



Ejercicio de transporte

Rodrigo Maranzana

Enunciado

Una empresa tiene **3 plantas productoras** y **4 puntos de almacenamiento**.

- Se conocen los costos de envío desde una planta a cada punto de almacenamiento.
- Los puntos de almacenamientos y plantas tienen demandas y ofertas fijas.



Almacenamiento:

- Villa Martelli
- Parque Patricios
- Flores
- Ciudadela

Plantas productoras:

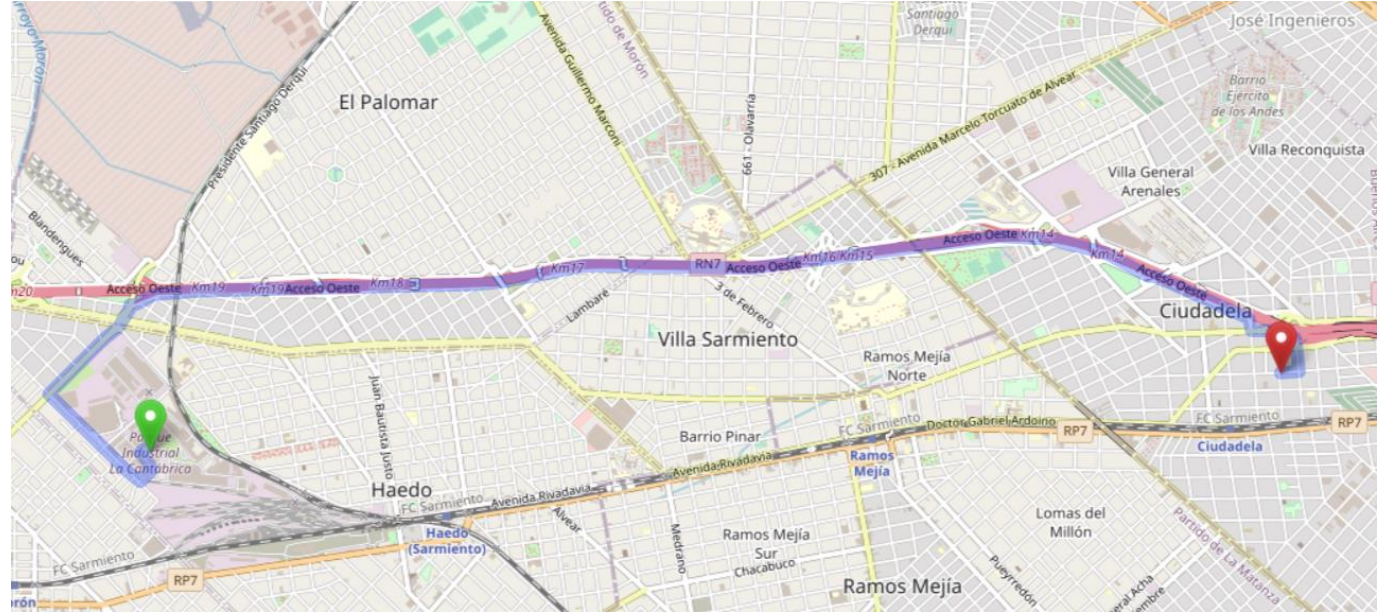
- Parque industrial “La Cantábrica”
- Mini Parque industrial Vergara
- Parque industrial Pilar

Datos

1. Se calcula la distancia entre cada origen y destino utilizando un **algoritmo de camino más corto**.

Una vez calculada, cada distancia, se considera lineal entre planta y almacén.

2. Para este ejemplo, se considera que el costo del transporte depende de la distancia recorrida y factores de carga.



La Cantábrica → Almacén Ciudadela: **9.2km**

Datos

Tabla de costos

origen / destino	Villa Martelli	Parque Patricios	Flores	Ciudadela
Parque industrial "La Cantábrica"	434	523	640	850
Mini Parque industrial Vergara	323	480	670	770
Parque industrial Pilar	997	680	390	590

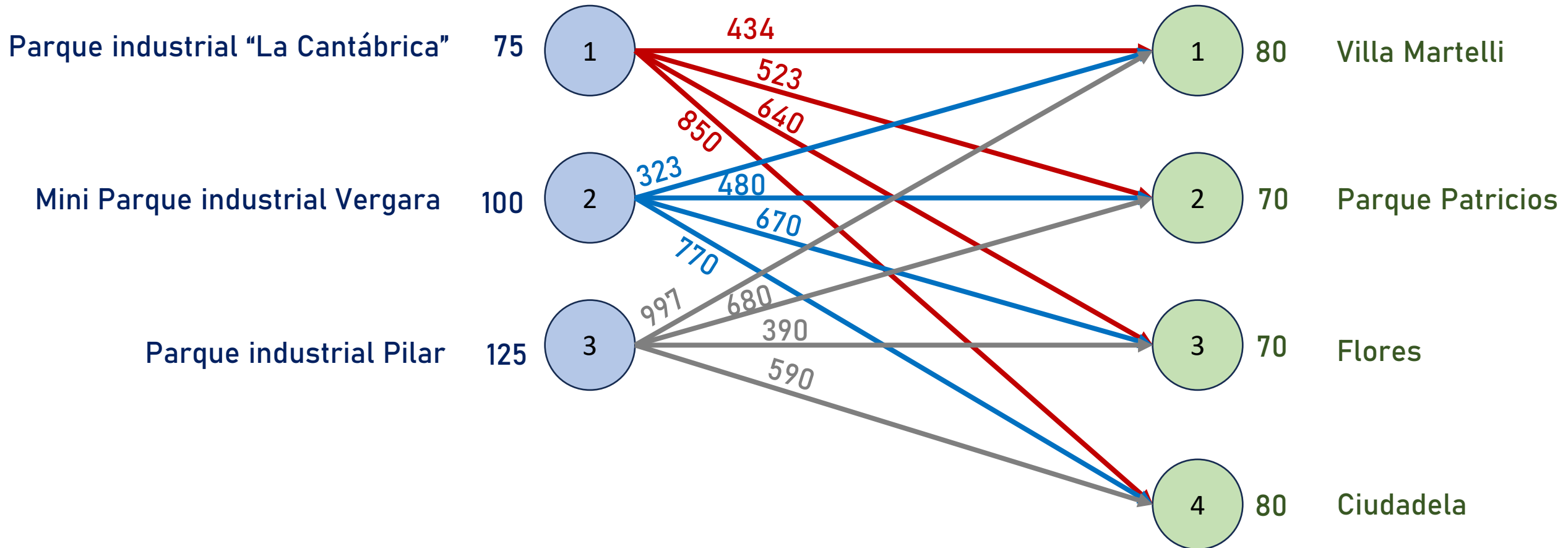
Tabla de cantidades a enviar

	Villa Martelli	Parque Patricios	Flores	Ciudadela	oferta
Parque industrial "La Cantábrica"	??	??	??	??	75
Mini Parque industrial Vergara	??	??	??	??	100
Parque industrial Pilar	??	??	??	??	125
					total oferta: 300
demanda	80	70	70	80	total demanda: 300

Consignas

- 1- Armar grafo de la red de transporte.
- 2- Armar modelo de transporte.
- 3- Armar modelo de transporte como FMC matricial.
- 4- Resolver la forma matricial con python.

1-Grafo



2-Modelo de transporte balanceado

Dado que la oferta es igual a la demanda, es un tipo de problema de transporte balanceado.

$$\text{Min} \sum_i \sum_j c_{ij} x_{ij}$$

s.t.

$$\sum_j x_{ij} = a_i \quad \forall i$$

$$\sum_i x_{ij} = b_j \quad \forall j$$

$$x \geq 0; x \in \mathbb{R}$$

Conjuntos (sets)

i : nodos oferentes

j : nodos demandantes

Parámetros

b_j : demanda

a_i : oferta

c_{ij} : costo del arco de i a j

Variables de decisión

x_{ij} : cantidad de producto a enviar de i a j

2-Modelo de transporte: sets

Conjuntos (sets)

i : nodos oferentes

j : nodos demandantes

Nodos i (Plantas)	Descripción
1	Parque industrial “La Cantábrica”
2	Mini Parque industrial Vergara
3	Parque industrial Pilar

Nodos j (Almacenes)	Descripción
1	Villa Martelli
2	Parque Patricios
3	Flores
4	Ciudadela

2-Modelo de transporte: parámetros y variables

Costos

C_{ij}	$j = 1$	$j = 2$	$j = 3$	$j = 4$
$i = 1$	434	523	640	850
$i = 2$	323	480	670	770
$i = 3$	997	680	390	590

Oferta

	a_i
$i = 1$	75
$i = 2$	100
$i = 3$	125

Demanda

	b_j
$j = 1$	80
$j = 2$	70
$j = 3$	70
$j = 4$	80

$$X = \begin{bmatrix} x_{11} \\ x_{12} \\ x_{13} \\ x_{14} \\ x_{21} \\ x_{22} \\ x_{23} \\ x_{24} \\ x_{31} \\ x_{32} \\ x_{33} \\ x_{34} \end{bmatrix}$$

Nodos i (Plantas)	Descripción
1	Parque industrial "La Cantábrica"
2	Mini Parque industrial Vergara
3	Parque industrial Pilar

Nodos j (Almacenes)	Descripción
1	Villa Martelli
2	Parque Patricios
3	Flores
4	Ciudadela

2-Modelo de transporte

$$\begin{aligned}
 & \text{Min} \sum_i \sum_j c_{ij} x_{ij} \\
 & \text{s.t.} \\
 & \sum_j x_{ij} = a_i \quad \forall i \\
 & \sum_i x_{ij} = b_j \quad \forall j \\
 & x \geq 0; x \in \mathbb{R}
 \end{aligned}$$

c_{ij}	$j = 1$	$j = 2$	$j = 3$	$j = 4$		a_i		b_j
$i = 1$	434	523	640	850	$i = 1$	75	$j = 1$	80
$i = 2$	323	480	670	770	$i = 2$	100	$j = 2$	70
$i = 3$	997	680	390	590	$i = 3$	125	$j = 3$	70
							$j = 4$	80

Función
objetivo

$$\begin{aligned}
 & \text{Min} \quad 434x_{11} + 523x_{12} + 640x_{13} + 850x_{14} + \\
 & \quad 323x_{21} + 480x_{22} + 670x_{23} + 770x_{24} + \\
 & \quad 997x_{31} + 680x_{32} + 390x_{33} + 590x_{34} +
 \end{aligned}$$

s.t.

Restricciones de
cumplimiento de oferta

$$\begin{aligned}
 & x_{11} + x_{12} + x_{13} + x_{14} = 75 \\
 & x_{21} + x_{22} + x_{23} + x_{24} = 100 \\
 & x_{31} + x_{32} + x_{33} + x_{34} = 125
 \end{aligned}$$

Restricciones de
cumplimiento de
demanda

$$\begin{aligned}
 & x_{11} + x_{21} + x_{31} = 80 \\
 & x_{12} + x_{22} + x_{32} = 70 \\
 & x_{13} + x_{23} + x_{33} = 70 \\
 & x_{14} + x_{24} + x_{34} = 80
 \end{aligned}$$

$$x \geq 0; x \in \mathbb{R}$$

3-Modelo de transporte como FMC matricial

$$\text{Min } C^T X$$

st:

$$AX = b$$

$$\text{cota inferior} \leq X \leq \text{cota superior}$$

3-FMC: vector de costos y variables de decisión

$$X = \begin{bmatrix} x_{11} \\ x_{12} \\ x_{13} \\ x_{14} \\ x_{21} \\ x_{22} \\ x_{23} \\ x_{24} \\ x_{31} \\ x_{32} \\ x_{33} \\ x_{34} \end{bmatrix} \quad C = \begin{bmatrix} 434 \\ 523 \\ 640 \\ 850 \\ 323 \\ 480 \\ 670 \\ 770 \\ 997 \\ 680 \\ 390 \\ 590 \end{bmatrix}$$

$$0 \leq X$$

3-FMC: matriz nodo arco y vector b

Matriz nodo arco

	x_{11}	x_{12}	x_{13}	x_{14}	x_{21}	x_{22}	x_{23}	x_{24}	x_{31}	x_{32}	x_{33}	x_{34}	
Restricciones de Oferta	1	1	1	1									75
					1	1	1	1					100
									1	1	1	1	125
Restricciones de Demanda	-1				-1				-1				-80
		-1				-1				-1			-70
			-1				-1				-1		-70
				-1				-1				-1	-80

$A =$

$b =$

4-Solución con scipy.optimize.linprog

```
import numpy as np
from scipy.optimize import linprog

# Matriz nodo-arco:
Aeq = np.array([[ 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0],
                 [ 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0],
                 [ 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1],
                 [-1, 0, 0, 0, -1, 0, 0, 0, -1, 0, 0, 0],
                 [ 0, -1, 0, 0, 0, -1, 0, 0, 0, -1, 0, 0],
                 [ 0, 0, -1, 0, 0, 0, -1, 0, 0, 0, -1, 0],
                 [ 0, 0, 0, -1, 0, 0, 0, -1, 0, 0, 0, -1]])

# Vector de costos por arco:
C = np.array([434, 523, 640, 850, 323, 480, 670, 770, 997, 680, 390, 590])

# Vector de oferta y demanda:
beq = np.array([75, 100, 125, -80, -70, -70, -80])

# Cotas:
bounds = tuple([(0, None) for arcs in range(0, C.shape[0])])
```

```
# Optimizamos:
res = linprog(C, A_eq=Aeq, b_eq=beq, bounds=bounds)

print('Cantidad para cada arco:', res.x)
print('Costo mínimo total:', res.fun)
```

4-Solución con scipy.optimize.linprog

```
>>Cantidad para cada arco: [ 0. 70.  5.  0. 80.  0.  0. 20.  0.  0. 65. 60.]  
>>Costo mínimo total: 141800.0
```

