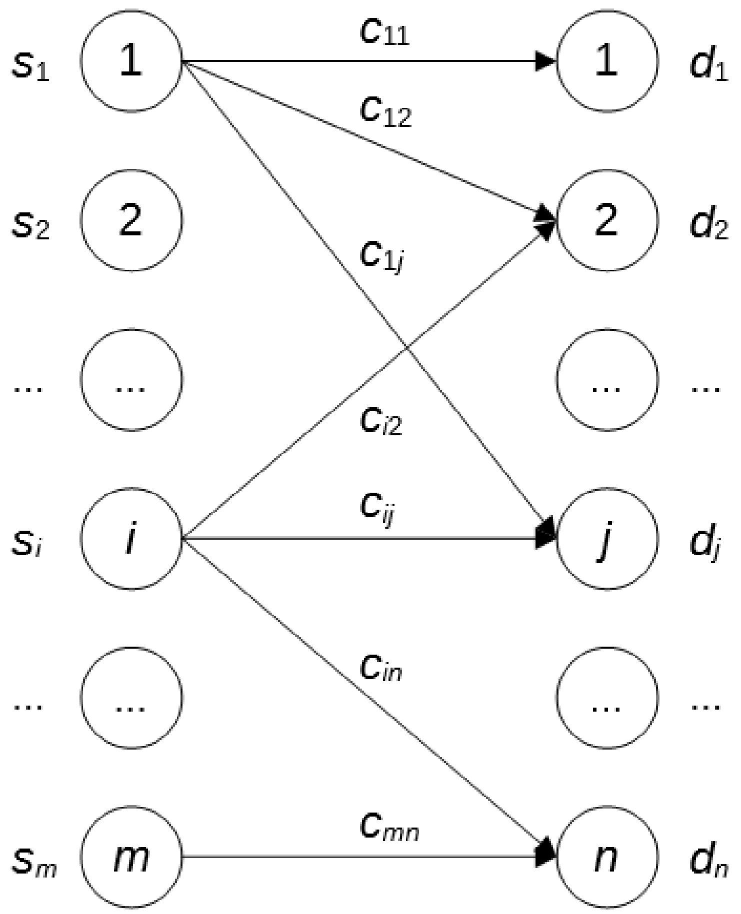
Elementos del modelo

- Índices:
  - Para orígenes: $i = 1, 2, \ldots, m$.
  - Para destinos: $j = 1, 2, \ldots, n$.

Elementos del modelo

- Parámetros (en total, $2 + m + n + m\,n$):
  - m: Cantidad de orígenes.
  - n: Cantidad de destinos.
  - s_i: Disponibilidad en el origen i.
  - d_j: Demanda del destino j.
  - c_{ij}: Costo unitario de transporte de i a j.
- Variables de decisión (en total, $m\,n$):
  - x_{ij}: Cantidad enviada de i a j.

Problema de transporte



x_{ij}

Casos del problema de transporte

1. Oferta igual a la demanda: $\sum_{i=1}^{m} s_i = \sum_{j=1}^{n} d_j$.
2. Oferta mayor que la demanda: $\sum_{i=1}^{m} s_i > \sum_{j=1}^{n} d_j$.
3. Oferta menor que la demanda: $\sum_{i=1}^{m} s_i < \sum_{j=1}^{n} d_j$.

Elementos del modelo

- Oferta igual a la demanda: $\sum_{i=1}^{m} s_i = \sum_{j=1}^{n} d_j$.
- Función objetivo:
  - Costo ($): $\sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij}\, x_{ij}$.
- Restricciones (en total, $m + n$ y las de no negatividad):
  - Origen *i* (caja): $\sum_{j=1}^{n} x_{ij} = s_i$.
  - Destino *j* (caja): $\sum_{i=1}^{m} x_{ij} = d_j$.

Problema de transporte. Caso 1

$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$$

s. a:

$$\sum_{j=1}^{n} x_{ij} = s_i \quad \forall i$$

$$\sum_{i=1}^{m} x_{ij} = d_j \quad \forall j$$

$$x_{ij} \geq 0 \quad \forall i, j$$

$$\sum_{i=1}^{m} s_i = \sum_{j=1}^{n} d_j$$

$$GL = m\,n - (m + n - 1)$$
$$= (m-1)(n-1)$$

Problema de transporte. Caso 2

$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$$

s. a:

$$\sum_{j=1}^{n} x_{ij} \leq s_i \quad \forall i$$

$$\sum_{i=1}^{m} s_i > \sum_{j=1}^{n} d_j$$

$$\sum_{i=1}^{m} x_{ij} = d_j \quad \forall j$$

$$x_{ij} \geq 0 \quad \forall i, j$$

Problema de transporte. Caso 3

$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$$

s. a:

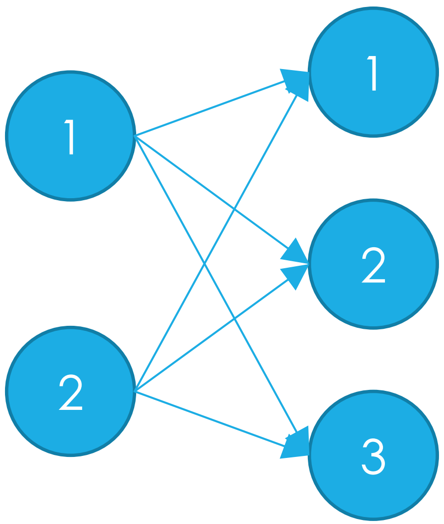
$$\sum_{j=1}^{n} x_{ij} = s_i \quad \forall i$$

$$\sum_{i=1}^{m} s_i < \sum_{j=1}^{n} d_j$$

$$\sum_{i=1}^{m} x_{ij} \le d_j \quad \forall j$$

$$x_{ij} \ge 0 \quad \forall i, j$$

Ejemplo de 2x3



$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{2}\sum_{j=1}^{3} c_{ij}\, x_{ij} = c_{1,1}\,x_{1,1} + c_{1,2}\,x_{1,2} + c_{1,3}\,x_{1,3}$$

$$+c_{2,1}\,x_{2,1} + c_{2,2}\,x_{2,2} + c_{2,3}\,x_{2,3}$$

s. a:

$$x_{1,1} + x_{1,2} + x_{1,3} = s_1$$

$$x_{2,1} + x_{2,2} + x_{2,3} = s_2$$

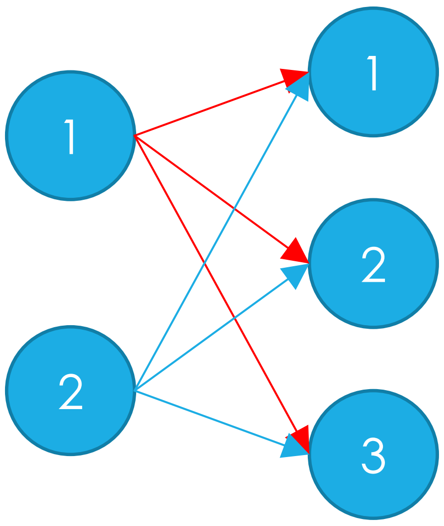
$$x_{1,1} + x_{2,1} = d_1$$

$$x_{1,2} + x_{2,2} = d_2$$

$$x_{1,3} + x_{2,3} = d_3$$

$$x_{ij} \geq 0 \quad \forall i, j$$

Ejemplo de 2x3



$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{2}\sum_{j=1}^{3} c_{ij}x_{ij} = c_{1,1}x_{1,1} + c_{1,2}x_{1,2} + c_{1,3}x_{1,3}$$

$$+c_{2,1}x_{2,1} + c_{2,2}x_{2,2} + c_{2,3}x_{2,3}$$

s. a:

$$x_{1,1} + x_{1,2} + x_{1,3} = s_1$$

$$x_{2,1} + x_{2,2} + x_{2,3} = s_2$$

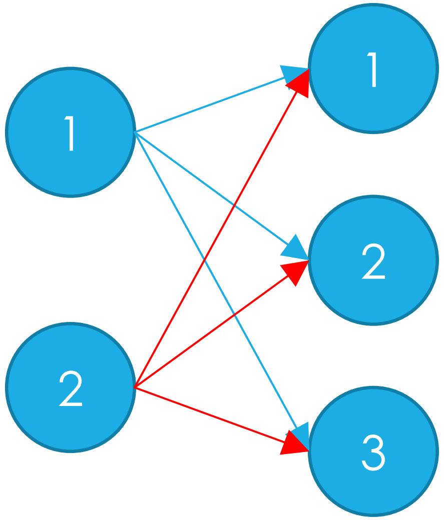
$$x_{1,1} + x_{2,1} = d_1$$

$$x_{1,2} + x_{2,2} = d_2$$

$$x_{1,3} + x_{2,3} = d_3$$

$$x_{ij} \geq 0 \quad \forall i, j$$

Ejemplo de 2x3



$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{2} \sum_{j=1}^{3} c_{ij} x_{ij} = c_{1,1} x_{1,1} + c_{1,2} x_{1,2} + c_{1,3} x_{1,3}$$

$$+ c_{2,1} x_{2,1} + c_{2,2} x_{2,2} + c_{2,3} x_{2,3}$$

s. a:

$$x_{1,1} + x_{1,2} + x_{1,3} = s_1$$

$$x_{2,1} + x_{2,2} + x_{2,3} = s_2$$

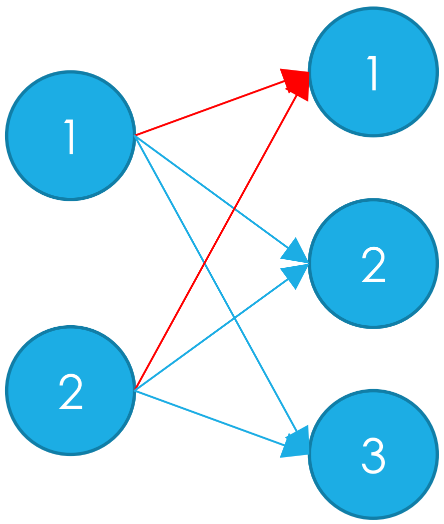
$$x_{1,1} + x_{2,1} = d_1$$

$$x_{1,2} + x_{2,2} = d_2$$

$$x_{1,3} + x_{2,3} = d_3$$

$$x_{ij} \geq 0 \quad \forall i, j$$

Ejemplo de 2x3



$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{2}\sum_{j=1}^{3} c_{ij}x_{ij} = c_{1,1}x_{1,1} + c_{1,2}x_{1,2} + c_{1,3}x_{1,3}$$

$$+c_{2,1}x_{2,1} + c_{2,2}x_{2,2} + c_{2,3}x_{2,3}$$

s. a:

$$x_{1,1} + x_{1,2} + x_{1,3} = s_1$$

$$x_{2,1} + x_{2,2} + x_{2,3} = s_2$$

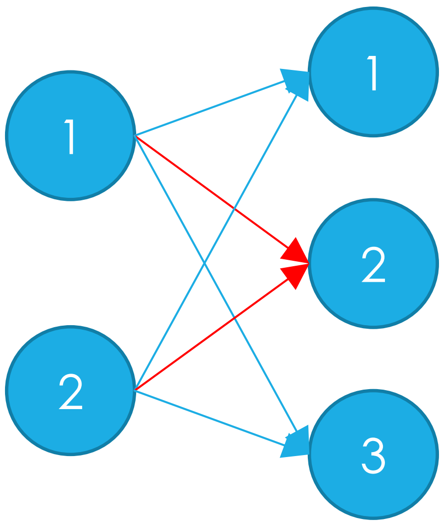
$$x_{1,1} + x_{2,1} = d_1$$

$$x_{1,2} + x_{2,2} = d_2$$

$$x_{1,3} + x_{2,3} = d_3$$

$$x_{ij} \geq 0 \quad \forall i, j$$

Ejemplo de 2x3



$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{2}\sum_{j=1}^{3} c_{ij}\,x_{ij} = c_{1,1}\,x_{1,1} + c_{1,2}\,x_{1,2} + c_{1,3}\,x_{1,3}$$

$$+c_{2,1}\,x_{2,1} + c_{2,2}\,x_{2,2} + c_{2,3}\,x_{2,3}$$

s. a:

$$x_{1,1} + x_{1,2} + x_{1,3} = s_1$$

$$x_{2,1} + x_{2,2} + x_{2,3} = s_2$$

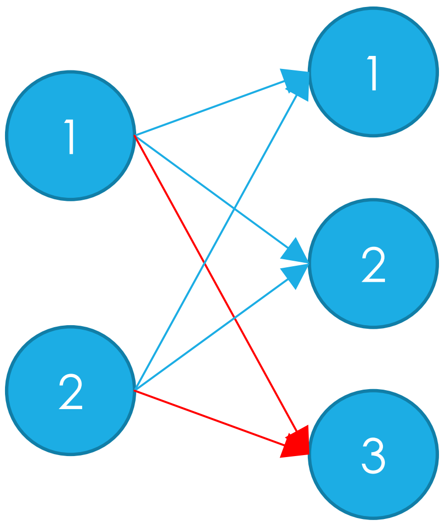
$$x_{1,1} + x_{2,1} = d_1$$

$$x_{1,2} + x_{2,2} = d_2$$

$$x_{1,3} + x_{2,3} = d_3$$

$$x_{ij} \geq 0 \quad \forall i, j$$

Ejemplo de 2x3



$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{2}\sum_{j=1}^{3} c_{ij}x_{ij} = c_{1,1}x_{1,1} + c_{1,2}x_{1,2} + c_{1,3}x_{1,3}$$

$$+c_{2,1}x_{2,1} + c_{2,2}x_{2,2} + c_{2,3}x_{2,3}$$

s. a:

$$x_{1,1} + x_{1,2} + x_{1,3} = s_1$$

$$x_{2,1} + x_{2,2} + x_{2,3} = s_2$$

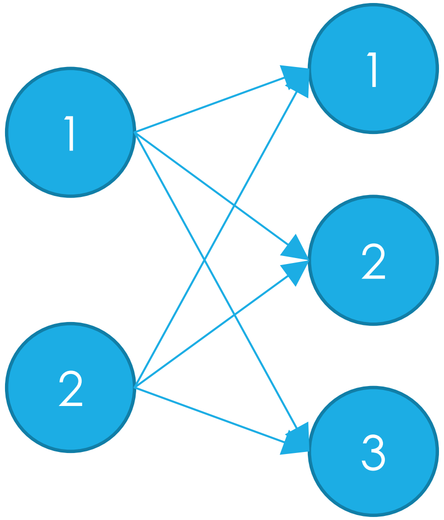
$$x_{1,1} + x_{2,1} = d_1$$

$$x_{1,2} + x_{2,2} = d_2$$

$$x_{1,3} + x_{2,3} = d_3$$

$$x_{ij} \geq 0 \quad \forall i, j$$

Ejemplo de 2x3



$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{2}\sum_{j=1}^{3} c_{ij}\,x_{ij} = c_{1,1}x_{1,1} + c_{1,2}x_{1,2} + c_{1,3}x_{1,3}$$

$$+c_{2,1}x_{2,1} + c_{2,2}x_{2,2} + c_{2,3}x_{2,3}$$

s. a:

$$x_{1,1} + x_{1,2} + x_{1,3} = s_1$$

$$x_{2,1} + x_{2,2} + x_{2,3} = s_2$$

$$x_{1,1} + x_{2,1} = d_1$$

$$x_{1,2} + x_{2,2} = d_2$$

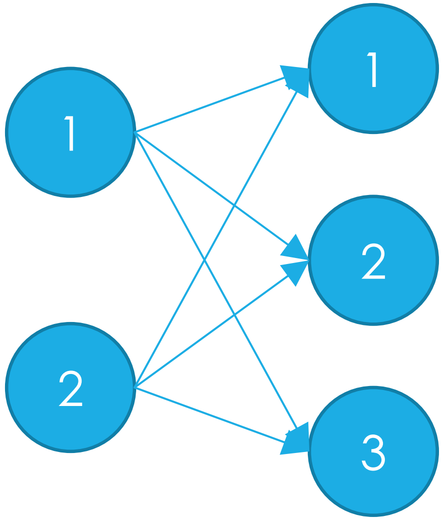
$$x_{1,3} + x_{2,3} = d_3$$

$$x_{ij} \geq 0 \quad \forall i, j$$

Total unimodularidad

- Una matriz es totalmente unimodular si todas sus submatrices cuadradas tienen determinante 0, 1 o -1.
- Si un problema LP tiene una matriz de restricciones totalmente unimodular y un vector de términos independientes enteros, cualquier vértice del poliedro de soluciones será entero.
- Por lo tanto, si los s_i y los d_j son enteros, $x_{ij} \in \mathbb{N}\ \forall i, \forall j$.

Ejemplo de 2x3



$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{2}\sum_{j=1}^{3} c_{ij}\,x_{ij} = c_{1,1}\,x_{1,1} + c_{1,2}\,x_{1,2} + c_{1,3}\,x_{1,3}$$

$$+c_{2,1}\,x_{2,1} + c_{2,2}\,x_{2,2} + c_{2,3}\,x_{2,3}$$

s. a:

$$x_{1,1} + x_{1,2} + x_{1,3} = s_1$$

$$x_{2,1} + x_{2,2} + x_{2,3} = s_2$$

$$x_{1,1} + x_{2,1} = d_1$$

$$x_{1,2} + x_{2,2} = d_2$$

$$x_{1,3} + x_{2,3} = d_3$$

$$x_{ij} \geq 0 \quad \forall i, j$$

Distribuidora de insumos

Distribuidora de insumos

Una distribuidora de insumos debe abastecer a 8 fábricas. Para ello, cuenta con 6 depósitos estratégicamente ubicados.

Distribuidora de insumos

Depósito	Disponibilidad (caja)
DE1	60
DE2	55
DE3	51
DE4	43
DE5	41
DE6	52
Total	**302**

Fábricas	Demanda (caja)
FA1	35
FA2	37
FA3	22
FA4	32
FA5	41
FA6	32
FA7	43
FA8	38
Total	**280**

Distribuidora de insumos

Costo de transporte ($/caja)

	FA1	FA2	FA3	FA4	FA5	FA6	FA7	FA8
DE1	6	2	6	7	4	2	5	9
DE2	4	9	5	3	8	5	8	2
DE3	5	2	1	9	7	4	3	3
DE4	7	6	7	3	9	2	7	1
DE5	2	3	9	5	7	2	6	5
DE6	5	5	2	2	8	1	4	3

Distribuidora de insumos.lg4

Distribuidora de insumos

$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}\, x_{ij}$$

s. a:

$$\sum_{j=1}^{n} x_{ij} \le s_i \quad \forall i$$

$$\sum_{i=1}^{m} s_i > \sum_{j=1}^{n} d_j$$

$$\sum_{i=1}^{m} x_{ij} = d_j \quad \forall j$$

$$x_{ij} \ge 0 \quad \forall i, j$$

```
!Distribuidora de insumos;
SETS:
  DEPOSITOS/DE1..DE6/: S;
  FABRICAS/FA1..FA8/: D;
  DXF(DEPOSITOS,FABRICAS): C,X;
ENDSETS

!Los datos;
DATA:
  S = 60 55 51 43 41 52;
  D = 35 37 22 32 41 32 43 38;
  C = 6 2 6 7 4 2 5 9
      4 9 5 3 8 5 8 2
      5 2 1 9 7 4 3 3
      7 6 7 3 9 2 7 1
      2 3 9 5 7 2 6 5
      5 5 2 2 8 1 4 3;
ENDDATA

!Función objetivo;
[FO] MIN = @SUM(DXF: C*X);

!Restricciones de orígenes;
@FOR(DEPOSITOS(I):
[RO] @SUM(FABRICAS(J): X(I,J)) <= S(I)
);

!Restricciones de destinos;
@FOR(FABRICAS(J):
[RD] @SUM(DEPOSITOS(I): X(I,J)) = D(J)
);
```

Modelo en LINGO

$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$$

s. a:

$$\sum_{j=1}^{n} x_{ij} \leq s_i \quad \forall i$$

$$\sum_{i=1}^{m} s_i > \sum_{j=1}^{n} d_j$$

$$\sum_{i=1}^{m} x_{ij} = d_j \quad \forall j$$

$$x_{ij} \geq 0 \quad \forall i, j$$

Lingo Model - Distribuidora de insumos

```
! Distribuidora de insumos;
SETS:
  DEPOSITOS/DE1..DE6/: S;
  FABRICAS/FA1..FA8/: D;
  DXF(DEPOSITOS,FABRICAS): C,X;
ENDSETS

! Los datos;
DATA:
  S = 60 55 51 43 41 52;
  D = 35 37 22 32 41 32 43 38;
  C = 6 2 6 7 4 2 5 9
      4 9 5 3 8 5 8 2
      5 2 1 9 7 4 3 3
      7 6 7 3 9 2 7 1
      2 3 9 5 7 2 6 5
      5 5 2 2 8 1 4 3;
ENDDATA

! Función objetivo;
[FO] MIN = @SUM(DXF: C*X);

! Restricciones de orígenes;
@FOR(DEPOSITOS(I):
  [RO] @SUM(FABRICAS(J): X(I,J)) <= S(I)
);

! Restricciones de destinos;
@FOR(FABRICAS(J):
  [RD] @SUM(DEPOSITOS(I): X(I,J)) = D(J)
);
```

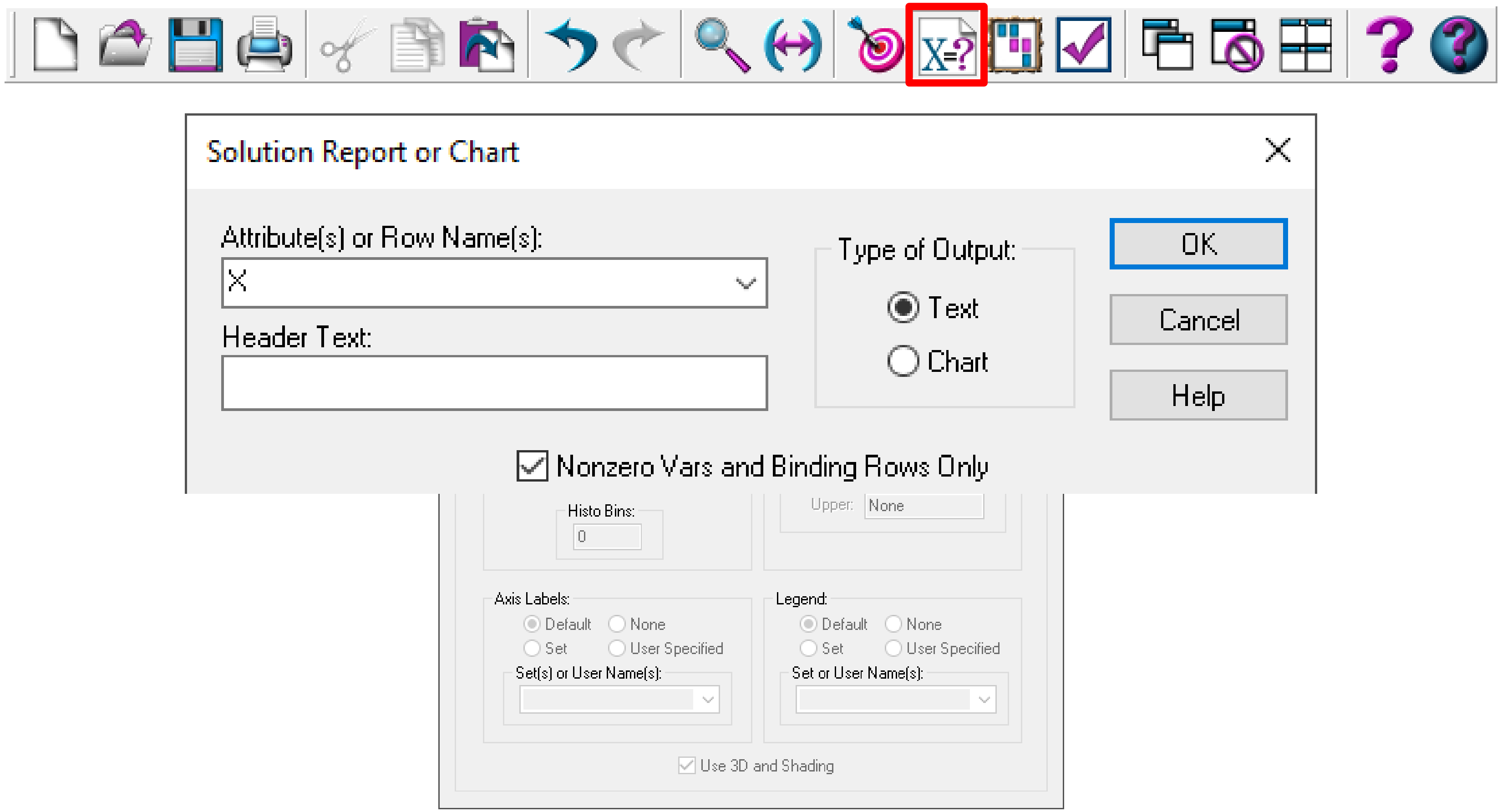
Resultados en LINGO

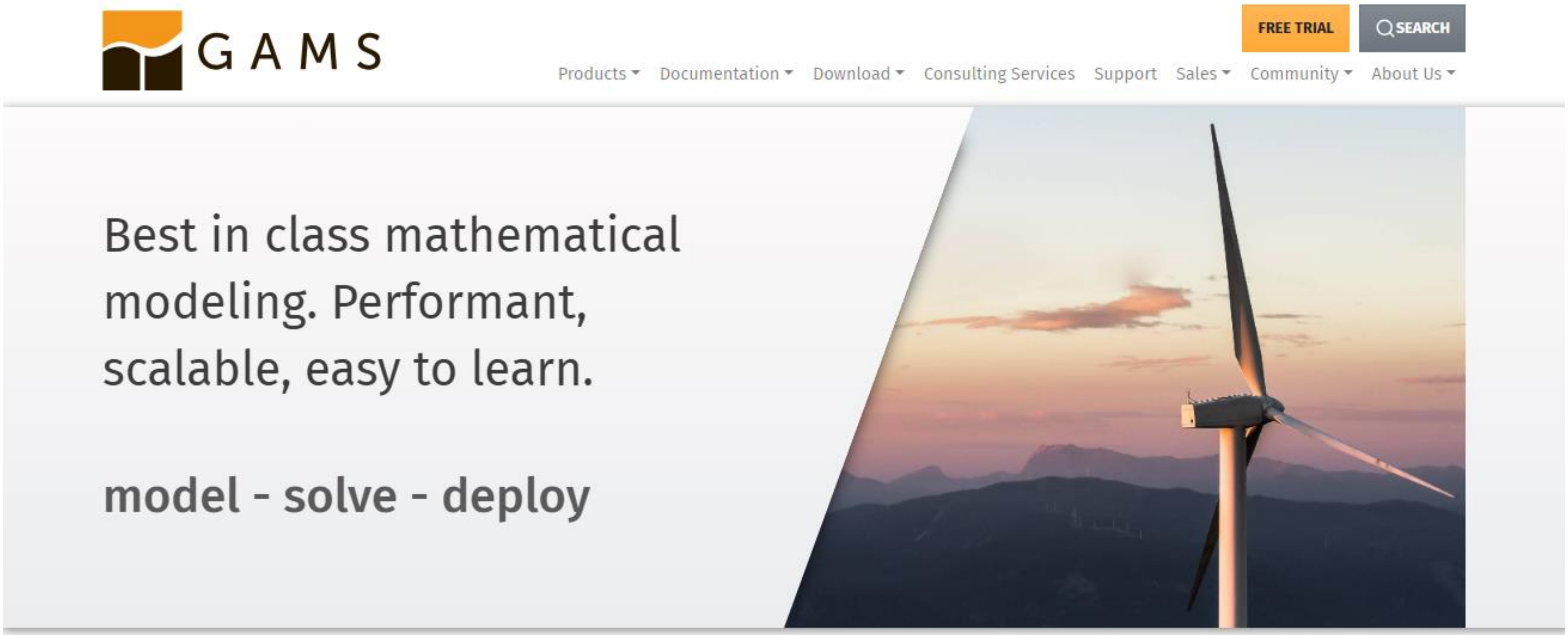
Variable	Value	Reduced Cost
X(DE1, FA2)	19.00000	0.000000
X(DE1, FA5)	41.00000	0.000000
X(DE2, FA1)	1.000000	0.000000
X(DE2, FA4)	32.00000	0.000000
X(DE3, FA2)	11.00000	0.000000
X(DE3, FA7)	40.00000	0.000000
X(DE4, FA6)	5.000000	0.000000
X(DE4, FA8)	38.00000	0.000000
X(DE5, FA1)	34.00000	0.000000
X(DE5, FA2)	7.000000	0.000000
X(DE6, FA3)	22.00000	0.000000
X(DE6, FA6)	27.00000	0.000000
X(DE6, FA7)	3.000000	0.000000

Resultados en LINGO

Row	Slack or Surplus	Dual Price
FO	664.0000	-1.000000
RO(DE1)	0.000000	3.000000
RO(DE2)	22.00000	0.000000
RO(DE3)	0.000000	3.000000
RO(DE4)	0.000000	1.000000
RO(DE5)	0.000000	2.000000
RO(DE6)	0.000000	2.000000
RD(FA1)	0.000000	-4.000000
RD(FA2)	0.000000	-5.000000
RD(FA3)	0.000000	-4.000000
RD(FA4)	0.000000	-3.000000
RD(FA5)	0.000000	-7.000000
RD(FA6)	0.000000	-3.000000
RD(FA7)	0.000000	-6.000000
RD(FA8)	0.000000	-2.000000

Resultados en LINGO



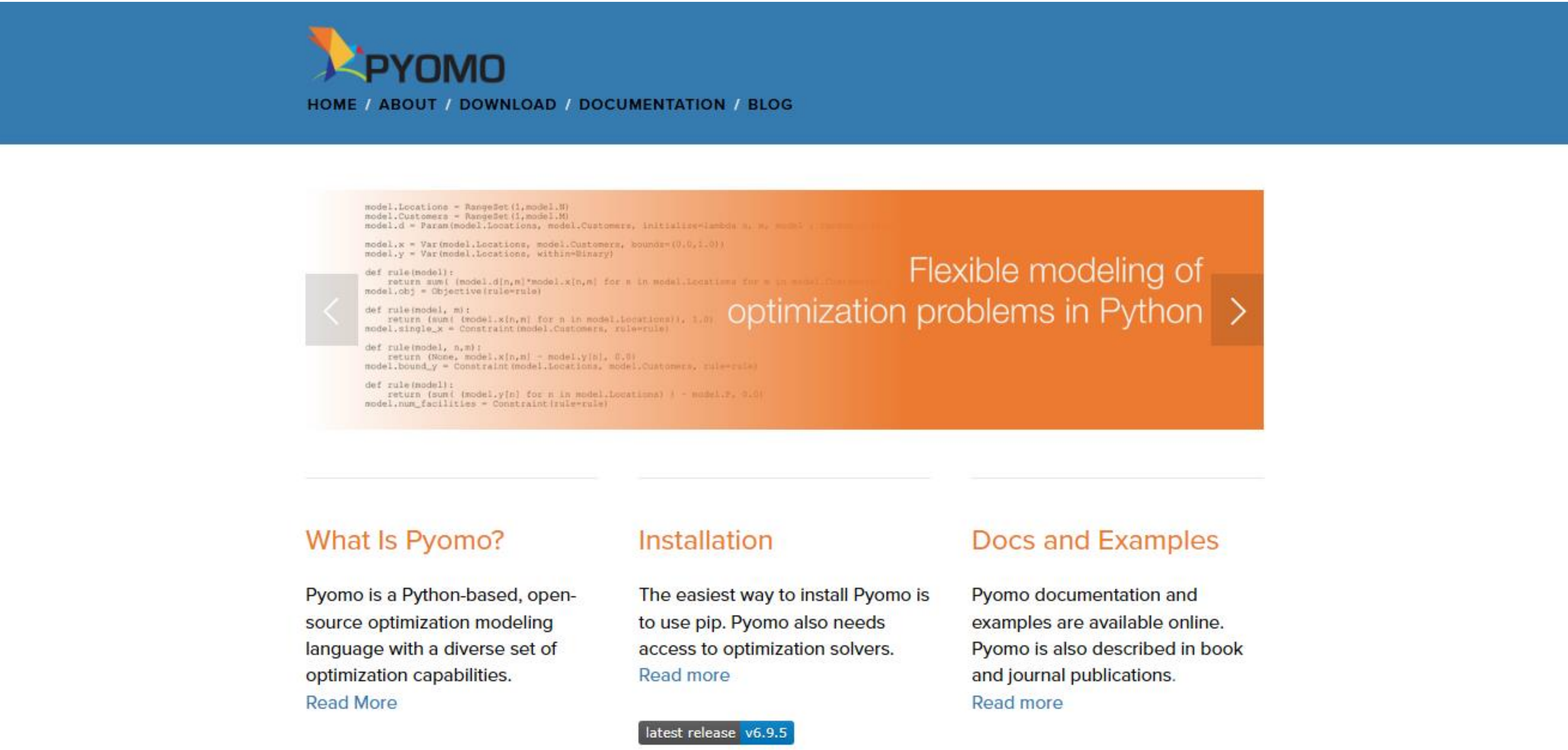

GAMS
FREE TRIAL
SEARCH
Products
Documentation
Download
Consulting Services
Support
Sales
Community
About Us
Best in class mathematical modeling. Performant, scalable, easy to learn.
model - solve - deploy

[GAMS]

```
$title A Transportation Problem (TRNSPORT,SEQ=1)

$onText
This problem finds a least cost shipping schedule that meets
requirements at markets and supplies at factories.

Keywords: linear programming, transportation problem, scheduling
$offText

Set
   i 'canning plants' / seattle,  san-diego /
   j 'markets'        / new-york, chicago, topeka /;

Parameter
   a(i) 'capacity of plant i in cases'
        / seattle    350
          san-diego  600 /

   b(j) 'demand at market j in cases'
        / new-york   325
          chicago    300
          topeka     275 /;

Table d(i,j) 'distance in thousands of miles'
              new-york  chicago  topeka
   seattle         2.5      1.7     1.8
   san-diego       2.5      1.8     1.4;

Scalar f 'freight in dollars per case per thousand miles' / 90 /;

Parameter c(i,j) 'transport cost in thousands of dollars per case';
c(i,j) = f*d(i,j)/1000;
```

```
Variable
   x(i,j) 'shipment quantities in cases'
   z      'total transportation costs in thousands of dollars';

Positive Variable x;

Equation
   cost      'define objective function'
   supply(i) 'observe supply limit at plant i'
   demand(j) 'satisfy demand at market j';

cost..      z =e= sum((i,j), c(i,j)*x(i,j));

supply(i).. sum(j, x(i,j)) =l= a(i);

demand(j).. sum(i, x(i,j)) =g= b(j);

Model transport / all /;

solve transport using lp minimizing z;

display x.l, x.m;
```

Ejemplo


PYOMO
HOME / ABOUT / DOWNLOAD / DOCUMENTATION / BLOG
Flexible modeling of optimization problems in Python
What Is Pyomo?
Pyomo is a Python-based, open-source optimization modeling language with a diverse set of optimization capabilities.
Read More
Installation
The easiest way to install Pyomo is to use pip. Pyomo also needs access to optimization solvers.
Read more
latest release v6.9.5
Docs and Examples
Pyomo documentation and examples are available online. Pyomo is also described in book and journal publications.
Read more

PYOMO

Objective and Solving

The definition of the objective is similar to those of the constrains, except that most solvers require a scalar objective function, hence a unique function, and we can specify the sense (direction) of the optimisation.

In [8]:

```
## Define Objective and solve ##
#  cost        define objective function
#  cost ..        z  =e=  sum((i,j), c(i,j)*x(i,j)) ;
#  Model transport /all/ ;
#  Solve transport using lp minimizing z ;
def objective_rule(model):
  return sum(model.c[i,j]*model.x[i,j] for i in model.i for j in model.j)
model.objective = Objective(rule=objective_rule, sense=minimize, doc='Define objective function')
```

As we are here looping over two distinct sets, we can see how list comprehension really simplifies the code. The objective function could have being written without list comprehension as:

In [9]:

```
def objective_rule(model):
  obj = 0.0
  for ki in model.i:
    for kj in model.j:
      obj += model.c[ki,kj]*model.x[ki,kj]
  return obj
```

Ejemplo

Problema de asignación

Problema de asignación

En este problema, *m* recursos se asignan a *n* actividades, uno a uno (cada recurso es asignado a lo sumo a una actividad, y cada actividad recibe a lo sumo un recurso). Existe un costo c_{ij} por asignar el recurso *i* a la actividad *j*. Se desea determinar para cada recurso *i* la actividad *j* a la que debe ser asignado con el fin de minimizar el costo total.

Es un problema de transporte con $s_i = 1$ y $d_j = 1$.

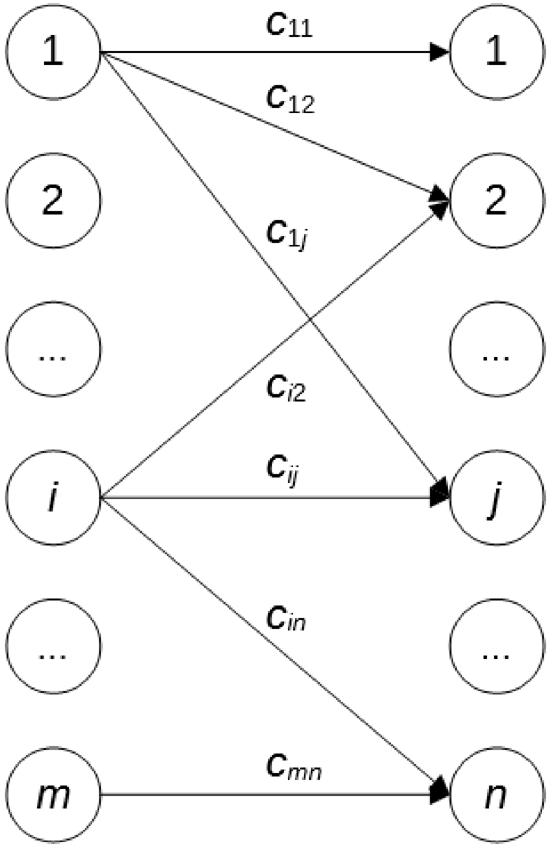
Elementos del modelo

- Índices:
  - Para recursos: $i = 1, 2, \ldots, m$.
  - Para actividades: $j = 1, 2, \ldots, n$.

Elementos del modelo

- Parámetros (en total, $2 + m\,n$):
  - *m*: Cantidad de recursos.
  - *n*: Cantidad de actividades.
  - c_{ij}: Costo de asignar el recurso *i* a la actividad *j*.
- Variables de decisión (en total, $m\,n$):
  - x_{ij}: 1 si el recurso *i* se asigna a la actividad *j*, si no 0.

Problema de asignación



$$x_{ij} = \begin{cases} 1 & \text{si } i \text{ se asigna a la actividad } j \\ 0 & \text{en caso contrario} \end{cases}$$

Casos del problema de asignación

1. Recursos iguales a las actividades: $m = n$.
2. Más recursos que actividades: $m > n$.
3. Menos recursos que actividades: $m < n$.

Elementos del modelo

- Recursos iguales a las actividades: $m = n$.
- Función objetivo:
  - Costo ($): $\sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij}\, x_{ij}$.
- Restricciones (en total, $m + n$ y las binarias):
  - Recurso *i* (persona): $\sum_{j=1}^{n} x_{ij} = 1$.
  - Actividad *j* (persona): $\sum_{i=1}^{m} x_{ij} = 1$.

Problema de asignación. Caso 1

$$\operatorname*{Min}_{x_{ij}} \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$$

s. a:

$$\sum_{j=1}^{n} x_{ij} = 1 \quad \forall i$$

$$\sum_{i=1}^{m} x_{ij} = 1 \quad \forall j$$

$$x_{ij} \in \{0,1\} \quad \forall i, j$$

$m = n$

$$GL = m\,n - (m + n - 1)$$
$$= (m-1)(n-1)$$

Problema de asignación. Caso 2

$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$$

s. a:

$$\sum_{j=1}^{n} x_{ij} \leq 1 \quad \forall i$$

$m > n$

$$\sum_{i=1}^{m} x_{ij} = 1 \quad \forall j$$

$$x_{ij} \in \{0,1\} \quad \forall i, j$$

Problema de asignación. Caso 3

$$\operatorname*{Min}_{x_{ij}} \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$$

s. a:

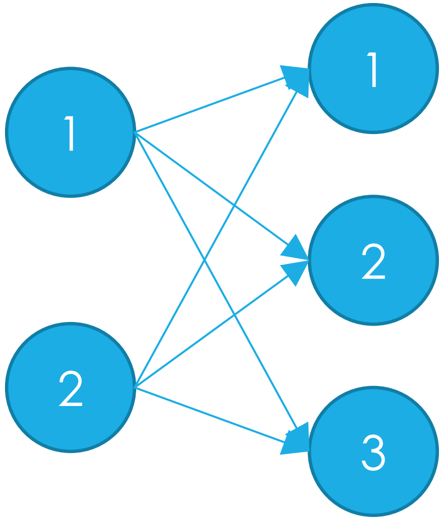
$$\sum_{j=1}^{n} x_{ij} = 1 \quad \forall i$$

$$\sum_{i=1}^{m} x_{ij} \leq 1 \quad \forall j$$

$$x_{ij} \in \{0,1\} \quad \forall i, j$$

$m < n$

Ejemplo de 2x3



$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{2}\sum_{j=1}^{3} c_{ij}x_{ij} = c_{1,1}x_{1,1} + c_{1,2}x_{1,2} + c_{1,3}x_{1,3}$$

$$+c_{2,1}x_{2,1} + c_{2,2}x_{2,2} + c_{2,3}x_{2,3}$$

s. a:

$$x_{1,1} + x_{1,2} + x_{1,3} = 1$$

$$x_{2,1} + x_{2,2} + x_{2,3} = 1$$

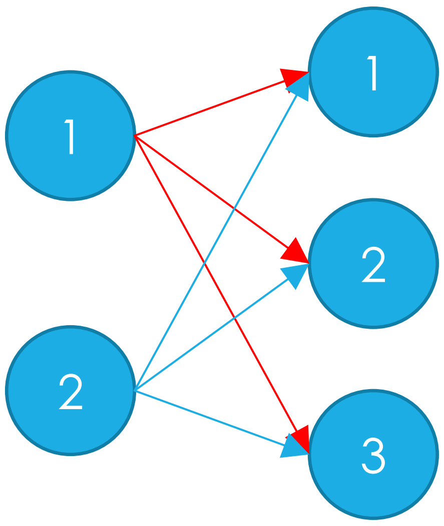
$$x_{1,1} + x_{2,1} \leq 1$$

$$x_{1,2} + x_{2,2} \leq 1$$

$$x_{1,3} + x_{2,3} \leq 1$$

$$x_{ij} \in \{0,1\} \quad \forall i, j$$

Ejemplo de 2x3



$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{2}\sum_{j=1}^{3} c_{ij}x_{ij} = c_{1,1}x_{1,1} + c_{1,2}x_{1,2} + c_{1,3}x_{1,3}$$

$$+c_{2,1}x_{2,1} + c_{2,2}x_{2,2} + c_{2,3}x_{2,3}$$

s. a:

$$x_{1,1} + x_{1,2} + x_{1,3} = 1$$

$$x_{2,1} + x_{2,2} + x_{2,3} = 1$$

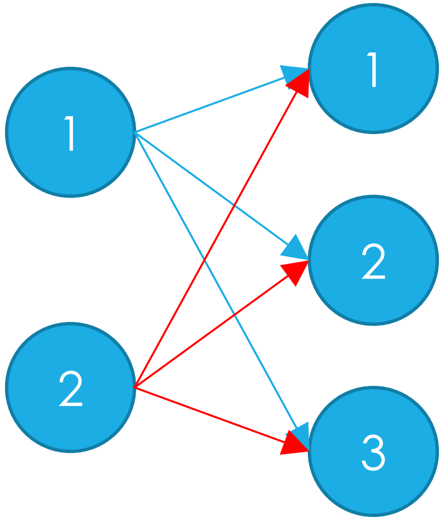
$$x_{1,1} + x_{2,1} \leq 1$$

$$x_{1,2} + x_{2,2} \leq 1$$

$$x_{1,3} + x_{2,3} \leq 1$$

$$x_{ij} \in \{0,1\} \quad \forall i, j$$

Ejemplo de 2x3



$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{2}\sum_{j=1}^{3} c_{ij}x_{ij} = c_{1,1}x_{1,1} + c_{1,2}x_{1,2} + c_{1,3}x_{1,3}$$

$$+c_{2,1}x_{2,1} + c_{2,2}x_{2,2} + c_{2,3}x_{2,3}$$

s. a:

$$x_{1,1} + x_{1,2} + x_{1,3} = 1$$

$$x_{2,1} + x_{2,2} + x_{2,3} = 1$$

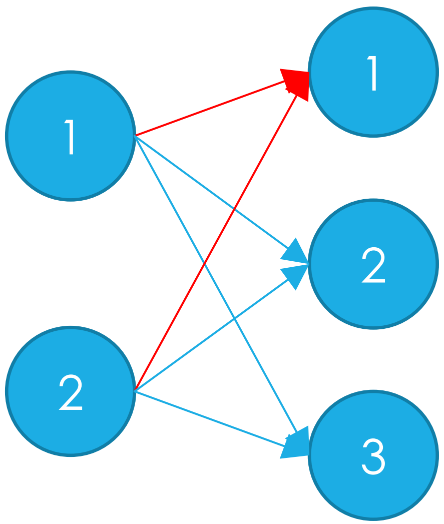
$$x_{1,1} + x_{2,1} \leq 1$$

$$x_{1,2} + x_{2,2} \leq 1$$

$$x_{1,3} + x_{2,3} \leq 1$$

$$x_{ij} \in \{0,1\} \quad \forall i, j$$

Ejemplo de 2x3



$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{2}\sum_{j=1}^{3} c_{ij}x_{ij} = c_{1,1}x_{1,1} + c_{1,2}x_{1,2} + c_{1,3}x_{1,3}$$

$$+c_{2,1}x_{2,1} + c_{2,2}x_{2,2} + c_{2,3}x_{2,3}$$

s. a:

$$x_{1,1} + x_{1,2} + x_{1,3} = 1$$

$$x_{2,1} + x_{2,2} + x_{2,3} = 1$$

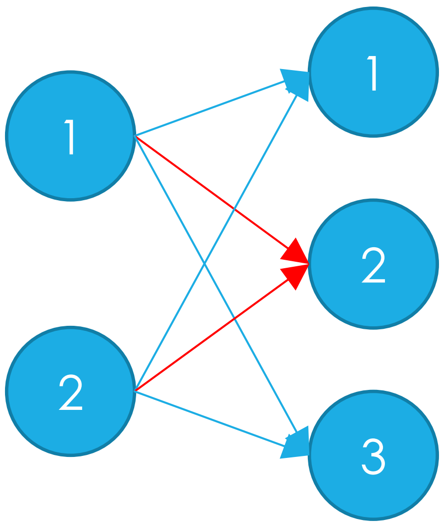
$$x_{1,1} + x_{2,1} \leq 1$$

$$x_{1,2} + x_{2,2} \leq 1$$

$$x_{1,3} + x_{2,3} \leq 1$$

$$x_{ij} \in \{0,1\} \quad \forall i, j$$

Ejemplo de 2x3



$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{2}\sum_{j=1}^{3} c_{ij}x_{ij} = c_{1,1}x_{1,1} + c_{1,2}x_{1,2} + c_{1,3}x_{1,3}$$

$$+c_{2,1}x_{2,1} + c_{2,2}x_{2,2} + c_{2,3}x_{2,3}$$

s. a:

$$x_{1,1} + x_{1,2} + x_{1,3} = 1$$

$$x_{2,1} + x_{2,2} + x_{2,3} = 1$$

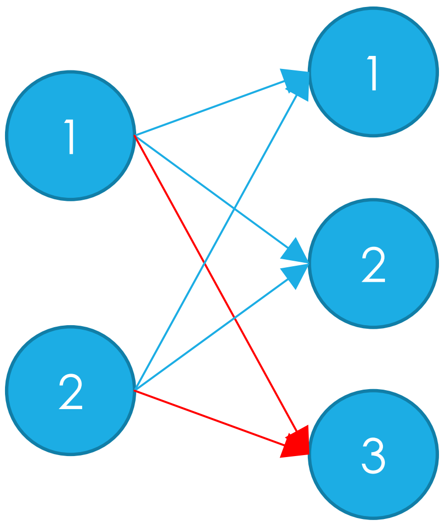
$$x_{1,1} + x_{2,1} \leq 1$$

$$x_{1,2} + x_{2,2} \leq 1$$

$$x_{1,3} + x_{2,3} \leq 1$$

$$x_{ij} \in \{0,1\} \quad \forall i, j$$

Ejemplo de 2x3



$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{2} \sum_{j=1}^{3} c_{ij} x_{ij} = c_{1,1} x_{1,1} + c_{1,2} x_{1,2} + c_{1,3} x_{1,3}$$

$$+ c_{2,1} x_{2,1} + c_{2,2} x_{2,2} + c_{2,3} x_{2,3}$$

s. a:

$$x_{1,1} + x_{1,2} + x_{1,3} = 1$$

$$x_{2,1} + x_{2,2} + x_{2,3} = 1$$

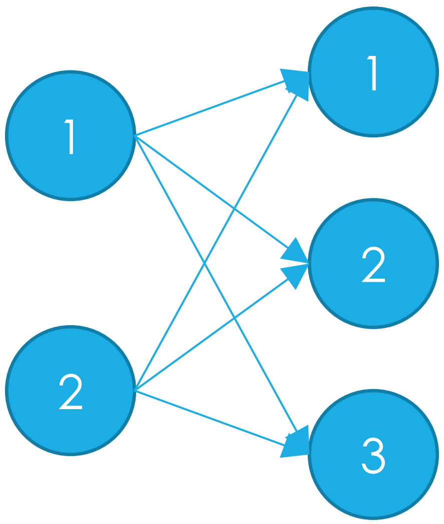
$$x_{1,1} + x_{2,1} \leq 1$$

$$x_{1,2} + x_{2,2} \leq 1$$

$$x_{1,3} + x_{2,3} \leq 1$$

$$x_{ij} \in \{0,1\} \quad \forall i, j$$

Ejemplo de 2x3



$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{2}\sum_{j=1}^{3} c_{ij}x_{ij} = c_{1,1}x_{1,1} + c_{1,2}x_{1,2} + c_{1,3}x_{1,3}$$

$$+c_{2,1}x_{2,1} + c_{2,2}x_{2,2} + c_{2,3}x_{2,3}$$

s. a:

$$x_{1,1} + x_{1,2} + x_{1,3} = 1$$

$$x_{2,1} + x_{2,2} + x_{2,3} = 1$$

$$x_{1,1} + x_{2,1} \leq 1$$

$$x_{1,2} + x_{2,2} \leq 1$$

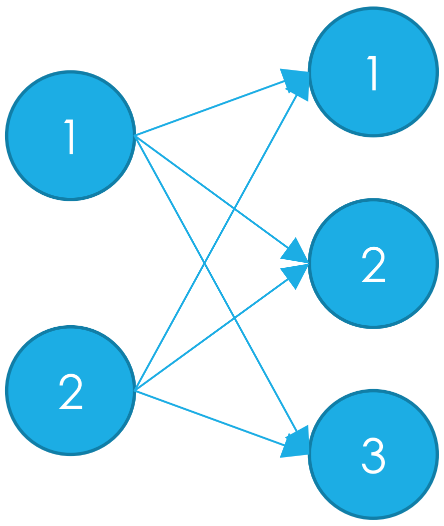
$$x_{1,3} + x_{2,3} \leq 1$$

$$x_{ij} \in \{0,1\} \quad \forall i, j$$

Total unimodularidad

- Una matriz es totalmente unimodular si todas sus submatrices cuadradas tienen determinante 0, 1 o -1.
- Si un problema LP tiene una matriz de restricciones totalmente unimodular y un vector de términos independientes enteros, cualquier vértice del poliedro de soluciones será entero.
- Por lo tanto, no es necesaria la restricción $x_{ij} \in \{0,1\} \; \forall i, \forall j$.

Ejemplo de 2x3



$$\underset{x_{ij}}{\text{Min}} \sum_{i=1}^{2}\sum_{j=1}^{3} c_{ij}x_{ij} = c_{1,1}x_{1,1} + c_{1,2}x_{1,2} + c_{1,3}x_{1,3}$$

$$+c_{2,1}x_{2,1} + c_{2,2}x_{2,2} + c_{2,3}x_{2,3}$$

s. a:

$$x_{1,1} + x_{1,2} + x_{1,3} = 1$$

$$x_{2,1} + x_{2,2} + x_{2,3} = 1$$

$$x_{1,1} + x_{2,1} \leq 1$$

$$x_{1,2} + x_{2,2} \leq 1$$

$$x_{1,3} + x_{2,3} \leq 1$$

Líderes de proyectos

Líderes de proyectos

Una empresa está planificando la ejecución de 4 proyectos. Con el fin de lograr la máxima empatía, los equipos deben elegir a sus líderes por votación de un conjunto de 6 candidatos potenciales.

C/E	E1	E2	E3	E4
C1	15	7	4	1
C2	5	8	3	1
C3	5	5	2	10
C4	1	2	2	8
C5	0	1	10	2
C6	0	2	5	2

Líderes de proyectos.lg4

Líderes de proyectos

$$\underset{x_{ij}}{\text{Max}} \sum_{i=1}^{n} \sum_{j=1}^{m} a_{ij} x_{ij}$$

s. a:

$$\sum_{j=1}^{m} x_{ij} \leq 1 \quad \forall i$$

$$\sum_{i=1}^{n} x_{ij} = 1 \quad \forall j$$

$$x_{ij} \in \{0,1\} \quad \forall i, j$$

$m > n$

```
! Problema de asignación;
Se desea asignar líderes a equipos de trabajo
de forma de maximizar la afinidad;

SETS:
  Candidatos/C1..C6/;
  Equipos/E1..E4/;
  CxE(Candidatos,Equipos): x,a;
ENDSETS

DATA:
! Afinidad a(i,j) del candidato i con el equipo j;
 a = 15  7  4  1
      5  8  3  1
      5  5  2 10
      1  2  2  8
      0  1 10  2
      0  2  5  2;
ENDDATA

! Maximizar la afinidad;
[FO] MAX = @sum(CxE: a*x);

! Asignar un equipo a lo sumo a cada candidato;
@For(Candidatos(i):
  [RC] @sum(Equipos(j): x(i,j)) <= 1
);

! Asignar un candidato a cada equipo;
@For(Equipos(j):
  [RE] @sum(Candidatos(i): x(i,j)) = 1
);
```

Modelo en LINGO

$$\max_{x_{ij}} \sum_{i=1}^{n} \sum_{j=1}^{m} a_{ij} x_{ij}$$

s. a:

$$\sum_{j=1}^{m} x_{ij} \leq 1 \quad \forall i$$

$$\sum_{i=1}^{n} x_{ij} = 1 \quad \forall j$$

$$x_{ij} \in \{0,1\} \quad \forall i, j$$

$m > n$

Lingo Model - Líderes de proyectos

```
! Problema de asignación
Se desea asignar líderes a equipos de trabajo
de forma de maximizar la afinidad;

SETS:
  Candidatos/C1..C6/;
  Equipos/E1..E4/;
  CxE(Candidatos,Equipos): x,a;
ENDSETS

DATA:
! Afinidad a(i,j) del candidato i con el equipo j;
 a = 15  7  4  1
      5  8  3  1
      5  5  2 10
      1  2  2  8
      0  1 10  2
      0  2  5  2;
ENDDATA

! Maximizar la afinidad;
[FO] MAX = @sum(CxE: a*x);

! Asignar un equipo a lo sumo a cada candidato;
@For(Candidatos(i):
  [RC] @sum(Equipos(j): x(i,j)) <= 1
);

! Asignar un candidato a cada equipo;
@For(Equipos(j):
  [RE] @sum(Candidatos(i): x(i,j)) = 1
);
```

Resultados en LINGO

Variable	Value	Reduced Cost
X(C1, E1)	1.000000	0.000000
X(C2, E2)	1.000000	0.000000
X(C3, E4)	1.000000	0.000000
X(C5, E3)	1.000000	0.000000

Resultados en LINGO

Row	Slack or Surplus	Dual Price
FO	43.00000	1.000000
RC(C1)	0.000000	0.000000
RC(C2)	0.000000	0.000000
RC(C3)	0.000000	0.000000
RC(C4)	1.000000	0.000000
RC(C5)	0.000000	0.000000
RC(C6)	1.000000	0.000000
RE(E1)	0.000000	15.00000
RE(E2)	0.000000	8.000000
RE(E3)	0.000000	10.00000
RE(E4)	0.000000	10.00000

Mapa curricular de programación lineal

1. Programación lineal
2. Problemas LP clásicos
3. Problema de mezcla
4. Problema de mezcla con composición
5. Problema de transporte
6. Casos del problema de transporte
7. Problema de asignación
8. Casos del problema de asignación