

Métodos Computacionais em Álgebra Linear via Python

Resolução de Sistemas Lineares com a biblioteca NumPy

João Alberto, Gabriel Lucas

Departamento de Matemática
Universidade Federal de Santa Maria

November 1, 2025

O objetivo das aulas aqui expostas é trazer maior elucidação acerca das ferramentas computacionais e possibilidades existentes para problemas frequentemente tratados de maneira manual e morosa nos cursos de Álgebra Linear. Será tomada uma abordagem prática e convenientemente concisa, de sorte que os alunos pensem e construam suas próprias implementações na linguagem Python.

Esse material será disponibilizado aos alunos para acesso.

Matrizes em Python

Seja A uma matriz 2×2 . Como podemos construí-la num *script* Python?

```
import numpy as np
# importação da biblioteca NumPy

A = np.array([
    [1, 2],
    [4, 5]])
```

Note que uma matriz é descrita por objetos do tipo **array**, presentes em qualquer linguagem de programação. A mesma declaração poderia ser feita usando a sintaxe padrão da linguagem Python, sem evocar o módulo NumPy. A utilização dessa biblioteca será justificada posteriormente.

Localização de Elementos numa Matriz por Índices

Para referenciar entradas de uma matriz A , lembremos que faz-se uso de subíndices $i, j \in \mathbb{N}$: a_{ij} é a entrada na i -ésima linha e j -ésima coluna de A .

Em Python, se A é uma matriz (array) construída como anteriormente, temos que a_{ij} é equivalente a

```
A = np.array([
    [1, 2],
    [4, 5]])

entry = A[0][1]
```

Atenção!

Os índices em Python começam em 0. Sendo assim, $A[0][1]$ refere-se à entrada na primeira linha (0) e segunda coluna (1).

Atividade Prática

Implemente um algoritmo que, dadas duas matrizes 2×2 , A e B , calcule e mostre a soma $A + B$.

Motivação para Bibliotecas como NumPy

Vejam a solução sem o uso da biblioteca NumPy...

```
A = [[1, 2], [1, 2]]
B = [[10, 10], [20, 20]]
C = [[0, 0], [0, 0]]

for i in range(len(A)):
    for j in range(len(A[i])):
        C[i][j] = A[i][j]+B[i][j]

print(C)

# output:
# [[11, 12],
#  [21, 22]]
```

Motivação para Bibliotecas como NumPy

Agora, com NumPy:

```
import numpy as np

A = np.array([[1, 2], [1, 2]])
B = np.array([[10, 10], [20, 20]])

print(A+B)

# output:
# [[11, 12],
#  [21, 22]]
```

Motivação para Bibliotecas como NumPy

De fato, nota-se que este é um exemplo em que a diferença entre o número de linhas de código é irrisória (ignorando-se a complexidade). No entanto, imagine se fosse proposta a implementação de um algoritmo que resolve sistemas lineares de dimensões arbitrárias; a complexidade e utilização de recursos computacionais decerto seria considerável. É aí que bibliotecas como NumPy tomam um valioso lugar.

Finalmente, vejamos como podemos resolver sistemas lineares usando as ferramentas que estudamos até aqui.

Sistemas Lineares

Das noções iniciais trazidas num curso usual de Álgebra Linear, sabemos que qualquer sistema linear

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = b_n \end{cases}$$

pode ser escrito na forma matricial:

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

Bullet Points

- Lorem ipsum dolor sit amet, consectetur adipiscing elit
- Aliquam blandit faucibus nisi, sit amet dapibus enim tempus eu
- Nulla commodo, erat quis gravida posuere, elit lacus lobortis est, quis porttitor odio mauris at libero
- Nam cursus est eget velit posuere pellentesque
- Vestibulum faucibus velit a augue condimentum quis convallis nulla gravida

Blocks of Highlighted Text

In this slide, some important text will be **highlighted** because it's important. Please, don't abuse it.

Block

Sample text

Alertblock

Sample text in red box

Examples

Sample text in green box. The title of the block is "Examples".

Heading

1. Statement
2. Explanation
3. Example

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer lectus nisl, ultricies in feugiat rutrum, porttitor sit amet augue. Aliquam ut tortor mauris. Sed volutpat ante purus, quis accumsan dolor.

Table

Treatments	Response 1	Response 2
Treatment 1	0.0003262	0.562
Treatment 2	0.0015681	0.910
Treatment 3	0.0009271	0.296

Table: Table caption

Theorem

Theorem (Mass–energy equivalence)

$$E = mc^2$$

Figure

Uncomment the code on this slide to include your own image from the same directory as the template .TeX file.

An example of the `\cite` command to cite within the presentation:

This statement requires citation [Smith, 2012].

References



Smith, J. (2012).

Title of the publication.

Journal Name, 12(3):45–678.

The End