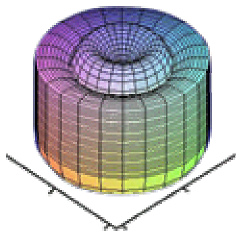
Cálculo Diferencial e Integral: um KIT de sobrevivência

www.dma.uem.br/kit



O Algoritmo de Leverrier – Faddeev

Jorge Ferreira de Lacerda- DMA- UEM

Resumo: O Algoritmo de Leverrier-Faddeev fornece uma alternativa ao cálculo do polinômio característico de uma matriz, sem a utilização de determinantes. Esta é uma alternativa computacionalmente interessante, entre outras razões, devido à redução no número de cálculos quando comparado aos métodos computacionais para o cálculo de determinantes.

1. Introdução

Um polinômio $p(t) = a_0 t^n + a_1 t^{n-1} + ... + a_{n-1} t + a_n$ determina uma função do conjunto das matrizes quadradas. Dada uma matriz quadrada A de ordem n, temos a matriz $p(A) = a_0 A^n + a_1 A^{n-1} + ... + a_{n-1} A + a_n I$.

O ***polinômio característico*** de uma matriz quadrada A de ordem n é dado por

$$p(t) = \det(tI - A) \qquad (1.1)$$

O polinômio característico de uma matriz A é caracterizado como o único polinômio mônico (o coeficiente de t^n é 1) de grau n tal que $p(A) = 0$.

As raízes do polinômio característico são os auto-valores da matriz.

$$p(\lambda) = 0 \Leftrightarrow \det(\lambda I - A) = 0 \Leftrightarrow \exists v \neq 0, \text{ tal que } Av = \lambda v$$

Exemplo 1: O polinômio característico da matriz $A = \begin{bmatrix} 3 & 1 & 5 \\ 3 & 3 & 1 \\ 4 & 6 & 4 \end{bmatrix}$ é

$$p(\lambda) = \det(\lambda I - A) = \begin{vmatrix} \lambda - 3 & -1 & -5 \\ -3 & \lambda - 3 & -1 \\ -4 & -6 & \lambda - 4 \end{vmatrix}$$

$$= (\lambda - 3)[(\lambda - 3)(\lambda - 4) - 6] + [-3(\lambda - 4) - 4] - 5[18 + 4(\lambda - 3)]$$

$$= (\lambda - 3)(\lambda^2 - 7\lambda + 6) + (-3\lambda + 8) - 5(6 + 4\lambda)$$

$$= \lambda^3 - 10\lambda^2 + 4\lambda + 40.$$

Para determinar o polinômio característico basta determinar seus coeficientes. No exemplo acima os coeficientes foram determinados calculando um determinante de ordem 3.

O algoritmo de Leverrier determina os coeficientes do polinômio característico de uma matriz a partir dos traços (*soma dos elementos da diagonal principal)* das potências desta matriz.

2. Algorítmo de Leverrier

Se

$$p(t) = \det(tI - A) = t^n + p_1 t^{n-1} + \ldots + p_{n-1} t + p_n, \qquad (1.2)$$

então os coeficientes $p_1,\ p_2,\ \ldots,\ p_n$ são dados por

$$\begin{aligned} -p_1 &= s_1 \\ -kp_k &= s_k + \sum_{i=1}^{k-1} p_i s_{k-i}, \quad 1 < k \leq n \end{aligned} \qquad \text{(L)}$$

onde $s_k = \operatorname{tr}(A^k), \quad 1 \leq k \leq n$

Exemplo 2: Apliquemos o algoritmo de Leverrier para a matriz A do exemplo 1.

$$A = \begin{bmatrix} 3 & 1 & 5 \\ 3 & 3 & 1 \\ 4 & 6 & 4 \end{bmatrix} \qquad A^2 = \begin{bmatrix} 32 & 36 & 36 \\ 22 & 18 & 22 \\ 46 & 46 & 42 \end{bmatrix} \qquad A^3 = \begin{bmatrix} 348 & 356 & 340 \\ 208 & 208 & 216 \\ 444 & 436 & 444 \end{bmatrix}$$

$$s_1 = \operatorname{tr}(A) = 3 + 3 + 4 = 10$$

$$s_2 = \operatorname{tr}(A^2) = 32 + 18 + 42 = 92$$

$$s_2 = \operatorname{tr}(A^3) = 348 + 208 + 444 = 1000.$$

Das relações (L) obtemos,

$$-p_1 = s_1 = 10 \Rightarrow p_1 = -10$$

$$-2p_2 = s_2 + p_1 s_1 = 92 + (-10)10 = -8 \Rightarrow p_2 = 4$$

$$-3p_3 = s_3 + p_1 s_2 + p_2 s_1 = 1000 + (-10)92 + 4(10) = 120 \Rightarrow p_3 = -40$$

Assim $p(t) = t^3 - 10t^2 + 4t - 40$. Observe que, aqui, nenhum determinante foi calculado.

D. K. Fadeev introduziu a seguinte melhoria no método de Leverrier:

3. Algoritmo de Leverrier-Faddeev

Se

$$p(t) = \det(tI - A) = t^n + p_1 t^{n-1} + \ldots + p_{n-1} t + p_n, \qquad (1.3)$$

então os coeficientes $p_1,\ p_2,\ \ldots,\ p_n$ são dados por

$$
\begin{aligned}
&F_1(A) = A, && p_1 = -\tfrac{1}{1}\operatorname{tr}(F_1(A)) \\
&F_2(A) = A(F_1(A) + p_1 I), && p_2 = -\tfrac{1}{2}\operatorname{tr}(F_2(A)) \\
&\ldots\ldots\ldots\ldots\ldots && \\
&F_k(A) = A(F_{k-1}(A) + p_{k-1} I), && p_k = -\tfrac{1}{k}\operatorname{tr}(F_k(A))
\end{aligned}
\qquad \text{(F)}
$$

Exemplo 3: Apliquemos o algoritmo de Leverrier-Faddeev para a matriz A do exemplo 1.

$$F_1(A) = A = \begin{bmatrix} 3 & 1 & 5 \\ 3 & 3 & 1 \\ 4 & 6 & 4 \end{bmatrix}$$

$$p_1 = -\tfrac{1}{1}\operatorname{tr}(F_1(A)) = -\operatorname{tr}(A) = -(3+3+4) = -10$$

$$F_2(A) = A(F_1(A) + p_1 I) = A(A - 10I) = \begin{bmatrix} 3 & 1 & 5 \\ 3 & 3 & 1 \\ 4 & 6 & 4 \end{bmatrix}\begin{bmatrix} -7 & 1 & 5 \\ 3 & -7 & 1 \\ 4 & 6 & -6 \end{bmatrix} = \begin{bmatrix} 2 & 26 & -14 \\ -8 & -12 & 12 \\ 6 & -14 & 2 \end{bmatrix}$$

$$p_2 = -\tfrac{1}{2}\operatorname{tr}(F_2(A)) = -\tfrac{1}{2}(2 - 12 + 2) = 4$$

$$F_3(A) = A(F_2(A) + p_2 I) = \begin{bmatrix} 3 & 1 & 5 \\ 3 & 3 & 1 \\ 4 & 6 & 4 \end{bmatrix}\begin{bmatrix} 6 & 1 & 5 \\ 3 & -8 & 1 \\ 4 & 6 & 6 \end{bmatrix} = \begin{bmatrix} 40 & 0 & 0 \\ 0 & 40 & 0 \\ 0 & 0 & 40 \end{bmatrix}$$

$$p_3 = -\tfrac{1}{3}\operatorname{tr}(F_3(A)) = -\tfrac{1}{3}(40 + 40 + 40) = -40$$

Observe que $F_3(A) + p_3 I = 0 \Rightarrow F_4(A) = 0$.

Assim, $p(t) = t^n - 10t^2 + 4t - 40$.

Observações: Como $F_3(A) + p_3 I = 0$ e $F_3(A) = A(F_2(A) + p_2 I)$, temos que $A(F_2(A) + p_2 I) + p_3 I = 0 \Rightarrow A \cdot \frac{1}{p_3}(F_2(A) + p_2 I) = I$. Logo, a matriz $\frac{1}{p_3}(F_2(A) + p_2 I)$ é a inversa da matriz A.

Além disso, no polinômio característico da matriz A, $p(t) = \det(tI - A) = t^n + p_1 t^{n-1} + \ldots + p_{n-1} t + p_n$, o termo constante p_n é, a menos de um sinal, igual ao determinante da matriz A. Precisamente, $p_n = (-1)^n \det A$.

Assim, a seqüência (F) inclui o determinante e a inversa da matriz A fornecendo uma maneira alternativa de calcular o determinante e a inversa de uma matriz.

4. Justificativa do algoritmo de Leverrier-Faddeev

i. As raízes do polinômio característico são os auto-valores da matriz

$$p(t) = 0 \Leftrightarrow \det(tI - A) = 0 \Leftrightarrow \exists v \neq 0, \text{ tal que } Av = tv$$

ii. Se λ é auto-valor de A, então λ^k é auto-valor de A^k.

$$Av = \lambda v \Rightarrow A^2 v = \lambda^2 v \Rightarrow \ldots \Rightarrow A^k v = \lambda^k v$$

iii. A soma dos autovalores de uma matriz é igual ao seu traço.

$$\lambda_1 + \lambda_2 + \ldots + \lambda_n = \mathrm{tr}(A)$$

De (ii) temos

$$s_k = \lambda_1^k + \lambda_2^k + \ldots + \lambda_n^k = \mathrm{tr}(A^k)$$

iv. Os coeficientes do polinômio característico são dados pelas *funções simétricas elementares* de suas raízes $\lambda_1, \lambda_2, \ldots, \lambda_n$.

Se $p(t) = (t - \lambda_1)(t - \lambda_2)\ldots(t - \lambda_n) = t^n + p_1 t^{n-1} + \ldots + p_{n-1} t + p_n$, então

$$p_k = (-1)^k \sigma_k, \text{ onde } \sigma_k = \sum_{i_1 < i_2 < \ldots < i_k} \lambda_{i_1} \lambda_{i_2} \ldots \lambda_{ik}$$

v. A partir das identidades de Newton que relacionam as funções simétricas elementares obtemos as seguintes relações que determinam os coeficientes p_k a partir dos traços de A^k.

$$-p_1 = s_1$$

$$-kp_k = s_k + \sum_{i=1}^{k-1} p_i s_{k-i}, \quad 1 < k \leq n \qquad \text{(L)}$$

5. Demonstração de (F) a partir de (L)

A seqüência de matrizes dada por (F) é da forma

$F_k(A) = A^k + a_1 A^{k-1} + \ldots + a_{k-1} A$ onde $a_k = -\frac{1}{k}\mathrm{tr}(F_k(A))$, $k \geq 1$.

Portanto $-ka_k = \mathrm{tr}(F_k(A)) = \mathrm{tr}(A^k) + a_1 \mathrm{tr}(A^{k-1}) + \ldots + a_{k-1}\mathrm{tr}(A) = s_k + \sum_{i=1}^{k-1} a_i s_{k-1}$.

Isto é, os coeficientes a_k verificam as relações (L). Logo $a_k = p_k$. Assim , $F_k(A) = A^k + p_1 A^{k-1} + \ldots + p_{k-1} A$.

Em particular,

$$F_n(A) = A^n + p_1 A^{n-1} + \ldots + p_{n-1} A \Rightarrow$$
$$F_n(A) + p_n I = p(A) = 0 \Rightarrow$$
$$F_{n+1}(A) = 0 \Rightarrow F_k(A) = 0, \text{ para } k > n.$$

6. Demonstração das relações (L) , para $1 \leq k \leq 3$

$\sigma_1 = \sum \lambda_i$

$\sigma_2 = \sum \lambda_i \lambda_j$

$\sigma_3 = \sum \lambda_j \lambda_j \lambda_k$

De (iv) temos,

$p_1 = -\sigma_1$

$p_2 = \sigma_2$

$p_3 = -\sigma_3$

Assim,

$$p_1 = -\sum_i \lambda_i = -s_1 \Rightarrow$$
$$-p_1 = s_1$$

$$\begin{aligned} p_1 s_1 = -s_1 s_1 &= -(\sum_i \lambda_i)(\sum_i \lambda_i) \\ &= -(\sum_i \lambda_i^2 + 2\sum_{i<j} \lambda_i \lambda_j) \\ &= -(s_2 + 2\sigma_2) = -s_2 - 2p_2 \Rightarrow \end{aligned}$$
$$-2p_2 = s_2 + p_1 s_1$$

$$p_1 = -\sigma_1 = -s_1 \Rightarrow$$
$$-p_1 = s_1$$

$$s_1\sigma_1 = (\sum \lambda_i)(\sum \lambda_i) = \sum \lambda_i^2 + 2\sigma_2 = s_2 + 2\sigma_2 \Rightarrow$$
$$-2\sigma_2 = s_2 - s_1\sigma_1 = s_2 + s_1 p_1$$

$$s_1 s_2 = (\sum \lambda_i)(\sum \lambda_i^2) = \sum \lambda_i^3 + \sum \lambda_i^2 \lambda_j = s_3 + \sum \lambda_i^2 \lambda_j$$
$$s_1\sigma_2 = (\sum \lambda_i)(\sum \lambda_i \lambda_j) = \sum \lambda_i^2 \lambda_j + 3\sigma_3$$
$$3\sigma_3 - s_3 = s_1\sigma_2 - s_1 s_2 \Rightarrow$$
$$-3p_3 - s_3 = s_1 p_2 + p_1 s_2 \Rightarrow$$
$$-3p_3 = s_3 + p_1 s_2 + p_2 s_1$$

Com uma pequena modificação o algoritmo (F) pode ser

$B_1 = I,$ $\qquad p_1 = -\frac{1}{1}\operatorname{tr}(AB_1)$

$B_2 = AB_1 + p_1 I,$ $\qquad p_2 = -\frac{1}{2}\operatorname{tr}(AB_2)$

....

$B_k = AB_{k-1} + p_{k-1} I,$ $\qquad p_k = -\frac{1}{k}\operatorname{tr}(AB_k)$

Uma demonstração de (F) utilizando a transformada de Laplace pode ser encontrada em **[2]**.

7. Alguns comentários

a) As operações do algoritmo de Leverrier-Faddeev são bastante simples: cálculo do traço e divisão por um inteiro ($p_k = \frac{1}{k}\operatorname{tr}(F_k(A))$) , modificação da diagonal (soma $F_k(A) + p_k I$) e multiplicação por uma matriz fixa, ($F_{k+1}(A) = A(F_k(A) + p_k I)$) .

b) A matriz calculada em uma etapa depende somente da matriz calculada na etapa anterior $F_{k+1}(A) = A(F_k(A) + p_k I)$. Portanto só precisamos guardar para a etapa seguinte a última matriz $F_k(A)$ calculada e os coeficientes p_k.

c) Se a matriz A tem entradas inteiras, sabemos que os coeficientes p_k devem ser inteiros e apenas números inteiros aparecem no algoritmo.

8. Procedimentos em Maple e MatLab

Ilustramos o algoritmo de Leverrier-Faddeev com os seguintes procedimentos em Maple e MatLab, disponíveis em [3]. Nesses procedimentos o sistema criará aleatoriamente uma matriz quadrada, em seguida é calculado o polinômio característico dessa matriz aleatória utilizando o algoritmo de Leverrier-Faddeeev. Esses procedimentos têm aplicação apenas didática. Veja [3] para outros exemplos.

Procedimento em MatLab:
Crie um arquivo LF.m com os seguintes comandos e o execute. O procedimento solicita a dimensão da matriz a ser criada e retornará os coeficientes do seu polinômio característico e suas raízes.

```
clear
n=input('Entre com o valor de n=');
A=randint(n,n),B=eye(n); for i=1:1:n
 a(i)=-trace(A*B)/i, B=A*B+ a(i)*eye(n),
 end, disp('O polinomio característo  é'); p=[1,a],
 disp('Suas raízes são'), roots(p),
```

Procedimento em Maple:
Crie um arquivo LF.mws com os seguintes comandos. O procedimento cria uma matriz quadrada de dimensão entre 5 e 10 e retornará os coeficientes do seu polinômio característico.

```
>  restart:  with(linalg):
> num:=rand(5..10): n:=num():
> A:=randmatrix(n, n); 'n'=n:
 > LF:=proc(A) local n,a,i,J,B:
n:=rowdim(A):J:=(i,j) ->matrix(n,n,(i,j)-
>piecewise(i<>j,0,i=j,1)):J:=J(n,n):B:=J:for i from 1 to n do
a[i]:=-(trace(A&*B))/i: B:=evalm(A&*B+(a[i])*J):  od:print(`o
polinômio  característico da matriz é`, x^n+sum('a[i]*x^(n-
i)','i'=1..n)): end:

> LF(A);
```

Referências

[1] D. K. Faddeev and V. N. Faddeeva, Computational Methods of Linear Algebra, Freeman, San Francisco, 1963.

[2] Shui-Hung Hou, On The Leverrier-Faddeev algorithm, Dept of Applied Math. , Hong Kong Polytechnic University.

[3] D. Andrade. Cálculo diferencial e Integral: um KIT de sobrevivência. http://www.dma.uem.br/kit/LeverrierFaddeev1.html, 2007.