

Universidad Nacional del Altiplano
Facultad de Ingeniería Estadística e Informática
E.P.de Ingeniería Estadística e Informática

Docente: Ing. Torres Cruz Fred

Presentado por: Quispe Ito Luz Leidy

Restricciones y Sistemas de Ecuaciones

Ejercicio 1: Desarrollador de Software

Enunciado:

Un desarrollador tiene 15 horas semanales para dedicar al desarrollo de software de *front-end* (x) y *back-end* (y). Además:

- Debe dedicar al menos 5 horas al desarrollo de *front-end* para cumplir con los entregables del cliente.
- El tiempo total no puede exceder 15 horas por restricciones del sprint.

Formule las restricciones, represéntelas gráficamente e identifique las combinaciones posibles de tiempo a invertir en cada actividad.

Restricciones:

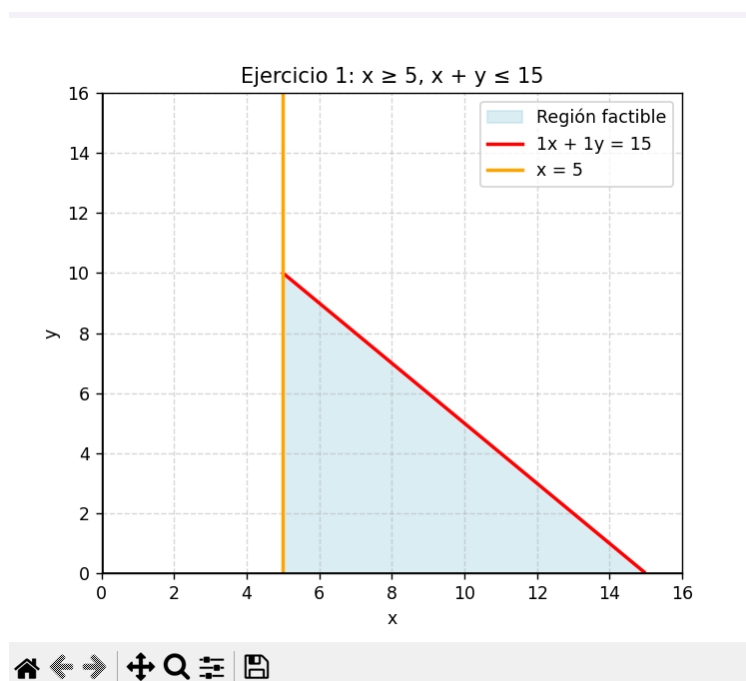
$$\begin{cases} x \geq 5 & (\text{mínimo tiempo en front-end}) \\ x + y \leq 15 & (\text{tiempo total disponible}) \\ x, y \geq 0 & (\text{no se puede trabajar tiempo negativo}) \end{cases}$$

Gráfico de la región factible:

```

1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 def ejercicio1():
5     # Rango de valores
6     x = np.linspace(0, 16, 200)
7     y = 15 - x # Línea:  $x + y = 15$ 
8
9     # Crear figura
10    plt.figure(figsize=(7,6))
11
12    # Dibujar región factible:  $x \geq 5$ ,  $y \geq 0$ ,  $x + y \leq 15$ 
13    x_fill = np.linspace(5, 15, 200)
14    y_fill = 15 - x_fill
15    plt.fill_between(x_fill, 0, y_fill, color='lightblue', alpha=0.5)
16
17    # Dibujar líneas de restricción
18    plt.plot(x, y, 'r-', label='x + y = 15')
19    plt.axvline(5, color='orange', label='x = 5')
20
21    # Ejes
22    plt.axhline(0, color='black')
23    plt.axvline(0, color='black')
24
25    # Configuración del gráfico
26    plt.xlim(0, 16)
27    plt.ylim(0, 16)
28    plt.xlabel("x (front-end)")
29    plt.ylabel("y (back-end)")
30    plt.title("Restricciones:  $x \geq 5$ ,  $x + y \leq 15$ ,  $x, y \geq 0$ ")
31    plt.legend()
32    plt.grid(True, linestyle='--', alpha=0.6)
33
34    # Mostrar
35    plt.show()
36
37 # Ejecutar
38 ejercicio1()
39
40

```



Interpretación:

La región factible está formada por todas las combinaciones de horas dedicadas al front-end y back-end que cumplen con las restricciones. El área se encuentra a la derecha de $x = 5$ y por debajo de la línea $x + y = 15$. El desarrollador puede distribuir el tiempo de distintas maneras, por ejemplo: (5,10), (6,9), (8,7) o (10,5). La elección depende de los objetivos de trabajo semanales.

Ejercicio 2: Ingeniero de Datos - Servidores

Enunciado:

Un ingeniero de datos debe asignar horas para la configuración de servidores locales (x) y en la nube (y). Cada hora en servidores locales cuesta 3 unidades y en la nube cuesta 5. El presupuesto total no debe exceder 20 unidades monetarias.

- Cada hora en servidores locales cuesta 3 unidades.
- Cada hora en la nube cuesta 5 unidades.
- El presupuesto máximo es 20 unidades.

Restricciones:

$$\begin{cases} 3x + 5y \leq 20 & \text{(presupuesto máximo)} \\ x, y \geq 0 & \text{(horas no negativas)} \end{cases}$$

Gráfico de la región factible:

```
import numpy as np
import matplotlib.pyplot as plt

def ejercicio2():
    x = np.linspace(0, 8, 200)
    y = (20 - 3*x) / 5 # Línea de restricción:  $3x + 5y = 20$ 

    plt.figure(figsize=(7,6))

    # Región factible:  $3x + 5y \leq 20$ ,  $x \geq 0$ ,  $y \geq 0$ 
    x_fill = np.linspace(0, 8, 200)
    y_fill = np.maximum(0, (20 - 3*x_fill)/5)
    plt.fill_between(x_fill, 0, y_fill, color='lightgreen', alpha=0.5)

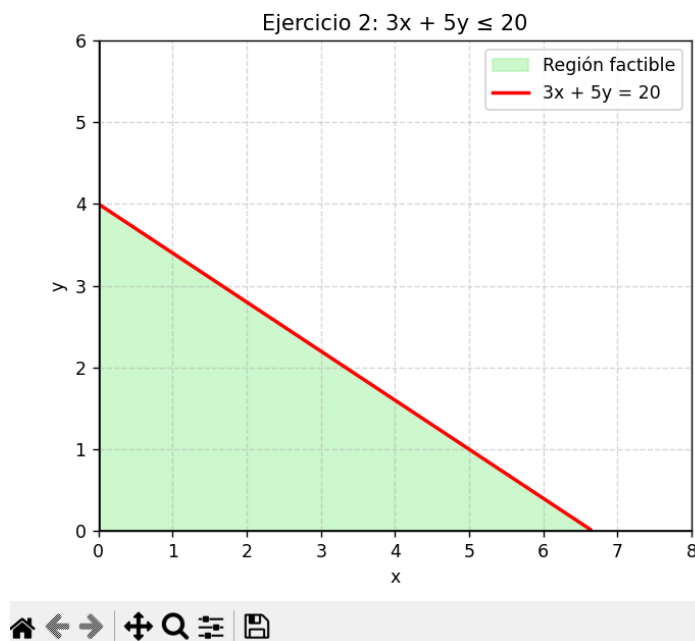
    plt.plot(x, y, 'r-', linewidth=2, label='3x + 5y = 20')

    plt.axhline(0, color='black')
    plt.axvline(0, color='black')

    plt.xlim(0, 8)
    plt.ylim(0, 6)
    plt.xlabel("x (horas servidor A)")
    plt.ylabel("y (horas servidor B)")
    plt.title("Restricción:  $3x + 5y \leq 20$ ,  $x, y \geq 0$ ")
    plt.legend()
    plt.grid(True, linestyle='--', alpha=0.6)

    plt.show()

ejercicio2()
```



Interpretación:

El área factible representa las combinaciones posibles de horas dedicadas a servidores locales y en la nube que no superan el presupuesto total. Por ejemplo, si se dedican 2 horas a la nube ($y = 2$), se pueden destinar hasta $3x = 10$ unidades, es decir, $x = 3,3$. El ingeniero debe elegir un punto dentro de la región factible según la eficiencia deseada o prioridad de trabajo.

Ejercicio 3: Administrador de Proyectos

Enunciado:

Un administrador de proyectos debe distribuir el tiempo semanal entre reuniones (x) y documentación (y). Debe cumplir las siguientes condiciones:

- Asistir al menos a 4 horas de reuniones.
- Dedicarse al menos 6 horas a la documentación.
- No puede exceder un total de 12 horas.

Restricciones:

$$\begin{cases} x \geq 4 & (\text{mínimo de reuniones}) \\ y \geq 6 & (\text{mínimo de documentación}) \\ x + y \leq 12 & (\text{tiempo total disponible}) \\ x, y \geq 0 & (\text{horas no negativas}) \end{cases}$$

Gráfico de la región factible:

```

import numpy as np
import matplotlib.pyplot as plt

def ejercicio3():

    x = np.linspace(0, 13, 200)
    y_linea = 12 - x # Línea de restricción  $x + y = 12$ 

    plt.figure(figsize=(7,6))

    x_fill = np.linspace(4, 6, 200)
    y_fill = 12 - x_fill
    plt.fill_between(x_fill, 6, y_fill, color='lightcoral', alpha=0.5)

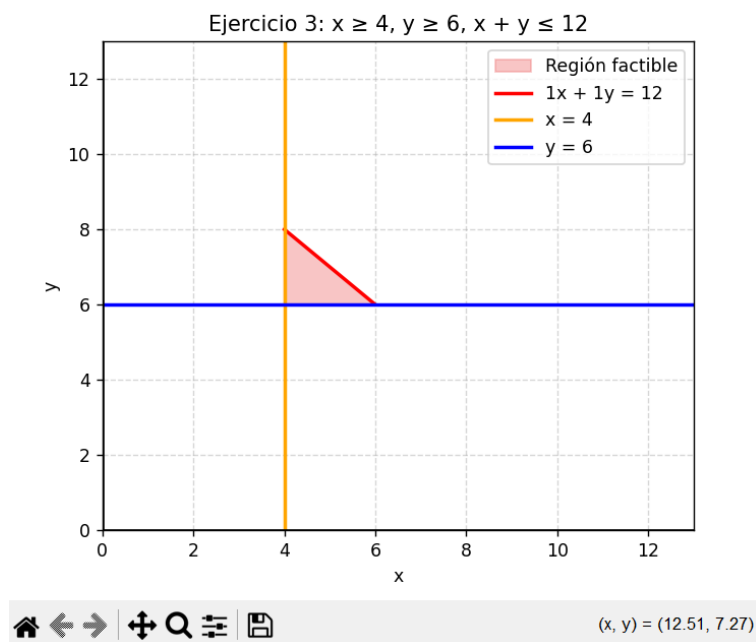
    plt.axvline(4, color='orange', linewidth=2, label='x = 4')
    plt.axhline(6, color='blue', linewidth=2, label='y = 6')
    plt.plot(x, y_linea, 'r-', linewidth=2, label='x + y = 12')

    # Ejes
    plt.axhline(0, color='black')
    plt.axvline(0, color='black')

    # Configuración del gráfico
    plt.xlim(0, 13)
    plt.ylim(0, 13)
    plt.xlabel("x (horas de reuniones)")
    plt.ylabel("y (horas de documentación)")
    plt.title("Restricciones:  $x \geq 4$ ,  $y \geq 6$ ,  $x + y \leq 12$ ")
    plt.legend()
    plt.grid(True, linestyle='--', alpha=0.6)

    plt.show()
|
ejercicio3()

```



Interpretación:

La región factible es un pequeño triángulo en el plano donde se cumplen todas las condiciones. El administrador puede combinar los tiempos siempre que sume como máximo 12 horas. Algunos ejemplos de combinaciones válidas son (4,8), (5,7) o (6,6). Esto le permite equilibrar las horas de reuniones y documentación sin sobrepasar el límite semanal.

Ejercicio 4: Empresa de Videojuegos

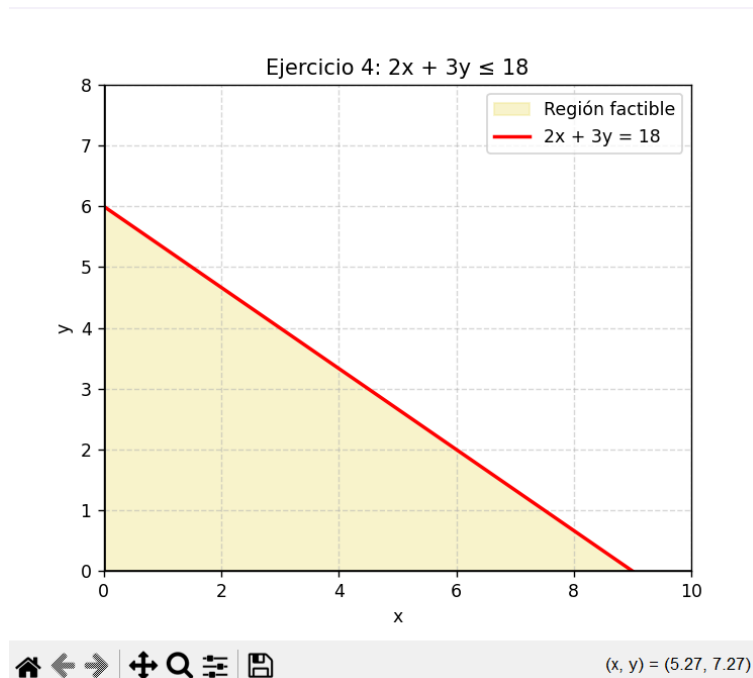
Enunciado:

Una empresa de videojuegos planifica la producción de dos tipos de *assets*: modelos 3D (x) y animaciones (y). Cada modelo requiere 2 horas y cada animación 3 horas. El total de tiempo disponible es 18 horas.

- Cada modelo 3D requiere 2 horas.
- Cada animación requiere 3 horas.
- El tiempo total disponible es 18 horas.

Restricciones:

$$\begin{cases} 2x + 3y \leq 18 & \text{(tiempo total de producción)} \\ x, y \geq 0 & \text{(no se pueden producir valores negativos)} \end{cases}$$

Gráfico de la región factible:

```

import numpy as np
import matplotlib.pyplot as plt

def ejercicio4():

    # Rango de valores
    x = np.linspace(0, 10, 200)
    y = (18 - 2*x) / 3 # Línea de restricción

    # Crear figura
    plt.figure(figsize=(7,6))

    # Región factible: debajo de la línea  $y \geq 0$ 
    x_fill = np.linspace(0, 9, 200)
    y_fill = np.maximum(0, (18 - 2*x_fill) / 3)
    plt.fill_between(x_fill, 0, y_fill, color='lightgreen', alpha=0.5)

    # Dibujar la línea de restricción
    plt.plot(x, y, 'r-', linewidth=2, label='2x + 3y = 18')

    # Ejes
    plt.axhline(0, color='black')
    plt.axvline(0, color='black')

    # Configuración del gráfico
    plt.xlim(0, 10)
    plt.ylim(0, 8)
    plt.xlabel("x (tiempo en assets 2D)")
    plt.ylabel("y (tiempo en assets 3D)")
    plt.title("Restricción:  $2x + 3y \leq 18$ ,  $x, y \geq 0$ ")
    plt.legend()
    plt.grid(True, linestyle='--', alpha=0.6)

    # Mostrar
    plt.show()

# Ejecutar
ejercicio4()

```

Interpretación:

La gráfica muestra todas las combinaciones posibles de modelos y animaciones que se pueden producir sin superar las 18 horas disponibles. Por ejemplo, si se hacen 3 animaciones ($y = 3$), quedan 9 horas disponibles para modelos, es decir, $x = 4,5$. El área factible es la parte del plano por debajo de la línea $2x + 3y = 18$.

Ejercicio 5: Startup de Hardware

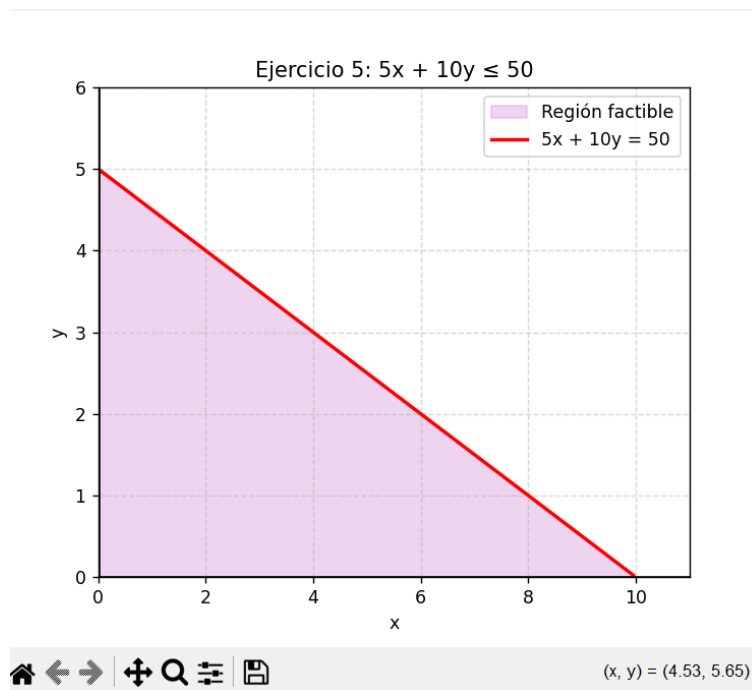
Enunciado:

Una startup fabrica dos tipos de dispositivos electrónicos: sensores (x) y controladores (y). Cada sensor requiere 5 unidades de componentes y cada controlador requiere 10 unidades. La empresa dispone de un máximo de 50 unidades de componentes.

- Cada sensor utiliza 5 unidades de componentes.
- Cada controlador utiliza 10 unidades.
- El total de componentes no debe exceder 50 unidades.

Restricciones:

$$\begin{cases} 5x + 10y \leq 50 & (\text{límite de componentes disponibles}) \\ x, y \geq 0 & (\text{producción no negativa}) \end{cases}$$

Gráfico de la región factible:**Interpretación:**

La región factible representa las posibles combinaciones de sensores y controladores que pueden fabricarse sin superar las 50 unidades de componentes disponibles. Por ejemplo, fabricar 5 sensores y 3 controladores implica $5(5) + 10(3) = 55$ unidades, lo cual excede el límite; mientras que $(5,2)$ o $(8,1)$ son combinaciones válidas. El área por debajo de la línea $5x + 10y = 50$ muestra todas las opciones de producción posibles.

```
import numpy as np
import matplotlib.pyplot as plt

def ejercicio5():
    |
    # Rango de valores
    x = np.linspace(0, 11, 200)
    y = (50 - 5*x) / 10 # Línea de restricción

    # Crear figura
    plt.figure(figsize=(7,6))

    # Región factible: debajo de la línea  $y \geq 0$ 
    x_fill = np.linspace(0, 10, 200)
    y_fill = np.maximum(0, (50 - 5*x_fill) / 10)
    plt.fill_between(x_fill, 0, y_fill, color='lightblue', alpha=0.5)

    # Dibujar la línea de restricción
    plt.plot(x, y, 'r-', linewidth=2, label='5x + 10y = 50')

    # Ejes
    plt.axhline(0, color='black')
    plt.axvline(0, color='black')

    # Configuración del gráfico
    plt.xlim(0, 11)
    plt.ylim(0, 6)
    plt.xlabel("x (dispositivos tipo A)")
    plt.ylabel("y (dispositivos tipo B)")
    plt.title("Restricción:  $5x + 10y \leq 50$ ,  $x, y \geq 0$ ")
    plt.legend()
    plt.grid(True, linestyle='--', alpha=0.6)

    # Mostrar
    plt.show()

# Ejecutar
ejercicio5()
```