Ejemplos de CPLEX (y algo de Simplex)

Programación Lineal - Problema de la dieta

- Motivación: Para mantener una dieta equilibrada el nutricionista de María le recomendó consumir diarimente una cierta cantidad de calorías, vitaminas, minerales, grasas, sodio y colesterol. Como María es muy ahorrativa, quiere cumplir lo recomendado por el nutricionista pero gastando lo menos posible.

Programación Lineal - Problema de la dieta

- Motivación: Para mantener una dieta equilibrada el nutricionista de María le recomendó consumir diarimente una cierta cantidad de calorías, vitaminas, minerales, grasas, sodio y colesterol. Como María es muy ahorrativa, quiere cumplir lo recomendado por el nutricionista pero gastando lo menos posible.
- Planteo: Dado un conjunto de alimentos, la información de nutrientes y el costo por porción de cada uno, el objetivo es seleccionar la cantidad de porciones de cada alimento para comprar (y consumir) a fin de minimizar el costo de la comida, satisfaciendo las necesidades nutricionales determinadas. Los requerimientos nutricionales se expresan como un nivel máximo o mínimo permisible. También para mejorar la calidad del menú, para ciertos alimentos es posible espedificar un número mínimo y/o máximo de porciones que se pueden incluir.

Programación Lineal - Problema de la dieta

Ejemplo:

Supongamos la nutricionista nos recomendó consumir diariamente entre 2000 y 2300 calorías y al menos 1200 mgr de calcio, y que contamos con 3 tipos de alimentos, maíz, leche y pan. La tabla muestra la información nutricional de cada alimento y su costo por porción.

Alimento	Calorías	Calcio	Costo
maíz	170	3	1.8
leche	50	400	2.3
pan	300	40	1.5

También sabemos que no debemos consumir más de 3 porciones de pan diariamente y por lo menos queremos que nuestra dieta tenga 2 porciones de leche.

Programación Lineal - Problema de la dieta

Representamos con:

- x_m a la cantidad de porciones diarias de maíz de nuestra dieta
- x_l a la cantidad de porciones diarias de leche de nuestra dieta
- x_p a la cantidad de porciones diarias de pan de nuestra dieta

Programación Lineal - Problema de la dieta

Representamos con:

- x_m a la cantidad de porciones diarias de maíz de nuestra dieta
- x_l a la cantidad de porciones diarias de leche de nuestra dieta
- x_p a la cantidad de porciones diarias de pan de nuestra dieta

¿Cuánto gastamos?

Programación Lineal - Problema de la dieta

Representamos con:

- x_m a la cantidad de porciones diarias de maíz de nuestra dieta
- x_l a la cantidad de porciones diarias de leche de nuestra dieta
- x_p a la cantidad de porciones diarias de pan de nuestra dieta

¿Cuánto gastamos?

$$1.8x_m + 2.3x_l + 1.5x_p$$

Programación Lineal - Problema de la dieta

Representamos con:

- x_m a la cantidad de porciones diarias de maíz de nuestra dieta
- x_l a la cantidad de porciones diarias de leche de nuestra dieta
- x_p a la cantidad de porciones diarias de pan de nuestra dieta

Programación Lineal - Problema de la dieta

Representamos con:

- x_m a la cantidad de porciones diarias de maíz de nuestra dieta
- x_l a la cantidad de porciones diarias de leche de nuestra dieta
- x_p a la cantidad de porciones diarias de pan de nuestra dieta

¿Cómo aseguramos cumplir el requerimiento mínimo de calorías?

Programación Lineal - Problema de la dieta

Representamos con:

- x_m a la cantidad de porciones diarias de maíz de nuestra dieta
- x_l a la cantidad de porciones diarias de leche de nuestra dieta
- x_p a la cantidad de porciones diarias de pan de nuestra dieta

¿Cómo aseguramos cumplir el requerimiento mínimo de calorías?

$$170x_m + 50x_l + 300x_p \geq 2000$$

Programación Lineal - Problema de la dieta

Representamos con:

- x_m a la cantidad de porciones diarias de maíz de nuestra dieta
- x_l a la cantidad de porciones diarias de leche de nuestra dieta
- x_p a la cantidad de porciones diarias de pan de nuestra dieta

¿Cómo aseguramos cumplir el requerimiento mínimo de calorías?

$$170x_m + 50x_l + 300x_p \geq 2000$$

¿Y el máximo?

Programación Lineal - Problema de la dieta

Representamos con:

- x_m a la cantidad de porciones diarias de maíz de nuestra dieta
- x_l a la cantidad de porciones diarias de leche de nuestra dieta
- x_p a la cantidad de porciones diarias de pan de nuestra dieta

¿Cómo aseguramos cumplir el requerimiento mínimo de calorías?

$$170x_m + 50x_l + 300x_p \geq 2000$$

¿Y el máximo?

$$170x_m + 50x_l + 300x_p \leq 2300$$

Programación Lineal - Problema de la dieta

Representamos con:

- x_m a la cantidad de porciones diarias de maíz de nuestra dieta
- x_l a la cantidad de porciones diarias de leche de nuestra dieta
- x_p a la cantidad de porciones diarias de pan de nuestra dieta

Programación Lineal - Problema de la dieta

Representamos con:

- x_m a la cantidad de porciones diarias de maíz de nuestra dieta
- x_l a la cantidad de porciones diarias de leche de nuestra dieta
- x_p a la cantidad de porciones diarias de pan de nuestra dieta

¿Cómo aseguramos el requerimiento mínimo de calcio?

Programación Lineal - Problema de la dieta

Representamos con:

- x_m a la cantidad de porciones diarias de maíz de nuestra dieta
- x_l a la cantidad de porciones diarias de leche de nuestra dieta
- x_p a la cantidad de porciones diarias de pan de nuestra dieta

¿Cómo aseguramos el requerimiento mínimo de calcio?

$$3x_m + 400x_l + 40x_p \geq 1200$$

Programación Lineal - Problema de la dieta

Representamos con:

- x_m a la cantidad de porciones diarias de maíz de nuestra dieta
- x_l a la cantidad de porciones diarias de leche de nuestra dieta
- x_p a la cantidad de porciones diarias de pan de nuestra dieta

Programación Lineal - Problema de la dieta

Representamos con:

- x_m a la cantidad de porciones diarias de maíz de nuestra dieta
- x_l a la cantidad de porciones diarias de leche de nuestra dieta
- x_p a la cantidad de porciones diarias de pan de nuestra dieta

¿Y que no tengamos más de 3 porciones diarias de pan?

Programación Lineal - Problema de la dieta

Representamos con:

- x_m a la cantidad de porciones diarias de maíz de nuestra dieta
- x_l a la cantidad de porciones diarias de leche de nuestra dieta
- x_p a la cantidad de porciones diarias de pan de nuestra dieta

¿Y que no tengamos más de 3 porciones diarias de pan?

$$x_p \leq 3$$

Programación Lineal - Problema de la dieta

Representamos con:

- x_m a la cantidad de porciones diarias de maíz de nuestra dieta
- x_l a la cantidad de porciones diarias de leche de nuestra dieta
- x_p a la cantidad de porciones diarias de pan de nuestra dieta

¿Y que no tengamos más de 3 porciones diarias de pan?

$$x_p \leq 3$$

¿Y por lo menos 2 de leche?

Programación Lineal - Problema de la dieta

Representamos con:

- x_m a la cantidad de porciones diarias de maíz de nuestra dieta
- x_l a la cantidad de porciones diarias de leche de nuestra dieta
- x_p a la cantidad de porciones diarias de pan de nuestra dieta

¿Y que no tengamos más de 3 porciones diarias de pan?

$$x_p \leq 3$$

¿Y por lo menos 2 de leche?

$$x_l \geq 2$$

Programación Lineal - Problema de la dieta

Representamos con:

- x_m a la cantidad de porciones diarias de maíz de nuestra dieta
- x_l a la cantidad de porciones diarias de leche de nuestra dieta
- x_p a la cantidad de porciones diarias de pan de nuestra dieta

¿Y que no tengamos más de 3 porciones diarias de pan?

$$x_p \leq 3$$

¿Y por lo menos 2 de leche?

$$x_l \geq 2$$

Y por supuesto que la cantidad de porciones de maíz, pan y leche debe ser positiva:

Programación Lineal - Problema de la dieta

Representamos con:

- x_m a la cantidad de porciones diarias de maíz de nuestra dieta
- x_l a la cantidad de porciones diarias de leche de nuestra dieta
- x_p a la cantidad de porciones diarias de pan de nuestra dieta

¿Y que no tengamos más de 3 porciones diarias de pan?

$$x_p \leq 3$$

¿Y por lo menos 2 de leche?

$$x_l \geq 2$$

Y por supuesto que la cantidad de porciones de maíz, pan y leche debe ser positiva:

$$x_m, x_l, x_p \geq 0$$

Programación Lineal - Problema de la dieta

Representamos con:

- x_m a la cantidad de porciones diarias de maíz de nuestra dieta
- x_l a la cantidad de porciones diarias de leche de nuestra dieta
- x_p a la cantidad de porciones diarias de pan de nuestra dieta

Programación Lineal - Problema de la dieta

Representamos con:

- x_m a la cantidad de porciones diarias de maíz de nuestra dieta
- x_l a la cantidad de porciones diarias de leche de nuestra dieta
- x_p a la cantidad de porciones diarias de pan de nuestra dieta

$$
\begin{array}{llcll}
\text{Minimizar} & 1.8x_m + 2.3x_l + 1.5x_p & & & \text{costo de compra} \\
\text{Sujeto a} & 170x_m + 50x_l + 300x_p & \geq & 2000 & \text{min. calorías} \\
 & 170x_m + 50x_l + 300x_p & \leq & 2300 & \text{max. calorías} \\
 & 3x_m + 400x_l + 40x_p & \geq & 1200 & \text{min. calcio} \\
 & x_p & \leq & 3 & \text{max. porciones pan} \\
 & x_l & \geq & 2 & \text{min. porciones leche} \\
 & x_m, x_l, x_p & \geq & 0 &
\end{array}
$$

Programación Lineal - ¿Qué es un Programa Lineal?

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

Sujeto a

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i \quad i = 1, \ldots, m$$

$$l_j \leq x_j \leq u_j \quad j = 1, \ldots, n$$

donde $c_j, a_{ij}, b_i \in \mathbb{R}$ para $j = 1, \ldots, n$ y $i = 1, \ldots, m$.

Programación Lineal - ¿Qué es un Programa Lineal?

- n variables de decisión:

$$x_1, \ldots, x_n$$

- función objetivo:

$$f(x) = \sum_{j=1}^{n} c_j x_j$$

- m restricciones:

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i \qquad i = 1, \ldots, m$$

  Las restricciones pueden ser por mayor o igual, por menor o igual o igualdades.

- restricciones de cota:

$$l_j \leq x_j \leq u_j \qquad j = 1, \ldots, n$$

  Las variables pueden ser no negativas y/o acotadas.

Programación Lineal - ¿Qué es un Programa Lineal?

- solución factible:

  $x = (x_1, \ldots, x_n)$ satisface las m desigualdades y las cotas

- región facible (poliedro convexo):

$$P = \{x \in R^n : x \text{ es factible}\}$$

- solución óptima (problema de minimización):

$$x^* = (x_1^*, \ldots, x_n^*) \in P \text{ tal que } f(x^*) \leq f(x) \qquad \forall x \in P$$

- valor óptimo:

$$z^* = f(x^*)$$

Programación Lineal - Asunciones

- Proporcionalidad: cuando el nivel de una actividad es multiplicada por un factor, entonces la contribución de esa actividad a la función objetivo o a cualquiera de las restricciones donde aparece es multiplicada por el mismo factor.

- Adición: la suma de las contribuciones de las actividades es igual a la contribución total, tanto en la función objetivo como en las restricciones, independientemente de los niveles de las otras actividades.

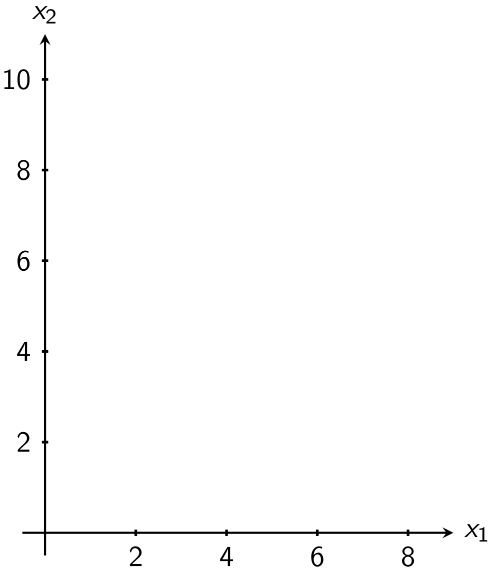
- Divisibilidad: Los niveles de las actividades pueden tomar valores enteros o no enteros.

Programación Lineal - Resolución gráfica

$$
\begin{array}{lrcl}
\text{Max} & z = 3x_1 + 2x_2 & & \\
\text{s.a.} & 2x_1 + x_2 & \leq & 10 \\
 & x_1 + x_2 & \leq & 8 \\
 & x_1 & \leq & 4 \\
 & x_1, x_2 & \geq & 0
\end{array}
$$

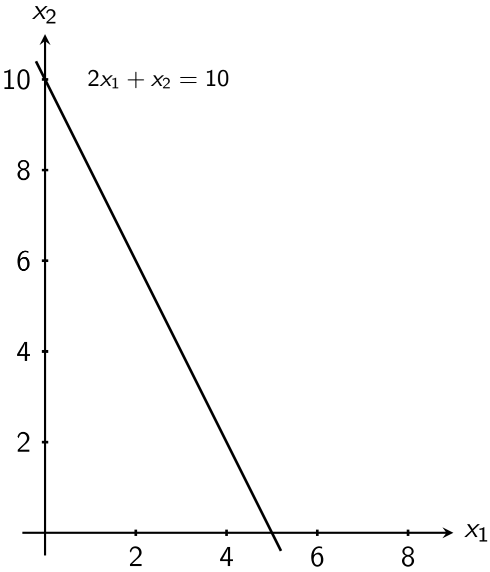
Programación Lineal - Resolución gráfica

$$
\begin{array}{llcl}
\text{Max} & z = 3x_1 + 2x_2 & & \\
\text{s.a.} & 2x_1 + x_2 & \leq & 10 \\
 & x_1 + x_2 & \leq & 8 \\
 & x_1 & \leq & 4 \\
 & x_1, x_2 & \geq & 0
\end{array}
$$



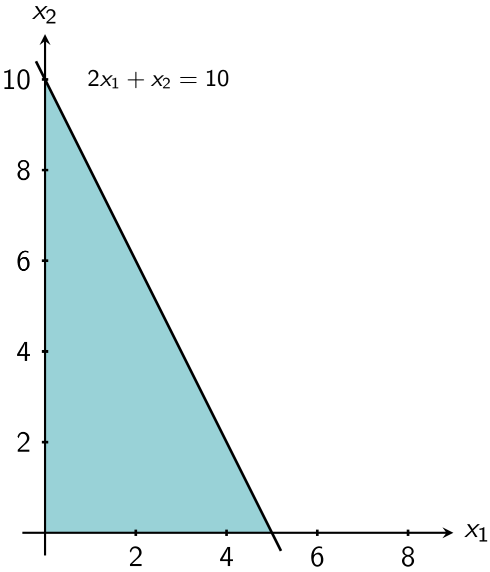
Programación Lineal - Resolución gráfica

$$\begin{array}{lrcl} \text{Max} & z = 3x_1 + 2x_2 & & \\ \text{s.a.} & 2x_1 + x_2 & \leq & 10 \\ & x_1 + x_2 & \leq & 8 \\ & x_1 & \leq & 4 \\ & x_1, x_2 & \geq & 0 \end{array}$$



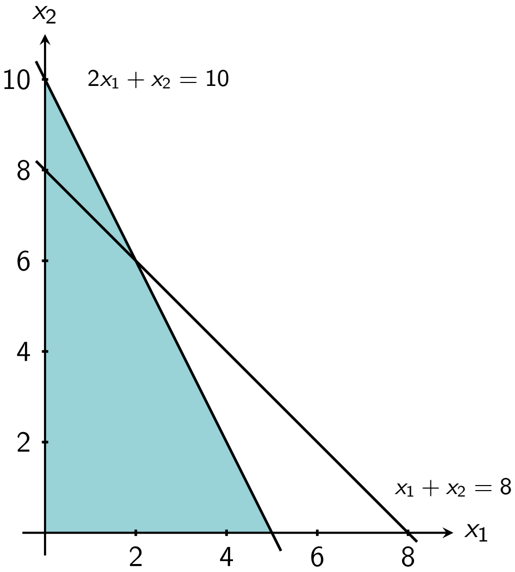
Programación Lineal - Resolución gráfica

$$
\begin{array}{lrcl}
\text{Max} & z = 3x_1 + 2x_2 & & \\
\text{s.a.} & 2x_1 + x_2 & \leq & 10 \\
 & x_1 + x_2 & \leq & 8 \\
 & x_1 & \leq & 4 \\
 & x_1, x_2 & \geq & 0
\end{array}
$$



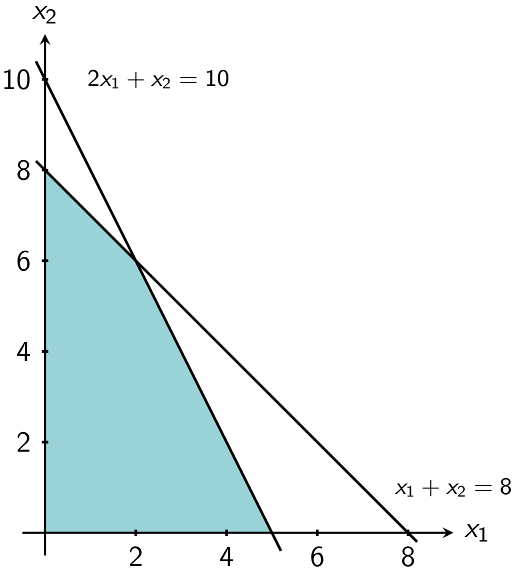
Programación Lineal - Resolución gráfica

$$
\begin{array}{lrcl}
\text{Max} & z = 3x_1 + 2x_2 & & \\
\text{s.a.} & 2x_1 + x_2 & \leq & 10 \\
 & x_1 + x_2 & \leq & 8 \\
 & x_1 & \leq & 4 \\
 & x_1, x_2 & \geq & 0
\end{array}
$$



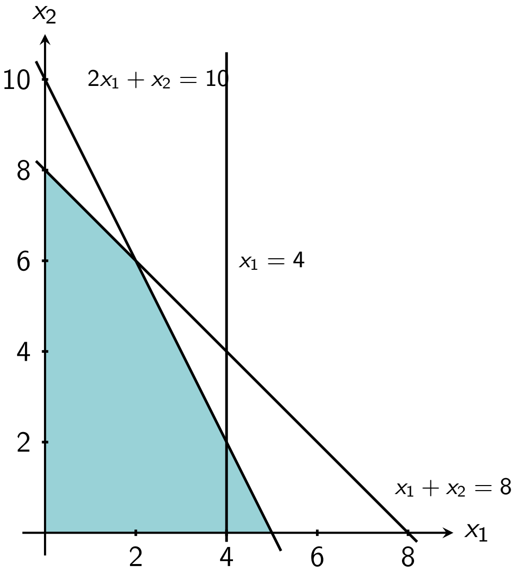
Programación Lineal - Resolución gráfica

$$
\begin{array}{lrcl}
\text{Max} & z = 3x_1 + 2x_2 & & \\
\text{s.a.} & 2x_1 + x_2 & \leq & 10 \\
 & x_1 + x_2 & \leq & 8 \\
 & x_1 & \leq & 4 \\
 & x_1, x_2 & \geq & 0
\end{array}
$$



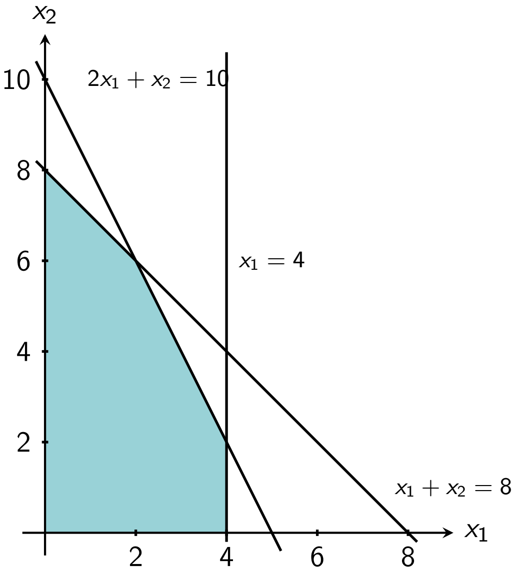
Programación Lineal - Resolución gráfica

$$\begin{array}{lrcl} \text{Max} & z = 3x_1 + 2x_2 & & \\ \text{s.a.} & 2x_1 + x_2 & \leq & 10 \\ & x_1 + x_2 & \leq & 8 \\ & x_1 & \leq & 4 \\ & x_1, x_2 & \geq & 0 \end{array}$$



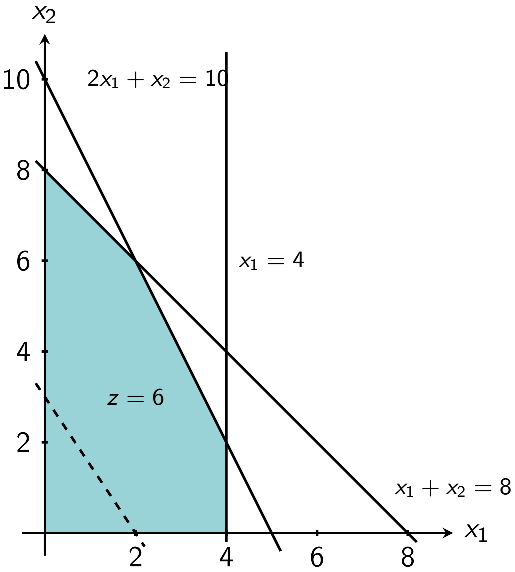
Programación Lineal - Resolución gráfica

$$\begin{array}{llcl} \text{Max} & z = 3x_1 + 2x_2 & & \\ \text{s.a.} & 2x_1 + x_2 & \leq & 10 \\ & x_1 + x_2 & \leq & 8 \\ & x_1 & \leq & 4 \\ & x_1, x_2 & \geq & 0 \end{array}$$



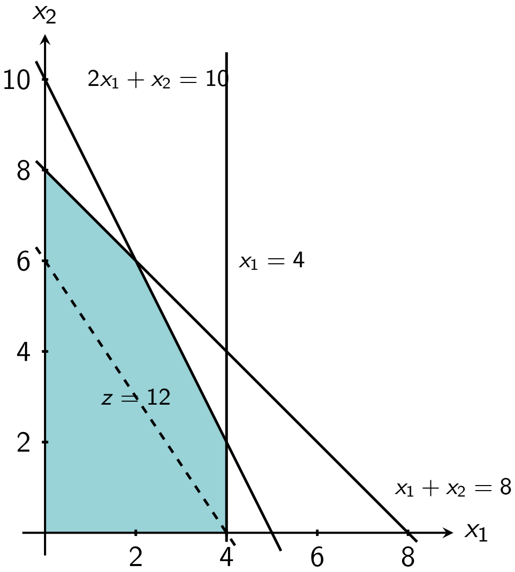
Programación Lineal - Resolución gráfica

$$
\begin{array}{llcl}
\text{Max} & z = 3x_1 + 2x_2 & & \\
\text{s.a.} & 2x_1 + x_2 & \leq & 10 \\
 & x_1 + x_2 & \leq & 8 \\
 & x_1 & \leq & 4 \\
 & x_1, x_2 & \geq & 0
\end{array}
$$



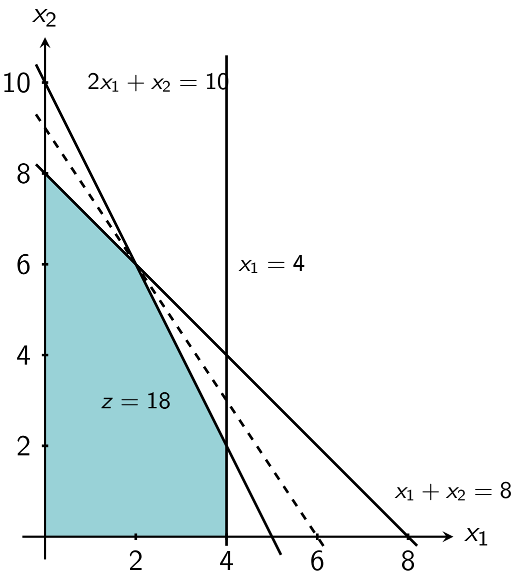
Programación Lineal - Resolución gráfica

$$\begin{array}{llcl} \text{Max} & z = 3x_1 + 2x_2 & & \\ \text{s.a.} & 2x_1 + x_2 & \leq & 10 \\ & x_1 + x_2 & \leq & 8 \\ & x_1 & \leq & 4 \\ & x_1, x_2 & \geq & 0 \end{array}$$



Programación Lineal - Resolución gráfica

$$
\begin{array}{llcl}
\text{Max} & z = 3x_1 + 2x_2 & & \\
\text{s.a.} & 2x_1 + x_2 & \leq & 10 \\
 & x_1 + x_2 & \leq & 8 \\
 & x_1 & \leq & 4 \\
 & x_1, x_2 & \geq & 0
\end{array}
$$

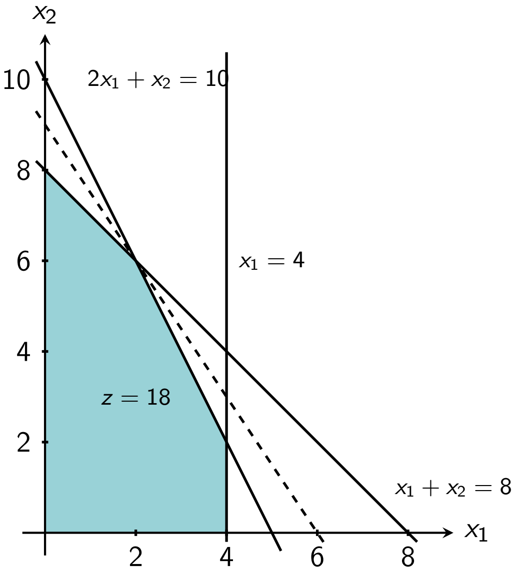


Programación Lineal - Resolución gráfica

$$\begin{array}{lrcl} \text{Max} & z = 3x_1 + 2x_2 & & \\ \text{s.a.} & 2x_1 + x_2 & \leq & 10 \\ & x_1 + x_2 & \leq & 8 \\ & x_1 & \leq & 4 \\ & x_1, x_2 & \geq & 0 \end{array}$$

$$(x_1^*, x_2^*) = (2, 6)$$

$$z^* = 18$$



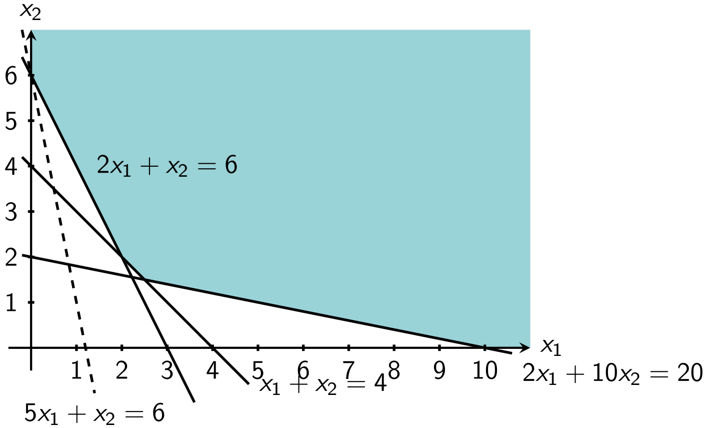
Programación Lineal - Alternativas

Un PL puede

- tener único óptimo
- tener infinitos óptimos
- no tener óptimo:
  - no tiene soluciones factibles
  - ser no acotado

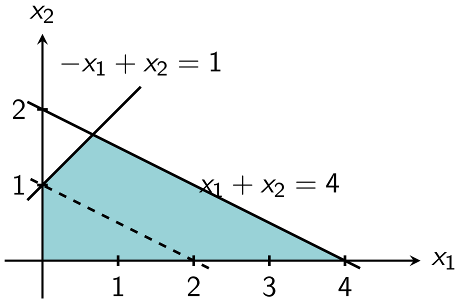
Programación Lineal - único óptimo

$$
\begin{array}{lrl}
\text{Min} & 5\,x_1 + x_2 & \\
\text{s.a.} & 2\,x_1 + x_2 & \geq 6 \\
& x_1 + x_2 & \geq 4 \\
& 2\,x_1 + 10 x_2 & \geq 20 \\
& x_1, \quad x_2 & \geq 0
\end{array}
$$



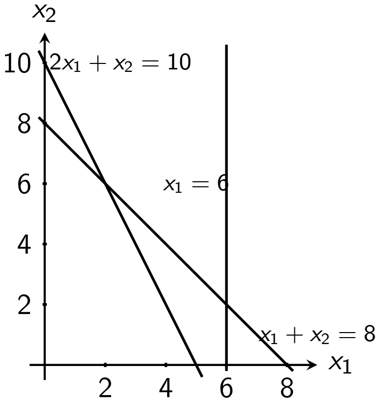
Programación Lineal - infinitos óptimos

$$\begin{array}{llr} \text{Max} & 2\ x_1 + 4x_2 & \\ \text{s.a.} & x_1 + 2x_2 \le 4 & \\ & -\ x_1 + \ x_2 \le 1 & \\ & x_1, \quad x_2 \ge 0 & \end{array}$$



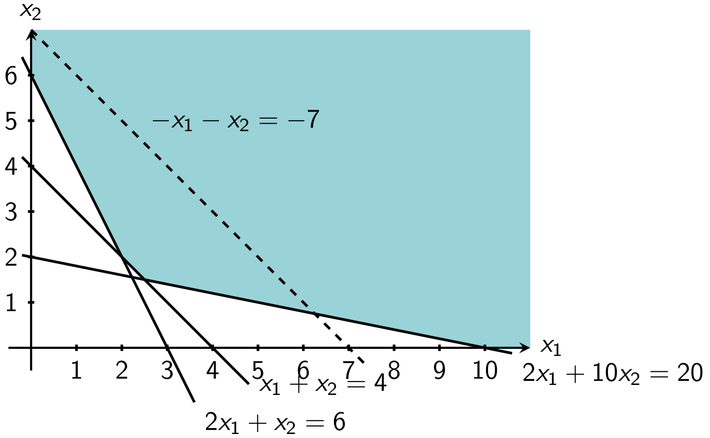
Programación Lineal - no factible

$$
\begin{array}{lrcl}
\max & z = 3x_1 + 2x_2 & & \\
\text{s.a.} & 2x_1 + x_2 & \leq & 10 \\
 & x_1 + x_2 & \leq & 8 \\
 & x_1 & \geq & 6 \\
 & x_1, x_2 & \geq & 0
\end{array}
$$



Programación Lineal - no acotado

$$\begin{array}{rl} \text{Max} & x_1 + x_2 \\ \text{s.a.} & 2x_1 + x_2 \geq 6 \\ & x_1 + x_2 \geq 4 \\ & 2x_1 + 10x_2 \geq 20 \\ & x_1, \quad x_2 \geq 0 \end{array}$$



Programación Lineal - Algoritmos

- SIMPLEX (George Dantzig 1947):
  Peor caso exponencial. Recorre los extremos del poliedro. Problemas con miles de variables y restricciones pueden ser resueltos en tiempo computacionales razonables. En la práctica es el algoritmo más utilizado.

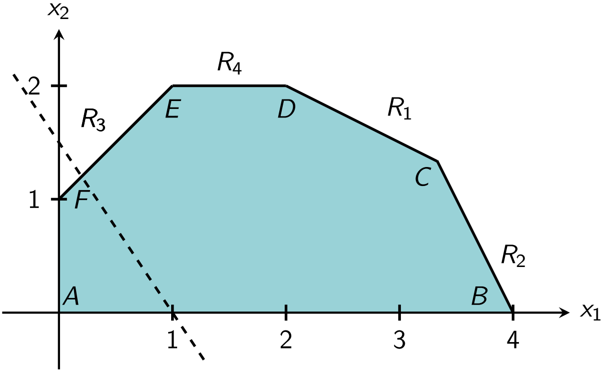
- Algoritmo de punto interior - Método del elipsoide (Leonid Khachiyan 1979):
  Polinomial en el peor caso. Se mueve en el interior del poliedro. No eficiente en la práctica.

- Algoritmo de punto interior - Método proyectivo (Narendra Karmarkar 1984):
  Polinomial en el peor caso. Según el tipo específico de LP es más eficiente que el método SIMPLEX en la práctica.

Programación Lineal - SIMPLEX (maximización)

Ilustremos el simplex con el siguinte ejemplo:

$$
(1) \qquad \begin{array}{rrcrcr}
\text{Max} & 3x_1 & + & 2x_2 & & \\
\text{sujeto a} & x_1 & + & 2x_2 & \leq & 6 \\
 & 2x_1 & + & x_2 & \leq & 8 \\
 & -x_1 & + & x_2 & \leq & 1 \\
 & & & x_2 & \leq & 2 \\
 & & & x_1, x_2 \geq 0 & &
\end{array}
$$



Programación Lineal - SIMPLEX (maximización)

Ilustremos el simplex con el siguinte ejemplo:

$$
(1) \qquad \begin{array}{rrcrcr}
\text{Max} & 3x_1 & + & 2x_2 & & \\
\text{sujeto a} & x_1 & + & 2x_2 & \leq & 6 \\
 & 2x_1 & + & x_2 & \leq & 8 \\
 & -x_1 & + & x_2 & \leq & 1 \\
 & & & x_2 & \leq & 2 \\
 & & & x_1, x_2 \geq 0 & &
\end{array}
$$

Programación Lineal - SIMPLEX (maximización)

Ilustremos el simplex con el siguinte ejemplo:

$$
(1) \qquad \begin{array}{rrcrcr}
\text{Max} & 3x_1 & + & 2x_2 & & \\
\text{sujeto a} & x_1 & + & 2x_2 & \leq & 6 \\
 & 2x_1 & + & x_2 & \leq & 8 \\
 & -x_1 & + & x_2 & \leq & 1 \\
 & & & x_2 & \leq & 2 \\
 & & x_1, x_2 \geq 0 & & &
\end{array}
$$

- Convertimos las restricciones en igualdades (salvo las restricciones de no negatividad), mediante la introducción de unas variables que llamaremos variables de holgura. Cada variable de holgura representa la diferencia entre el lado derecho de una restricción y el término independiente correspondiente.

Por ejemplo, la restricción

$$2x_1 + x_2 \leq 8$$

es equivalente al par de restricciones

$$2x_1 + x_2 + x_3 = 8 \quad \text{y} \quad x_3 \geq 0.$$

Programación Lineal - SIMPLEX (maximización)

El sistema de restricciones puede reescribirse de la siguiente manera:

$$
\begin{array}{rrcrcrcrcrcrcr}
\text{Max} & 3x_1 & + & 2x_2 & & & & & & & & & & \\
\text{sujeto a} & x_1 & + & 2x_2 & + & x_3 & & & & & & & = & 6 \\
(2) & 2x_1 & + & x_2 & & & + & x_4 & & & & & = & 8 \\
 & -x_1 & + & x_2 & & & & & + & x_5 & & & = & 1 \\
 & & & x_2 & & & & & & & + & x_6 & = & 2 \\
\end{array}
$$

$$x_1, x_2, x_3, x_4, x_5, x_6 \geq 0$$

- A las variables x_3, x_4, x_5 y x_6 las denominamos variables de holgura y a x_1 y x_2 variables de decisión.

Programación Lineal - SIMPLEX (maximización)

El sistema de restricciones puede reescribirse de la siguiente manera:

$$
\begin{array}{rrcrcrcrcrcrcr}
\text{Max} & 3x_1 & + & 2x_2 & & & & & & & & & & \\
\text{sujeto a} & x_1 & + & 2x_2 & + & x_3 & & & & & & & = & 6 \\
(2) & 2x_1 & + & x_2 & & & + & x_4 & & & & & = & 8 \\
 & -x_1 & + & x_2 & & & & & + & x_5 & & & = & 1 \\
 & & & x_2 & & & & & & & + & x_6 & = & 2 \\
\end{array}
$$

$$x_1, x_2, x_3, x_4, x_5, x_6 \geq 0$$

- A las variables x_3, x_4, x_5 y x_6 las denominamos variables de holgura y a x_1 y x_2 variables de decisión.
- Ambas forman de representar el modelo son equivalentes, es decir, cada solución de (1) puede ser extendida a una solución de (2) y viceversa.

Programación Lineal - SIMPLEX (maximización)

- La estrategia del método **simplex** es iterar consiguiendo mejores soluciones para la función objetivo. Dada una solución factible $x_1 \dots x_5$ se trata de hallar otra $\tilde{x_1} \dots \tilde{x_5}$ tal que

$$3\tilde{x_1} + 2\tilde{x_2} > 3x_1 + 2x_2.$$

Programación Lineal - SIMPLEX (maximización)

- La estrategia del método **simplex** es iterar consiguiendo mejores soluciones para la función objetivo. Dada una solución factible $x_1 \dots x_5$ se trata de hallar otra $\tilde{x_1} \dots \tilde{x_5}$ tal que

$$3\tilde{x_1} + 2\tilde{x_2} > 3x_1 + 2x_2.$$

- Repitiendo este proceso un número finito de veces se tratará de llegar a una solución óptima.

Programación Lineal - SIMPLEX (maximización)

- La estrategia del método **simplex** es iterar consiguiendo mejores soluciones para la función objetivo. Dada una solución factible $x_1 \dots x_5$ se trata de hallar otra $\tilde{x_1} \dots \tilde{x_5}$ tal que

$$3\tilde{x_1} + 2\tilde{x_2} > 3x_1 + 2x_2.$$

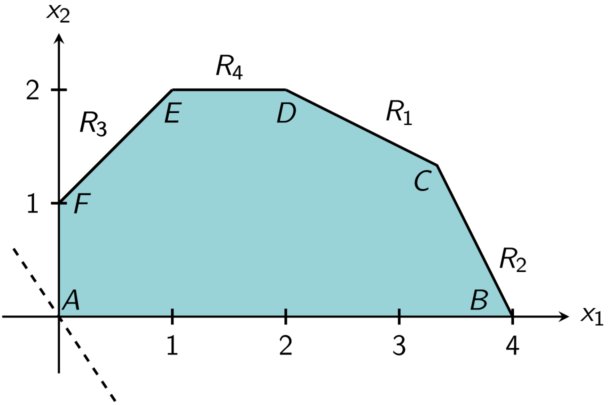
- Repitiendo este proceso un número finito de veces se tratará de llegar a una solución óptima.
- Para comenzar, se necesita una solución factible. En nuestro caso, basta tomar

$$\begin{array}{rclcrcl} x_1 & = & 0 & \qquad & x_2 & = & 0 \\ x_3 & = & 6 & & x_4 & = & 8 \\ x_5 & = & 1 & & x_6 & = & 2 \\ z & = & 0. & & & & \end{array}$$

Programación Lineal - SIMPLEX (maximización)

Solución A:

$$
\begin{aligned}
x_1 &= 0 & \qquad x_2 &= 0 \\
x_3 &= 6 & \qquad x_4 &= 8 \\
x_5 &= 1 & \qquad x_6 &= 2 \\
z &= 0.
\end{aligned}
$$



Programación Lineal - SIMPLEX (maximización)

- Buscamos una solución factible con mayor valor de z.

Por ejemplo, si mantenemos el valor de $x_2 = 0$ e incrementamos x_1 se obtiene $z = 3x_1$.

¿Pero cuánto podemos incrementar x_1?

Programación Lineal - SIMPLEX (maximización)

- Buscamos una solución factible con mayor valor de z.

Por ejemplo, si mantenemos el valor de $x_2 = 0$ e incrementamos x_1 se obtiene $z = 3x_1$.

¿Pero cuánto podemos incrementar x_1?

- $x_1 = 1 \longrightarrow x_3 = 5 \qquad x_4 = \quad 6 \qquad x_5 = 2 \qquad x_6 = 2 \qquad z = 3$

Programación Lineal - SIMPLEX (maximización)

- Buscamos una solución factible con mayor valor de z.

Por ejemplo, si mantenemos el valor de $x_2 = 0$ e incrementamos x_1 se obtiene $z = 3x_1$.

¿Pero cuánto podemos incrementar x_1?

- $x_1 = 1 \longrightarrow x_3 = 5 \qquad x_4 = \quad 6 \qquad x_5 = 2 \qquad x_6 = 2 \qquad z = 3$
- $x_1 = 3 \longrightarrow x_3 = 3 \qquad x_4 = \quad 2 \qquad x_5 = 4 \qquad x_6 = 2 \qquad z = 9$

Programación Lineal - SIMPLEX (maximización)

- Buscamos una solución factible con mayor valor de z.

Por ejemplo, si mantenemos el valor de $x_2 = 0$ e incrementamos x_1 se obtiene $z = 3x_1$.

¿Pero cuánto podemos incrementar x_1?

- $x_1 = 1 \longrightarrow x_3 = 5 \quad x_4 = \quad 6 \quad x_5 = 2 \quad x_6 = 2 \quad z = 3$
- $x_1 = 3 \longrightarrow x_3 = 3 \quad x_4 = \quad 2 \quad x_5 = 4 \quad x_6 = 2 \quad z = 9$
- $x_1 = 4 \longrightarrow x_3 = 2 \quad x_4 = \quad 0 \quad x_5 = 5 \quad x_6 = 2 \quad z = 12$

Programación Lineal - SIMPLEX (maximización)

- Buscamos una solución factible con mayor valor de z.

Por ejemplo, si mantenemos el valor de $x_2 = 0$ e incrementamos x_1 se obtiene $z = 3x_1$.

¿Pero cuánto podemos incrementar x_1?

- $x_1 = 1 \longrightarrow x_3 = 5 \quad x_4 = 6 \quad x_5 = 2 \quad x_6 = 2 \quad z = 3$
- $x_1 = 3 \longrightarrow x_3 = 3 \quad x_4 = 2 \quad x_5 = 4 \quad x_6 = 2 \quad z = 9$
- $x_1 = 4 \longrightarrow x_3 = 2 \quad x_4 = 0 \quad x_5 = 5 \quad x_6 = 2 \quad z = 12$
- $x_1 = 5 \longrightarrow x_3 = 1 \quad x_4 = -2 \quad x_5 = 6 \quad x_6 = 2 \quad z = 15$

Programación Lineal - SIMPLEX (maximización)

- En el último caso, perdemos factibilidad, porque x_4 toma valor negativo.

Programación Lineal - SIMPLEX (maximización)

- En el último caso, perdemos factibilidad, porque x_4 toma valor negativo.
- No podemos incrementar x_1 demasiado.

Programación Lineal - SIMPLEX (maximización)

- En el último caso, perdemos factibilidad, porque x_4 toma valor negativo.
- No podemos incrementar x_1 demasiado.
- La pregunta es, ¿cuánto podemos incrementar x_1 (manteniendo $x_2 = 0$) y manteniendo factibilidad ($x_3, x_4, x_5, x_6 \geq 0$)?

Programación Lineal - SIMPLEX (maximización)

- En el último caso, perdemos factibilidad, porque x_4 toma valor negativo.
- No podemos incrementar x_1 demasiado.
- La pregunta es, ¿cuánto podemos incrementar x_1 (manteniendo $x_2 = 0$) y manteniendo factibilidad $(x_3, x_4, x_5, x_6 \geq 0)$?
- Para encontrar esta respuesta de forma fácil, despejamos x_3, x_4, x_5, x_6 y z en función de x_1 y x_2.

$$
\begin{array}{rcrcrcr}
x_3 & = & 6 & - & x_1 & - & 2x_2 \\
x_4 & = & 8 & - & 2x_1 & - & x_2 \\
x_5 & = & 1 & + & x_1 & - & x_2 \\
x_6 & = & 2 & & & - & x_2 \\
z & = & & & 3x_1 & + & 2x_2
\end{array}
$$

Programación Lineal - SIMPLEX (maximización)

- La condición $x_3 \geq 0$ implica $6 - x_1 \geq 0 \longrightarrow x_1 \leq 6$.

Programación Lineal - SIMPLEX (maximización)

- La condición $x_3 \geq 0$ implica $6 - x_1 \geq 0 \longrightarrow x_1 \leq 6$.

- Para $x_4 \geq 0 \longrightarrow x_1 \leq 8/2 = 4$.

Programación Lineal - SIMPLEX (maximización)

- La condición $x_3 \geq 0$ implica $6 - x_1 \geq 0 \longrightarrow x_1 \leq 6$.
- Para $x_4 \geq 0 \longrightarrow x_1 \leq 8/2 = 4$.
- La condición $x_5 \geq 0$ implica $1 + x_1 \geq 0 \longrightarrow$ no impone cota superior para el crecimiento de x_1.

Programación Lineal - SIMPLEX (maximización)

- La condición $x_3 \geq 0$ implica $6 - x_1 \geq 0 \longrightarrow x_1 \leq 6$.
- Para $x_4 \geq 0 \longrightarrow x_1 \leq 8/2 = 4$.
- La condición $x_5 \geq 0$ implica $1 + x_1 \geq 0 \longrightarrow$ no impone cota superior para el crecimiento de x_1.
- Para $x_6 \geq 0 \longrightarrow$ no impone cota superior para el crecimiento de x_1.

Programación Lineal - SIMPLEX (maximización)

- La condición $x_3 \geq 0$ implica $6 - x_1 \geq 0 \longrightarrow x_1 \leq 6$.
- Para $x_4 \geq 0 \longrightarrow x_1 \leq 8/2 = 4$.
- La condición $x_5 \geq 0$ implica $1 + x_1 \geq 0 \longrightarrow$ no impone cota superior para el crecimiento de x_1.
- Para $x_6 \geq 0 \longrightarrow$ no impone cota superior para el crecimiento de x_1.

El mayor valor que puede tomar x_1 manteniendo factibilidad es $x_1 = 4$, y la nueva solución factible

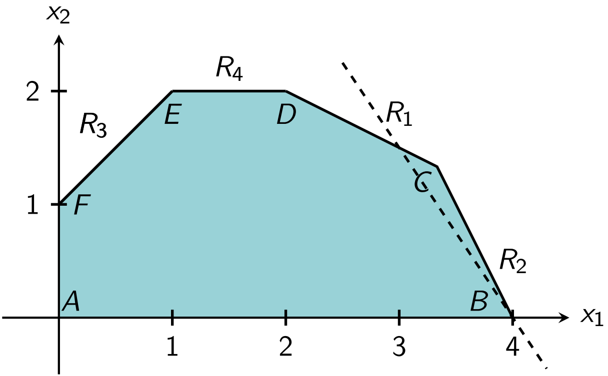
$$\begin{array}{rclcrcl} x_1 & = & 4 & \qquad & x_2 & = & 0 \\ x_3 & = & 2 & & x_4 & = & 0 \\ x_5 & = & 5 & & x_6 & = & 2 \\ z & = & 12 & & & & \end{array}$$

incrementa el valor de z.

Programación Lineal - SIMPLEX (maximización)

Solución B:

$$\begin{aligned} x_1 &= 4 & \qquad x_2 &= 0 \\ x_3 &= 2 & \qquad x_4 &= 0 \\ x_5 &= 5 & \qquad x_6 &= 2 \\ z &= 12 \end{aligned}$$



Programación Lineal - SIMPLEX (maximización)

- La decisión para mejorar el valor de z se vio facilitada por la expresión de las ecuaciones y su relación con la solución factible.

Programación Lineal - SIMPLEX (maximización)

- La decisión para mejorar el valor de z se vio facilitada por la expresión de las ecuaciones y su relación con la solución factible.
- Necesitaríamos tener un nuevo sistema que se relacione con nuestra nueva solución factible de manera similar al anterior.

Programación Lineal - SIMPLEX (maximización)

- La decisión para mejorar el valor de z se vio facilitada por la expresión de las ecuaciones y su relación con la solución factible.
- Necesitaríamos tener un nuevo sistema que se relacione con nuestra nueva solución factible de manera similar al anterior.
- Las variables con valor > 0 están expresadas en término de las que tienen valor 0. Necesitaríamos entonces tener x_1, x_3, x_5 y x_6 en términos de x_2 y x_4.

Programación Lineal - SIMPLEX (maximización)

- La decisión para mejorar el valor de z se vio facilitada por la expresión de las ecuaciones y su relación con la solución factible.
- Necesitaríamos tener un nuevo sistema que se relacione con nuestra nueva solución factible de manera similar al anterior.
- Las variables con valor > 0 están expresadas en término de las que tienen valor 0. Necesitaríamos entonces tener x_1, x_3, x_5 y x_6 en términos de x_2 y x_4.

Haciendo despejes y substituciones obtenemos:

$$
\begin{array}{rcrcrcr}
x_1 & = & 4 & - & 1/2x_2 & - & 1/2x_4 \\
x_3 & = & 2 & - & 3/2x_2 & + & 1/2x_4 \\
x_5 & = & 5 & - & 3/2x_2 & - & 1/2x_4 \\
x_6 & = & 2 & - & x_2 & & \\
z & = & 12 & + & 1/2x_2 & - & 3/2x_4
\end{array}
$$

Programación Lineal - SIMPLEX (maximización)

- Repitiendo el procedimiento anterior, buscamos aumentar nuevamente el valor de z.

Programación Lineal - SIMPLEX (maximización)

- Repitiendo el procedimiento anterior, buscamos aumentar nuevamente el valor de z.
- Para eso elegiremos una variable entre x_2 y x_4 para llevarla a valor positivo mientras mantenemos nula la otra.

- Repitiendo el procedimiento anterior, buscamos aumentar nuevamente el valor de z.
- Para eso elegiremos una variable entre x_2 y x_4 para llevarla a valor positivo mientras mantenemos nula la otra.
- Notar que incrementos en el valor de x_4 producen decrementos en z, luego sólo queda x_2 para elegir.

Programación Lineal - SIMPLEX (maximización)

- Repitiendo el procedimiento anterior, buscamos aumentar nuevamente el valor de z.
- Para eso elegiremos una variable entre x_2 y x_4 para llevarla a valor positivo mientras mantenemos nula la otra.
- Notar que incrementos en el valor de x_4 producen decrementos en z, luego sólo queda x_2 para elegir.
- ¿Cuánto podemos incrementar x_2 manteniendo factibilidad?

Programación Lineal - SIMPLEX (maximización)

Razonando como antes:

- $x_1 = 4 - 1/2x_2 \geq 0 \quad \longrightarrow \quad 4 - 1/2x_2 \geq 0 \quad \longrightarrow \quad x_2 \leq \frac{4}{1/2} = 8$
- $x_3 = 2 - 3/2x_2 \geq 0 \quad \longrightarrow \quad 2 - 3/2x_2 \geq 0 \quad \longrightarrow \quad x_2 \leq \frac{2}{3/2} = 4/3$
- $x_5 = 5 - 3/2x_2 \geq 0 \quad \longrightarrow \quad 5 - 3/2x_2 \geq 0 \quad \longrightarrow \quad x_2 \leq \frac{5}{3/2} = 10/3$
- $x_6 = 2 - x_2 \geq 0 \quad \longrightarrow \quad 2 - x_2 \geq 0 \quad \longrightarrow \quad x_2 \leq 2$

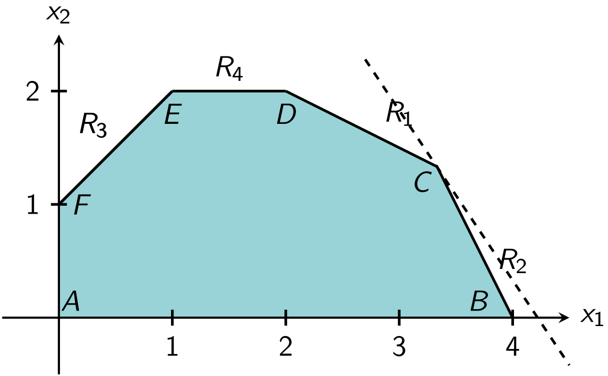
Luego $x_2 = 4/3$ es el mayor incremento y la nueva solución factible es:

$$\begin{array}{rclcrcl} x_1 & = & 10/3 & \qquad & x_2 & = & 4/3 \\ x_3 & = & 0 & & x_4 & = & 0 \\ x_5 & = & 3 & & x_6 & = & 2/3 \\ z & = & 38/3 & & & & \end{array}$$

Programación Lineal - SIMPLEX (maximización)

Solución C:

$$\begin{array}{lllllll} x_1 & = & 10/3 & \qquad & x_2 & = & 4/3 \\ x_3 & = & 0 & & x_4 & = & 0 \\ x_5 & = & 3 & & x_6 & = & 2/3 \\ z & = & 38/3 & & & & \end{array}$$



Programación Lineal - SIMPLEX (maximización)

- Teniendo la nueva solución, expresamos las variables x_1, x_2, x_5 y x_6 en función de x_3 y x_4. Substituyendo:

$$\begin{array}{rcrcrcr}
x_1 & = & 10/3 & + & 1/3x_3 & - & 2/3x_4 \\
x_2 & = & 4/3 & - & 2/3x_3 & + & 1/3x_4 \\
x_5 & = & 3 & - & x_3 & - & x_4 \\
x_6 & = & 2/3 & + & 2/3x_3 & - & 1/3x_4 \\
z & = & 38/3 & - & 1/3x_3 & - & 4/3x_4
\end{array}$$

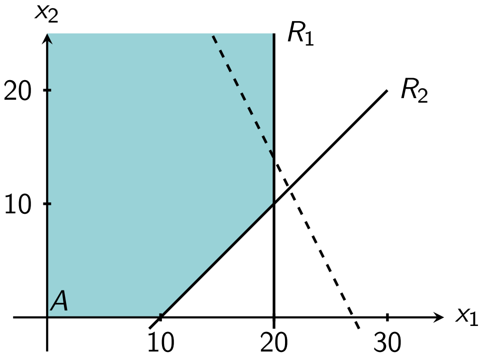
- Debemos ahora elegir una variable entre x_3 y x_4 cuyo incremento aumente el valor de z, pero dada su expresión, ninguna lo logra.
- Si incrementamos cualquiera de las variables x_3 o x_4, el valor de z disminuirá.
- Nuestro trabajo termino aquí. ¿Por qué?

Programación Lineal - SIMPLEX (maximización)

- La última solución tiene $z = 38/3$.
- Si esta solución es óptima, debe cumplirse que cualquier solución factible verifique $z \leq 38/3$.
- Pero como toda solución factible debe tener x_3 y $x_4 \geq 0$ y $z = 38/3 - 1/3x_3 - 4/3x_4$, entonces $z \leq 38/3$ para toda solución factible.
- Luego la solución obtenida es óptima.

Programación Lineal - SIMPLEX (maximización)

$$
\begin{array}{rl}
\text{Max} & 2\,x_1 + x_2 \\
\text{s.a.} & x_1 - x_2 \leq 10 \\
 & 2\,x_1 \qquad \leq 40 \\
 & x_1, \quad x_2 \geq 0
\end{array}
$$



Programación Lineal - SIMPLEX (maximización)

$$\begin{array}{rcrcrcr} x_3 & = & 10 & - & x_1 & + & 2x_2 \\ x_4 & = & 40 & - & 2x_1 & & \\ z & = & 0 & + & 2x_1 & + & x_2 \end{array}$$

$$\text{Sol. inicial: } (0, 0, 10, 40) \longrightarrow A$$

Si elegimos x_2 para incrementar su valor, no hay cota para lo máximo que puede crecer $\Longrightarrow$ LP no acotado!

Programación Lineal - SIMPLEX (maximización)

Generalización: Dado un problema

$$
\begin{array}{llcll}
\text{Maximizar} & \sum_{j=1}^{n} c_j x_j & & & \\
\text{sujeto a} & \sum_{j=1}^{n} a_{ij} x_j & \leq & b_i & i = 1, \ldots, m \qquad (*) \\
 & x_j & \geq & 0 & j = 1, \ldots, n
\end{array}
$$

primero introducimos las variables de holgura $x_{n+1} \ldots x_{n+m}$, las despejamos en función de las variables de decisión y denotamos a la función objetivo como z:

$$
\begin{array}{lclcll}
x_{n+i} & = & b_i & - & \sum_{j=1}^{n} a_{ij} x_j & i = 1, \ldots, m \\
 & & & & x_j \geq 0 & j = 1, \ldots, n+m \qquad (**) \\
z & = & & & \sum_{j=1}^{n} c_j x_j &
\end{array}
$$

Cada solución factible de (*) corresponde a una solución de (**).

Programación Lineal - SIMPLEX (maximización)

- En cada iteración el método simplex se mueve de una solución factible $x_1 \ldots x_{n+m}$ a otra $\tilde{x_1} \ldots \tilde{x_{n+m}}$ que es mejor en el sentido que

$$\sum_{j=1}^{n} c_j \tilde{x_j} > \sum_{j=1}^{n} c_j x_j.$$

- Es conveniente asociar a cada solución factible un sistema lineal de ecuaciones en el cual sea fácil de encontrar una solución factible mejor.

- Las ecuaciones del sistema deben expresar m de las variables $x_1 \ldots x_{n+m}$ y la función objetivo z en términos de las restantes n variables.

Programación Lineal - SIMPLEX (maximización)

- Estos sistemas se llaman diccionarios. Si el término independiente de cada ecuación es ≥ 0 (salvo para z), el diccionario se llama diccionario factible.
- Las variables que aparecen en el lado izquierdo de un diccionario son llamadas variables básicas y el resto de las variables no básicas.
- Las variables básicas constituyen una base, que cambia en cada iteración. Llamamos B al conjunto de variables básicas y N al de variables no básicas.

Programación Lineal - SIMPLEX (maximización)

Entonces un diccionario tiene la siguiente expresión:

$$
\begin{array}{ccccll}
x_i & = & \bar{b}_i & - & \sum_{j\in N} \bar{a_{ij}} x_j & i \in B \\
z & = & \bar{z} & + & \sum_{j\in N} \bar{c}_j x_j &
\end{array}
$$

- A cada diccionario se asocia el punto $x_j = 0$ para $j \in N$ y $x_i = \bar{b}_i$ para $i \in B$.
- Como todos los diccionarios son equivalentes (sólo hicimos manipulación algebraica) este punto es solución del sistema original, es decir, es una solución factible.
- No toda solución factible puede ser asociada a un diccionario.
- Las soluciones factibles que pueden ser representadas por un diccionario se llaman soluciones básicas.
- El método simplex sólo explora soluciones básicas, ignorando al resto.

Programación Lineal - SIMPLEX (maximización)

- En cada iteración primero se elige una variables no básica que entrará a la base y luego una variable básica que saldrá de la base.
- La elección de la variable de entrada está motivada por el deseo de incrementar el valor de z.
- Para esto seleccionamos una variable no básica x_j con $\bar{c}_j > 0$.
- Si no existe, el presente diccionario describe una solución óptima ($x_j = 0$, $j \in N$ y $x_i = \bar{b}_i$, $i \in B$), en cuyo caso el método finaliza.

Programación Lineal - SIMPLEX (maximización)

- La determinación de la variable de salida está basada en el requerimiento que todas las variables deben ser ≥ 0.
- La variable de salida es la variable básica cuya no negatividad impone la cota superior más restrictiva para el incremento de la variable de entrada (la menor de las cotas superiores).
- Si no existe cota sobre el crecimiento de la variable de entrada (coeficientes de la columna de entrada ≤ 0), esta puede crecer tanto como querramos y por lo tanto la función objetivo es no acotada. En este caso, el método termina.
- La ecuación en la que aparece la variable de salida se llama fila pivote y el proceso de construir un nuevo diccionario es referido como pivoteo.

Programación Lineal - SIMPLEX (maximización)

Dado un problema

$$
\begin{array}{llcll}
\text{Maximizar} & \sum_{j=1}^{n} c_j x_j & & & \\
\text{sujeto a} & \sum_{j=1}^{n} a_{ij} x_j & \leq & b_i & i = 1, \ldots, m \\
& x_j & \geq & 0 & j = 1, \ldots, n
\end{array}
$$

los pasos del algoritmo simplex son:

Programación Lineal - SIMPLEX (maximización)

PASO 1 - Inicialización: Introducir las variables de holgura $x_{n+1}, \ldots, x_{n+m}$ y considerar como base inicial este conjunto. El diccionario factible inicial es:

$$\begin{array}{llllll} x_{n+i} & = & b_i & - & \sum_{j=1}^{n} a_{ij} x_j & i = 1, \ldots, m \\ \\ z & = & & & \sum_{j=1}^{n} c_j x_j & \end{array}$$

Este diccionario es factible si $b_i \geq 0$, en cuyo caso $x_1 = \ldots = x_n = 0$ es una solución factible básica (cuando esto no sucede se aplica SIMPLEX en dos fases).

Programación Lineal - SIMPLEX (maximización)

PASO 2 - Variable de entrada: Si el diccionario actual es:

$$
\begin{array}{ccccl}
x_i & = & \bar{b}_i & - & \sum_{j \in N} \bar{a}_{ij} x_j \quad i \in B \\
z & = & \bar{z} & + & \sum_{j \in N} \bar{c}_j x_j
\end{array}
$$

- Si $\bar{c}_j \leq 0 \ \forall j \in N$, la solución asociada al diccionario actual es óptima, PARAR.
- Si no, elegir x_k, $k \in N$ con $\bar{c}_k > 0$ como variable de entrada.

Programación Lineal - SIMPLEX (maximización)

PASO 3 - Variable de salida:

- Si $\bar{a_{ik}} \leq 0 \ \forall i \in B$, el problema es no acotado, PARAR.
- Si no,

$$t = \min_{i \in B, \bar{a_{ik}} > 0} \frac{\bar{b_i}}{\bar{a_{ik}}} = \frac{\bar{b_s}}{\bar{a_{is}}}$$

  es el incremento que tendrá x_k y la variable de salida x_s será la (o una de las) que define t.

PASO 4 - Actualización: Calcular la nueva base $B = B \cup \{x_k\} \setminus \{x_s\}$, la solución básica y el diccionario correspondiente.

PASO 5: Ir a PASO 2.

Programación Lineal - SIMPLEX (maximización)

Los problemas que se pueden presentar en las distintas etapas son:

- Inicialización: ¿Cómo obtener una solución factible básica inicial? El diccionario inicial

$$\begin{array}{rcl} x_{n+i} & = & b_i - \sum_{j=1}^{n} a_{ij} x_j \quad i = 1, \ldots, m \\ z & = & \sum_{j=1}^{n} c_j x_j \end{array}$$

  es factible sólo si $b_i \geq 0$, en cuyo caso $x_1 = \ldots = x_n = 0$ es una solcución factible básica. Si esto no ocurre mediante el método SIMPLEX en dos fases es posible encontrar una solución factible básica inicial o establecer que el PL es infactible.

Programación Lineal - SIMPLEX (maximización)

- Iteración: Dado un diccionario factible con conjunto de variables básicas B y no básicas N,

$$
\begin{array}{ccccl}
x_i & = & \bar{b}_i & - & \sum_{j \in N} \bar{a}_{ij} x_j \quad i \in B \\
z & = & \bar{z} & + & \sum_{j \in N} \bar{c}_j x_j
\end{array}
$$

  hay que seleccionar una variable de entrada, una de salida y construir el diccionario correspondiente a la nueva solución básica por pivoteo. ¿Siempre se puede seleccionar variable de entrada y de salida?
- Terminación: ¿Siempre se alcanza el óptimo después de finitas iteraciones?

Programación Lineal - SIMPLEX (maximización)

- Hay problemas de programación lineal para los cuales el método simplex realiza un número exponencial de iteraciones.
- En 1972, Klee y Minty mostraron que para resolver

$$
\begin{array}{llcccl}
\text{Maximizar} & \sum_{j=1}^{n} 10^{n-j} x_j & & & & \\
\text{sujeto a} & 2\sum_{j=1}^{i-1} 10^{i-j} x_j & + \quad x_i & \leq & 100^{i-1} & i = 1, \ldots, n \\
 & & x_j & \geq & 0 & j = 1, \ldots, n
\end{array}
$$

el método simplex realiza $2^n - 1$ iteraciones si se selecciona como variable de entrada la de mayor coeficiente. Si asumimos que es posible realizar 1000 iteraciones por segundo, para $n = 50$ necesitaríamos 30.000 años para resolverlo.

Programación Lineal - SIMPLEX primera fase

Sea

$$
\begin{array}{llcll}
\text{Maximizar} & \sum_{j=1}^{n} c_j x_j & & & \\
\text{sujeto a} & \sum_{j=1}^{n} a_{ij} x_j & \leq & b_i & i = 1, \ldots, m \\
 & x_j & \geq & 0 & j = 1, \ldots, n
\end{array}
$$

- En el caso que exista algún $b_i < 0$ nos vemos en el problema de encontrar una solución factible.
- Más aún, no podemos afirmar la existencia de solución factible.

Programación Lineal - SIMPLEX primera fase

Construimos un problema auxiliar:

$$
\begin{array}{llcll}
\text{Minimizar} & x_0 & & & \\
\text{sujeto a} & \sum_{j=1}^{n} a_{ij}x_j - x_0 & \leq & b_i & i = 1, \ldots, m \\
 & x_j & \geq & 0 & j = 0, \ldots, n
\end{array}
$$

- Una solución factible del problema auxiliar es: $x_j = 0$ $j = 1, \ldots, n$ y x_0 lo suficientemente grande.
- El problema original tiene solución factible si y sólo si el problema auxiliar tiene solución factible con $x_0 = 0$.
- Es decir, el problema original tiene solución factible si y sólo si el valor óptimo del problema auxiliar es cero.
- El objetivo entonces será resolver el problema auxiliar.

Programación Lineal - SIMPLEX primera fase

Ejemplo:

$$\begin{array}{rrcrcrcr}
\text{Max} & x_1 & - & x_2 & + & x_3 & & \\
\text{sujeto a} & 2x_1 & - & x_2 & + & 2x_3 & \leq & 4 \\
 & 2x_1 & - & 3x_2 & + & x_3 & \leq & -5 \\
 & -x_1 & + & x_2 & - & 2x_3 & \leq & -1 \\
\end{array}$$

$$x_1, x_2, x_3 \geq 0$$

Programación Lineal - SIMPLEX primera fase

Ejemplo:

$$\begin{array}{rrcrcrcr}
\text{Max} & x_1 & - & x_2 & + & x_3 & & \\
\text{sujeto a} & 2x_1 & - & x_2 & + & 2x_3 & \leq & 4 \\
 & 2x_1 & - & 3x_2 & + & x_3 & \leq & \text{-5} \\
 & -x_1 & + & x_2 & - & 2x_3 & \leq & \text{-1} \\
\end{array}$$

$$x_1, x_2, x_3 \geq 0$$

El problema auxiliar resulta

$$\begin{array}{rrcrcrcrcr}
\text{Max} & -x_0 & & & & & & & & \\
\text{sujeto a} & 2x_1 & - & x_2 & + & 2x_3 & - & x_0 & \leq & 4 \\
 & 2x_1 & - & 3x_2 & + & x_3 & - & x_0 & \leq & \text{-5} \\
 & -x_1 & + & x_2 & - & 2x_3 & - & x_0 & \leq & \text{-1} \\
\end{array}$$

$$x_0, x_1, x_2, x_3 \geq 0$$

Programación Lineal - SIMPLEX primera fase

El primer diccionario

$$
\begin{array}{ccccccccccc}
x_4 & = & 4 & - & 2x_1 & + & x_2 & - & 2x_3 & + & x_0 \\
x_5 & = & -5 & - & 2x_1 & + & 3x_2 & - & x_3 & + & x_0 \\
x_6 & = & -1 & + & x_1 & - & x_2 & + & 2x_3 & + & x_0 \\
w & = & & & & & & & & - & x_0
\end{array}
$$

Pero no es factible. Haciendo pivote en x_0 como variable de entrada y x_5 de salida se obtiene

$$
\begin{array}{ccccccccccc}
x_0 & = & 5 & + & 2x_1 & - & 3x_2 & + & x_3 & + & x_5 \\
x_4 & = & 9 & & & - & 2x_2 & - & x_3 & + & x_5 \\
x_6 & = & 4 & + & 3x_1 & - & 4x_2 & + & 3x_3 & + & x_5 \\
w & = & -5 & - & 2x_1 & + & 3x_2 & - & x_3 & - & x_5
\end{array}
$$

que es un diccionario factible para comenzar la primer fase del SIMPLEX.

Programación Lineal - SIMPLEX primera fase

En general, el problema auxiliar puede ser escrito como

$$
\begin{array}{llcll}
\text{Minimizar} & x_0 & & & \\
\text{sujeto a} & \sum_{j=1}^{n} a_{ij}x_j - x_0 & \leq & b_i & i = 1, \ldots, m \\
& x_j & \geq & 0 & j = 0, \ldots, n
\end{array}
$$

Agregando las variables de holgura y llamando w a la función objetivo, resulta

$$
\begin{array}{lcccccl}
x_{n+i} & = & b_i & - & \sum_{j=1}^{n} a_{ij}x_j & + & x_0 \quad i = 1, \ldots, m \\
w & = & & & & - & x_0
\end{array}
$$

pero no es un diccionario factible.

Programación Lineal - SIMPLEX primera fase

- Pivoteando en este diccionario tomando como variable de entrada a x_0 y de salida la "más infactible" de las x_{n+i}, obtenemos un diccionario factible.
- Si x_{n+k} es la variable de salida, después de pivotear x_0 asume el valor $-b_k$ y x_{n+i} el valor $b_i - b_k$ que es ≥ 0.
- Ahora estamos en condiciones de aplicar el método simplex al problema auxiliar.

Programación Lineal - SIMPLEX primera fase

Siguiendo con el ejemplo se obtiene

$$
\begin{array}{rcrcrcrcrcr}
x_2 & = & 1 & + & 0.75x_1 & + & 0.75x_3 & + & 0.25x_5 & - & 0.25x_6 \\
x_0 & = & 2 & - & 0.25x_1 & - & 1.25x_3 & + & 0.25x_5 & + & 0.75x_6 \\
x_4 & = & 7 & - & 1.5x_1 & - & 2.5x_3 & + & 0.5x_5 & + & 0.5x_6 \\
w & = & \text{-2} & + & 0.25x_1 & + & 1.25x_3 & - & 0.25x_5 & - & 0.75x_6
\end{array}
$$

Y después de una iteración:

$$
\begin{array}{rcrcrcrcrcr}
x_3 & = & 1.6 & - & 0.2x_1 & + & 0.2x_5 & + & 0.6x_6 & - & 0.8x_0 \\
x_2 & = & 2.2 & + & 0.6x_1 & + & 0.4x_5 & + & 0.2x_6 & - & 0.6x_0 \\
x_4 & = & 3 & - & x_1 & & & - & x_6 & + & 2x_0 \\
w & = & & & & & & & & - & x_0
\end{array}
$$

Que es un diccionario óptimo y da una solución factible básica para el problema original $x_1 = 0$, $x_2 = 2.2$, $x_3 = 1.6$, que se relaciona con el diccionario:

$$
\begin{array}{rcrcrcrcr}
x_3 & = & 1.6 & - & 0.2x_1 & + & 0.2x_5 & + & 0.6x_6 \\
x_2 & = & 2.2 & + & 0.6x_1 & + & 0.4x_5 & + & 0.2x_6 \\
x_4 & = & 3 & - & x_1 & & & - & x_6 \\
z & = & \text{-}0.6 & + & 0.2x_1 & - & 0.2x_5 & + & 0.4x_6
\end{array}
$$

Programación Lineal - IBM ILOG CPLEX

Software de Optimización para resolver problemas de:

- Programación lineal.
- Programación lineal entera mixta.
- Programación cuadrática (función objetivo cuadrática y restricciones lineales).
- Programación con restricciones cuadráticas convexas.

Formas de uso:

- Interfase interactiva con el usuario.
- Interfase con lenguajes de programación (C, C++, Java).

Programación Lineal - IBM ILOG CPLEX

Archivo dieta.lp:

```
\xm: cantidad de porciones diarias de maiz de nuestra dieta
\xl: cantidad de porciones diarias de leche de nuestra dieta
\xp: cantidad de porciones diarias de pan de nuestra dieta

Min 1.8xm + 2.3xl + 1.5xp \costo de compra

Subject to

170xm + 50xl + 300xp >= 2000 \min. calorias
170xm + 50xl + 300xp <= 2300 \max. calorias
3xm + 400xl + 40xp >= 1200 \min. calcio

Bounds
xp <= 3 \max. porciones pan
xl >= 2 \min. porciones leche
\No es necesario 0 <=, lo toma por defecto
End
```

Programación Lineal - IBM ILOG CPLEX

2:pongo.cuartos.inv.dc.uba.ar - default* - SSH Secure Shell

File Edit View Window Help

Quick Connect Profiles

```
[usuario@pongo ejcplex]$ cplex

Welcome to IBM(R) ILOG(R) CPLEX(R) Interactive Optimizer 12.4.0.0
  with Simplex, Mixed Integer & Barrier Optimizers
5725-A06 5725-A29 5724-Y48 5724-Y49 5724-Y54 5724-Y55
Copyright IBM Corp. 1988, 2011.  All Rights Reserved.

Type 'help' for a list of available commands.
Type 'help' followed by a command name for more
information on commands.

CPLEX> r dieta.lp
Problem 'dieta.lp' read.
Read time =    0.00 sec.
CPLEX> opt
Tried aggregator 1 time.
No LP presolve or aggregator reductions.
Presolve time =    0.00 sec.

Iteration log . . .
Iteration:     1   Dual objective     =            19.688235

Dual simplex - Optimal:  Objective =  2.0852100221e+01
Solution time =    0.00 sec.  Iterations = 2 (0)
Deterministic time = 0.01 ticks  (5.30 ticks/sec)

CPLEX> dis sol var -
Variable Name           Solution Value
xm                            5.689020
xl                            2.657332
xp                            3.000000
CPLEX>
```

Programación Lineal - Ejemplo 1

Combinación de productos

El granjero Jones prepara dos tipos de masas (chocolate y vainilla) que vende para obtener ingresos extra. Cada kilogramo de masas de chocolate se puede vender a $20, y cada kilogramo de masas de vainilla se puede vender a $15. Un kilo de masas de chocolate requiere de 20 minutos de cocción y 4 huevos. En cambio cada kilo de masas de vainilla requiere 40 minutos de cocción y 1 huevo. Jones tiene disponibles 8 horas de cocción y 30 huevos semanalmente y sabe que a lo sumo puede vender 5.3 kilogramos de masas de chocolate en una semana. Formular un LP para maximizar las ganancias de Jones.

Programación Lineal - Ejemplo 1

Combinación de productos

Variables:

x_c: kilogramos de masas de chocolate que se fabrican semanales

x_v: kilogramos de masas de vainilla que se fabrican semanales

Programación Lineal - Ejemplo 1

Combinación de productos

$$\text{Max} \quad 20x_c + 15x_v \qquad \text{ganancia}$$

s.a. Tiempo de cocción:

$$20x_c + 40x_v \leq 8{*}60$$

Huevos disponibles:

$$4x_c + x_v \leq 30$$

Máxima venta de masas de chocolate:

$$x_c \leq 5.3$$

$$x_c, x_v \geq 0$$

Programación Lineal - Ejemplo 1

Combinación de productos

Max $\quad 20x_c + 15x_v$ ganancia

s.a. Tiempo de cocción:

$$20x_c + 40x_v \leq 8*60$$

Huevos disponibles:

$$4x_c + x_v \leq 30$$

Máxima venta de masas de chocolate:

$$x_c \leq 5.3$$

$$x_c, x_v \geq 0$$

$$x^* = (x_c^*, x_v^*) = (5.142857, 9.428571)$$

Programación Lineal - Ejemplo 2

Planificación de la producción

Una empresa produce listones de madera en 4 medidas: chico, mediano, grande y extra grande. Estos listones pueden producirse en tres máquinas: A, B y C. La cantidad de metros que puede producir por hora cada máquina es:

	A	B	C
chico	300	500	800
mediano	250	400	640
grande	200	320	500
extra grande	100	200	250

Supongamos que las máquinas A y B puede ser usada 50 horas semanales y 30 la máquina C, que el costo operativo por hora de cada una es \$30, \$50 y \$80 respectivamente y que por restricciones técnicas a lo suma se pueden fabricar 7000 metros de listones medianos enla máquina A. Si se necesitan 10000, 8000, 6000 y 4000 metros de cada tipo de listones por semana, formular un modelo para minimizar costos.

Programación Lineal - Ejemplo 2

Planificación de la producción

Variables:

$x_{cA/B/C}$: metros de listones chicos fabricados en la máquina $A/B/C$

$x_{mA/B/C}$: metros de listones mediasnos fabricados en la máquina $A/B/C$

$x_{gA/B/C}$: metros de listones grandes fabricados en la máquina $A/B/C$

$x_{eA/B/C}$: metros de listones extragrandes fabricados en la máquina $A/B/C$

Programación Lineal - Ejemplo 2

Planificación de la producción

$$
\begin{aligned}
\text{Min} \quad & 30\left(\frac{x_{cA}}{300} + \frac{x_{mA}}{250} + \frac{x_{gA}}{200} + \frac{x_{eA}}{100}\right) + && \text{hs. usadas de la máquina } A \\
& 50\left(\frac{x_{cB}}{500} + \frac{x_{mB}}{400} + \frac{x_{gB}}{320} + \frac{x_{eB}}{200}\right) + && \text{hs. usadas de la máquina } B \\
& 80\left(\frac{x_{cC}}{800} + \frac{x_{mC}}{640} + \frac{x_{gC}}{500} + \frac{x_{eC}}{250}\right) && \text{hs. usadas de la máquina } C
\end{aligned}
$$

s.a. Restricciones de demanda:

$$
\begin{array}{ccccccc}
x_{cA} & + & x_{cB} & + & x_{cC} & = & 10000 \\
x_{mA} & + & x_{mB} & + & x_{mC} & = & 8000 \\
x_{gA} & + & x_{gB} & + & x_{gC} & = & 6000 \\
x_{eA} & + & x_{eB} & + & x_{eC} & = & 4000
\end{array}
$$

Restricciones de disponibilidad:

$$
\begin{array}{ccccccccc}
\frac{x_{cA}}{300} & + & \frac{x_{mA}}{250} & + & \frac{x_{gA}}{200} & + & \frac{x_{eA}}{100} & \leq & 50 \\
\frac{x_{cB}}{500} & + & \frac{x_{mB}}{400} & + & \frac{x_{gB}}{320} & + & \frac{x_{eB}}{200} & \leq & 50 \\
\frac{x_{cC}}{800} & + & \frac{x_{mC}}{640} & + & \frac{x_{gC}}{500} & + & \frac{x_{eC}}{250} & \leq & 30
\end{array}
$$

$$x_{mA} \leq 7000 \qquad x \geq 0$$

Programación Lineal - Ejemplo 3

Producción multiperíodo (inventario)

Una compañía produce harina. Considerando un horizonte de planificación de 3 meses, quiere decidir cuántos kilogramos de harina producir cada mes. El pronóstico de demanda en kilogramos para los siguientes 3 meses es de 10000, 35000 y 30000. La compañía quiere abastecer de forma completa esta demanda. Tiene 5000 kilogramos de harina en inventario al comenzar el mes 1 y puede usar la producción de un mes para cubrir la demanda de un mes posterior. La capacidad máxima de producción mensual es de 30000 kilogramos y la capacidad máxima de almacenaje es de 10000 al finalizar el mes. El costo pronosticado de producción para los próximos 3 meses por kilogramos es de \$8, \$8.1 y \$8.3 respectivamente y el costo de almacenamiento por kilogramos al finalizar un mes es del 5% del costo de producción para ese mes. Al finalizar el mes 3 la compañía quiere tener en inventario por lo menos 4000 kilogramos de harina.

Programación Lineal - Ejemplo 3

Producción multiperíodo (inventario)

Variables:

p_i: cantidad de kilogramos de harina producidos en el mes $i = 1, \ldots, 3$

a_i: cantidad de kilogramos de harina almacendos al finalizar el mes $i = 1, \ldots, 3$

Programación Lineal - Ejemplo 3

Producción multiperíodo (inventario)

$$
\begin{array}{lrcll}
\text{Min} & & & 8p_1 + 8.1p_2 + 8.3p_3 + & \text{costo de producción} \\
& & & 0.05(8a_1 + 8.1a_2 + 8.3a_3) & \text{costo de almacenamiento}
\end{array}
$$

s.a. Máxima producción:

$$p_i \leq 30000 \qquad i = 1, \ldots, 3$$

Máximo almacenaje:

$$a_i \leq 10000 \qquad i = 1, \ldots, 3$$

Balance mes 1:

$$p_1 + 5000 = 10000 + a_1$$

Balance mes 2:

$$p_2 + a_1 = 15000 + a_2$$

Balance mes 3:

$$p_3 + a_2 = 30000 + a_3$$

Mínimo inventario mes 3:

$$a_3 \geq 4000$$

Positividad:

$$p_i, a_i \geq 0 \qquad i = 1, \ldots, 3$$

Programación Lineal - Ejemplo 4

Mezcla

Derco procesa petróleo para producir combustible para aviones y aceite de máquina.
Cuesta \$40 comprar 1000 barriles de petróleo, que luego destilados producen 500 barriles de combustible para aviones y 500 de aceite.
Lo que se obtiene de la destilación puede ser vendido directamente o ser procesado nuevamente con un fraccionador catalítico.
Si se vende sin el segundo proceso, el combustible para aviones se vende a \$60 por 1000 barriles y el aceite se vende a \$40 por 1000 barriles.
Lleva 1 hora procesar 1000 barriles de combustible para aviones en el fraccionador catalítico, y esos 1000 barriles se venden a \$130.
El mismo proceso demora 45 minutos para 1000 barriles de aceite, y esos 1000 barriles se venden a \$90.
Cada día, se pueden comprar a lo sumo 20000 barriles de petróleo, y se tienen disponibles 8 horas del fraccionador catalítico.
Formular un LP que maximice los beneficios de Derco.

Programación Lineal - Ejemplo 4

Mezcla

Variables:

X: barriles de petróleo comprados por día x 1000

x_c: barriles de combustible sin proceso catalítico vendidos por día x 1000

x_a: barriles de aceite sin proceso catalítico vendidos por día x 1000

x_{cp}: barriles de combustible con proceso catalítico vendidos por día x 1000

x_{ap}: barriles de aceite con proceso catalítico vendidos por día x 1000

Programación Lineal - Ejemplo 4

Mezcla

$$
\begin{array}{lrl}
\text{Max} & -40X + & \text{costo de compra} \\
 & 60x_c + 40x_a + & \text{ganancia venta directa} \\
 & 130x_{cp} + 90x_{ap} & \text{ganancia venta procesado}
\end{array}
$$

s.a. Tiempo de procesamiento:

$$3/4x_{ap} + x_{cp} \leq 8$$

500 de aceite y 500 de combustible
por cada 1000 de petróleo:

$$x_c + x_a + x_{cp} + x_{ap} = X$$

$$x_c + x_{cp} = x_a + x_{ap}$$

Máxima disponibilidad:

$$X \leq 20$$

$$x_c, x_a, x_{cp}, x_{ap} \geq 0$$

Programación Lineal - Ejemplo 5

Distribución de mercadería (continuo)

Variables:

x_{ij}: flujo de mercadería desde fábrica i al depósito j

y_{jk}: flujo de mercadería desde el depoósito j al cliente k

Programación Lineal - Ejemplo 5

Distribución de mercadería (continuo)

Min $\sum_{i \in F, j \in D} c_{ij} x_{ij} +$ costo de transporte entre fábricas y depósitos

$\sum_{j \in D, k \in C} c_{jk} y_{jk}$ costo de transporte entre depósitos y clientes

s.a.

Capacidad fábrica: $\sum_{j \in D} x_{ij} \leq f_i \quad \forall i \in F$

Capacidad depósito: $\sum_{i \in F} x_{ij} \leq d_j \quad \forall j \in D$

Preservación de mercadería: $\sum_{i \in F} x_{ij} = \sum_{k \in C} y_{jk} \quad \forall j \in D$

Demanda de clientes: $\sum_{j \in D} y_{jk} = r_k \quad \forall k \in C$

$x_{ij}, y_{jk} \geq 0 \quad \forall i \in F, j \in D, k \in C$

Programación Lineal

Limitaciones:

- ¿Qué sucede si sólo tiene sentido que las variables tomen valores enteros? Por ejemplo si lo que fabricamos son autos, o si representan personas.
- Limitada para modelar toma de decisiones.