Programación Lineal: Método SIMPLEX con tablas

Rodrigo Maranzana

Concepto

El método SIMPLEX inventado por George Dantzig (1947, US Air Force); para reducir el número de soluciones a analizar de un problema.

- Uno de los algoritmos más importantes del siglo XX.
- Variantes se usan al día de hoy en solvers.

La resolución con tablas es un tipo de representación del algoritmo.

- Resolución iterativa de problemas lineales.
- Forma didáctica: permite entender cada operación.
- Definición clara de cada componente en las operaciones (ej: variables básicas y no básicas)

Modelo lineal (LP)

$$\min C^T X$$

s.t.

$$AX = B$$
$$X \geq 0$$

$X \in \mathbb{R}^n$	Vector de variables de decisión.
$b \in \mathbb{R}^m$	Término del lado derecho.
$A \in \mathbb{R}^{m \times n}$	Matriz de coeficientes tecnológicos.
$C \in \mathbb{R}^n$	Vector de costos.

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} \quad b = \begin{bmatrix} b_1 \\ b_2 \\ \dots \\ b_m \end{bmatrix} \quad C = \begin{bmatrix} c_1 \\ c_2 \\ \dots \\ c_n \end{bmatrix} \quad X = \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{bmatrix}$$

Algoritmo SIMPLEX: variables básicas y no básicas

- Seleccionar m variables, creando el subconjunto B de variables llamadas básicas: X_B.
- Las variables básicas están asociadas a la matriz A_B, columnas de A en la posición de X_B.

- Como seleccionamos m variables y esa matriz ya tiene m filas de restricciones. La matriz es cuadrada $A_B \in \mathbb{R}^{m\times m}$
- La selección de las m variables debe ser tal que A_B sea invertible.

- $A_N \in \mathbb{R}^{m\times(n-m)}$ corresponde a las variables no básicas X_N.

$$A = \begin{array}{c} \begin{array}{ccccccc} & & \boldsymbol{A_B} & & & & \boldsymbol{A_N} \end{array} \\ \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1m} & \dots & a_{14} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2m} & \dots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \dots & a_{3m} & \dots & a_{3n} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & a_{m3} & \dots & a_{mm} & \dots & a_{mn} \end{bmatrix} \end{array} \qquad X = \begin{array}{cc} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \dots \\ x_m \\ \dots \\ x_n \end{bmatrix} & \begin{array}{c} \boldsymbol{X_B} \\ \\ \\ \\ \\ \boldsymbol{X_N} \end{array} \end{array}$$

UTN.BA

Algoritmo SIMPLEX: variables básicas y no básicas

Una solución factible contiene ambas, variables básicas y no básicas.

- Las variables básicas: X_B, forman parte de la base y se permite que tengan un valor distinto de 0.
- Las variables no básicas X_N tendrán valor 0.

$$
A = \begin{array}{c}
\begin{array}{ccccccc} & & A_B & & & & A_N \end{array} \\
\begin{bmatrix}
a_{11} & a_{12} & a_{13} & \dots & a_{1m} & \dots & a_{14} \\
a_{21} & a_{22} & a_{23} & \dots & a_{2m} & \dots & a_{2n} \\
a_{31} & a_{32} & a_{33} & \dots & a_{3m} & \dots & a_{3n} \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots \\
a_{m1} & a_{m2} & a_{m3} & \dots & a_{mm} & \dots & a_{mn}
\end{bmatrix}
\end{array}
\qquad
X = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \dots \\ x_m \\ \dots \\ x_n \end{bmatrix}
\begin{array}{l} \\ X_B \\ \\ \\ \\ X_N \\ \end{array}
$$

Algoritmo SIMPLEX: variables básicas y no básicas

Podemos escribir las restricciones en términos de variables básicas y no básicas:

$$\boldsymbol{AX = B}$$

$$A_B X_B + A_N X_N = B$$

Despejando X_B:

$$X_B = A_B^{-1}B - A_B^{-1}A_N X_N \qquad [1]$$

Hacemos lo mismo con la función objetivo:

$$\boldsymbol{Z = C^T X}$$

$$Z = C_B^T X_B + C_N^T X_N$$

Reemplazamos X_B con [1]

$$Z = C_B^T (A_B^{-1}B - A_B^{-1}A_N X_N) + C_N^T X_N$$

Distribuimos y simplificamos:

$$Z = C_B^T A_B^{-1}B + (C_N^T - C_B^T A_B^{-1}A_N) X_N \qquad [2]$$

Algoritmo SIMPLEX: expresiones

Expresiones:

[1] $X_B = A_B^{-1}B - A_B^{-1}A_N X_N$

[2] $\mathrm{Z} = C_B^T A_B^{-1}B + (C_N^T - C_B^T A_B^{-1}A_N)X_N$

Conclusiones:

- Si las variables no básicas se anulan, $X_N = 0$, entonces
  - con [1] X_B resulta $s = A_B^{-1}B$
  - con [2] Z resulta $\mathrm{Z}_0 = C_B^T s$

- Si llamamos $\mathrm{H} = A_B^{-1}A_N$
  - con [1] $X_B = s - HX_N$
  - con [2] $Z = C_B^T s + (C_N^T - C_B^T H)X_N$

Algoritmo SIMPLEX: costo de oportunidad

$$Z = C_B^T s + (C_N^T - C_B^T H) X_N$$

Llamamos costo de oportunidad de X_N al término $r^T = (C_N^T - C_B^T H)$.

Es la contribución que podría hacer X_N a la función objetivo.

Por una cuestión de acuerdo de signos con la bibliografía vamos a tomar al término como $-r^T$:

$$C_B^T H - C_N^T$$

En las tablas va a llevar el nombre de: $Z_j - C_j$

Aporte al objetivo:

- Maximización: cuanto más negativo mayor aporte.
- Minimización: cuanto más positivo mayor aporte.

Algoritmo SIMPLEX: costo de oportunidad

$$C_B^T H - C_N^T$$

$$\boxed{Z_j - C_j}$$

Este término nos va a permitir saber:

- Qué variable no básica X_N tiene el mejor costo de oportunidad.
- Qué variable no básica debe entrar a la base.

Dado que A_B debe ser siempre cuadrada. ¿Quién sale de la base?

Algoritmo SIMPLEX: Ratio Test

Por acuerdos con las tablas y la bibliografía:

- Llamamos A_{ij} al término $\mathrm{H} = A_B^{-1} A_N$
- Llamamos B_k al término $s = A_B^{-1} B$

Se conoce a la proporción B_k / A_{ij} como Ratio Test.

Siendo "j" la variable no básica X_N que entra a la base.
Siendo "i" la restricción que vamos a estresar.

> $B_{i=k} / A_{ij}$ cuánto debo incrementar la variable entrante "j" para satisfacer la restricción "i"

Algoritmo SIMPLEX: Ratio Test

$B_{i=k}/A_{ij}$ cuánto debo incrementar la variable entrante “j” para satisfacer la restricción “i”

Buscamos seleccionar siempre $\min B_{i=k}/A_{ij} \geq 0$

Esto quiere decir que:

- Con un aumento mínimo de la variable entrante cumplo con la restricción “i”.
- Evito salir del recinto factible, estresando alguna restricción de más.

Algoritmo SIMPLEX: regiones de la tabla

			$\boldsymbol{C_B}$	$\boldsymbol{C_N}$	
$\boldsymbol{C_B}$		$\boldsymbol{s = B_k}$	$\boldsymbol{I}$	$\boldsymbol{H}$	B_k / A_{ij}
$\boldsymbol{Z_0}$			$\bar{0}$	$\boldsymbol{Z_j - C_j}$	

$\mathrm{H} = A_B^{-1} A_N$ (Llamado A_{ij})

Resultado variable básica: $B_k = s = A_B^{-1} B$

Costo de Oportunidad: $Z_j - C_j = C_B^T H \; - \; C_N^T$

Resultado función objetivo: $\mathrm{Z}_0 = C_B^T s$

Ratio Test: B_k / A_{ij}

I: matriz identidad

$\bar{0}$: vector nulo

$\boldsymbol{C_B}$: vector de costos básicos

$\boldsymbol{C_N}$: vector de costos no básicos

Ejemplo

Una empresa fabrica el producto A, con una utilidad de 2 $/u, y el producto B, con una ganancia de 3 $/u.

El producto A requiere para su fabricación 2 kg de cobre y 1 kg de aluminio. El producto B, en cambio, requiere 1 kg de cobre y 2 kg de aluminio. El máximo disponible de cobre y aluminio es 160 kg y 180 kg, respectivamente.

- Plantear modelo matemático.
- Resolver mediante método de tablas SIMPLEX.

Componentes del modelo

Función objetivo: Maximizar la utilidad de un mix de productos A y B.
Tipo: Lineal

Variables de decisión: Cantidad de producto A ($\boldsymbol{X_1}$) y B ($\boldsymbol{X_2}$)
Tipo: Lineal

Restricciones:
- Máximo de materia prima de cobre ($\boldsymbol{Y_1}$) y aluminio ($\boldsymbol{Y_2}$)
- Restricciones y variables de decisión Reales
- Positividad

Métodos de resolución posibles:
- Método gráfico
- Algoritmo SIMPLEX
- Algoritmo de punto interior
- Otros algoritmos específicos de asignación de recursos.
- Algoritmos heurísticos.

Método elegido: SIMPLEX

Construcción del modelo

Una empresa fabrica el producto A, que le aumenta su utilidad 2 $ por unidad, y el producto B, que le aumenta la utilidad 3 $ por unidad.

El producto A requiere de 2 kg de cobre y 1 kg de aluminio. El producto B requiere de 1 kg de cobre y 2 kg de aluminio. El máximo disponible de cobre es 160 kg y el máximo disponible de aluminio es de 180 kg.

$$\boldsymbol{Max\ Z = 2X_1 + 3X_2}$$

$$\boldsymbol{sujeto\ a:}$$

$$\boldsymbol{Y_1: 2X_1 + 1X_2 \leq 160}$$
$$\boldsymbol{Y_2: 1X_1 + 2X_2 \leq 180}$$

$$\boldsymbol{X_1, X_2 \geq 0}$$

Modelo extendido

$$\boldsymbol{Max\ Z = 2X_1 + 3X_2}$$
$$\boldsymbol{sujeto\ a}:$$
$$\boldsymbol{Y_1: 2X_1 + X_2 \leq 160}$$
$$\boldsymbol{Y_2: X_1 + 2X_2 \leq 180}$$
$$\boldsymbol{X_1, X_2 \geq 0}$$

Modelo Extendido

$$\boldsymbol{Max\ Z = 2X_1 + 3X_2}$$
$$\boldsymbol{sujeto\ a}:$$
$$\boldsymbol{2X_1 + X_2 + X_3 = 160}$$
$$\boldsymbol{X_1 + 2X_2 + X_4 = 180}$$
$$\boldsymbol{X_1, X_2 \geq 0}$$

Forma matricial

$$\boldsymbol{Max\ Z = 2X_1 + 3X_2}$$
sujeto a:

$$\boldsymbol{2X_1 + X_2 + X_3 = 160}$$
$$\boldsymbol{X_1 + 2X_2 + X_4 = 180}$$

$$\boldsymbol{X_1, X_2 \geq 0}$$

Modelo Extendido Matricial

$$\boldsymbol{Max\ Z = C^T X}$$
sujeto a:

$$\boldsymbol{AX = b}$$

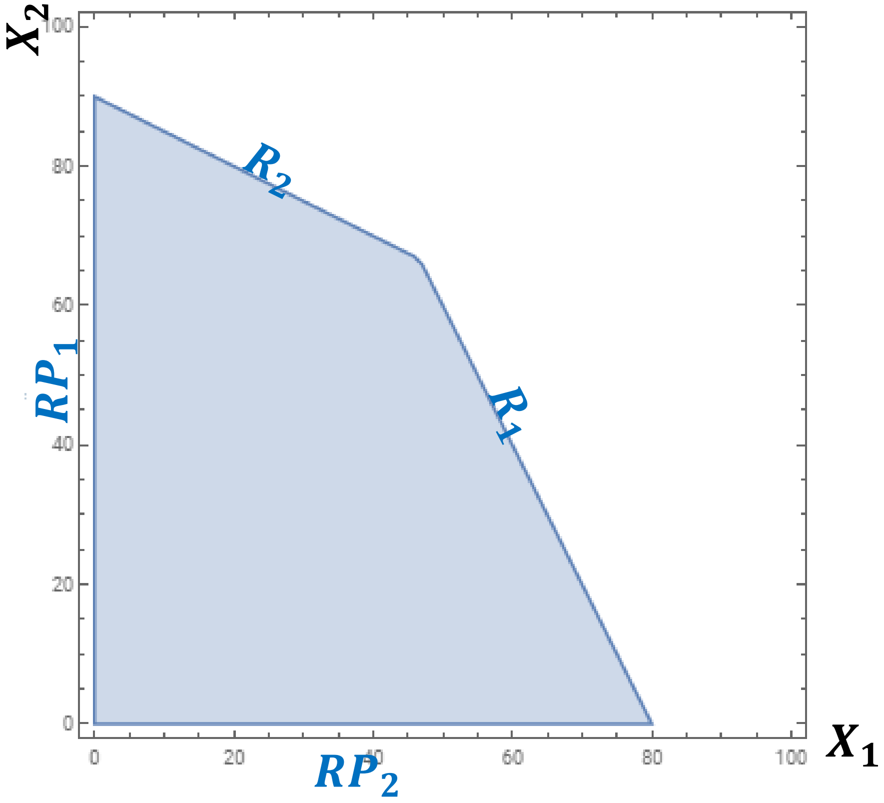
Valores de matrices:

$$A = \begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & 2 & 0 & 1 \end{bmatrix}$$

$$b = \begin{bmatrix} 160 \\ 180 \end{bmatrix}$$

$$C = \begin{bmatrix} 2 \\ 3 \\ 0 \\ 0 \end{bmatrix} X = \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}$$

Representación gráfica



Estructura de tabla SIMPLEX

C_j							B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k					
Z	$Z_j - C_j$						

Configuración inicial (#0)

$$\boldsymbol{Max\ Z = 2X_1 + 3X_2}$$
$$\boldsymbol{sujeto\ a:}$$
$$\boldsymbol{2X_1 + X_2 + X_3 = 160}$$
$$\boldsymbol{X_1 + 2X_2 + X_4 = 180}$$
$$\boldsymbol{X_1, X_2 \geq 0}$$

$$\boldsymbol{Max\ Z = C^T X}$$
$$\boldsymbol{sujeto\ a:}$$
$$\boldsymbol{AX = b}$$

$$C = \begin{bmatrix} 2 \\ 3 \\ 0 \\ 0 \end{bmatrix} \quad X = \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}$$

$$A = \begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & 2 & 0 & 1 \end{bmatrix} \quad b = \begin{bmatrix} 160 \\ 180 \end{bmatrix}$$

$\boldsymbol{C_j}$							B_k / A_{ij}
$\boldsymbol{C_j\ Base}$	$\boldsymbol{X_j\ Base}$	$\boldsymbol{B_k}$					
$\boldsymbol{Z}$	$Z_j - C_j$						

Configuración inicial (#0)

$$\boldsymbol{Max\ Z = 2X_1 + 3X_2 + 0X_3 + 0X_4}$$
$\boldsymbol{sujeto\ a}$:

$$\boldsymbol{2X_1 + X_2 + X_3 = 160}$$

$$\boldsymbol{X_1 + 2X_2 + X_4 = 180}$$

$$\boldsymbol{X_1, X_2 \geq 0}$$

$$\boldsymbol{Max\ Z = C^T X}$$
$\boldsymbol{sujeto\ a}$:

$$\boldsymbol{AX = b}$$

$$C = \begin{bmatrix} 2 \\ 3 \\ 0 \\ 0 \end{bmatrix} \quad X = \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}$$

$$A = \begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & 2 & 0 & 1 \end{bmatrix} \quad b = \begin{bmatrix} 160 \\ 180 \end{bmatrix}$$

C_j			2	3	0	0	B_k / A_{ij}
C_j Base	X_j Base	B_k					
Z	$Z_j - C_j$						

Configuración inicial (#0)

$$\begin{aligned} &\boldsymbol{Max\ Z = 2X_1 + 3X_2 + 0X_3 + 0X_4} \\ &\boldsymbol{sujeto\ a:} \\ &\quad \boldsymbol{2X_1 + 1X_2 + 1X_3 \qquad = 160} \\ &\quad \boldsymbol{1X_1 + 2X_2 \qquad + 1X_4 = 180} \\ &\qquad\qquad \boldsymbol{X_1, X_2 \geq 0} \end{aligned}$$

$$\begin{aligned} &\boldsymbol{Max\ Z = C^T X} \\ &\boldsymbol{sujeto\ a:} \\ &\quad \boldsymbol{AX = b} \end{aligned}$$

$$C = \begin{bmatrix} 2 \\ 3 \\ 0 \\ 0 \end{bmatrix} \quad X = \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}$$

$$A = \begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & 2 & 0 & 1 \end{bmatrix} \quad b = \begin{bmatrix} 160 \\ 180 \end{bmatrix}$$

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
Z	$Z_j - C_j$						

Configuración inicial (#0)

$$\boldsymbol{Max\ Z = 2X_1 + 3X_2}$$
$$\boldsymbol{sujeto\ a:}$$
$$\boldsymbol{2X_1 + 1X_2 + 1X_3 \qquad = 160}$$
$$\boldsymbol{1X_1 + 2X_2 \qquad + 1X_4 = 180}$$
$$\boldsymbol{X_1, X_2 \geq 0}$$

$$\boldsymbol{Max\ Z = C^T X}$$
$$\boldsymbol{sujeto\ a:}$$
$$\boldsymbol{AX = b}$$

$$C = \begin{bmatrix} 2 \\ 3 \\ 0 \\ 0 \end{bmatrix} \quad X = \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}$$

$$A = \begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & 2 & 0 & 1 \end{bmatrix} \quad b = \begin{bmatrix} 160 \\ 180 \end{bmatrix}$$

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
			2	1	1	0	
			1	2	0	1	
Z	$Z_j - C_j$						

Configuración inicial (#0)

$$\boldsymbol{Max\ Z = 2X_1 + 3X_2}$$
$$\boldsymbol{sujeto\ a}:$$
$$\boldsymbol{2X_1 + X_2 + X_3 = 160}$$
$$\boldsymbol{X_1 + 2X_2 + X_4 = 180}$$
$$\boldsymbol{X_1, X_2 \geq 0}$$

$$\boldsymbol{Max\ Z = C^T X}$$
$$\boldsymbol{sujeto\ a}:$$
$$\boldsymbol{AX = b}$$

$$C = \begin{bmatrix} 2 \\ 3 \\ 0 \\ 0 \end{bmatrix} \quad X = \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}$$

$$A = \begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & 2 & 0 & 1 \end{bmatrix} \quad b = \begin{bmatrix} 160 \\ 180 \end{bmatrix}$$

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
		160	2	1	1	0	
		180	1	2	0	1	
Z	$Z_j - C_j$						

Configuración inicial (#0)

$$\textbf{Max } Z = 2X_1 + 3X_2$$
$$\textbf{sujeto a}:$$
$$2X_1 + X_2 + X_3 = 160$$
$$X_1 + 2X_2 + X_4 = 180$$
$$X_1, X_2 \geq 0$$

$$\textbf{Max } Z = C^T X \qquad C = \begin{bmatrix} 2 \\ 3 \\ 0 \\ 0 \end{bmatrix} \quad X = \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}$$
$$\textbf{sujeto a}:$$
$$AX = b$$

$$A = \begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & 2 & 0 & 1 \end{bmatrix} \quad b = \begin{bmatrix} 160 \\ 180 \end{bmatrix}$$

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
	X_3	160	2	1	1	0	
	X_4	180	1	2	0	1	
Z	$Z_j - C_j$						

Configuración inicial (#0)

$$\boldsymbol{Max\ Z = 2X_1 + 3X_2 + 0X_3 + 0X_4}$$

$\boldsymbol{sujeto\ a}$:

$$\boldsymbol{2X_1 + X_2 + X_3 = 160}$$

$$\boldsymbol{X_1 + 2X_2 + X_4 = 180}$$

$$\boldsymbol{X_1, X_2 \geq 0}$$

$$\boldsymbol{Max\ Z = C^T X}$$

$\boldsymbol{sujeto\ a}$:

$$\boldsymbol{AX = b}$$

$$C = \begin{bmatrix} 2 \\ 3 \\ 0 \\ 0 \end{bmatrix} \quad X = \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}$$

$$A = \begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & 2 & 0 & 1 \end{bmatrix} \quad b = \begin{bmatrix} 160 \\ 180 \end{bmatrix}$$

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	160	2	1	1	0	
0	X_4	180	1	2	0	1	
Z	$Z_j - C_j$						

Configuración inicial (#0)

C_j			2	3	0	0	B_k/A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	160	2	1	1	0	
0	X_4	180	1	2	0	1	
Z	$Z_j - C_j$		-2	-3	0	0	

$$\boxed{\boldsymbol{Z_1 = C_3 * A_{11} + C_4 * A_{21} = 0 * 2 + 0 * 1 = 0}}$$

$$\boxed{\boldsymbol{C_1 = 2}}$$

$$\boldsymbol{Z_1 - C_1 = 0 - 2 = -2}$$

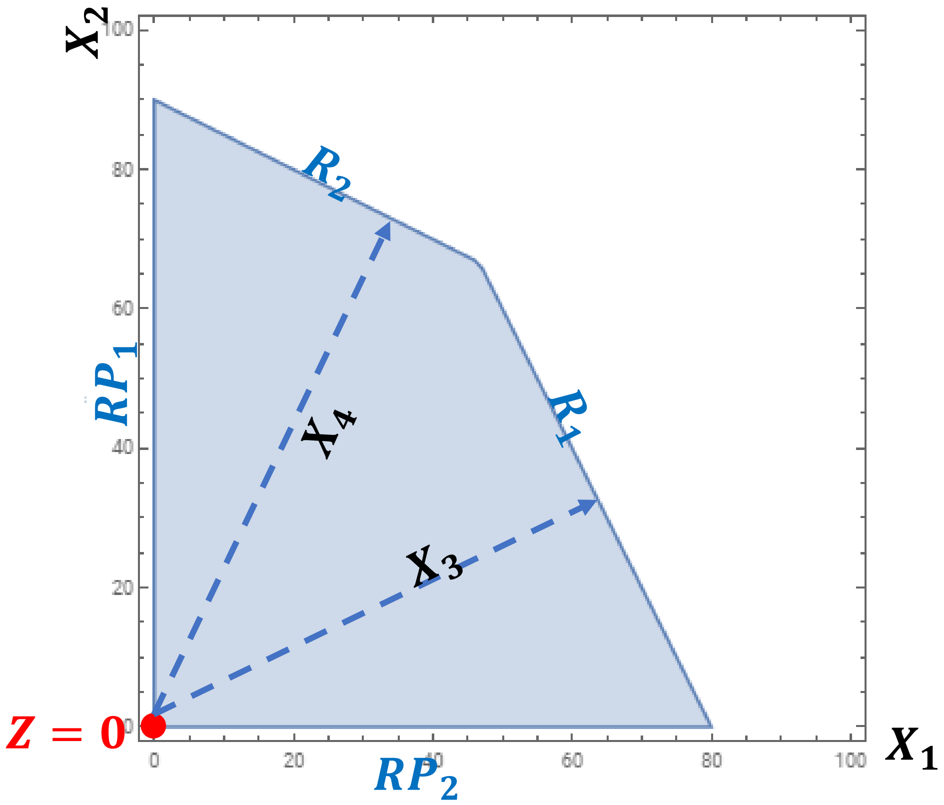
Configuración inicial (#0)

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	160	2	1	1	0	
0	X_4	180	1	2	0	1	
0	$Z_j - C_j$		-2	-3	0	0	

$$\boldsymbol{Z = 0 * 160 + 0 * 180 = 0}$$

¡Hay valores negativos, puede mejorar!

Representación gráfica (#0)



$$X_1 = 0$$
$$X_2 = 0$$

Optimización (#0)

C_j			2	3	0	0	B_k/A_{ij}
$\boldsymbol{C_j}$ **Base**	$\boldsymbol{X_j}$ **Base**	$\boldsymbol{B_k}$	X_1	X_2	X_3	X_4	
0	X_3	160	2	1	1	0	
0	X_4	180	1	2	0	1	
0	$Z_j - C_j$		-2	-3	0	0	

Columna pivote: $\min(Z_j - C_j)$

X_2 el más negativo, entra a la base. ¿Quién sale?

Optimización (#0)

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	160	2	1	1	0	160
0	X_4	180	1	2	0	1	90
0	$Z_j - C_j$		-2	-3	0	0	

$B_k\ /A_{ij}$ (de la columna pivote) $= B_k\ /A_{i2}$

Optimización (#0)

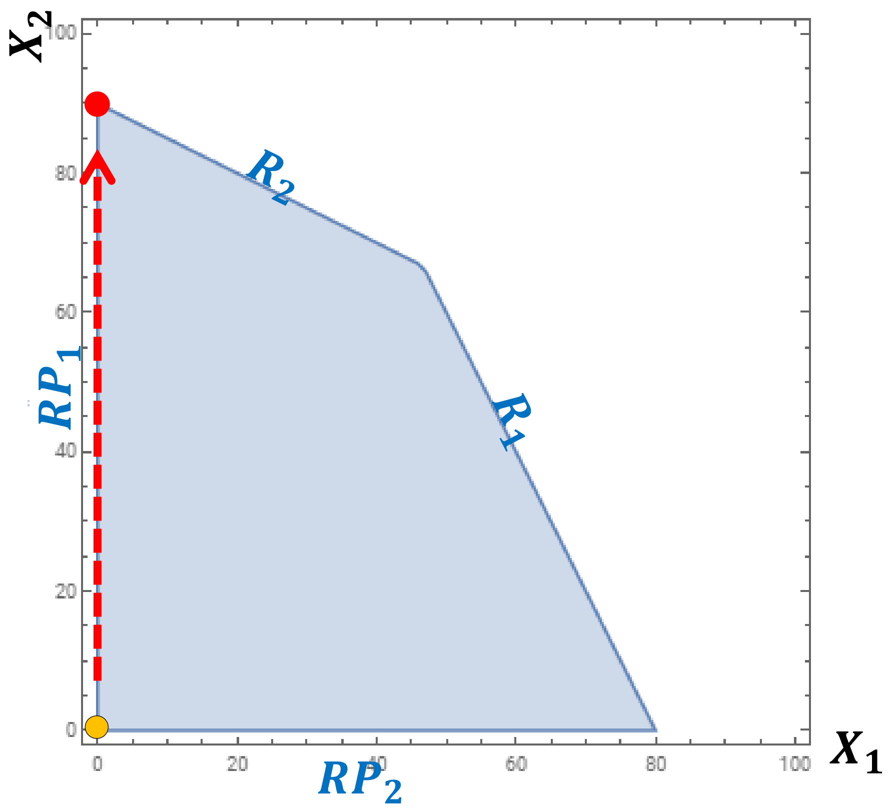
C_j			2	3	0	0	B_k / A_{ij}
$\boldsymbol{C_j\ Base}$	$\boldsymbol{X_j\ Base}$	$\boldsymbol{B_k}$	X_1	X_2	X_3	X_4	
0	X_3	160	2	1	1	0	160
0	X_4	180	1	2	0	1	90
0	$Z_j - C_j$		-2	-3	0	0	

pivote

Fila pivote: $\min\left(\frac{B_k}{A_{ij}}\right), \quad \text{si } \frac{B_k}{A_{ij}} > 0$

X_4 Sale de la base, entra X_2

Representación gráfica (#0 a #1)



Optimización (#0 a #1)

Tabla iteración 0

C_j			2	3	0	0	B_k / A_{ij}
C_j Base	X_j Base	B_k	X_1	X_2	X_3	X_4	
0	X_3	160	2	1	1	0	
0	X_4	180	1	$\boxed{2}$	0	1	
0	$Z_j - C_j$		-2	-3	0	0	

Tabla iteración 1

C_j			2	3	0	0	B_k / A_{ij}
C_j Base	X_j Base	B_k	X_1	X_2	X_3	X_4	
0	X_3						
3	X_2						
0	$Z_j - C_j$						

Actualización (#1)

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	160	2	1	1	0	
0	X_4	180	1	$\boxed{2}$	0	1	
0	$Z_j - C_j$		-2	-3	0	0	

Actualizar valores de la fila pivote:

$$B'_{k_p} = B_{k_p} / A_{i_p j_p}$$
$$A'_{i_p j} = A_{i_p j} / A_{i_p j_p}$$

Valores de la fila pivote

Valor pivote

Actualización (#1)

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	160	2	1	1	0	
0	X_4	180 90	1 0.5	$\boxed{2}$ 1	0 0	1 0.5	
0	$Z_j - C_j$		-2	-3	0	0	

Actualizar valores de la fila pivote: $B'_{k_p} = B_{k_p} / A_{i_p j_p}$

$A'_{i_p j} = A_{i_p j} / A_{i_p j_p}$

Valores de la fila pivote   Valor pivote

Actualización (#1)

Tabla iteración 0

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	160	2	1	1	0	
0	X_4	180	1	2	0	1	
0	$Z_j - C_j$		-2	-3	0	0	

Tabla iteración 1

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3						
3	X_2	90	0.5	1	0	0.5	
Z	$Z_j - C_j$						

Actualización (#1)

C_j			2	3	0	0	B_k / A_{ij}
$\boldsymbol{C_j}$ **Base**	$\boldsymbol{X_j}$ **Base**	$\boldsymbol{B_k}$	X_1	X_2	X_3	X_4	
0	X_3	160	2	1	1	0	
0	X_4	180	1	$\boxed{2}$	0	1	
0	$Z_j - C_j$		-2	-3	0	0	

Actualizar valores del resto de las filas

Actualización (#1)

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	160 70	2	1	1	0	
0	X_4	180	1	2	0	1	
0	$Z_j - C_j$		-2	-3	0	0	

Actualizar valores del resto de las filas:

Valor de la fila pivote

$$B'_k = B_k - \frac{B_{k_p} * A_{i j_p}}{A_{i_p j_p}}$$

Valor de la columna pivote

Valor pivote

Valor a actualizar

Actualización (#1)

Tabla iteración 0

C_j			2	3	0	0	B_k $/A_{ij}$
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	160	2	1	1	0	
0	X_4	180	1	2	0	1	
0	$Z_j - C_j$		-2	-3	0	0	

Tabla iteración 1

C_j			2	3	0	0	B_k $/A_{ij}$
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	70					
3	X_2	90	0.5	1	0	0.5	
Z	$Z_j - C_j$						

Actualización (#1)

C_j			2	3	0	0	B_k / A_{ij}
$\boldsymbol{C_j}$ **Base**	$\boldsymbol{X_j}$ **Base**	$\boldsymbol{B_k}$	X_1	X_2	X_3	X_4	
0	X_3	160	2	1	1	0	
0	X_4	180	1	2	0	1	
0	$Z_j - C_j$		-2	-3	0	0	

Actualizar valores del resto de las filas:

Valor de la fila pivote

$$A'_{ij} = A_{ij} - \frac{A_{i_p j} * A_{i j_p}}{A_{i_p j_p}}$$

Valor de la columna pivote

Valor pivote

Valor a actualizar

Actualización (#1)

C_j			2	3	0	0	B_k $/A_{ij}$
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	160	2	1.5 1	1	0	
0	X_4	180	1	$\boxed{2}$	0	1	
0	$Z_j - C_j$		-2	-3	0	0	

Actualizar valores del resto de las filas:

Valor de la fila pivote

$$A'_{ij} = A_{ij} - \frac{A_{i_p j} * A_{i j_p}}{A_{i_p j_p}}$$

Valor de la columna pivote

Valor pivote

Valor a actualizar

Actualización (#1)

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	160	2 1.5	1 0	1 1	0 -0.5	
0	X_4	180	1	2	0	1	
0	$Z_j - C_j$		-2	-3	0	0	

Actualizar valores del resto de las filas:

Valor de la fila pivote

$$A'_{ij} = A_{ij} - \frac{A_{i_p j} * A_{i j_p}}{A_{i_p j_p}}$$

Valor de la columna pivote

Valor pivote

Valor a actualizar

Actualización (#1)

Tabla iteración 0

C_j			2	3	0	0	B_k/A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	160	2	1	1	0	
0	X_4	180	1	$\boxed{2}$	0	1	
0	$Z_j - C_j$		-2	-3	0	0	

Tabla iteración 1

C_j			2	3	0	0	B_k/A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	70	1.5	0	1	-0.5	
3	X_2	90	0.5	1	0	0.5	
Z	$Z_j - C_j$						

Actualización (#1)

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	160	2	1	1	0	
0	X_4	180	1	$\boxed{2}$	0	1	
0	$Z_j - C_j$		-2	-3	0	0	

Actualizar valores del resto de las filas:

Valor de la fila pivote

Valor de la columna pivote

$$(Z_j - C_j)' = (Z_j - C_j) - \frac{A_{i_p j} * (Z_{j_p} - C_{j_p})}{A_{i_p j_p}}$$

Valor a actualizar

Valor pivote

Actualización (#1)

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	160	2	1	1	0	
0	X_4	180	1	$\boxed{2}$	0	1	
0	$Z_j - C_j$		-2	-3	0	0	

-0.5

Actualizar valores del resto de las filas:

Valor de la fila pivote

Valor de la columna pivote

$$(Z_j - C_j)' = (Z_j - C_j) - \frac{A_{i_p j} * (Z_{j_p} - C_{j_p})}{A_{i_p j_p}}$$

Valor a actualizar

Valor pivote

Actualización (#1)

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	160	2	1	1	0	
0	X_4	180	1	$\boxed{2}$	0	1	
0	$Z_j - C_j$		-2	-3	0	0	
			-0.5	0	0	1.5	

Actualizar valores del resto de las filas:

Valor de la fila pivote

Valor de la columna pivote

$$(Z_j - C_j)' = (Z_j - C_j) - \frac{A_{i_p j} * (Z_{j_p} - C_{j_p})}{A_{i_p j_p}}$$

Valor a actualizar

Valor pivote

Actualización (#1)

Tabla iteración 0

C_j			2	3	0	0	B_k / A_{ij}
C_j Base	X_j Base	B_k	X_1	X_2	X_3	X_4	
0	X_3	160	2	1	1	0	
0	X_4	180	1	$\boxed{2}$	0	1	
0	$Z_j - C_j$		-2	-3	0	0	

Tabla iteración 1

C_j			2	3	0	0	B_k / A_{ij}
C_j Base	X_j Base	B_k	X_1	X_2	X_3	X_4	
0	X_3	70	1.5	0	1	-0.5	
3	X_2	90	0.5	1	0	0.5	
Z	$Z_j - C_j$		-0.5	0	0	1.5	

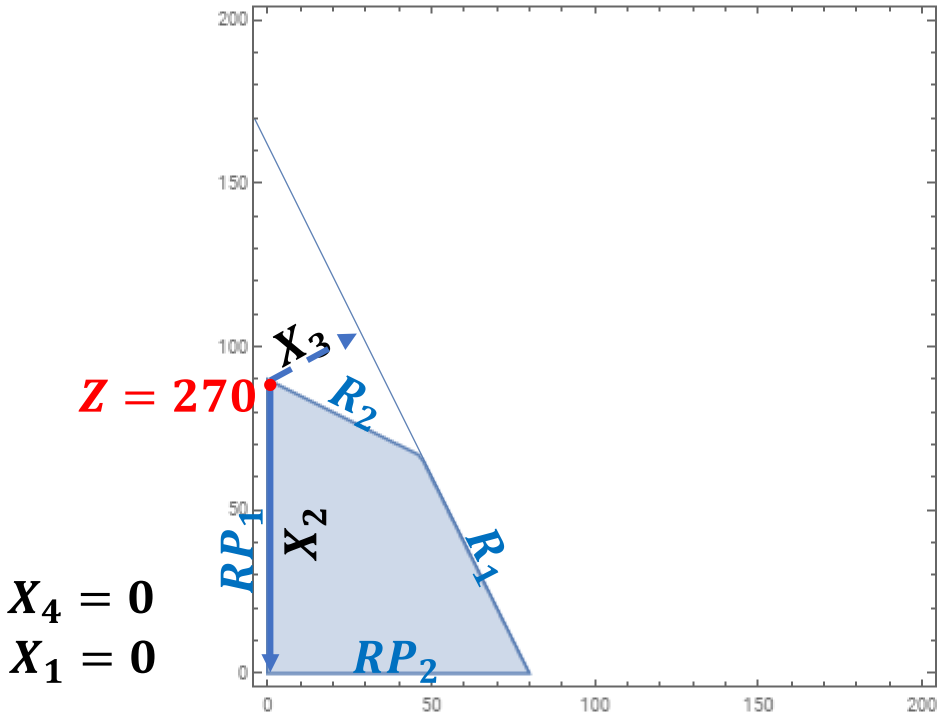
Optimización (#1)

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	70	1.5	0	1	-0.5	
3	X_2	90	0.5	1	0	0.5	
270	$Z_j - C_j$		$\boxed{-0.5}$	0	0	1.5	

$$Z = 0 * 70 + 3 * 90 = 270$$

¡Hay valores negativos, puede mejorar!

Representación gráfica (#1)



Optimización (#1)

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	70	1.5	0	1	-0.5	
3	X_2	90	0.5	1	0	0.5	
270	$Z_j - C_j$		-0.5	0	0	1.5	

X_1 Columna pivote, entra a la base

Optimización (#1)

C_j			2	3	0	0	B_k / A_{ij}
$\boldsymbol{C_j}$ **Base**	$\boldsymbol{X_j}$ **Base**	$\boldsymbol{B_k}$	X_1	X_2	X_3	X_4	
0	X_3	70	1.5	0	1	-0.5	46.67
3	X_2	90	0.5	1	0	0.5	180.00
270	$Z_j - C_j$		-0.5	0	0	1.5	

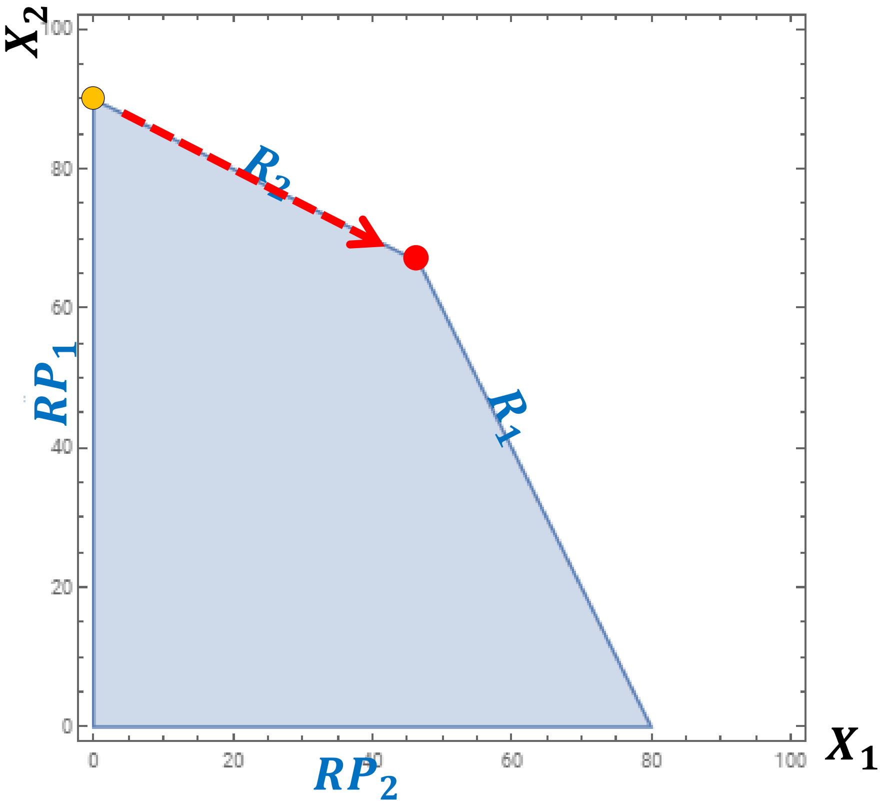
Calculamos B_k / A_{ij}

Optimización (#1)

C_j			2	3	0	0	B_k $/A_{ij}$
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	70	1.5	0	1	-0.5	46.67
3	X_2	90	0.5	1	0	0.5	180.00
270	$Z_j - C_j$		-0.5	0	0	1.5	

El menor positivo B_k $/A_{ij}$ es el saliente, X_3. Entra X_1

Representación gráfica (#1 a #2)



Optimización (#1 a #2)

Tabla iteración 1

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	70	$\boxed{1.5}$	0	1	-0.5	46.67
3	X_2	90	0.5	1	0	0.5	180.00
270	$Z_j - C_j$		-0.5	0	0	1.5	

Tabla iteración 2

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
2	X_1						
3	X_2						
$\boldsymbol{Z}$	$Z_j - C_j$						

Actualización (#2)

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	70 46.67	$\boxed{1.5}$ 1	0 0	1 0.67	-0.5 -0.33	46.67
3	X_2	90	0.5	1	0	0.5	180.00
270	$Z_j - C_j$		-0.5	0	0	1.5	

Actualizamos la fila pivote

$$B'_{k_p} = B_{k_p} / A_{i_p j_p}$$
$$A'_{i_p j} = A_{i_p j} / A_{i_p j_p}$$

Valores de la fila pivote

Valor pivote

Actualización (#2)

Tabla iteración 1

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	70	$\boxed{1.5}$	0	1	-0.5	46.67
3	X_2	90	0.5	1	0	0.5	180.00
270	$Z_j - C_j$		-0.5	0	0	1.5	

Tabla iteración 2

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
2	X_1	46.67	1	0	0.67	-0.33	
3	X_2						
Z	$Z_j - C_j$						

Actualización (#2)

C_j			2	3	0	0	B_k/A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	70	1.5	0	1	-0.5	46.67
3	X_2	90 66.67	0.5 0	1 1	0 -0.33	0.5 0.67	180.00
270	$Z_j - C_j$		-0.5	0	0	1.5	

Actualizamos el resto de las filas:

Valor de la fila pivote

Valor de la columna pivote

$$B'_k = B_k - \frac{B_{k_p} * A_{ij_p}}{A_{i_p j_p}} \qquad A'_{ij} = A_{ij} - \frac{A_{i_p j} * A_{ij_p}}{A_{i_p j_p}}$$

Valor a actualizar

Valor pivote

Actualización (#2)

Tabla iteración 1

C_j			2	3	0	0	B_k $/A_{ij}$
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	70	1.5	0	1	-0.5	46.67
3	X_2	90	0.5	1	0	0.5	180.00
270	$Z_j - C_j$		-0.5	0	0	1.5	

Tabla iteración 2

C_j			2	3	0	0	B_k $/A_{ij}$
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
2	X_1	46.67	1	0	0.67	-0.33	
3	X_2	66.67	0	1	-0.33	0.67	
Z	$Z_j - C_j$						

Actualización (#2)

C_j			2	3	0	0	B_k / A_{ij}
C_j Base	X_j Base	B_k	X_1	X_2	X_3	X_4	
0	X_3	70	$\boxed{1.5}$	0	1	-0.5	46.67
3	X_2	90	0.5	1	0	0.5	180.00
270	$Z_j - C_j$		-0.5	0	0	1.5	
			0	0	0.33	1.33	

Actualizamos el resto de las filas:

Valor de la fila pivote

Valor de la columna pivote

$$(Z_j - C_j)' = (Z_j - C_j) - \frac{A_{i_p j} * (Z_{j_p} - C_{j_p})}{A_{i_p j_p}}$$

Valor a actualizar

Valor pivote

Actualización (#2)

Tabla iteración 1

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
0	X_3	70	1.5	0	1	-0.5	46.67
3	X_2	90	0.5	1	0	0.5	180.00
270	$Z_j - C_j$		-0.5	0	0	1.5	

Tabla iteración 2

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
2	X_1	46.67	1	0	0.67	-0.33	
3	X_2	66.67	0	1	-0.33	0.67	
$\boldsymbol{Z}$	$Z_j - C_j$		0	0	0.33	1.33	

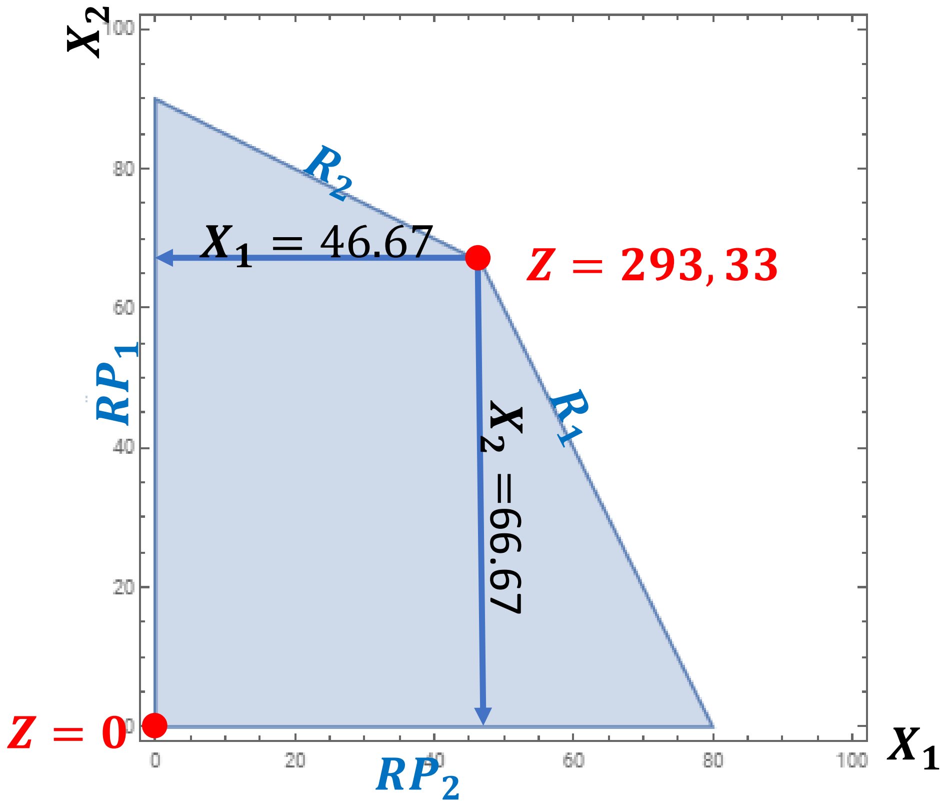
Optimización (#2)

C_j			2	3	0	0	B_k / A_{ij}
C_j **Base**	X_j **Base**	B_k	X_1	X_2	X_3	X_4	
2	X_1	46.67	1	0	0.67	-0.33	
3	X_2	66.67	0	1	-0.33	0.67	
$\mathbf{293.33}$	$Z_j - C_j$		0	0	0.33	1.33	

$$\boldsymbol{Z = 2 * 46.67 + 3 * 66.67 = 293.33}$$

No hay valores negativos, las variables slack salieron de la base, ¡es el óptimo!

Representación gráfica (#2)



$X_3 = 0$
$X_4 = 0$

Conclusión

Dado el modelo formulado, bajo las suposiciones tomadas al principio:

Se logró maximizar la solución para cantidades de producto A y B de $X_1^* = 46.67$ y $X_2^* = 66.67$ respectivamente; con un ingreso máximo de $Z^* = \$\,293.33$

Check con Python PuLP

```
# Problema de producción
import pulp

lp01 = pulp.LpProblem("Intro SIMPLEX", pulp.LpMaximize)

# Variables:
x = pulp.LpVariable('x', lowBound=0, cat='Continuous')
y = pulp.LpVariable('y', lowBound=0, cat='Continuous')

# Función objetivo:
lp01 += 2*x + 3*y, "Z"

# Restricciones:
lp01 +=  2*x + 1*y ⩽ 160
lp01 +=  1*x + 2*y ⩽ 180
```

```
# Resolvemos:
lp01.solve()

# Imprimimos status:
pulp.LpStatus[lp01.status]
print(pulp.LpStatus[lp01.status])

# Imprimimos variables:
for variable in lp01.variables():
    print("%s = %.2f" % (variable.name,
variable.varValue))

# Imprimimos objetivo:
print(pulp.value(lp01.objective))
```

```
>> Optimal
>> x = 46.67
>> y = 66.67
>> 293.333335
```