ESCUELA POLITÉCNICA NACIONAL
MÉTODOS NUMÉRICOS

Leandro Bravo GR1CC

Tarea 09 - Eliminación Gaussiana vs Gauss-Jordan

```
%load_ext autoreload
import numpy as np
from src import eliminacion_gaussiana_redondeo,eliminacion_gaussiana, multiplicar_matriz_vector
from src import gauss_jordan
```

```
The autoreload extension is already loaded. To reload it, use:
  %reload_ext autoreload
```

Conjunto de Ejercicios

1. Para cada uno de los siguientes sistemas lineales, obtenga, de ser posible, una solución con métodos gráficos. Explique los resultados desde un punto de vista geométrico.

a. $x_1 + 2x_2 = 0,$
$x_1 - x_2 = 0.$
$x_1 - 3x_2 = 5$

b. $x_1 + 2x_2 = 3,$
$-2x_1 - 4x_2 = 6.$

c. $2x_1 + x_2 = -1,$
$x_1 + x_2 = 2,$

d. $2x_1 + x_2 + x_3 = 1,$
$2x_1 + 4x_2 - x_3 = -1.$

2. Utilice la eliminación gaussiana con sustitución hacia atrás y aritmética de redondeo de dos dígitos para resolver los siguientes sistemas lineales. No reordene las ecuaciones. (La solución exacta para cada sistema es $x_1 = -1$, $x_2 = 1, x_3 = 3$)

a. $-x_1 + 4x_2 + x_3 = 8,$
$\frac{5}{3}x_1 + \frac{2}{3}x_2 + \frac{2}{3}x_3 = 1,$
$2x_1 + x_2 + 4x_3 = 11.$

b. $4x_1 + 2x_2 - x_3 = -5,$
$\frac{1}{9}x_1 + \frac{1}{9}x_2 - \frac{1}{3}x_3 = -1,$
$x_1 + 4x_2 + 2x_3 = 9$

```
print("Literal a:\n")
%autoreload 2
A = [[-1,4,1,8],[5/3,2/3,2/3,1],[2,1,4,11]]
sol_a = eliminacion_gaussiana_redondeo(A)
print("La solución es:",sol_a)

print("\nLiteral b:\n")
B = [[4,2,-1,-5],[1/9,1/9,-1/3,-1],[1,4,2,9]]
sol_b = eliminacion_gaussiana_redondeo(B)
print("La solución es:",sol_b)
```

```
Literal a:

[07-28 11:36:49][INFO]
[[ 2.00e+00  1.00e+00  4.00e+00  1.10e+01]
 [ 1.00e-02 -1.60e-01 -2.65e+00 -8.13e+00]
 [ 0.00e+00  4.50e+00  3.00e+00  1.35e+01]]
[07-28 11:36:49][INFO]
[[ 2.00e+00  1.00e+00  4.00e+00  1.10e+01]
 [ 0.00e+00  4.50e+00  3.00e+00  1.35e+01]
 [ 1.00e-02  2.00e-02 -2.53e+00 -7.59e+00]]
La solución es: [-1.  1.  3.]

Literal b:

[07-28 11:36:49][INFO]
[[ 4.000e+00  2.000e+00 -1.000e+00 -5.000e+00]
 [-1.000e-02  5.000e-02 -3.000e-01 -8.500e-01]
 [ 0.000e+00  3.500e+00  2.250e+00  1.025e+01]]
[07-28 11:36:49][INFO]
[[ 4.000e+00  2.000e+00 -1.000e+00 -5.000e+00]
 [ 0.000e+00  3.500e+00  2.250e+00  1.025e+01]
 [-1.000e-02  1.000e-02 -3.200e-01 -9.500e-01]]
La solución es: [-1.02     1.02     2.96875]
```

3. Utilice el algoritmo de eliminación gaussiana para resolver, de ser posible, los siguientes sistemas lineales, y determine si se necesitan intercambios de fila:

a. $x_1 - x_2 + 3x_3 = 2,$
$3x_1 - 3x_2 + x_3 = -1,$
$x_1 + x_2 \quad = 3.$

b. $2x_1 - 1.5x_2 + 3x_3 = 1,$
$-x_1 + 2x_3 \quad = 3,$
$4x_1 - 4.5x_2 + 5x_3 = 1,$

b. $2x_1 \quad = 3,$
$x_1 + 1.5x_2 \quad = 4.5,$
$-3x_2 + 0.5x_3 \quad = -6.6,$
$2x_1 - 2x_2 + x_3 + x_4 = 0.8.$

d. $x_1 + x_2 + x_4 \quad = 2,$
$2x_1 + x_2 - x_3 + x_4 \quad = 1,$
$4x_1 - x_2 - 2x_3 + 2x_4 = 0,$
$3x_1 - x_2 - x_3 + 2x_4 \quad = -3.$

```python
%autoreload 2
print("Literal a:\n")
A = [[1,-1,3,2],[3,-3,1,-1],[1,1,0,3]]
sol_a = eliminacion_gaussiana(A)
print("\nLa solución es:",sol_a)

print("\nLiteral b:\n")
B = [[2,-1.5,3,1],[-1,0,2,3],[4,-4.5,5,1]]
sol_b = eliminacion_gaussiana(B)
print("\nLa solución es:",sol_b)

print("\nLiteral c:\n")
C = [[2,0,0,0,3],[1,1.5,0,0,4.5],[0,-3,0.5,0,-6.6],[2,-2,1,1,0.8]]
sol_c = eliminacion_gaussiana(C)
print("\nLa solución es:",sol_c)

print("\nLiteral d:\n")
D = [[1,1,0,1,2],[2,1,-1,1,1],[4,-1,-2,2,0],[3,-1,-1,2,-3]]
sol_d = eliminacion_gaussiana(D)
```

```
Literal a:

[07-28 11:36:49][INFO]
[[ 3.         -3.          1.         -1.        ]
 [ 0.          0.          2.6666667   2.3333333 ]
 [ 0.          2.         -0.33333334  3.3333333 ]]
[07-28 11:36:49][INFO]
[[ 3.         -3.          1.         -1.        ]
 [ 0.          2.         -0.33333334  3.3333333 ]
 [ 0.          0.          2.6666667   2.3333333 ]]

La solución es: [1.1875     1.8125     0.87499994]

Literal b:

[07-28 11:36:49][INFO]
[[ 4.    -4.5    5.     1.   ]
 [ 0.    -1.125  3.25   3.25 ]
 [ 0.     0.75   0.5    0.5  ]]
[07-28 11:36:49][INFO]
[[ 4.        -4.5        5.         1.       ]
 [ 0.        -1.125      3.25       3.25     ]
 [ 0.         0.         2.6666667  2.6666667]]

La solución es: [-1. -0.  1.]

Literal c:

[07-28 11:36:49][INFO]
[[ 2.   0.   0.   0.   3. ]
 [ 0.   1.5  0.   0.   3. ]
 [ 0.  -3.   0.5  0.  -6.6]
 [ 0.  -2.   1.   1.  -2.2]]
[07-28 11:36:49][INFO]
[[ 2.          0.          0.          0.          3.        ]
 [ 0.         -3.          0.5         0.         -6.6       ]
 [ 0.          0.          0.25        0.         -0.29999995]
 [ 0.          0.          0.6666666   1.          2.2       ]]
[07-28 11:36:49][INFO]
[[ 2.          0.          0.          0.          3.        ]
 [ 0.         -3.          0.5         0.         -6.6       ]
 [ 0.          0.          0.6666666   1.          2.2       ]
 [ 0.          0.          0.         -0.37500003 -1.125     ]]

La solución es: [ 1.5        2.        -1.1999997  2.9999998]

Literal d:

[07-28 11:36:49][INFO]
```

```
[[ 4.   -1.   -2.    2.    0.  ]
 [ 0.    1.5   0.    0.    1.  ]
 [ 0.    1.25  0.5   0.5   2.  ]
 [ 0.   -0.25  0.5   0.5  -3.  ]]
[07-28 11:36:49][INFO]
[[ 4.        -1.        -2.         2.         0.       ]
 [ 0.         1.5        0.         0.         1.       ]
 [ 0.         0.         0.5        0.5        1.1666667]
 [ 0.         0.         0.5        0.5       -2.8333333]]
[07-28 11:36:49][INFO]
[[ 4.        -1.        -2.         2.         0.       ]
 [ 0.         1.5        0.         0.         1.       ]
 [ 0.         0.         0.5        0.5        1.1666667]
 [ 0.         0.         0.         0.        -4.       ]]

No existe solución.
```

4. Use el algoritmo de eliminación gaussiana y la aritmética computacional de precisión de 32 bits para resolver los siguientes sistemas lineales.

a.
$$\begin{aligned}\frac{1}{4}x_1+\frac{1}{5}x_2+\frac{1}{6}x_3&=9,\\ \frac{1}{3}x_1+\frac{1}{4}x_2+\frac{1}{5}x_3&=8,\\ \frac{1}{2}x_1+x_2+2x_3&=8.\end{aligned}$$

b.
$$\begin{aligned}3.333x_1+15920x_2-10.333x_3&=15913,\\ 2.222x_1+16.71x_2+9.612x_3&=28.544,\\ 1.5611x_1+5.1791x_2+1.6852x_3&=8.4254.\end{aligned}$$

b.
$$\begin{aligned}x_1+\frac{1}{2}x_2+\frac{1}{3}x_3+\frac{1}{4}x_4&=\frac{1}{6},\\ \frac{1}{2}x_1+\frac{1}{3}x_2+\frac{1}{4}x_3+\frac{1}{5}x_4&=\frac{1}{7},\\ \frac{1}{3}x_1+\frac{1}{4}x_2+\frac{1}{5}x_3+\frac{1}{6}x_4&=\frac{1}{8},\\ \frac{1}{4}x_1+\frac{1}{5}x_2+\frac{1}{6}x_3+\frac{1}{7}x_4&=\frac{1}{9}.\end{aligned}$$

d.
$$\begin{aligned}2x_1+x_2-x_3+x_4-3x_5&=7,\\ x_1+2x_3-x_4+x_5&=2,\\ -2x_2-x_3+x_4-x_5&=-5,\\ 3x_1+x_2-4x_3+5x_5&=6,\\ x_1-x_2-x_3-x_4+x_5&=-3\end{aligned}$$

```python
%autoreload 2
print("Literal a:\n")
A = [[1/4,1/5,1/6,9],[1/3,1/4,1/5,8],[1/2,1,2,8]]
sol_a = eliminacion_gaussiana(A)
print("\nLa solución es:",sol_a)

print("\nLiteral b:\n")
B = [[3.333,15920,-10.333,15913],[2.222,16.71,9.612,28.544],[1.5611,5.1791,1.6852,8.4254]]
sol_b = eliminacion_gaussiana(B)
print("\nLa solución es:",sol_b)

print("\nLiteral c:\n")
C = [[1,1/2,1/3,1/4,1/6],[1/2,1/3,1/4,1/5,1/7],[1/3,1/4,1/5,1/6,1/8],[1/4,1/5,1/6,1/7,1/9]]
sol_c = eliminacion_gaussiana(C)
print("\nLa solución es:",sol_c)

print("\nLiteral d:\n")
D = [[2,1,-1,1,-3,7],[1,0,2,-1,1,2],[0,-2,-1,1,-1,-5],[3,1,-4,0,5,6],[1,-1,-1,-1,1,-3]]
sol_d = eliminacion_gaussiana(D)
print("\nLa solución es:",sol_d)
```

```
Literal a:

[07-28 11:36:50][INFO]
[[ 0.5        1.         2.         8.       ]
 [ 0.        -0.4166667 -1.1333333  2.6666665]
 [ 0.        -0.3       -0.8333333  5.       ]]
[07-28 11:36:50][INFO]
[[ 0.5         1.          2.          8.        ]
 [ 0.         -0.4166667  -1.1333333   2.6666665 ]
 [ 0.          0.         -0.01733333  3.0800002 ]]

La solución es: [-227.07697  476.92322 -177.69237]

Literal b:

[07-28 11:36:50][INFO]
[[ 3.3329999e+00  1.5920000e+04 -1.0333000e+01  1.5913000e+04]
 [ 0.0000000e+00 -1.0596623e+04  1.6500668e+01 -1.0580122e+04]
 [ 0.0000000e+00 -7.4513804e+03  6.5249376e+00 -7.4448555e+03]]
[07-28 11:36:50][INFO]
[[ 3.3329999e+00  1.5920000e+04 -1.0333000e+01  1.5913000e+04]
 [ 0.0000000e+00 -1.0596623e+04  1.6500668e+01 -1.0580122e+04]
```

```
 [ 0.0000000e+00  0.0000000e+00 -5.0780745e+00 -5.0786133e+00]]

La solución es: [0.9997431 1.0000001 1.0001061]

Literal c:

[07-28 11:36:50][INFO]
[[1.         0.5        0.33333334 0.25       0.16666667]
 [0.         0.08333334 0.08333333 0.075      0.05952381]
 [0.         0.08333333 0.08888888 0.08333334 0.06944444]
 [0.         0.075      0.08333334 0.08035715 0.06944445]]
[07-28 11:36:50][INFO]
[[1.         0.5        0.33333334 0.25       0.16666667]
 [0.         0.08333334 0.08333333 0.075      0.05952381]
 [0.         0.         0.00555557 0.00833335 0.00992064]
 [0.         0.         0.00833335 0.01285715 0.01587302]]
[07-28 11:36:50][INFO]
[[ 1.0000000e+00  5.0000000e-01  3.3333334e-01  2.5000000e-01
   1.6666667e-01]
 [ 0.0000000e+00  8.3333343e-02  8.3333328e-02  7.5000003e-02
   5.9523813e-02]
 [ 0.0000000e+00  0.0000000e+00  8.3333477e-03  1.2857154e-02
   1.5873022e-02]
 [ 0.0000000e+00  0.0000000e+00  0.0000000e+00 -2.3809634e-04
  -6.6138618e-04]]

La solución es: [-0.03174745  0.59525675 -2.3809996   2.7778091 ]

Literal d:

[07-28 11:36:50][INFO]
[[ 3.          1.         -4.          0.          5.          6.        ]
 [ 0.         -0.33333334  3.3333335  -1.         -0.66666675  0.        ]
 [ 0.         -2.         -1.          1.         -1.         -5.        ]
 [ 0.          0.3333333   1.6666667   1.         -6.3333335   3.        ]
 [ 0.         -1.3333334   0.33333337 -1.         -0.66666675 -5.        ]]
[07-28 11:36:50][INFO]
[[ 3.0000000e+00  1.0000000e+00 -4.0000000e+00  0.0000000e+00
   5.0000000e+00  6.0000000e+00]
 [ 0.0000000e+00 -2.0000000e+00 -1.0000000e+00  1.0000000e+00
  -1.0000000e+00 -5.0000000e+00]
 [ 0.0000000e+00  0.0000000e+00  3.5000002e+00 -1.1666666e+00
  -5.0000006e-01  8.3333337e-01]
 [ 0.0000000e+00  0.0000000e+00  1.5000001e+00  1.1666666e+00
  -6.5000000e+00  2.1666667e+00]
 [ 0.0000000e+00  0.0000000e+00  1.0000000e+00 -1.6666667e+00
  -5.9604645e-08 -1.6666665e+00]]
[07-28 11:36:50][INFO]
[[ 3.          1.         -4.          0.          5.          6.        ]
 [ 0.         -2.         -1.          1.         -1.         -5.        ]
 [ 0.          0.          3.5000002  -1.1666666  -0.50000006  0.8333334 ]
 [ 0.          0.          0.          1.6666666  -6.285714    1.8095238 ]
 [ 0.          0.          0.         -1.3333335   0.14285709 -1.9047618 ]]
[07-28 11:36:50][INFO]
[[ 3.          1.         -4.          0.          5.          6.        ]
 [ 0.         -2.         -1.          1.         -1.         -5.        ]
 [ 0.          0.          3.5000002  -1.1666666  -0.50000006  0.8333334 ]
 [ 0.          0.          0.          1.6666666  -6.285714    1.8095238 ]
 [ 0.          0.          0.          0.         -4.885715   -0.45714247]]

La solución es: [1.8830409  2.8070176  0.730994   1.4385961  0.09356716]
```

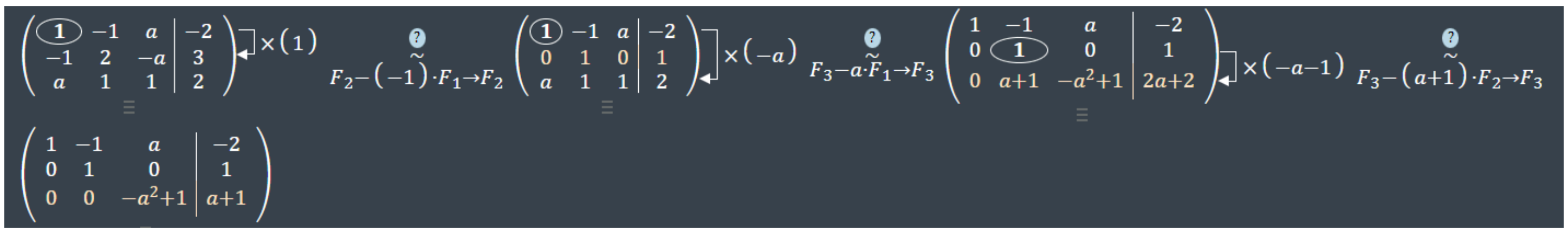
5. Dado el sistema lineal:

$$x_1 - x_2 + \alpha x_3 = -2,$$
$$-x_1 + 2x_2 - \alpha x_3 = 3,$$
$$\alpha x_1 + x_2 + x_3 = 2$$

.

Encuentre el valor(es) de α para los que el sistema no tiene soluciones.
Encuentre el valor(es) de α para los que el sistema tiene un número infinito de soluciones.
Suponga que existe una única solución para una a determinada, encuentre la solución.

Resolución del sistema

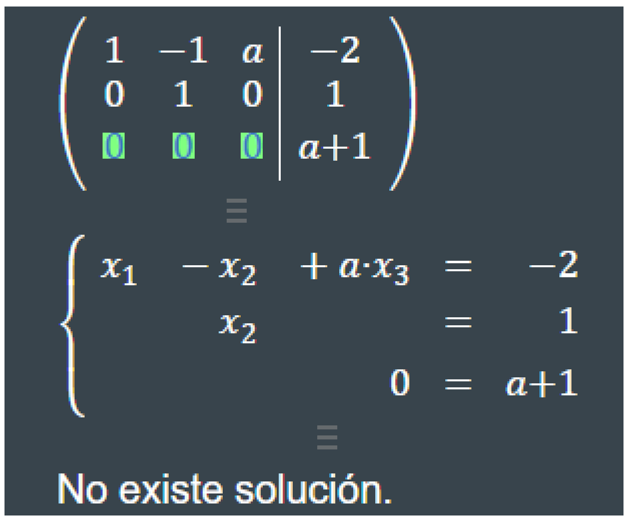


$$\left(\begin{array}{ccc|c} 1 & -1 & a & -2 \\ -1 & 2 & -a & 3 \\ a & 1 & 1 & 2 \end{array}\right) \underset{F_2-(-1)\cdot F_1\to F_2}{\sim} \left(\begin{array}{ccc|c} 1 & -1 & a & -2 \\ 0 & 1 & 0 & 1 \\ a & 1 & 1 & 2 \end{array}\right) \underset{F_3-a\cdot F_1\to F_3}{\sim} \left(\begin{array}{ccc|c} 1 & -1 & a & -2 \\ 0 & 1 & 0 & 1 \\ 0 & a+1 & -a^2+1 & 2a+2 \end{array}\right) \underset{F_3-(a+1)\cdot F_2\to F_3}{\sim}$$

$$\left(\begin{array}{ccc|c} 1 & -1 & a & -2 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & -a^2+1 & a+1 \end{array}\right)$$

image.png

1. a = 1;



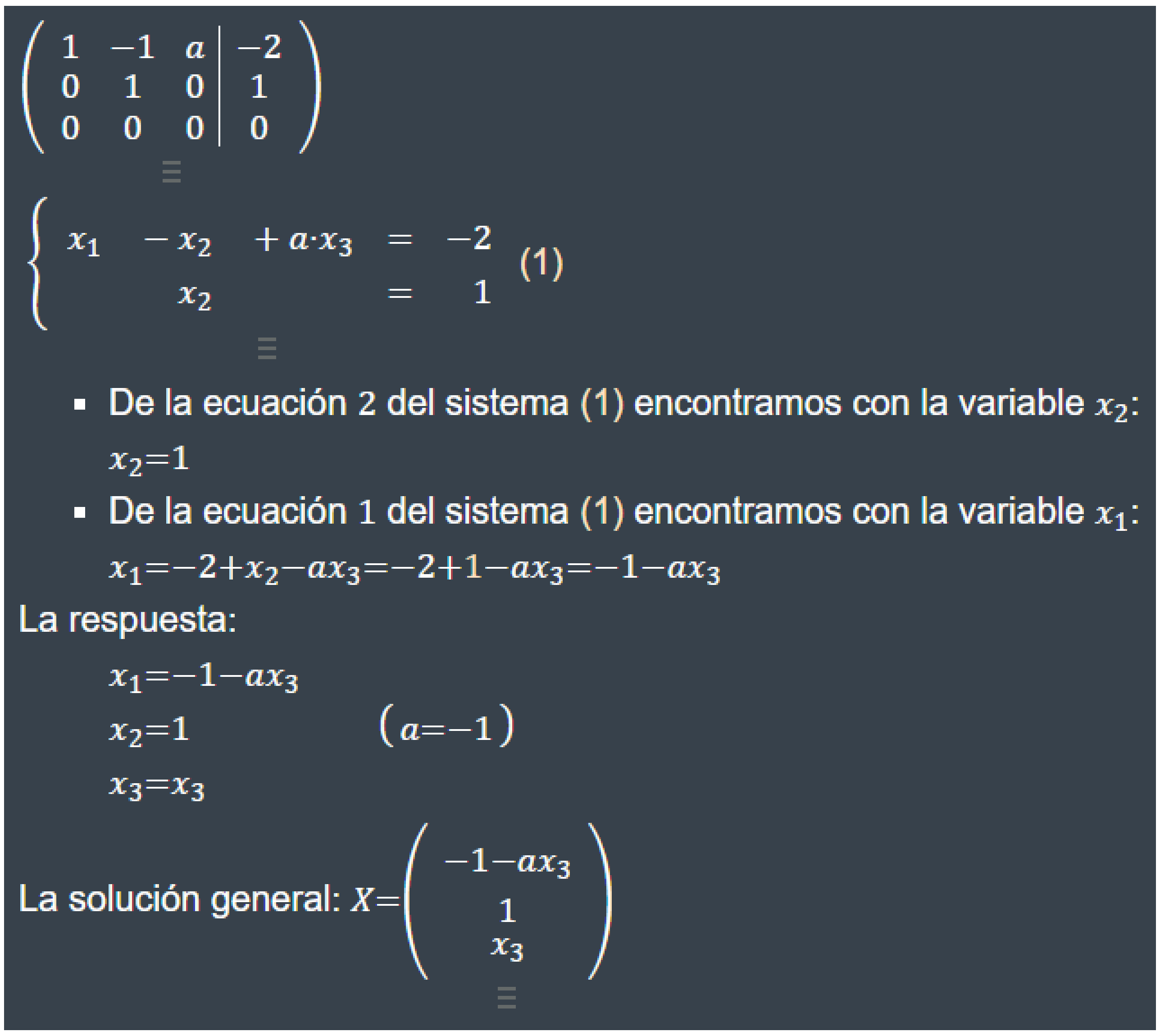
$$\left(\begin{array}{ccc|c} 1 & -1 & a & -2 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & a+1 \end{array}\right)$$

$$\begin{cases} x_1 - x_2 + a\cdot x_3 = -2 \\ x_2 = 1 \\ 0 = a+1 \end{cases}$$

No existe solución.

image.png

2. a = -1;



$$\left(\begin{array}{ccc|c} 1 & -1 & a & -2 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{array}\right)$$

$$\begin{cases} x_1 - x_2 + a\cdot x_3 = -2 \\ x_2 = 1 \end{cases} \quad (1)$$

- De la ecuación 2 del sistema (1) encontramos con la variable x_2:
  $x_2=1$
- De la ecuación 1 del sistema (1) encontramos con la variable x_1:
  $x_1=-2+x_2-ax_3=-2+1-ax_3=-1-ax_3$

La respuesta:

$$x_1=-1-ax_3$$
$$x_2=1 \qquad (a=-1)$$
$$x_3=x_3$$

La solución general: $X=\begin{pmatrix} -1-ax_3 \\ 1 \\ x_3 \end{pmatrix}$

image.png

Ejercicios Aplicados

6. Suponga que en un sistema biológico existen n especies de animales y m fuentes de alimento. Si x_j representa la población de las j-ésimas especies, para cada $j = 1, 2, \cdots$; b_i; representa el suministro diario disponible del i-ésimo alimento y a_{ij} representa la cantidad del i-ésimo alimento.

$$a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1,$$

$$a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2,$$

$$\vdots \qquad \vdots \qquad \qquad \vdots \qquad \vdots$$

$$a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n = b_m,$$

representa un equilibrio donde existe un suministro diario de alimento para cumplir con precisión con el promedio diario de consumo de cada especie.

a. Si

$$A = [a_{ij}] = \begin{bmatrix} 1 & 2 & 0 & 3 \\ 1 & 0 & 2 & 2 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

$ = (x_j)=[1000, 500, 350, 400],$ y $\mathbf{b} = (b_i) = [3500, 2700, 900]$. ¿Existe suficiente alimento para satisfacer el consumo promedio diario?
b. ¿Cuál es el número máximo de animales de cada especie que se podría agregar de forma individual al sistema con el suministro de alimento que cumpla con el consumo?
c. Si la especie 1 se extingue, ¿qué cantidad de incremento individual de las especies restantes se podría soportar?
d. Si la especie 2 se extingue, ¿qué cantidad de incremento individual de las especies restantes se podría soportar?

a. Para saber si existe suficiente alimento para satisfacer el consumo promedio diario, debemos multiplicar la matriz A por el número de especies x.

```
%autoreload 2
A = np.array([[1,2,0,3],[1,0,2,2],[0,0,1,1]])
x = np.array([1000,500,350,400])

b_obtenido =  multiplicar_matriz_vector(A,x)
print("El vector de consumo promedio diario es de:",b_obtenido)
```

```
El vector de consumo promedio diario es de: [3200. 2500.  750.]
```

Por tanto, sí existe suficiente alimento para satisfacer el consumo promedio diario para las diferentes poblaciones de animales.

b. Para saber el número máximo de animales de cada especie que se podría agregar de forma individual, es necesario hallar las soluciones usando A y b.

```
%autoreload 2
A = np.array([[1,2,0,3],[1,0,2,2],[0,0,1,1]])
x = np.array([1005,511,355,491])

b_obtenido =  multiplicar_matriz_vector(A,x)
print("El vector de consumo promedio diario es de:",b_obtenido)
```

```
El vector de consumo promedio diario es de: [3500. 2697.  846.]
```

Por tanto, para que el alimento alcancé para todas las especies estás pueden crecer de la siguiente manera:
- Especie 1: Crece máximo en 5 individuos.
- Especie 2: Crece máximo en 11 individios.
- Especie 3: Crece máximo en 5 individuos.
- Especie 4: Crece máximo en 91 individuos.

c. Que la especie uno se extinga nos quiere decir que $x_1 = 0$. Con lo que nos queda el siguiente sistema de ecuaciones:

$$\begin{aligned} 2x_2 + 3x_4 &= 3500 \\ 2x_3 + 2x_4 &= 2700 \\ x_3 + x_4 &= 900 \end{aligned}$$

```
%autoreload 2
A = np.array([[1,2,0,3],[1,0,2,2],[0,0,1,1]])
x = np.array([0,1000,400,500])

b_obtenido =  multiplicar_matriz_vector(A,x)
print("El vector de consumo promedio diario es de:",b_obtenido)
```

```
El vector de consumo promedio diario es de: [3500. 1800.  900.]
```

Por tanto, si la especie 1 se extingue, las otras especies tener los siguientes incrementos de su población:
- Especie 2: Se incrementa en 500.
- Especie 3: Se incrementa en 50.
- Especie 4: Se incrementa en 100.

d. Que la especie dos se extinga nos quiere decir que $x_2 = 0$. Con lo que nos queda el siguiente sistema de ecuaciones:

$$\begin{aligned} x_1 + 3x_4 &= 3500 \\ x_1 + 2x_4 &= 2700 \\ x_3 + x_4 &= 900 \end{aligned}$$

```
%autoreload 2
A = np.array([[1,2,0,3],[1,0,2,2],[0,0,1,1]])
```

```
x = np.array([1080,0,380,430])

b_obtenido =  multiplicar_matriz_vector(A,x)
print("El vector de consumo promedio diario es de:",b_obtenido)
```

```
El vector de consumo promedio diario es de: [2370. 2700.  810.]
```

Por tanto, si la especie 2 se extingue, las otras especies tener los siguientes incrementos de su población:
- Especie 1: Se incrementa en 80.
- Especie 3: Se incrementa en 30.
- Especie 4: Se incrementa en 30.

Estos incrementos son máximos ya que si se varían un poco, el alimento dispuesto para las especies restante no es suficiente.

Ejercicios Teóricos

7. Repita el ejercicio 4 con el método Gauss-Jordan

```
%autoreload 2
print("Literal a:\n")
A = [[1/4,1/5,1/6,9],[1/3,1/4,1/5,8],[1/2,1,2,8]]
sol_a = gauss_jordan(A)
print("\nLa solución es:",sol_a)

print("\nLiteral b:\n")
B = [[3.333,15920,-10.333,15913],[2.222,16.71,9.612,28.544],[1.5611,5.1791,1.6852,8.4254]]
sol_b = gauss_jordan(B)
print("\nLa solución es:",sol_b)

print("\nLiteral c:\n")
C = [[1,1/2,1/3,1/4,1/6],[1/2,1/3,1/4,1/5,1/7],[1/3,1/4,1/5,1/6,1/8],[1/4,1/5,1/6,1/7,1/9]]
sol_c = gauss_jordan(C)
print("\nLa solución es:",sol_c)

print("\nLiteral d:\n")
D = [[2,1,-1,1,-3,7],[1,0,2,-1,1,2],[0,-2,-1,1,-1,-5],[3,1,-4,0,5,6],[1,-1,-1,-1,1,-3]]
sol_d = gauss_jordan(D)
print("\nLa solución es:",sol_d)
```

```
Literal a:

[07-28 11:36:52][INFO]
[[ 1.0000000e+00  8.0000001e-01  6.6666669e-01  3.6000000e+01]
 [ 0.0000000e+00 -1.6666681e-02 -2.2222236e-02 -4.0000000e+00]
 [ 0.0000000e+00  6.0000002e-01  1.6666666e+00 -1.0000000e+01]]
[07-28 11:36:52][INFO]
[[   1.           0.          -0.3999998 -155.99985  ]
 [  -0.           1.           1.333333   239.9998   ]
 [   0.           0.           0.8666668 -153.9999   ]]
[07-28 11:36:52][INFO]
[[   1.         0.         0.      -227.07668]
 [  -0.         1.         0.       476.9226 ]
 [   0.         0.         1.      -177.69215]]

La solución es: [-227.07668  476.9226  -177.69215]

Literal b:

[07-28 11:36:52][INFO]
[[ 1.0000000e+00  3.3175967e+00  1.0794952e+00  5.3970919e+00]
 [ 0.0000000e+00  9.3382998e+00  7.2133622e+00  1.6551662e+01]
 [ 0.0000000e+00  1.5908942e+04 -1.3930958e+01  1.5895012e+04]]
[07-28 11:36:52][INFO]
[[ 1.0000000e+00  0.0000000e+00 -1.4831798e+00 -4.8318005e-01]
 [ 0.0000000e+00  1.0000000e+00  7.7244920e-01  1.7724493e+00]
 [ 0.0000000e+00  0.0000000e+00 -1.2302780e+04 -1.2302781e+04]]
[07-28 11:36:52][INFO]
[[ 1.         0.         0.         0.9999999 ]
 [ 0.         1.         0.         0.99999994]
 [-0.        -0.         1.         1.0000001 ]]

La solución es: [0.9999999  0.99999994 1.0000001 ]

Literal c:

[07-28 11:36:52][INFO]
[[ 1.         0.8        0.6666667  0.5714286  0.44444445]
```

```
 [ 0.         -0.06666666 -0.08333334 -0.0857143  -0.07936507]
 [ 0.         -0.01666668 -0.02222224 -0.02380954 -0.02314815]
 [ 0.         -0.3        -0.33333334 -0.3214286  -0.2777778 ]]
[07-28 11:36:52][INFO]
[[ 1.          0.         -0.3999998  -0.57142836 -0.6666658 ]
 [-0.          1.          1.333333    1.4285711   1.3888878 ]
 [ 0.          0.          0.00555552  0.00952377  0.01322744]
 [ 0.          0.          0.06666657  0.10714275  0.13888857]]
[07-28 11:36:52][INFO]
[[ 1.          0.          0.          0.1142875   0.28571594]
 [-0.          1.          0.         -0.857149   -1.7857188 ]
 [ 0.          0.          1.          1.7142905   2.3809555 ]
 [ 0.          0.          0.         -0.00714312 -0.01984157]]
[07-28 11:36:52][INFO]
[[ 1.          0.          0.          0.         -0.03174219]
 [-0.          1.          0.          0.          0.5951971 ]
 [ 0.          0.          1.          0.         -2.3808553 ]
 [-0.         -0.         -0.          1.          2.7777152 ]]

La solución es: [-0.03174219  0.5951971  -2.3808553   2.7777152 ]

Literal d:

[07-28 11:36:52][INFO]
[[  1.   0.   2.  -1.   1.   2.]
 [  0.   1.  -5.   3.  -5.   3.]
 [  0.  -2.  -1.   1.  -1.  -5.]
 [  0.   1. -10.   3.   2.   0.]
 [  0.  -1.  -3.   0.   0.  -5.]]
[07-28 11:36:52][INFO]
[[  1.   0.   2.  -1.   1.   2.]
 [  0.   1.  -5.   3.  -5.   3.]
 [  0.   0. -11.   7. -11.   1.]
 [  0.   0.  -5.   0.   7.  -3.]
 [  0.   0.  -8.   3.  -5.  -2.]]
[07-28 11:36:52][INFO]
[[  1.          0.          0.          -1.          3.8
    0.79999995]
 [  0.          1.          0.           3.        -12.
    6.        ]
 [ -0.         -0.          1.          -0.         -1.4
    0.6       ]
 [  0.          0.          0.           7.        -26.4
    7.6000004 ]
 [  0.          0.          0.           3.        -16.2
    2.8000002 ]]
[07-28 11:36:52][INFO]
[[ 1.         0.         0.         0.        -1.6000001  1.7333333]
 [ 0.         1.         0.         0.         4.200001   3.1999998]
 [ 0.         0.         1.         0.        -1.4        0.6      ]
 [ 0.         0.         0.         1.        -5.4        0.9333334]
 [ 0.         0.         0.         0.        11.4        1.0666666]]
[07-28 11:36:52][INFO]
[[1.         0.         0.         0.         0.         1.8830409 ]
 [0.         1.         0.         0.         0.         2.8070173 ]
 [0.         0.         1.         0.         0.         0.73099416]
 [0.         0.         0.         1.         0.         1.4385965 ]
 [0.         0.         0.         0.         1.         0.09356725]]

La solución es: [1.8830409  2.8070173  0.73099416 1.4385965  0.09356725]
```

Link del repositorio:

https://github.com/ElAlfa3007/M-todos_Num-ricos.git