

Programación lineal: el problema dual

Rodrigo Maranzana

Repaso problema de programación lineal

Algebraica:

$$\text{Max o Min } \sum_j c_j x_j$$

s.t.

$$\sum_j a_{ij} x_j \begin{matrix} \leq \\ = \\ \geq \end{matrix}$$

$$x_j \geq 0$$

Matricial:

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

s.t.

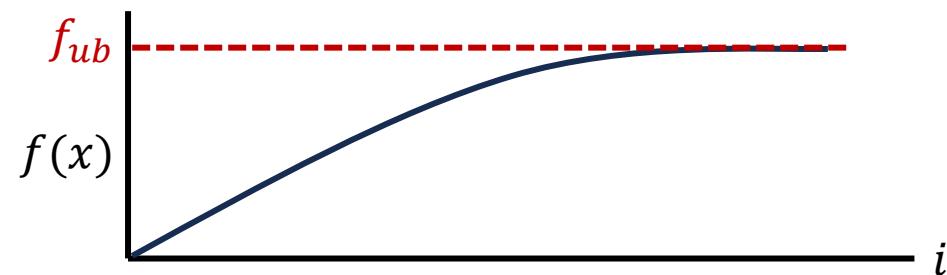
$$\begin{matrix} \leq \\ AX=B \\ \geq \end{matrix}$$

$$X \geq 0$$

Cotas de un problema LP

Una cota superior de la función objetivo del problema LP en maximización (f_{ub}) nos daría una información valiosa:

- Durante la optimización, es un límite que **no se puede superar**.
- Nos da una idea aproximada de la **distancia al óptimo**.
- Si en un paso de optimización i resulta $f_{ub} = f_i(x)$, entonces **encontramos el óptimo**.



Nota: Aplica lo mismo para minimización pero con cota inferior.

Cotas de un problema LP

Partiendo del siguiente problema, que vamos a llamar **primal**:

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

s.t.

$$AX \leq B$$

$$X \geq 0$$

Ej: maximizar una ganancia de un mix comercial sujeta a restricciones de recursos disponibles.

Buscamos encontrar una cota superior del problema.

Cotas de un problema LP

En primer lugar, buscamos una cota superior de los coeficientes C (ganancia unitaria de cada producto comercializado) de la función objetivo del primal.

Siendo \mathbf{Y} un vector

$$\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \cdots \\ \mathbf{y}_i \\ \cdots \\ \mathbf{y}_n \end{bmatrix} \geq \mathbf{0}$$

Podemos entender \mathbf{y}_i como el precio por unidad de cada recurso.

Vamos a usar las variables \mathbf{y}_i para ponderar **cada coeficiente tecnológico** del problema primal y **usarlo como cota superior de C**.

$$A^T Y \geq C$$

Cotas de un problema LP: intuición



Cotas de un problema LP: intuición

$$AX \leq B$$

Recuso 1	$a_{11}x_1 + a_{12}x_2 + \cdots + a_{1m} \leq b_1$
Recuso 2	$a_{21}x_1 + a_{22}x_2 + \cdots + a_{1m} \leq b_2$
...	
Recuso n	$a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nm} \leq b_n$

Producto 1 Producto 2 ... Producto m

$$A^T Y \geq C$$

$a_{11}y_1 + a_{21}y_2 + \cdots + a_{n1} \geq c_1$	Producto 1
$a_{12}y_1 + a_{22}y_2 + \cdots + a_{n2} \geq c_2$	Producto 2
...	
$a_{1m}y_1 + a_{2m}y_2 + \cdots + a_{nm} \geq c_m$	Producto m

Recuso 1 Recuso 2 ... Recuso n

Cotas de un problema LP: intuición

$$AX \leq B$$

Recurso 1	$a_{11}x_1 + a_{12}x_2 + \cdots + a_{1m} \leq b_1$
Recurso 2	$a_{21}x_1 + a_{22}x_2 + \cdots + a_{1m} \leq b_2$
...	
Recurso n	$a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nm} \leq b_n$

Producto 1 Producto 2 Producto m



Vendedor de productos



Cliente

Conceptos:

x_j : cantidad total de producto "j" fabricado.

a_{ij} : u. de recurso "i" por u. de producto "j".

b_i : cantidad total de recurso "i" disponible.

La multiplicación $a_{ij} x_j$ representa todo el recurso "i" que hace falta para fabricar la cantidad necesaria de producto "j"

La suma de $a_{ij} x_j$ no puede ser mayor que la capacidad total b_i

Cotas de un problema LP: intuición

Conceptos:

y_i : contribución marginal. \$ por u. de recurso "i"

a_{ji} : u. de recurso "i" por u. de producto "j".

c_i : ganancia unitaria del producto "i".

La multiplicación $a_{ji} y_i$ resulta en \$ por u. de producto "j". Es el aporte total en \$ del recurso "i" por cada unidad de producto "j".

La suma de la contribución $a_{ji} y_i$ debe ser mayor o igual que la ganancia unitaria c_j de cada producto "j" que saca el vendedor.

De esta forma, el proveedor siempre elige vender los recursos por separado y no dedicarse a vender el producto armado.

$$A^T Y \geq C$$

$a_{11}y_1 + a_{21}y_2 + \cdots + a_{n1} \geq c_1$	Producto 1
$a_{12}y_1 + a_{22}y_2 + \cdots + a_{n2} \geq c_2$	Producto 2
...	
$a_{1m}y_1 + a_{2m}y_2 + \cdots + a_{nm} \geq c_m$	Producto m

Recurso 1 Recurso 2 Recurso n



Proveedor de recursos



Vendedor de productos

Cotas de un problema LP

$$C \leq A^T Y$$

Si reorganizamos:

$$C^T \leq Y^T A$$

Dado que en el primal $X \geq 0$, podemos multiplicar ambos términos por X :

$$C^T X \leq Y^T AX$$

Función objetivo primal

Restricciones primales

$$AX \leq B$$

Cotas de un problema LP

$$C^T X \leq Y^T(AX) \leq Y^T B$$
$$C^T X \leq Y^T(B)$$

$$C^T X \leq B^T Y$$

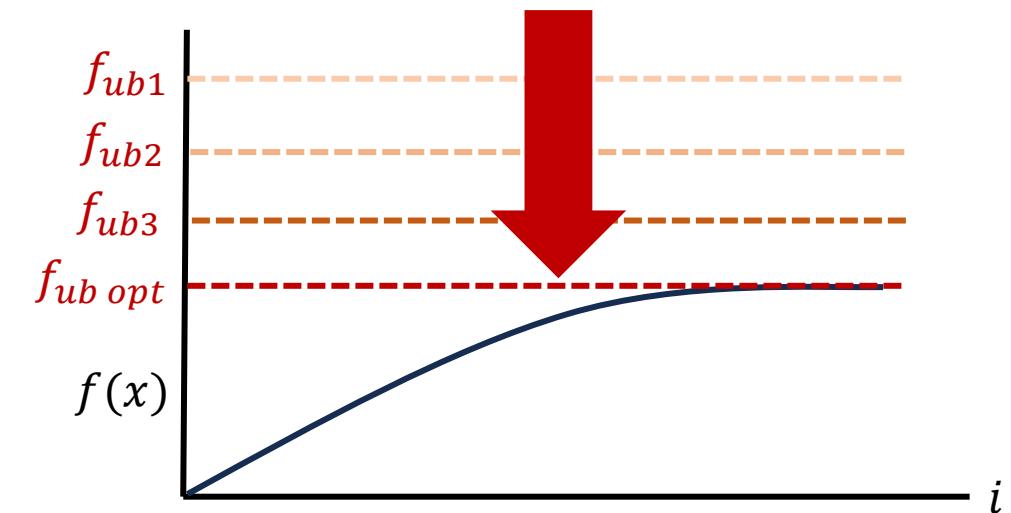
Vemos que $B^T Y$ resulta cota superior de la función objetivo primal.
Pero Y es variable, ¿Qué valor seleccionamos?

Cotas de un problema LP

$$C^T X \leq B^T Y$$

La cota superior más próxima al objetivo primal. Minimizamos.

$$\boxed{\text{Min } B^T Y}$$



Problema dual LP

El problema de optimización resultante:

$$\text{Min } B^T Y$$

s.t.

$$A^T Y \geq C$$

$$Y \geq 0$$

Se conoce como **problema dual**.

Repasamos la intuición



Proveedor de recursos

Resuelve problema Dual

Minimiza el costo total de los recursos.

Sujeto a que la contribución en \$/producto sea mayor que la ganancia unitaria del vendedor.

$$\text{Min } B^T Y$$

s.t.

$$A^T Y \geq C$$

$$Y \geq 0$$

Vendedor de productos

Resuelve problema Primal

Maximiza la ganancia total de sus productos.

Sujeto a que la cantidad de cada recurso invertido no supere la capacidad total.

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

s.t.

$$AX \leq B$$

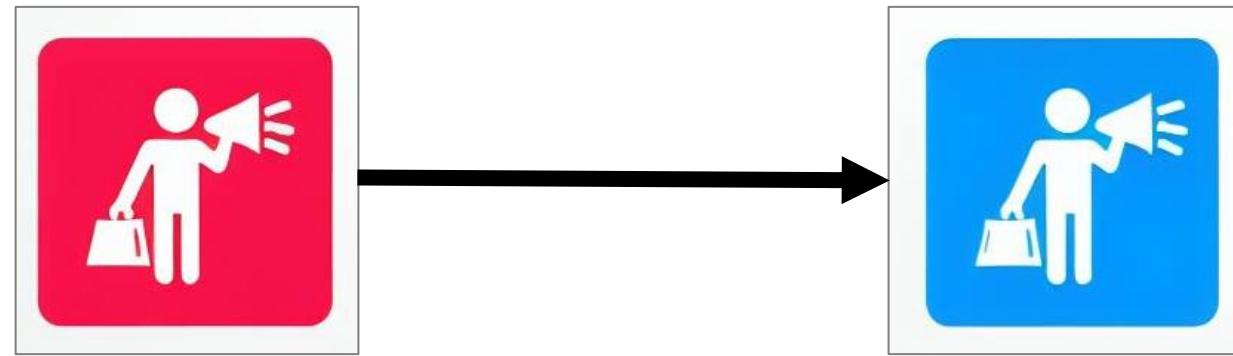
$$X \geq 0$$

Cliente

Relación entre el dual y el primal: cotas

Las variables Y (dual) y X (primal) tienen unidades diferentes.

El valor de la función objetivo Z (primal) y W (dual), tiene la misma unidad.



Podemos imaginar como si ambos estuvieran negociando un precio.

- El proveedor intenta minimizar el costo de la venta.
- El vendedor intenta maximizar su ganancia.

Dualidad débil

Concepto de dualidad: un problema tiene dos formas análogas: primal y dual.

Además, se conoce como teorema de **dualidad débil** (weak duality theorem), a la fórmula derivada en la slide anterior:

$$C^T X \leq B^T Y$$

Función objetivo del problema primal



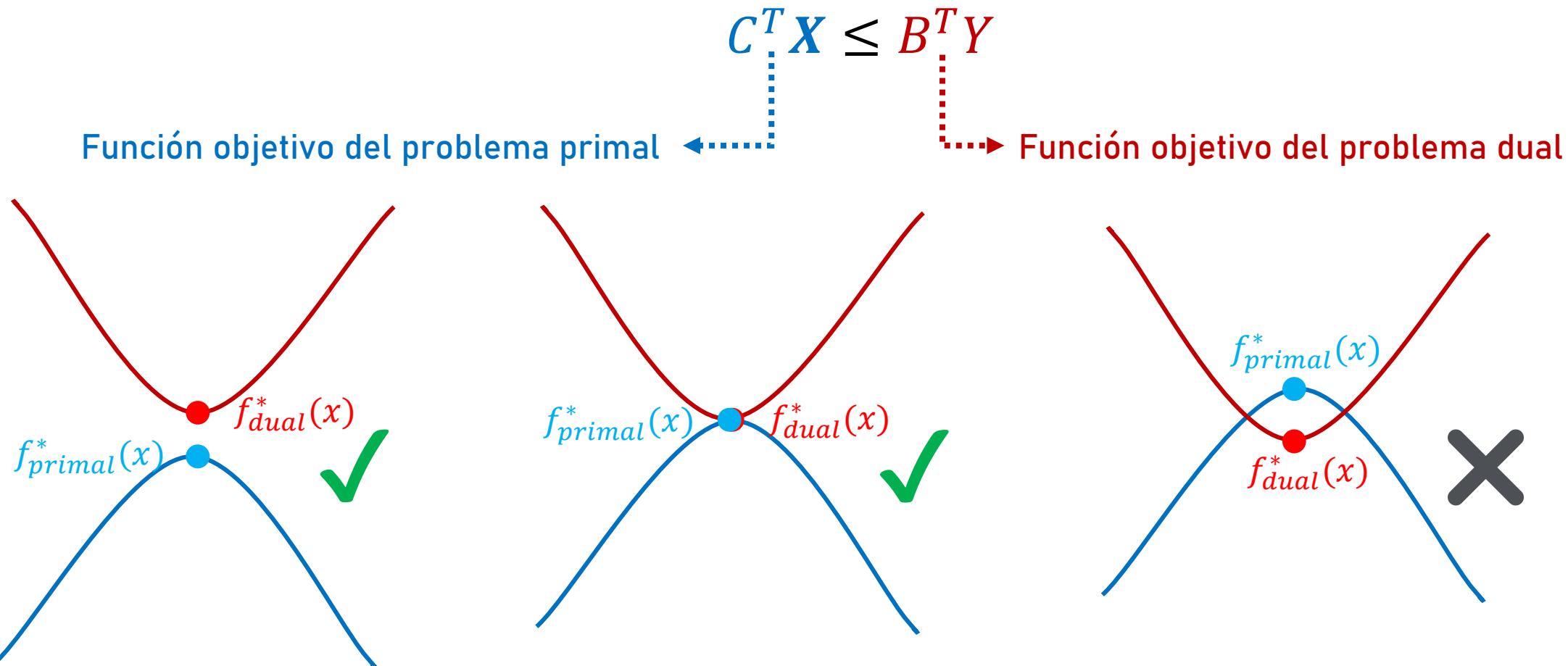
Función objetivo del problema dual



Nota: En caso de estar minimizando en el primal, se volteá el signo, y el dual pasa a ser una cota inferior.

Dualidad débil

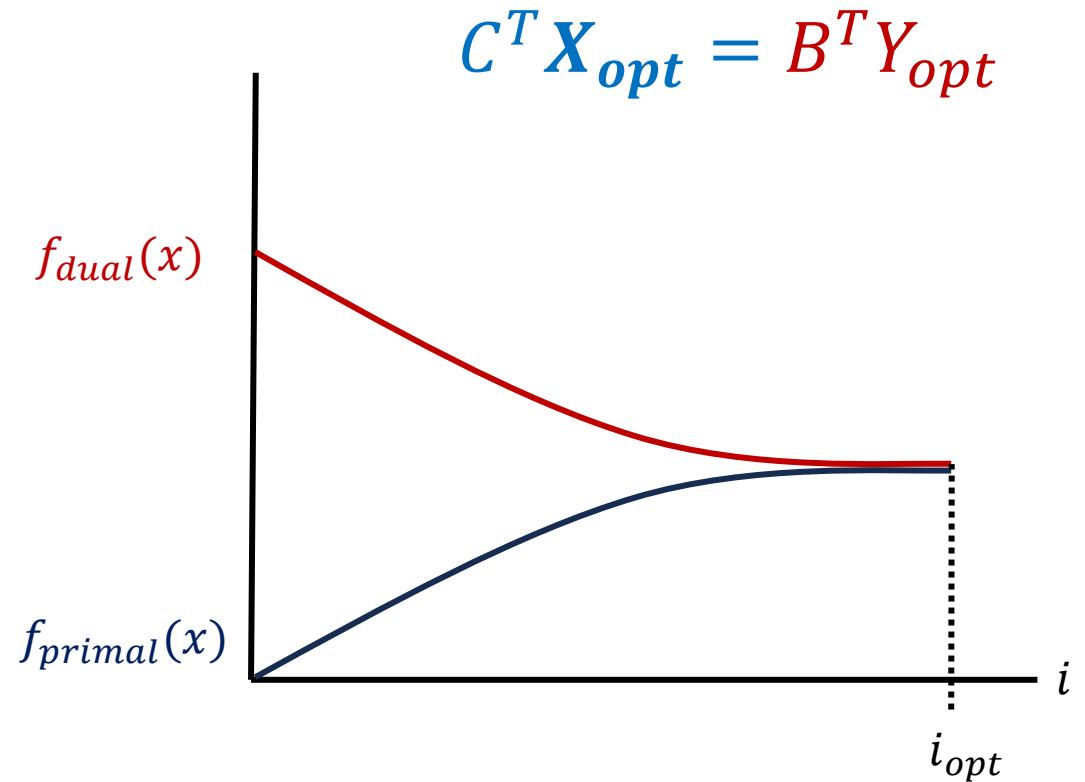
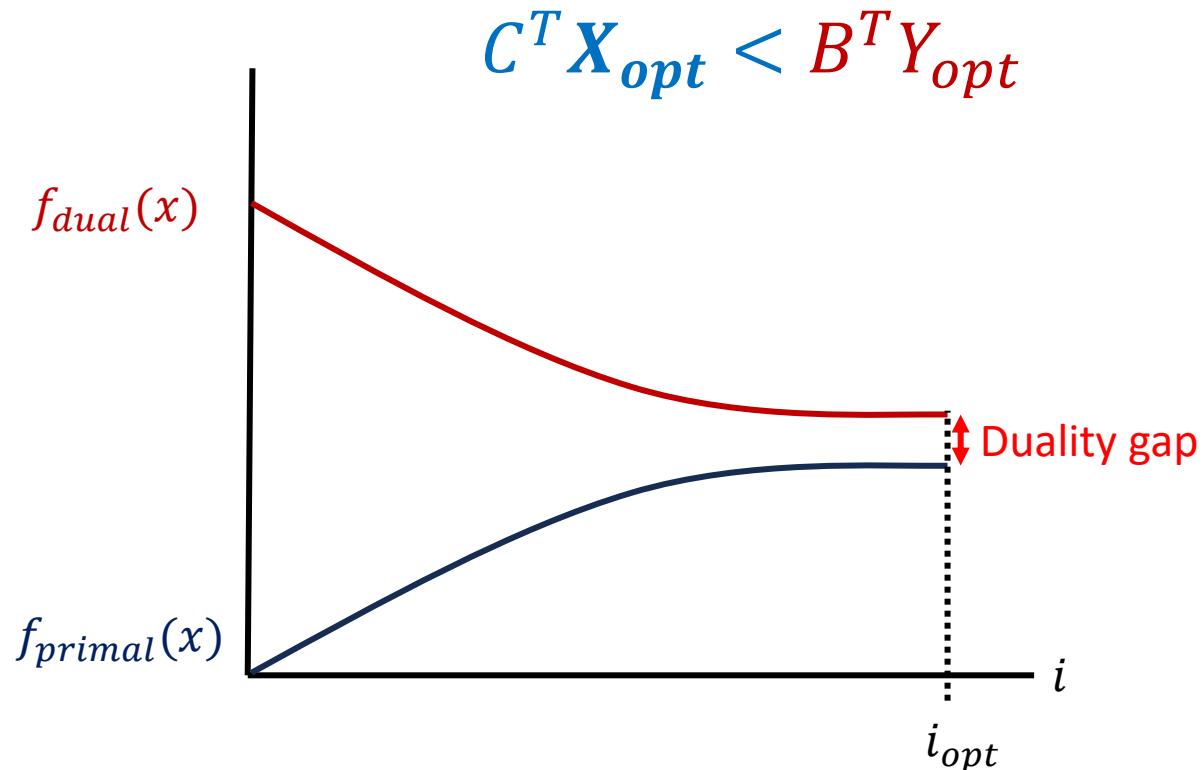
La dualidad débil implica que no puede conseguirse un óptimo primal por encima del óptimo dual:



Duality gap

Tanto en primal como dual, siempre estoy resolviendo un problema LP que implica maximización o minimización. Uno es la cota del otro.

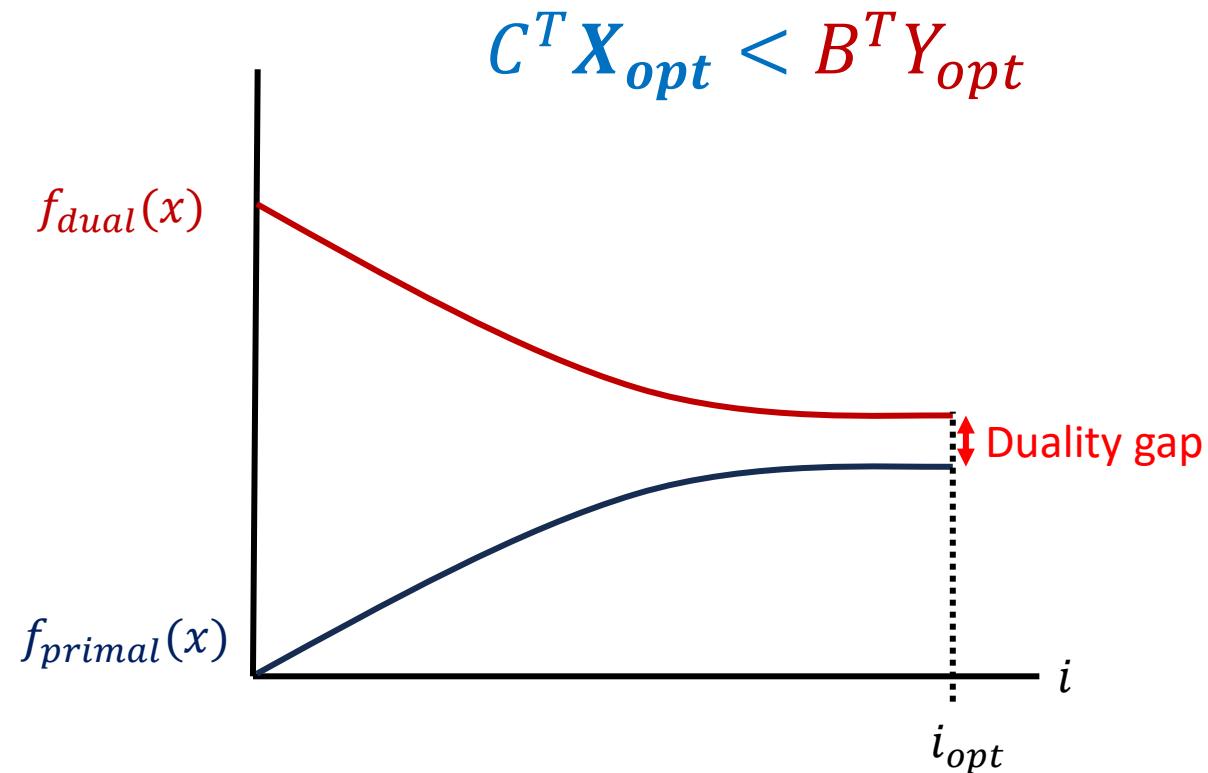
Duality gap es la diferencia entre ambos funcionales primal y dual en el óptimo.



Duality gap

Si existe Duality Gap implica que para resolver un problema, debe conocerse el resultado del primal y el dual.

Es decir, resolver 2 problemas por separado.



Dualidad fuerte

Se conoce como dualidad fuerte (strong duality) al caso en el que no existe duality gap:

$$C^T X_{opt} = B^T Y_{opt}$$

En programación lineal:

Si el primal tiene solución óptima factible, entonces el dual también y es la misma. Es decir cumple con la dualidad fuerte.

Nota: la demostración se puede lograr mediante el lema de Farkas*.

*Strong Duality, École Polytechnique Fédérale de Lausanne https://sma.epfl.ch/~niemeier/opt09/opt09_ch05.pdf

Ejemplo construcción dual con max

$$\max z = 72x_1 + 33x_2$$

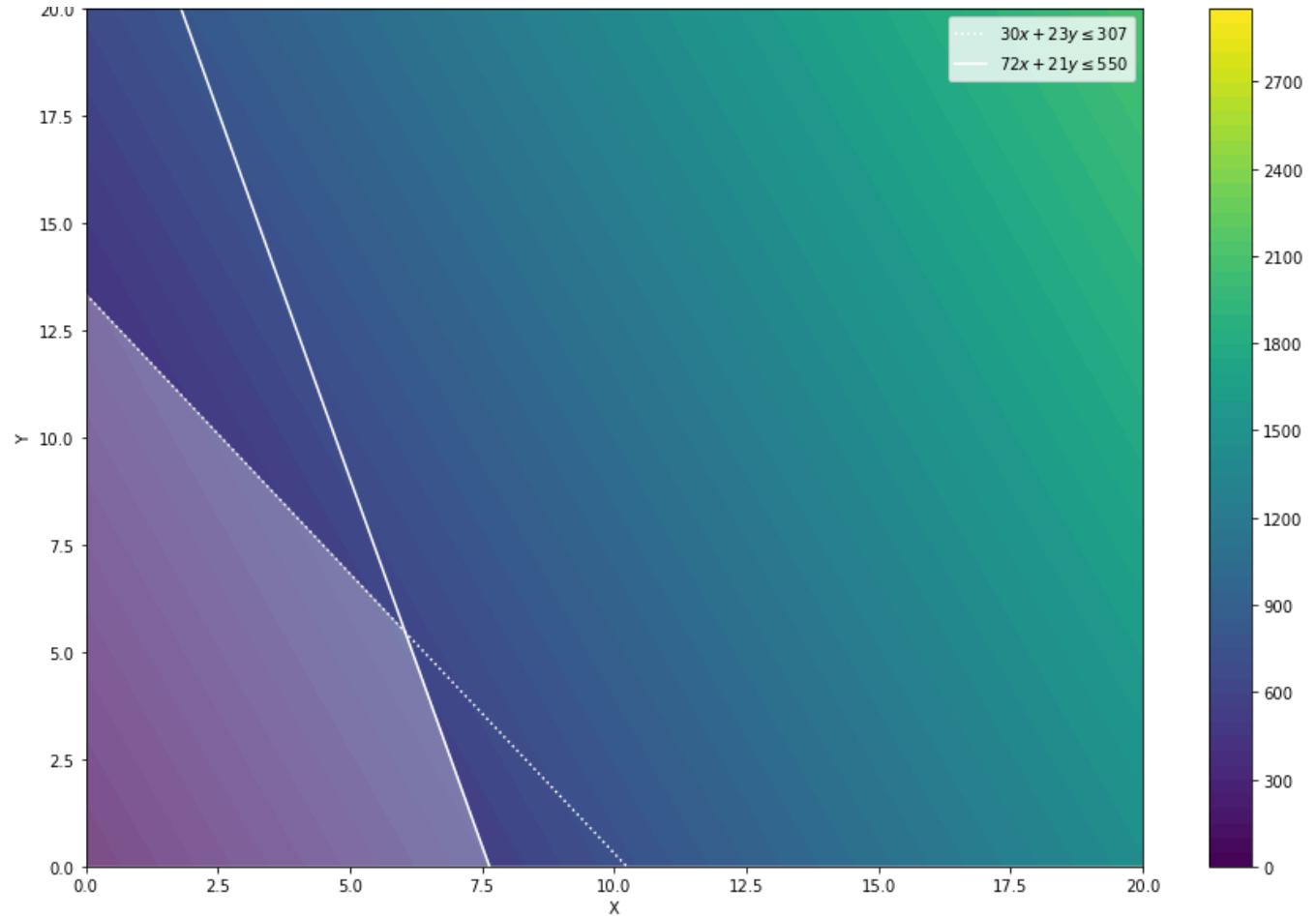
s.t.

$$30x_1 + 23x_2 \leq 307$$

$$72x_1 + 21x_2 \leq 550$$

$$x_1, x_2 \geq 0$$

A este problema lo vamos a llamar: problema LP primal



Ejemplo construcción dual con max

Función objetivo primal $\max z = 72x_1 + 33x_2$

The diagram illustrates the construction of the dual problem from the given primal problem. The primal problem is:

$$\max z = 72x_1 + 33x_2$$

subject to:

$$30x_1 + 23x_2 \leq 307$$
$$72x_1 + 21x_2 \leq 550$$

The dual problem is constructed as follows:

Restricción 1 primal (left):

$$30y_1 + 72y_2 \geq 72$$

Restricción 2 primal (right):

$$23y_1 + 21y_2 \geq 33$$

Dashed lines connect the coefficients of x_1 and x_2 in the primal constraints to the corresponding y_1 and y_2 variables in the dual constraints. The right-hand side values 307 and 550 are also connected to their respective dual constraint equations.

Ejemplo construcción dual con max

Ejemplo: construcción de problema dual.

$$\min z = 307y_1 + 550y_2$$

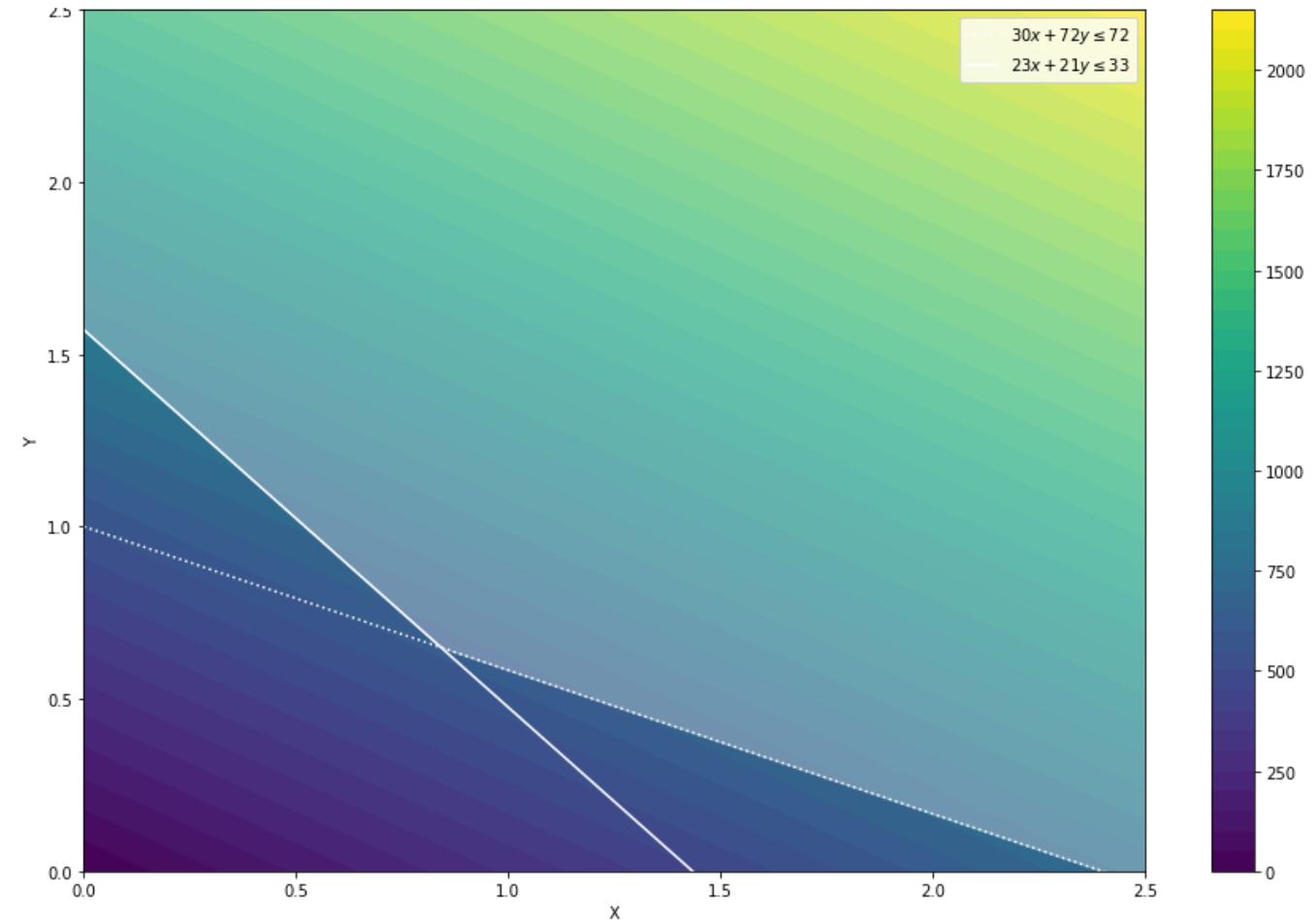
s.t.

$$30y_1 + 72y_2 \geq 72$$

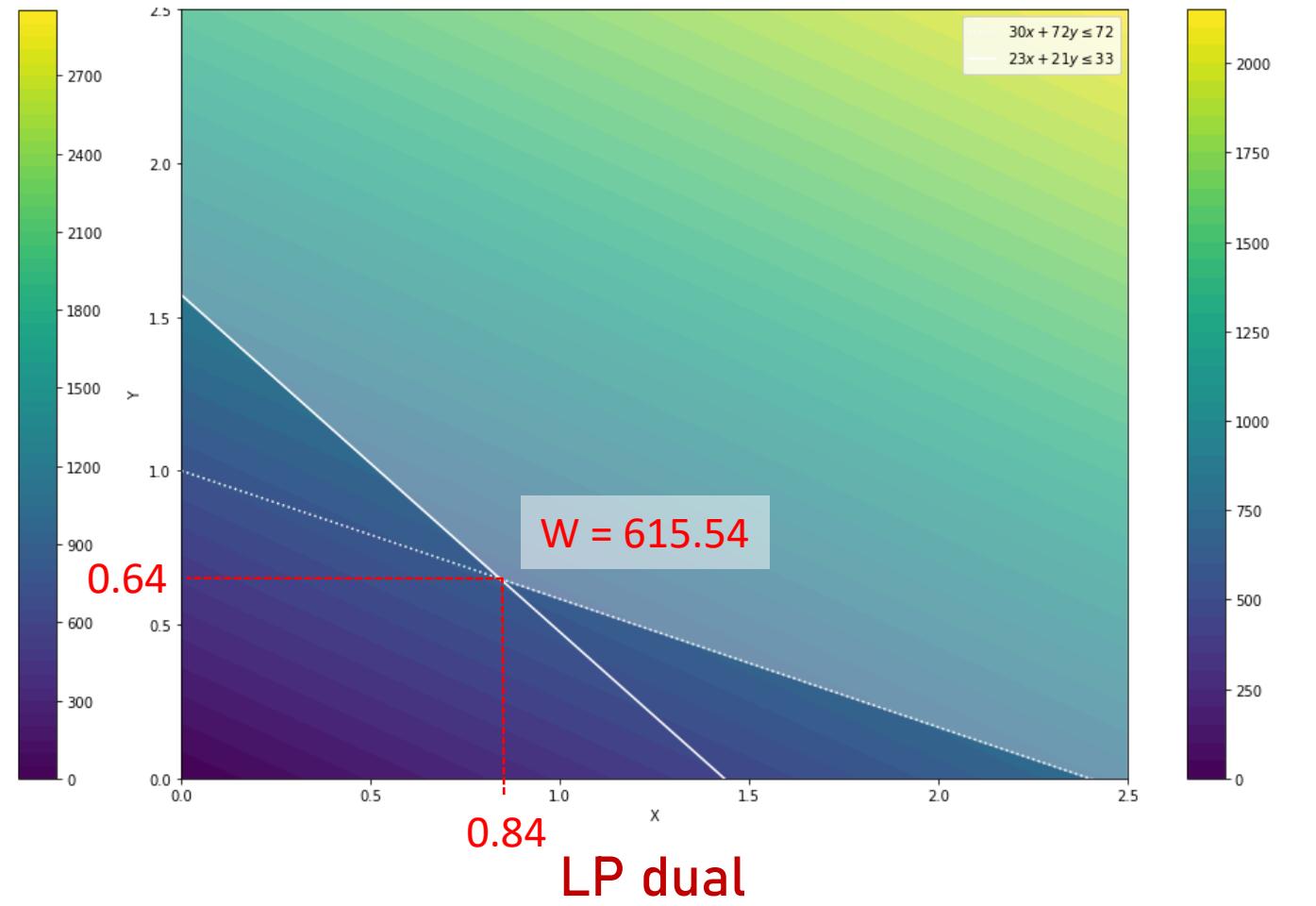
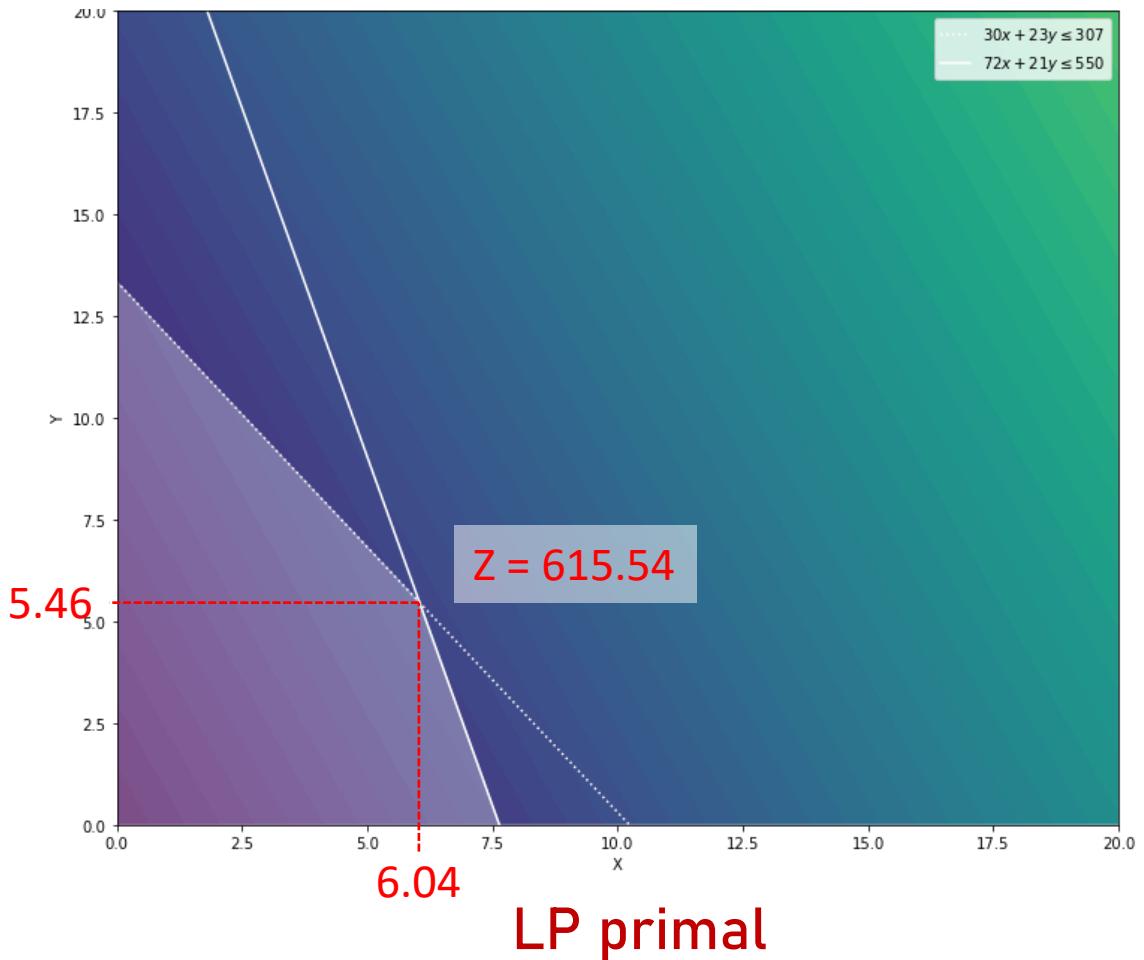
$$23y_1 + 21y_2 \geq 33$$

$$y_1, y_2 \geq 0$$

A este problema lo vamos a llamar: problema LP dual



Ejemplo construcción dual con max



Transformación primal/dual

Primal (maximización)	Dual (minimización)
restricción "i"	variable "i"
\leq	≥ 0
\geq	≤ 0
$=$	<i>irrestricta</i>
variable "j"	restricción "j"
≥ 0	\geq
≤ 0	\leq
<i>irrestricta</i>	$=$

Posibles resultados

Primal	Dual	Dualidad
Factible	Factible	Dualidad fuerte
No acotado	Incompatible	X
Incompatible	No acotado	X
Soluciones alternativas	Solución degenerada	X
Solución degenerada	Soluciones alternativas	X

Ejemplo construcción dual con min

$$\min z = 35x_1 + 71x_2$$

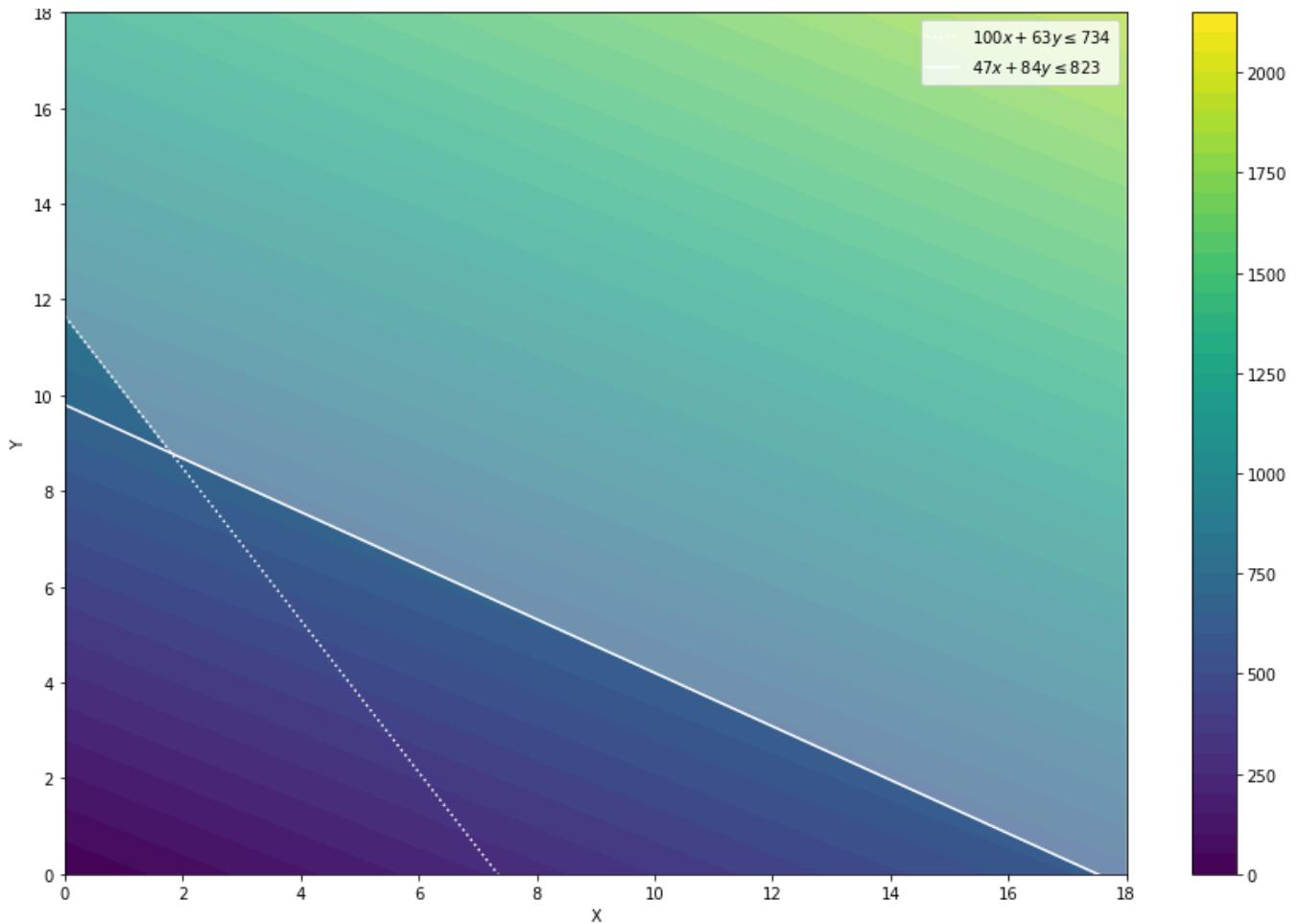
s.t.

$$100x_1 + 63x_2 \geq 734$$

$$47x_1 + 84x_2 \geq 823$$

$$x_1, x_2 \geq 0$$

A este problema lo vamos a llamar: problema LP primal



Ejemplo construcción dual con min

$$\min z = 35x_1 + 71x_2$$

s.t.

$$100x_1 + 63x_2 \geq 734$$

$$47x_1 + 84x_2 \geq 823$$

$$x_1, x_2 \geq 0$$

LP primal

$$\max z = 734y_1 + 823y_2$$

s.t.

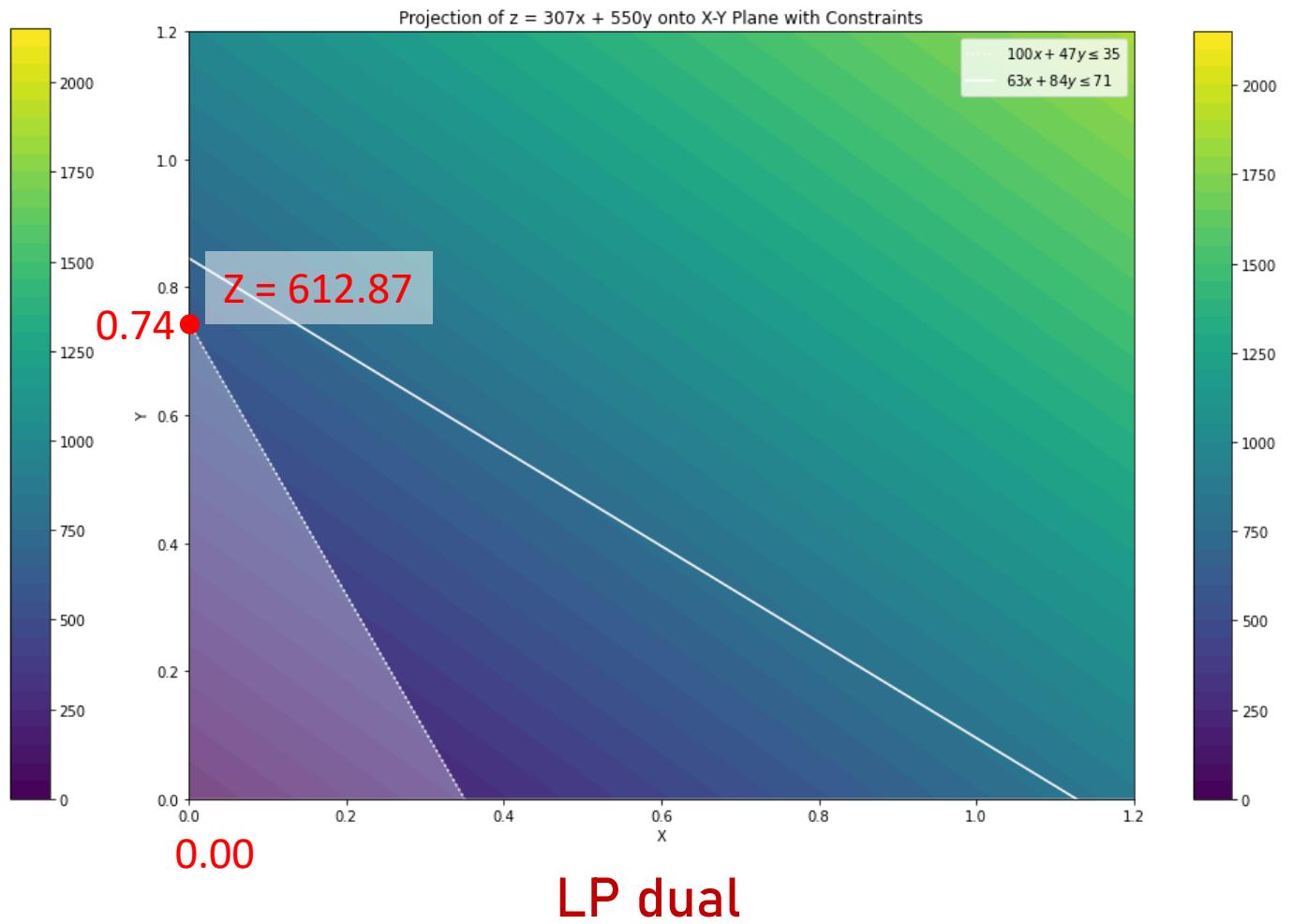
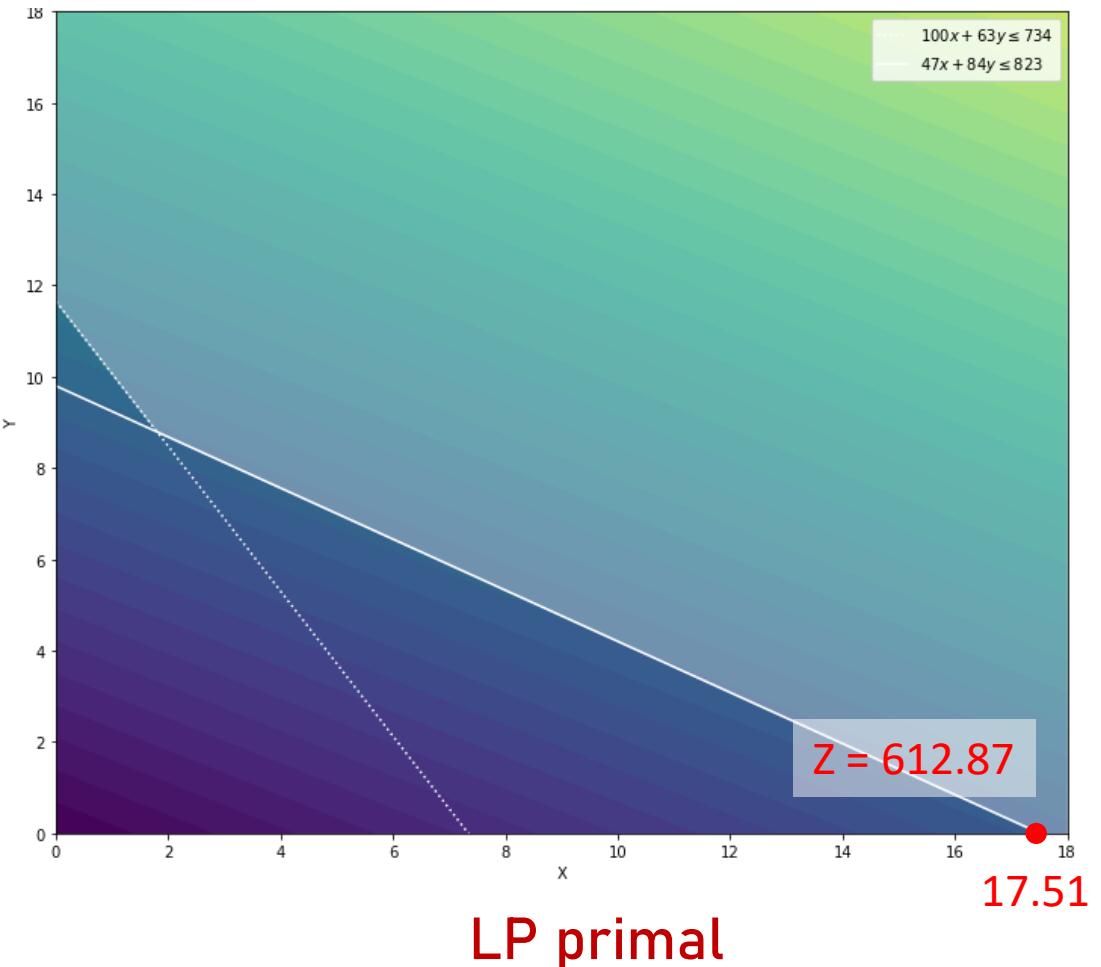
$$100y_1 + 47y_2 \leq 35$$

$$63y_1 + 84y_2 \leq 71$$

$$y_1, y_2 \geq 0$$

LP dual

Ejemplo construcción dual con min

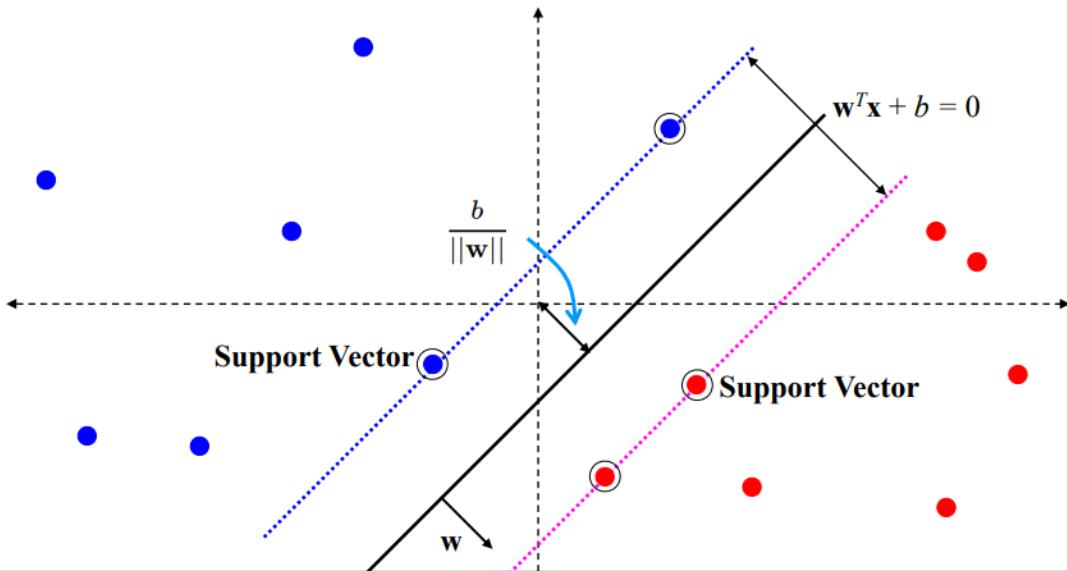


Selección dual/primal

- **Eficiencia en resolución:** Mientras uno tiene “n” restricciones y “m” variables, en el otro se traspone. Depende del contexto.
- **Significado** de cada problema.
- **Análisis de sensibilidad:** las variables duales corresponden a los precios sombra (shadow prices)
- El dual arroja información sobre **cotas** del primal.

Caso dual en Machine Learning

Support Vector Machine



CS19 Machine Learning (2015) Lecture 3: SVM dual, kernels and regression, University of Oxford.

<https://www.robots.ox.ac.uk/~az/lectures/ml/lect3.pdf>

Support Vector Machine es un modelo supervisado de Machine Learning usado para clasificación y regresión.

Implica encontrar un hiperplano que maximice el margen entre clases (ej: fraude y no fraude)

Los puntos que caen en cada margen se llaman vectores de soporte.

La formulación dual del problema:

- Las **variables duales** están asociadas con cada uno de los datos. **Representa la contribución marginal de cada dato al límite de decisión.**
- El problema dual resulta ser de **programación cuadrática** (convexo), es computacionalmente eficiente.