

# Introdução à Computação em Física

AUTOVALORES E  
AUTOVETORES  
PROF. WALBER

Refs.:

Compact numerical methods for computers , Linear Algebra and function minimisation, J. C. Nash  
Applied numerical linear algebra, J. W. Demmel.

# Autovalores e autovetores

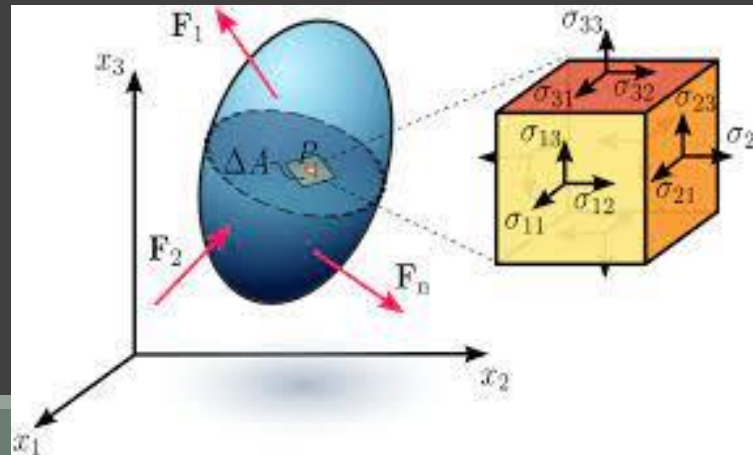
---

Em diversas ocasiões, temos problemas que são descritos por uma equação de autovalores e autovetores:

$$Ax = \lambda x$$

A matriz quadrada  
 $x$  é um autovetor  
 $\lambda$  autovalor;

Estudo de vibrações, stress em sistemas contínuos, circuitos elétricos, etc.



# Autovalores e autovetores

---

Em diversas ocasiões, temos problemas que são descritos por uma equação de autovalores e autovetores:

$$Ax = \lambda x$$

A matriz quadrada  
X é um autovetor  
 $\lambda$  autovalor;

Def.: Seja uma matriz A (n x n). O escalar  $\lambda$  é dito autovalor de A se existe um vetor x não nulo tal que  $Ax = \lambda x$ . O vetor x é chamado de autovetor associado a  $\lambda$ .

Iremos ver brevemente alguns métodos que nos permitem obter os diferentes valores de  $\lambda$  para uma dada matriz A, obtendo em alguns casos os autovetores x.

## Polinômio característico

---

Nesse contexto, podemos obter os autovalores  $\lambda$  a partir dos zeros do chamado polinômio característico

$$p(\lambda) = \det(A - \lambda I)$$

Assim sendo, se  $p$  for o polinômio característico de  $A$ , os zeros de  $p$  são os autovalores de  $A$ .

Uma vez que os autovalores  $\lambda$  tenham sido determinados, pode-se obter os autovetores pela solução de

$$(A - \lambda I)x = 0$$

## Matriz diagonal

---

Vale lembrar que:

1. Se uma matriz quadrada  $A$  de ordem  $n$ , tem  $n$  autovetores linearmente independentes, então escolhendo estes autovetores como base obtemos uma matriz diagonal similar a  $A$ ;
2. Toda matriz quadrada com autovalores distintos pode ser reduzida a uma matriz diagonal, em que os elementos da diagonal são seus autovalores

$$A = \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{pmatrix}$$

Onde  $\lambda_1$ ,  $\lambda_2$  e  $\lambda_3$  são os autovalores de  $A$ .

# Métodos numéricos

---

Exceto nos casos de matrizes menores, o cálculo do determinante envolvendo o polinômio característico pode se tornar problemático.

Nesse sentido, iremos estudar métodos que não fazem uso do polinômio característico. Tais métodos podem ser classificados em:

1. Métodos que determinam alguns autovalores: veremos os chamados método das potências;
2. Métodos que determinam todos autovalores: Nesse caso veremos o método QR;

Métodos para se determinar alguns autovalores

---

Seja  $A$  uma matriz real de ordem  $n$  e sejam  $\lambda_1, \lambda_2, \dots, \lambda_n$  seus autovalores e  $u_1, u_2, \dots, u_n$  seus correspondentes autovetores. Suponha que os autovetores são linearmente independentes e que:

$$|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$$

Seja a sequência  $y_k$  definida por:

$$y_{k+1} = Ay_k (k = 0, 1, 2, \dots)$$

onde  $y_0$  é um vetor arbitrário que permite a expansão:

$$y_0 = \sum_{j=1}^n c_j u_j,$$

com  $c_j$  escalares quaisquer e  $c_1 \neq 0$ , então:

$$\lambda_1 = \lim_{k \rightarrow \infty} \frac{(y_{k+1})_r}{(y_k)_r},$$

onde  $r$  indica a  $r$ -ésima componente. Além disso, quando  $k \rightarrow \infty$ ,  $y_k$  tende ao autovetor correspondente a  $\lambda_1$ .

## Método da potência

O método da potência consiste em um método iterativo para se obter o autovalor de uma dada matriz com maior módulo.

Este método se baseia no teorema ao lado.



# Método da potência

---

Seja

$$Au_i = \lambda_i u_i$$

$$y_0 = c_1 u_1 + c_2 u_2 + \dots + c_n u_n$$

$$|\lambda_1| \geq |\lambda_2| \geq \dots > |\lambda_n|$$

De nosso teorema, obtemos que:

$$y_1 = Ay_0 = c_1 Au_1 + c_2 Au_2 + \dots + c_n Au_n$$

$$y_1 = \lambda_1 \left[ c_1 u_1 + c_2 \frac{\lambda_2}{\lambda_1} u_2 + \dots + c_n \frac{\lambda_n}{\lambda_1} u_n \right]$$



k iterações

$$y_k = \lambda_1^k \left[ c_1 u_1 + c_2 \left( \frac{\lambda_2}{\lambda_1} \right)^k u_2 + \dots + c_n \left( \frac{\lambda_n}{\lambda_1} \right)^k u_n \right]$$

Razões de autovalores irão se anular no limite de k grande


$$\lambda_1 = \lim_{k \rightarrow \infty} \frac{(y_{k+1})_r}{(y_k)_r}$$

$$r = 1, 2, \dots, n$$

# Método da potência

---

Assim sendo, podemos determinar  $\lambda_1$  através dos seguintes passos:


$$\begin{aligned}z_1 &= Ay_0 \\y_1 &= \frac{z_1}{\alpha_1} = \frac{Ay_0}{\alpha_1} \\z_2 &= Ay_1 = \frac{A^2y_0}{\alpha_1} \\y_2 &= \frac{z_2}{\alpha_2} = \frac{A^2y_0}{\alpha_1\alpha_2} \\z_3 &= Ay_2 = \frac{A^3y_0}{\alpha_1\alpha_2} \\&\dots\end{aligned}$$

$$\begin{aligned}y_k &= \frac{z_k}{\alpha_k} = \frac{A^k y_0}{\alpha_1 \alpha_2 \dots \alpha_k} \\z_{k+1} &= Ay_k = \frac{A^{k+1} y_0}{\alpha_1 \alpha_2 \dots \alpha_k}\end{aligned}$$

Onde  
 $y_0$  é um vetor arbitrário não nulo.

$$\alpha_{k+1} = \max_{1 \leq r \leq n} |(z_{k+1})_r|$$

No final das iterações, o valor de  $\lambda_1$  converge

$$\lambda_1 = \lim_{k \rightarrow \infty} \frac{(z_{k+1})_r}{(y_k)_r}$$

## Método da potência

---

Vale destacar que se algum vetor resultar no vetor nulo então o método falha (associado a hipóteses);

Método tem a desvantagem de que não se sabe, no início, se a matriz  $A$  tem ou não um único autovalor dominante;

Também não se sabe como  $y_0$  poderia ser escolhido de modo a garantir que a representação contenha uma contribuição não nula do autovetor associado ao autovalor dominante, caso ele exista.

## Método da potência inversa

---

Podemos fazer uso da matriz inversa de  $A$  para obter o menor autovalor (menor módulo).

Método similar ao visto anteriormente, embora queremos encontrar  $\lambda_n$ ,

$$|\lambda_1| \geq |\lambda_2| \geq \dots > |\lambda_n|$$

Temos que essencialmente substituir  $A$  por  $A^{-1}$  nos passos feitos anteriormente. Note que no final iremos obter  $1/\lambda_n$ .

## Atividade prática:

---

1. Implemente o método da potência e potência inversa para calcular o maior e menor autovalor da matriz

$$A = \begin{pmatrix} 2 & 1 & 0 \\ 2 & 5 & 3 \\ 0 & 1 & 6 \end{pmatrix} \quad \text{Considere } y_0 = [1, 1, 1]$$

Obs: O maior autovalor é 7.4437 e o menor é 1.3375

Métodos para se determinar todos autovalores

---

## Método QR (método de Francis)

---

A ideia principal do método QR é determinar os autovalores de uma matriz  $A$  sem determinar seu polinômio característico;

Método consiste em obter uma sequência de matrizes  $A_1, A_2, \dots$ , da seguinte maneira:

$$\begin{aligned} A_1 &= A = Q_1 R_1 \\ A_2 &= R_1 Q_1 = Q_2 R_2 \\ &\dots \\ A_k &= R_{k-1} Q_{k-1} = Q_k R_k \\ &\dots \end{aligned}$$

Onde  $Q_i$  é uma matriz ortogonal e  $R_i$  uma matriz triangular superior

$$Q^t Q = Q Q^t = I$$

$$Q^t = Q^{-1}$$

# Método QR

---

Em suma,

$$\begin{aligned} A_1 &= A = Q_1 R_1 \\ A_2 &= R_1 Q_1 = Q_2 R_2 \\ &\dots \\ A_k &= R_{k-1} Q_{k-1} = Q_k R_k \\ &\dots \end{aligned}$$

Onde  $Q_i$  é uma matriz ortogonal e  $R_i$  uma matriz triangular superior

Uma matriz ortogonal  $Q$  é caracterizada por:

$$Q^t Q = Q Q^t = I$$

$$Q^t = Q^{-1}$$

$Q^t$  é a transposta de  $Q$   
 $I$  é a matriz identidade de ordem  $n$



# Método QR


---

A sequência  $A_k$  converge para uma matriz triangular superior;

Os elementos diagonais da matriz  $A_k$  são os autovalores procurados;

O processo termina quando o elemento de maior valor absoluto da matriz  $A_k$  (abaixo da diagonal principal) for menor que uma dada precisão pré-fixada.

Diversos métodos para se obter as matrizes Q e R (não iremos entrar em detalhes).

$$\begin{aligned} A_1 &= A = Q_1 R_1 \\ A_2 &= R_1 Q_1 = Q_2 R_2 \\ &\dots \\ A_k &= R_{k-1} Q_{k-1} = Q_k R_k \\ &\dots \end{aligned}$$


## Atividade prática

---

1. Implemente o método QR para calcular todos autovalores da matriz abaixo

$$A = \begin{pmatrix} 2 & 1 & 0 \\ 2 & 5 & 3 \\ 0 & 1 & 6 \end{pmatrix}$$

Para obtenção das matrizes Q e R, utilize a biblioteca Numpy

```
from numpy.linalg import qr
```

```
Q, R = qr(A)
```

Mais informações em:

<https://numpy.org/doc/stable/reference/generated/numpy.linalg.qr.html>

## Obtenção geral via Numpy

---

De maneira geral podemos obter os autovalores e autovetores de uma matriz via Numpy, utilizando

```
import numpy as np
from numpy.linalg import eig
```

```
A = np.array([ [2.0, 1.0, 0.0], [2.0, 5.0, 3.0], [0.0, 1.0, 6.0]])
```

```
autova, autovet = eig(A)
```

```
print ' Autovalores = ', autova
print ' Autovetores = ', autovet
```

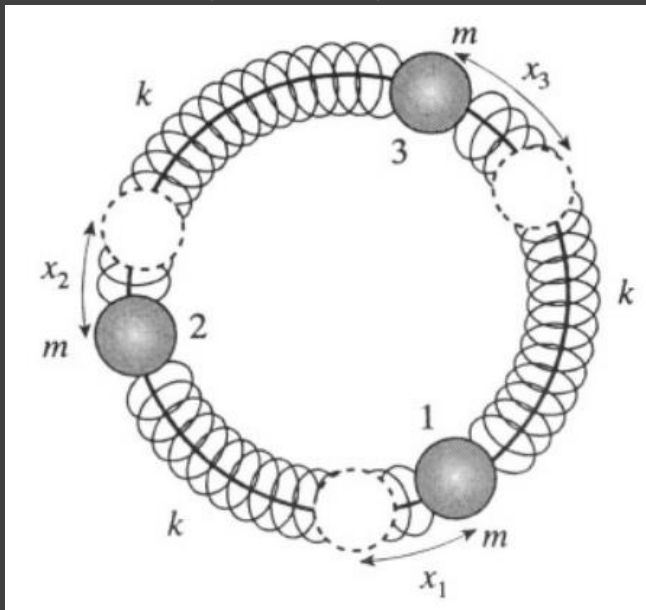
Existe função similar na SciPy.

Mais informações:

<https://numpy.org/doc/stable/reference/generated/numpy.linalg.eig.html#numpy.linalg.eig>

## Exemplo: massas acopladas por molas

3 massas acopladas por molas:



Equações de movimento para cada massa:

$$\frac{d^2 x_1}{dt^2} + \frac{2k}{m} x_1 - \frac{k}{m} (x_2 + x_3) = 0$$

$$\frac{d^2 x_2}{dt^2} + \frac{2k}{m} x_2 - \frac{k}{m} (x_1 + x_3) = 0$$

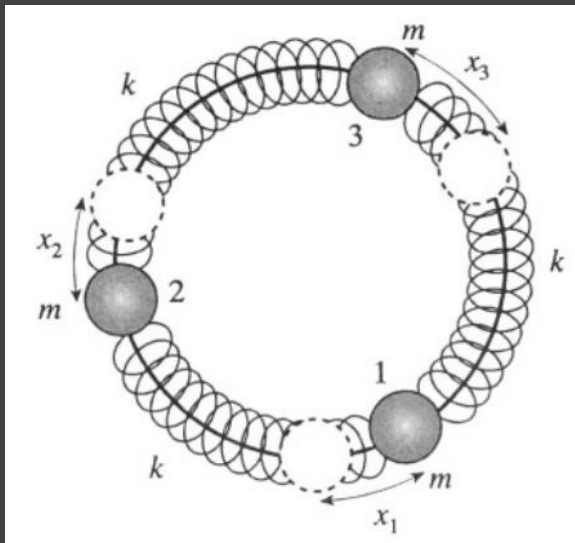
$$\frac{d^2 x_3}{dt^2} + \frac{2k}{m} x_3 - \frac{k}{m} (x_1 + x_2) = 0$$

Considerando solução harmônicas para descobrir as frequências de vibração

$$x_i = A_i e^{i\omega t}$$

# Aplicação: massas acopladas por molas

3 massas acopladas por molas:



$$\frac{d^2 x_1}{dt^2} + \frac{2k}{m} x_1 - \frac{k}{m} (x_2 + x_3) = 0$$

$$\frac{d^2 x_2}{dt^2} + \frac{2k}{m} x_2 - \frac{k}{m} (x_1 + x_3) = 0$$

$$\frac{d^2 x_3}{dt^2} + \frac{2k}{m} x_3 - \frac{k}{m} (x_1 + x_2) = 0$$



$$\frac{2k}{m} A_1 - \frac{k}{m} A_2 - \frac{k}{m} A_3 = \omega^2 A_1$$

$$-\frac{k}{m} A_1 + \frac{2k}{m} A_2 - \frac{k}{m} A_3 = \omega^2 A_2$$

$$-\frac{k}{m} A_1 - \frac{k}{m} A_2 + \frac{2k}{m} A_3 = \omega^2 A_3$$

$$\begin{pmatrix} \frac{2k}{m} & -\frac{k}{m} & -\frac{k}{m} \\ -\frac{k}{m} & \frac{2k}{m} & -\frac{k}{m} \\ -\frac{k}{m} & -\frac{k}{m} & \frac{2k}{m} \end{pmatrix} \begin{pmatrix} A_1 \\ A_2 \\ A_3 \end{pmatrix} = \omega^2 \begin{pmatrix} A_1 \\ A_2 \\ A_3 \end{pmatrix}$$

Equação de autovalores e autovetores (autovalores)