2 Resolução de sistemas de equações lineares

Resumo: *O problema fundamental da álgebra linear é resolver o sistema linear $A\mathbf{x} = \mathbf{b}$, dado que existe uma solução. Muitas abordagens para resolver este problema podem ser propostas, cada uma com diferentes prós e contras. Neste curso implementamos a decomposição LU e a usamos para resolver sistemas lineares quadrados. Também apresentamos a biblioteca `SciPy` do `Python`, juntamente com suas bibliotecas para álgebra linear e trabalho com matrizes esparsas. No entanto, nossa primeira implementação utilizaremos apenas a linguagem básica do `Python` para demonstrar que os métodos matemáticos de Álgebra Linear podem ser devidamente implementados na linguagem como um algoritmo.*

Introdução

Nesta unidade, olhamos para a solução de n equações algébricas lineares com n incógnitas. É de longe o tópico mais longo e sem dúvida o mais importante do curso porque muitas partes matemáticas e computacionais serão reutilizadas! Além disso, há outra boa razão — é quase impossível realizar análises numéricas de qualquer tipo sem encontrar a solução de equações simultâneas.

Geralmente é possível reduzir os requisitos de armazenamento e o tempo de execução explorando propriedades especiais da matriz de coeficientes, como a escassez (a maioria dos elementos de uma matriz esparsa é zero). Portanto, há muitos algoritmos dedicados à solução de grandes conjuntos de equações, cada um sendo adaptado a uma forma particular da matriz de coeficientes (simétrica, em faixas, esparsa etc.).

Não temos como discutir todos os algoritmos especiais no espaço limitado disponível. O melhor que podemos fazer é apresentar os métodos básicos de solução, suplementados por alguns algoritmos úteis para matrizes de coeficientes esparsos e em faixas.

Um sistema de equações algébricas tem a seguinte forma:

$$\left\{\begin{array}{c} A_{11}x_1 + A_{12}x_2 + \ldots + A_{1n}x_n \\ A_{21}x_1 + A_{22}x_2 + \ldots + A_{2n}x_n \\ \vdots \\ A_{n1}x_1 + A_{n2}x_2 + \ldots + A_{nn}x_n \end{array}\right. = \begin{array}{c} b_1 \\ b_2 \\ \vdots \\ b_n \end{array}$$

onde os coeficientes A_{ij} e as constantes b_j são conhecidos, e x_i representa as incógnitas. Na notação matricial as equações são escritas como:

$$\begin{bmatrix} A_{11} & A_{12} & \dots & A_{1n} \\ A_{21} & A_{22} & \dots & A_{2n} \\ \vdots & & & \\ A_{n1} & A_{n2} & \dots & A_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}.$$

Um sistema de n equações lineares em n incógnitas tem uma solução única, desde que o determinante da matriz coeficiente seja não singular; isto é, $|\mathbf{A}| = 0$. As linhas e colunas de uma matriz não singular são linearmente independentes no sentido de que nenhuma linha (ou coluna) é uma combinação linear de outras linhas (ou colunas).

Se a matriz coeficiente for singular, as equações podem ter um número infinito de soluções, ou nenhuma solução, dependendo do vetor constante. Como ilustração, pegue as equações:

$$2x + y = 3 \qquad 4x + 2y = 6$$

Como a segunda equação pode ser obtida multiplicando a primeira equação por dois, qualquer combinação de x e y que satisfaça a primeira equação também é uma solução da segunda equação. O número de tais combinações é infinito. Por outro lado, as equações:

$$2x + y = 3 \qquad 4x + 2y = 0$$

não tem solução porque a segunda equação, sendo equivalente a $2x + y = 0$, contradiz a primeira. Portanto, qualquer solução que satisfaça uma equação não pode satisfazer a outra.

Uma pergunta óbvia é: o que acontece quando a matriz de coeficientes é quase singular; ou seja, se $|\mathbf{A}|$ é muito pequeno? Para determinar se o determinante da matriz de coeficientes é "pequeno", precisamos de uma referência contra a qual o determinante pode ser medido. Essa referência é chamada de norma da matriz e é denotada por $\|\mathbf{A}\|$. Podemos então dizer que o determinante é pequeno se:

$$|\mathbf{A}| \ll \|\mathbf{A}\|$$

Uma medida formal de condicionamento é o número de condição da matriz, definido como:

$$\|\mathbf{A}\| = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n} \mathbf{A}_{ij}^2} \qquad \|\mathbf{A}\| = \max_{1\leqslant i\leqslant n} \sum_{j=1}^{n} |\mathbf{A}_{ij}| \tag{2.1}$$

Uma medida formal de condicionamento é o número de condição da matriz, definido como:

$$\mathtt{cond}(\mathbf{A}) = \|\mathbf{A}\| \left\|\mathbf{A}^{-1}\right\| \tag{2.2}$$

Se esse número estiver próximo da unidade, a matriz está **bem condicionada**. O número de condição aumenta com o grau de mau condicionamento, atingindo o infinito para uma matriz singular. Observe que o número de condição não é único, mas depende da escolha da matriz norma. Infelizmente, o número de condição é caro para calcular para matrizes grandes. Na maioria dos casos, é suficiente avaliar o condicionamento comparando o determinante com as magnitudes dos elementos na matriz.

Se as equações estiverem mal condicionadas, pequenas mudanças na matriz de coeficientes resultam em grandes mudanças na solução. Como ilustração, considere as equações

$$2x + y = 3 \qquad 2x + 1.001y = 0$$

que tem a solução $x = 1501.5$, $y = -3000$. Como $|\mathbf{A}| = 2(1.001) - 2(1) = 0.002$ é muito menor que os coeficientes, as equações são **mal-condicionadas**. O efeito do mal-condicionamento pode ser verificado alterando a segunda equação para $2x + 1.002y = 0$ e resolvendo novamente as equações, obtemos o resultado $x = 751.5$, $y = -1500$. Observe que uma mudança de 0.1% no coeficiente de y produziu uma mudança de 100% na solução!

Soluções numéricas de equações **mal-condicionadas** não são confiáveis. A razão é que os erros de arredondamento inevitáveis durante o processo de solução são equivalentes a introduzir pequenas mudanças na matriz de coeficientes. Isso, por sua vez, introduz grandes erros na solução, cuja magnitude depende da gravidade do mau condicionamento. Em casos suspeitos, o determinante da matriz de coeficientes deve ser computado para que o grau de mau condicionamento possa ser estimado. Isso pode ser feito durante ou após a solução com apenas um pequeno esforço computacional.

Métodos de Soluções

Existem duas classes de métodos para resolver sistemas de equações algébricas lineares: **métodos diretos** e **iterativos**. A característica comum dos métodos diretos é que eles transformam as equações originais em equações equivalentes (equações que têm a mesma solução) que podem ser resolvidas mais facilmente. A transformação é realizada aplicando as três operações listadas a seguir:

- Troca de duas equações (muda o sinal de $|\mathbf{A}|$).
- Multiplicação de uma equação por uma constante diferente de zero (multiplica $|mathbf A|$ pela mesma constante).
- Multiplicação de uma equação por uma constante diferente de zero e então subtração de outra equação (deixa $|\mathbf{A}|$ inalterado).

Essas chamadas operações elementares não alteram a solução, mas podem afetar o determinante da matriz coeficiente conforme indicado entre parênteses.

Métodos iterativos ou indiretos começam com uma estimativa da solução x e, então, refinam repetidamente a solução até que um certo critério de convergência seja alcançado. Métodos iterativos são geralmente menos eficientes do que suas contrapartes diretas devido ao grande número de iterações necessárias. Mas eles têm vantagens computacionais significativas se a matriz de coeficientes for muito grande e escassamente preenchida (a maioria dos coeficientes é zero).

Médodo	Forma Inicial	Forma Final
Eliminação Gaussiana	$\mathbf{Ax} = \mathbf{b}$	$\mathbf{Ux} = \mathbf{b}$
Decomposição **LU**	$\mathbf{Ax} = \mathbf{b}$	$\mathbf{LUx} = \mathbf{b}$
Eliminação Gauss-Jordan	$\mathbf{Ax} = \mathbf{b}$	$\mathbf{Ix} = \mathbf{b}$

Na tabela acima, **U** representa uma matriz triangular superior, **L** é uma matriz triangular inferior e **I** denota a matriz identidade. Uma matriz quadrada é chamada triangular se ela contiver apenas zero elementos em um lado da diagonal principal. Assim, uma **matriz triangular superior** 3×3 tem a forma:

$$\mathbf{U} = \begin{bmatrix} U_{11} & U_{12} & U_{13} \\ 0 & U_{22} & U_{23} \\ 0 & 0 & U_{33} \end{bmatrix}$$

e uma **matriz triangular inferior** 3×3 se parece como:

$$\mathbf{L} = \begin{bmatrix} L_{11} & 0 & 0 \\ L_{21} & L_{22} & 0 \\ L_{31} & L_{32} & L_{33} \end{bmatrix}$$

Matrizes triangulares desempenham um papel importante na álgebra linear, pois simplificam muitos cálculos. Por exemplo, considere as equações $\mathbf{Lx} = \mathbf{c}$, ou:

$$\begin{aligned} L_{11}x_1 &= c_1 \\ L_{21}x_1 + L_{22}x_2 &= c_2 \\ L_{31}x_1 + L_{32}x_2 + L_{33}x_3 &= c_3 \\ \vdots \quad &= \vdots \end{aligned}$$

Se resolvermos as equações em termos das variáveis x_i, começando com a primeira equação, os cálculos são muito fáceis, já que cada equação contém apenas uma incógnita por vez. A solução assim procederia da seguinte forma:

$$\begin{aligned} x_1 &= c_1/L_{11} \\ x_2 &= (c_2 - L_{21}x_1)/L_{22} \\ x_3 &= (c_3 - L_{31}x_1 + L_{32}x_2)/L_{33} \\ \vdots &= \quad \vdots \end{aligned}$$

Este procedimento é conhecido como substituição direta. De forma semelhante, $\mathbf{Ux} = \mathbf{c}$, encontrado na eliminação de Gauss, pode ser facilmente resolvido por substituição reversa, que começa com a última equação e prossegue para trás através das equações.

As equações $\mathbf{LUx} = \mathbf{b}$, que são associadas à decomposição $\mathbf{LU}$, também podem ser resolvidas rapidamente se as substituirmos por dois conjuntos de equações equivalentes: $\mathbf{Ly} = \mathbf{b}$ e $\mathbf{Ux} = \mathbf{y}$. Agora $\mathbf{Ly} = \mathbf{b}$ pode ser resolvido para $\mathbf{y}$ por substituição direta, seguido pela solução de $\mathbf{Ux} = \mathbf{y}$ por meio de substituição reversa.

As equações $\mathbf{Ix} = \mathbf{c}$, que são produzidas pela eliminação de Gauss-Jordan, são equivalentes a $\mathbf{x} = \mathbf{c}$ (lembre-se da identidade $\mathbf{Ix} = \mathbf{x}$), de modo que $\mathbf{c}$ já é a solução.

Problema 1. Determine se a seguinte matriz é singular:

$$A = \begin{bmatrix} 2.1 & -0.6 & 1.1 \\ 3.2 & 4.7 & -0.8 \\ 3.1 & -6.5 & 4.1 \end{bmatrix}$$

Solução: O determinante de uma matriz pode ser encontrado através da relação:

$$|\mathbf{A}| = \sum_{k=1}^{n} (-1)^{k+1} A_{1k} M_{1k}$$

onde M_{1k} é o determinante da matriz $(n-1) \times (n-1)$ obtida deletando-se a i-ésima linha e a k-ésima coluna de A. O termo $(-1)^{k+1} M_{1k}$ é chamado de **cofator** de A. Assim:

$$\begin{aligned} |A| &= 2.1 \begin{vmatrix} 4.7 & -0.8 \\ -6.5 & 4.1 \end{vmatrix} - (-0.6) \begin{vmatrix} 3.2 & -0.8 \\ 3.1 & 4.1 \end{vmatrix} + 1.1 \begin{vmatrix} 3.2 & 4.7 \\ 3.1 & -6.5 \end{vmatrix} \\ &= 2.1(14.07) + 0.6(15.60) + 1.1(-35.37) \end{aligned}$$

Como o determinante é zero, a matriz é singular. Pode-se verificar que a singularidade é devido à seguinte dependência de linha: $L_3 \longleftarrow 3L_1 - L_2$.

Problema 2. Resolva a equação $\mathbf{Ax} = \mathbf{b}$, onde

$$A = \begin{bmatrix} 8 & -6 & 2 \\ -4 & 11 & -7 \\ 4 & -7 & 6 \end{bmatrix} \quad \text{e} \quad B = \begin{bmatrix} 28 \\ -40 \\ 33 \end{bmatrix}$$

sabendo que a decomposição **LU** da matriz coeficiente é (você deve verificar):

$$A = \mathbf{LU} = \begin{bmatrix} 2 & 0 & 0 \\ -1 & 2 & 0 \\ 1 & -1 & 1 \end{bmatrix} \begin{bmatrix} 4 & -3 & 1 \\ 0 & 4 & -3 \\ 0 & 0 & 2 \end{bmatrix}$$

Solução: Primeiro vamos resolver a equação $\mathbf{Ly} = \mathbf{b}$ pelo método da substituição:

$$\begin{aligned} 2y_1 &= 28 & y_1 &= 28/2 = 14 \\ -y_1 + 2y_2 &= -40 & y_2 &= (-40 + y_1)/2 = (-40 + 14)/2 = -13 \\ y_1 - y_2 + y_3 &= 33 & y_3 &= 33 - y_1 + y_2 = 33 - 14 - 13 = 6 \end{aligned}$$

Agora podemos obter $\mathbf{x}$ a partir da expressão $\mathbf{Ux} = \mathbf{y}$, através dos cálculos de retorno:

$$\begin{aligned} 2x_3 &= y_3 & x_3 &= y_2/2 = 6/2 = 3 \\ 4x_2 - 3x_3 &= y_2 & x_2 &= (y_2 + 3x_3)/4 = (-13 + 3(3))/4 = -1 \\ 4x_1 - 3x_2 + x_3 &= y_1 & x_1 &= (y_1 + 3x_2 - x_3)/4 = [14 + 3(-1) - 3]/4 = 2 \end{aligned}$$

E assim, a solução é $\mathbf{x} = [2 \quad -1 \quad 3]^T$.

Método de Eliminação Gauss-Jordan

Considere inicialmente o seguinte exemplo:

$$\begin{cases} x + y + 2z &= 9 \\ 2x + 4y - 3z &= 1 \\ 3x + 6y - 5z &= 0 \end{cases} \tag{2.3}$$

Para resolver o sistema podemos proceder de maneira muito simples, como já visto. A fim de eliminar a variável x nessa equação, podemos multiplicar a primeira por -2 e adicionarmos à segunda. Depois, multiplicamos novamente a primeira por -3 e adicionamos à terceira equação. Essas operações constituiem o que chamamos de combinação linear e são pertinentes a solução do problema. Implementando essas operações iremos obter:

$$\begin{cases} x + y + 2z &= 9 \\ 2y - 7z &= -17 \\ 3y - 11z &= -27 \end{cases}$$

Observe que a segunda e a terceira equações formam um sistema de duas equações nas incógnitas y e z. Novamente, podemos eliminar a variável y multiplicando a segunda equação por 3 e a terceira por 2 e efetuando a subtração da segunda equação pela terceira, obtendo:

$$\left\{\begin{array}{rcr} x+y+2z & = & 9 \\ 6y-21z & = & -51 \\ z & = & 3 \end{array}\right.$$

encontrando assim o valor de z, a saber $z = 3$. Substituindo nas duas primeiras equações, temos: $6y = 63 - 51 = 12$, ou $y = 2$, e finalmente, $x = 9 - 2 - 6 = 1$.

Portanto, temos que a solução do sistema dado é $x = 1$, $y = 2$ e $z = 3$, que neste caso é a única solução! $\triangleleft$

Como outro exemplo, vamos resolver o seguinte sistema:

$$\left\{\begin{array}{rcr} x+y+z & = & 1 \\ x+y-z & = & 0 \\ 2x+2y & = & 1 \end{array}\right. \tag{2.4}$$

Procedendo como anteriormente, multiplicamos a segunda equação por -2 e adicionamos à terceira para eliminarmos x na terceira equação. Em seguida, subtraímos a segunda equação da primeira para eliminarmos x na segunda equação, obtendo:

$$\left\{\begin{array}{rcr} x+y+z & = & 1 \\ 2z & = & 1 \\ 2z & = & 1 \end{array}\right.$$

Neste caso, ao eliminar a variável x, eliminamos também a variável y e obtemos a segunda equação identica à terceira. O sistema é, assim, equivalente ao seguinte:

$$\left\{\begin{array}{rcr} x+y+z & = & 1 \\ 2z & = & 1 \end{array}\right.$$

Facilmente, encontramos que $z = 1/2$. Porém, agora temos algo distinto do primeiro exemplo. Observe que para $z = 1/2$, $x = -t + 1/2$ é sempre solução para qualquer valor de $y = t$. Vemos, portanto, que o sistema possui infinitas soluções, ou seja: $x = -t + 1/2$, $y = t$ e $z = 1/2$. $\triangleleft$

Vejam que da forma que obtivemos as soluções nos exemplos anteriores, não temos exatamente um modelo de algoritmo para executar a busca das soluções de sistemas lineares. Por isso, precisamos aplicar em tais problemas o método da eliminação gaussiana se quisermos uma primeira modelagem computacional, que basicamente funciona a partir da aplicação três operações lineares: multiplicar uma linha por um valor escalar, trocar duas linhas ou adicionar um múltiplo escalar de uma linha a outra. Vamos retornar ao sistema linear apresentada na Equação (2.3).

$$\left\{\begin{array}{rcr} x+y+2z & = & 9 \\ 2x+4y-3z & = & 1 \\ 3x+6y-5z & = & 0 \end{array}\right.$$

A abordagem para resolver o sistema linear acima pelo método de Gauss exige escrever uma matriz associada ao sistema da seguinte forma: cada linha da matriz corresponderá a uma das equações.

Teremos, portanto, uma matriz com 3 linhas. Cada coeficiente da primeira equação corresponderá ordenadamente a uma entrada da primeira linha. O termo independente será a 4^a entrada desta primeira linha. Ela terá 4 entradas. Para a segunda e terceira linhas procedemos da mesma forma. Teremos, portanto, uma matriz 3×4 associada ao sistema, chamada de **matriz aumentada**, ou seja:

$$[A \mid \mathbf{b}] = \left[\begin{array}{ccr|c} 1 & 1 & 2 & 9 \\ 2 & 4 & -3 & 1 \\ 3 & 6 & -5 & 0 \end{array}\right] \tag{2.5}$$

A matriz acima se chama aumenta para se distinguir da matriz:

$$A = \begin{bmatrix} 1 & 1 & 2 \\ 2 & 4 & -3 \\ 3 & 6 & -5 \end{bmatrix}$$

denominada **matriz de coeficientes**.

A abordagem padrão para resolver o sistema linear $A\mathbf{x} = \mathbf{b}$ no papel é reduzir a matriz aumentada $[A \mid \mathbf{b}]$ para a forma escalonada por linha (*Row-Echelon Form* - REF) via *eliminação gaussiana*, então usando substituição reversa.

Nota

Uma matriz $m \times n$ se diz na forma escalonada reduzida por linhas quando:

1. Se uma linha não consistir só de zeros, então o primeiro número não nulo da linha é 1. Chamamos este número 1 de **líder** ou **pivô**.
2. Se existirem linhas constituídas somente de zeros, elas estão agrupadas nas linhas inferiores da matriz.
3. Em quaisquer duas linhas sucessivas que não consistem só de zeros, o líder da linha inferior ocorre mais à direita que o líder da linha superior.
4. Cada coluna que contém um líder tem zeros nas demais entradas.

Se apenas a condição 4 não é satisfeita, a matriz se diz na forma escalonada.

Vemos que a condição 1 já está satisfeita na nossa matriz aumentada. Portanto, precisamos realizar operações lineares sobre as linhas L_2 e L_3 para atendermos ao critério 2. As transformaçõs lineares estão notadas abaixo:

$$[A \mid \mathbf{b}] = \left[\begin{array}{ccr|c} 1 & 1 & 2 & 9 \\ 2 & 4 & -3 & 1 \\ 3 & 6 & -5 & 0 \end{array}\right] \begin{array}{lcl} \\ L_2 & \longleftarrow & L_2 - 2L_1 \\ L_3 & \longleftarrow & L_3 - 3L_1 \end{array}$$

Após a aplicação das transformações lineares, verificamos que ainda não temos o pivô definido na segunda linha. Novamente, por meio de uma transformação linear, o obtemos:

$$[A \mid \mathbf{b}] = \left[\begin{array}{ccr|r} 1 & 1 & 2 & 9 \\ 0 & 2 & -7 & -17 \\ 0 & 3 & -11 & -27 \end{array}\right] \quad L_2 \longleftarrow L_2/2$$

A linha L_2, com o pivô definido, passa a ser a nossa referência para a linha L_3. Mediante nova transformação linear, atendemos ao critério 2:

$$[A \mid \mathbf{b}] = \left[\begin{array}{ccr|c} 1 & 1 & 2 & 9 \\ 0 & 1 & -7/2 & -17/2 \\ 0 & 3 & -11 & -27 \end{array}\right] \quad L_3 \longleftarrow L_3 - 3L_2$$

Na linha L_3 precisamos definir o seu pivô, o que facilmente pode ser obtido mediante a transformação linear notada a seguir:

$$[A \mid \mathbf{b}] = \left[\begin{array}{ccr|c} 1 & 1 & 2 & 9 \\ 0 & 1 & -7/2 & -17/2 \\ 0 & 0 & -1/2 & -3/2 \end{array}\right] \quad L_3 \longleftarrow (-2)L_3$$

Logo, obtemos a nossa matriz aumentada na forma escalonada por linha.

$$[A \mid \mathbf{b}] = \left[\begin{array}{ccr|c} 1 & 1 & 2 & 9 \\ 0 & 1 & -7/2 & -17/2 \\ 0 & 0 & 1 & 3 \end{array}\right] \tag{2.6}$$

Uma vez que z está determinado, por substituição, encontramos os valores de x e y.

Para atendermos ao critério 4, precisamos garantir que todas as entradas que contém um lider tenham valor igual a zero. Logo, vamos aplicar as seguintes transformações lineares na Equação (2.6):

$$[A \mid \mathbf{b}] = \left[\begin{array}{ccr|c} 1 & 1 & 2 & 9 \\ 0 & 1 & -7/2 & -17/2 \\ 0 & 0 & 1 & 3 \end{array}\right] \quad \begin{array}{l} L1 \longleftarrow L1 - 2L_3 \\ L_2 \longleftarrow L_2 + 7/2L_3 \\ \end{array}$$

$$[A \mid \mathbf{b}] = \left[\begin{array}{ccc|c} 1 & 1 & 0 & 3 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 3 \end{array}\right] \quad \begin{array}{l} L1 \longleftarrow L1 - 2L_3 \\ L_2 \longleftarrow L_2 + 7/2L_3 \\ \end{array}$$

$$[A \mid \mathbf{b}] = \left[\begin{array}{ccc|c} 1 & 1 & 0 & 3 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 3 \end{array}\right] \quad \begin{array}{l} L1 \longleftarrow L1 - L_2 \\ \\ \end{array}$$

$$[A \mid \mathbf{b}] = \left[\begin{array}{ccc|c} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 3 \end{array}\right]$$

Tal que:

$$I = \left[\begin{array}{ccc} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{array}\right]$$

é a **matriz identidade**.

```python
# gauss_01.py

def print_matrix(M, decimals=3):
    """
    Print a matrix one row at a time
        :param M: The matrix to be printed
    """
    for row in M:
        print([round(x, decimals)+0 for x in row])


def zeros_matrix(rows, cols):
    """
    Cria uma matriz de zeros
        :param rows: numero de linhas
        :param cols: numero de colunas

        :return: matriz quadrada de zeros
    """
    M = []
    while len(M) < rows:
        M.append([])
        while len(M[-1]) < cols:
            M[-1].append(0.0)

    return M


def coef_matrix(augMat):
    """
    Recupera a matriz Coeficientes para o calculo do Determinante
        :param M: Matriz aumentada

        :return: Matriz Coeficiente
    """
    # Recuperando as dimensoes da matriz coeficiente
    rows = len(augMat)
    cols = len(augMat[0])

    # Section 2: Create a new matrix of zeros
    MC = zeros_matrix(rows, cols - 1)

    # Section 3: Copy values of M into the copy
    for i in range(rows):
        for j in range(cols - 1):
            MC[i][j] = augMat[i][j]

    return MC
```

```
def determinant(AM):
    """
    Calcula o determinante a partir da matriz triangular superior
        O produto da diagonal principal eh o valor do determinante

        :param AM: matriz de coeficientes

        :return: determinante da matriz
    """
    n = len(AM)

    # Reducao a forma triangular superior
    for fd in range(n):  # fd: foco diagonal values
        if AM[fd][fd] == 0:
            for j in range(fd + 1, n):
                if AM[j][fd] != 0:
                    AM[fd], AM[j] = AM[j], AM[fd]
                    break
            else:
                # If no non-zero pivot is found, the matrix might be singular
                #   Se um pivo nao for encontrado, a matriz eh singular
                raise ValueError("Matriz singular!.")

        for i in range(fd+1, n):  # skip row with fd in it.
            crScaler = AM[i][fd] / AM[fd][fd]  # cr stands for "current row".
            for j in range(n):  # cr - crScaler * fdRow, one element at a time.
                AM[i][j] = AM[i][j] - crScaler * AM[fd][j]

    # Uma vez que temos a forma triangular, calculamos o produto da diagonal ↩
        principal
    product = 1.0
    for i in range(n):
        product *= AM[i][i]
    return product

def verifica_non_singularidade(A):
    """
    Verifica se a matriz eh NAO SINGULAR
        :param A: Matriz a ser avaliada

        :return: boolean True ou raise ArithmeticError
    """
    det = determinant(A)
    if det != 0:
        return True
    else:
        raise ArithmeticError("Matriz Singular!")
```

```
def GaussJordanMethod(augMat):
    n = len(augMat)             #   Numero de linhas
    m = len(augMat[0])          #   Numero de colunas

    for i in range(n):
        #   Check if the pivot element is zero and swap rows if necessary
        #   Verifica se o pivo eh zero e muda a linha se necessario
        if augMat[i][i] == 0:
            for j in range(i + 1, n):
                if augMat[j][i] != 0:
                    augMat[i], augMat[j] = augMat[j], augMat[i]
                    break
            else:
                # If no non-zero pivot is found, the matrix might be singular
                #   Se um pivo nao for encontrado, a matriz eh singular
                raise ValueError("Matriz singular!.")

        #   Normalizando cada linha
        #   L_i <-- L_i/a_ii
        if augMat[i][i] != 1:
            divisor = augMat[i][i]
            for k in range(m):
                augMat[i][k] /= divisor
        else:
            pass

        #   Zera as entradas referentes aos pivos
        #   L_j <-- L_j - a_ji * L_i
        for j in range(n):
            if i != j:
                coef = augMat[j][i]
                for k in range(m):
                    augMat[j][k] -= coef * augMat[i][k]
            else:
                pass
    print(augMat)

matrix = [[3.0, 2.0, -4.0, 3.0], [2.0, 3.0, 3.0, 15.0], [5.0, -3, 1.0, 14.0]]
#matrix = [[0, 2, 0, 1, 0], [2, 2, 3, 2, -2], [4, -3, 0, 1, -7], [6, 1, -6, -5,
    6]]

mc = coef_matrix(matrix)
print_matrix(mc)
det = determinant(mc)
print(det)
result = verifica_non_singularidade(mc)
print(result)
GaussJordanMethod(matrix)
```

Exercícios

1. Encontre as soluções dos seguintes sistemas (se existirem), de forma analítica e computacional, utilizando o método de Gauss-Jordan. Caso não existam soluções, demonstre a justificativa!

a)

$$\begin{cases} 2x + 3y & = & 5 \\ 4x - y & = & 7 \end{cases}$$

b)

$$\begin{cases} 2x + 3y + z & = & 0 \\ x - 2y - z & = & 1 \\ x + 4y + z & = & 2 \end{cases}$$

c)

$$\begin{cases} 2x + 2y - 3z & = & 4 \\ x + 3y + z & = & 11 \\ 2x + 5y - 4z & = & 13 \end{cases}$$

d)

$$\begin{cases} x + y + z & = & 1 \\ x - y + z & = & 2 \\ 2x + 2y + 2z & = & 5 \end{cases}$$

e)

$$\begin{cases} x + y + z & = & 1 \\ x - y + z & = & 2 \\ 2x + 2z & = & 3 \end{cases}$$

f)

$$\begin{cases} 2x + y - 2z & = & 10 \\ 3x + 2y + 2z & = & 1 \\ 5x + 4y + 3z & = & 4 \end{cases}$$