

UNIVERSIDADE ESTADUAL DE CAMPINAS
INSTITUTO DE MATEMÁTICA, ESTATÍSTICA E
COMPUTAÇÃO CIENTÍFICA

MS512 - PROJETO COMPUTACIONAL II

**DETERMINAÇÃO NUMÉRICA DE
AUTOVALORES E AUTOVETORES E
RESOLUÇÃO NUMÉRICA DE SISTEMAS DE
EQUAÇÕES NÃO LINEARES**

Professor: Eduardo Cardoso de Abreu
Aluna: Ana Paula Magalhães dos Santos
RA: 101461

Campinas, Junho de 2016

Resumo

Neste projeto serão analisados métodos para obtenção de autovalores bem como métodos para resolução de sistemas não lineares. Primeiramente será feita uma análise da Decomposição SVD (*Singular Value Decomposition*). Uma grande vantagem deste tipo de fatoração é que ela pode ser realizada em qualquer tipo de matriz real ou complexa, quadrada ou retangular, e não apresenta as exigências de matrizes simétricas e positivas definitas, como na fatoração de Cholesky, por exemplo. Dentre suas aplicações para resolução de problemas físicos, podemos citar o papel desta ferramenta na compressão de imagens coloridas. Foram abordadas ainda as diferenças e similaridades entre a decomposição SVD e a decomposição espectral. Realizou-se também um sumário sobre as equações de Sylvester e sobre a demonstração da convergência do algoritmo QR, concluindo que este é equivalente ao método das iterações simultâneas. Ainda neste contexto, foram analisados alguns aspectos computacionais das rotações de Givens e reflexões de Householder no cálculo da decomposição QR. Finalmente, foram investigados alguns aspectos e aplicações dos métodos de Newton e quase-Newton, a saber, o método de Broyden.

1 Decomposição SVD

A Decomposição em Valores Singulares (SVD - *Singular Value Decomposition*) é um tópico bastante significativo da Álgebra Linear devido ao grande leque de aplicações para resolução de problemas teóricos e físicos que este tipo de fatoração proporciona.

Na Matemática, a utilização deste tipo de fatoração é extensa. Este tipo de decomposição pode ser utilizada no cálculo da pseudo-inversa, como auxílio na resolução de sistemas equações lineares homogêneas $Ax = b$ (principalmente em casos em que a matriz A é uma matriz não quadrada), na minimização de mínimos quadrados, na aproximação de uma matriz de baixo posto, dentre outros.

Dentre outras aplicações, pode-se destacar a utilização de SVD na compressão de imagens, na análise de quantidades elevadas de dados, na computação de matrizes que atingem dimensões na casa dos milhares, no processamento de sinais, na remoção de ruídos em imagens de ressonância magnética, na detecção de padrões como nos sistemas de reconhecimento facial, no estudo da expressão de genes na Biologia, dentre outros.

Consideremos uma matriz A , real ou complexa, de dimensão $m \times n$. A decomposição SVD fatora esta matriz A em:

$$A = U\Sigma V^t \tag{1}$$

onde U e V são matrizes ortogonais unitárias de dimensões $m \times m$ e $n \times n$, respectivamente, e Σ uma matriz diagonal de dimensão $m \times n$.

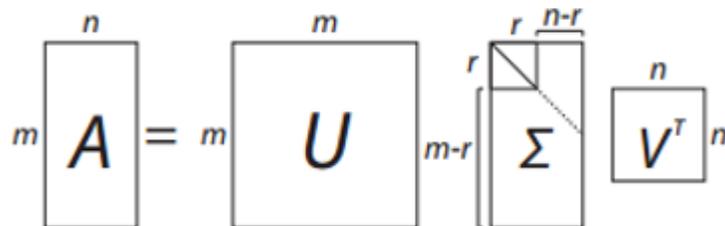


Figura 1: SVD completa de uma matriz $A \in \mathbb{R}^{m \times n}$ ($m \geq n$)

1.1 Teorema

Teorema 1.1. (*Decomposição SVD*): Consideremos uma matriz $A \in \mathbb{R}^{m \times n}$, então, existem matrizes ortogonais: $U = [u_1 | \dots | u_m] \in \mathbb{R}^{m \times m}$ e $V = [v_1 | \dots | v_m] \in \mathbb{R}^{n \times n}$ tal que: $U^t A V = \Sigma = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_p)$ onde $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_p \geq 0$ e $p = \min\{m, n\}$.

Prova: Sejam $x \in \mathbb{R}^n$ e $y \in \mathbb{R}^m$ vetores unitários segundo a norma-2 que satisfazem $Ax = \sigma y$, com $\sigma = \|A\|_2$. Consideremos agora o seguinte resultado:

Teorema 1.2. Se $V_1 \in \mathbb{R}^{n \times r}$ tem colunas ortonormais, então existe $V_2 \in \mathbb{R}^{m \times (n-r)}$ tal que $V = [V_1 | V_2]$ é ortogonal.

Segundo o Teorema (1.2) existem $V_2 \in \mathbb{R}^{n \times (n-1)}$ e $U_2 \in \mathbb{R}^{m \times (m-1)}$ tal que $V = [x | V_2] \in \mathbb{R}^{n \times n}$ e $U = [y | U_2] \in \mathbb{R}^{m \times m}$ são ortogonais. Desta maneira, podemos escrever:

$$U^T A V = \begin{bmatrix} \sigma & w^T \\ 0 & B \end{bmatrix} \equiv A_1$$

onde $w \in \mathbb{R}^{n-1}$ e $B \in \mathbb{R}^{(m-1) \times (n-1)}$. Como:

$$\left\| A_1 \begin{pmatrix} \sigma \\ w \end{pmatrix} \right\|_2^2 \geq (\sigma^2 + w^T w)^2 = \sqrt{(\sigma^2 + w^T w)} \left\| \begin{pmatrix} \sigma \\ w \end{pmatrix} \right\|_2^2$$

temos $\|A_1\|_2^2 \geq (\sigma^2 + w^T w)^2$. Mas, $\sigma^2 = \|A\|_2^2 = \|A_1\|_2^2$, de maneira que devemos ter $w = 0$. Se $m = 1$ e $n = 1$, está provado. Caso contrário, a submatriz B descreve a ação de A no subespaço ortogonal a v_1 . Pela hipótese de indução B , tem uma decomposição SVD: $B = U_2 \Sigma_2 V_2^T$. Daí, podemos verificar que:

$$A = U_1 \begin{bmatrix} 1 & 0 \\ 0 & U_2 \end{bmatrix} \begin{bmatrix} \sigma & 0 \\ 0 & \Sigma_2 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & V_2 \end{bmatrix}^T V_1^T$$

é uma decomposição SVD de A , concluindo a prova da existência. Para a

demonstração da unicidade podemos utilizar um argumento geométrico, cuja interpretação será explorada na subseção (1.4). \square

Os valores σ_i correspondem aos valores singulares da matriz A , u_i são os vetores singulares à esquerda de A e v_i os vetores singulares à direita de A .

1.2 SVD Reduzida

A forma reduzida de uma matriz $A \in \mathbb{R}^{m \times n}$ com $m \geq n$ de posto r consiste na remoção de colunas e linhas extras colocadas com intuito de transformar $U \in \mathbb{R}^{m \times r}$ e $V \in \mathbb{R}^{n \times r}$ em matrizes ortogonais $U \in \mathbb{R}^{m \times m}$ e $V \in \mathbb{R}^{n \times n}$. Essa remoção tem por objetivo fornecer apenas a informação oferecida pelo conjunto de vetores definidos pelo posto da matriz.

$$\begin{array}{c} n \\ \boxed{A} \\ m \end{array} = \begin{array}{c} r \\ \boxed{U} \\ m \end{array} \begin{array}{c} r \\ \boxed{\Sigma} \\ r \end{array} \begin{array}{c} n \\ \boxed{V^T} \\ r \end{array}$$

Figura 2: SVD reduzida de uma matriz $A \in \mathbb{R}^{m \times n}$ ($m \geq n$)

1.3 Uma Interpretação Geométrica

Para entender como a matriz A deforma o espaço, podemos considerar a ação desta sobre uma esfera unitária no \mathbb{R}^n . Um elemento arbitrário x desta esfera pode ser escrito como $x = x_1v_1 + x_2v_2 + \dots + x_nv_n$ com $\sum_1^n x_i^2 = 1$, a imagem deste elemento é $Ax = \sigma_1x_1u_1 + \sigma_2x_2u_2 + \dots + \sigma_kx_ku_k$, com $k < m$. Fazendo $y_i = \sigma_ix_i$ vemos que a imagem da esfera unitária consiste dos vetores $y_1u_1 + y_2u_2 + \dots + y_ku_k$, onde:

$$\frac{y_1^2}{\sigma_1^2} + \frac{y_2^2}{\sigma_2^2} + \dots + \frac{y_k^2}{\sigma_k^2} = \sum_{i=1}^k x_i^2 \leq 1$$

Se A tem posto completo, tal que $k = n$, a desigualdade acima é, na verdade, uma igualdade. Caso contrário, alguns dos x_i não estarão incluídos na equação da direita, e a soma pode assumir qualquer valor de 0 a 1. Isto mostra que A mapeia a esfera unitária do \mathbb{R}^n para um elipsóide k -dimensional com semi-eixos na direção de u_i e com magnitudes σ_i . Se $k = n$ a imagem consiste da superfície do elipsóide, caso contrário, é um elipsóide sólido. Em resumo, podemos visualizar a ação de A da seguinte maneira:

- A 'destruição' de $n - k$ dimensões do domínio;
- A distorção das dimensões restantes;
- A contração e distensão da esfera unitária de dimensão k a um elipsóide;
- Finalmente, A engloba o elipsóide em \mathbb{R}^n ;

Abaixo, uma demonstração da ação de A para $n = m = 3$ e $k = 2$:

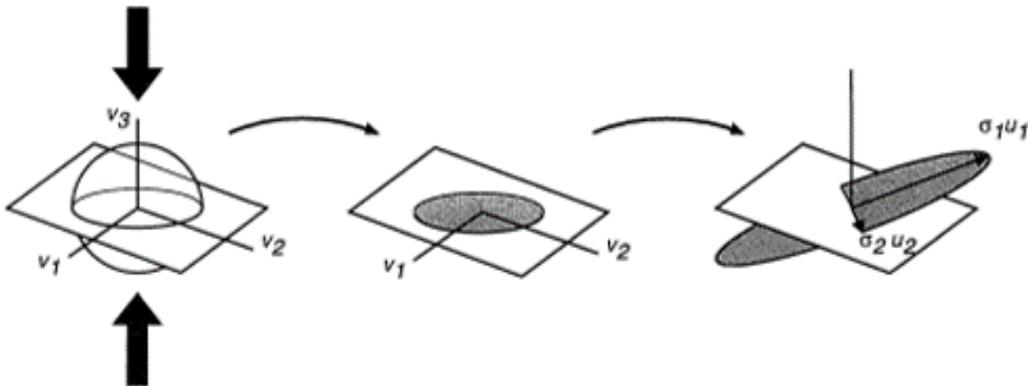


Figura 3: Como A deforma \mathbb{R}^n

1.4 Relação com a Decomposição em Autovalores (decomposição espectral)

A SVD é bastante geral, já que pode ser aplicada a qualquer matriz de dimensão $m \times n$. Já a decomposição em autovalores, ou espectral, pode apenas ser aplicada para algumas classes de matrizes quadradas.

Dada uma decomposição SVD de uma matriz $A \in \mathbb{R}^{m \times n}$, valem as seguintes propriedades:

$$\begin{aligned}A^T A &= (V \Sigma^T U^T)(U \Sigma V^T) = V(\Sigma^T \Sigma)V^T \\AA^T &= (U \Sigma V^T)(V \Sigma^T U^T) = U(\Sigma \Sigma^T)U^T\end{aligned}$$

Os lados direitos dessas relações descrevem a decomposição em autovalor dos lados esquerdos. Sendo assim:

- Os vetores singulares à esquerda de A são autovetores de AA^T ;
- Os vetores singulares à direita de A são autovetores de $A^T A$;
- Os valores singulares não-nulos de A ao longo de Σ são as raízes dos autovalores não-nulos de $A^T A$ ou de AA^T ;

No caso especial em que A é uma matriz normal, ou seja, quando vale $A^T A = AA^T$, que por definição deve ser quadrada, o teorema espectral diz que ela pode ser unitariamente diagonalizada usando-se uma base de autovalores, de forma que ela pode ser escrita para uma matriz unitária U e uma matriz diagonal D . Quando A é também definida positiva, a decomposição $A = UDU^{-1}$ é também uma SVD.

OBS: Uma discussão mais detalhada sobre a relação entre SVD e a decomposição espectral será desenvolvida na seção (2) subseção (2.3).

1.5 Compressão de Imagens

O processo de compressão de imagens lida com o problema de reduzir a quantidade de dados requeridos para representar uma imagem digital. A compressão é alcançada quando se remove três redundâncias básicas:

- Redundância de código;
- Redundância de interpixels, que resulta na correlação entre os pixels;;
- Redundância psicovisual, que ocorre por ocasião de múltiplos dados que são ignorados pelo olho humano;

A principal vantagem de SVD, no que concerne a compressão de imagens, é a possibilidade de armazenar matrizes de maneira que o espaço requerido para seu armazenamento seja reduzido. Na decomposição SVD de uma matriz A , a matriz Σ contém os valores singulares de A ao longo de sua diagonal. Para reduzir o espaço de armazenamento o número de valores singulares da matriz (ou seu posto) precisa ser reduzido.

Durante a compressão, os menores valores singulares são substituídos por zero. Isto reduz o custo de armazenamento ao mesmo tempo em que preserva o tamanho da matriz, de modo que a equação (1) possa ser continuamente utilizada.

Consideremos, então, uma imagem com $m \times n$ pixels. Para uma representação em escala cinza precisamos de apenas um número por pixel, o que pode ser representado por uma matriz de ordem $m \times n$.

Para imagens coloridas precisamos de três números por pixel, para cada cor *red*, *green* e *blue*: RGB. Cada cor pode ser representada por uma matriz de ordem $m \times n$, e, a imagem completa pode ser obtida por meio de uma matriz $m \times 3n$, onde cada 'porção' de cores pode ser colocada em uma coluna, uma ao lado da outra, da seguinte maneira:

$$A = [A_{red} \quad A_{green} \quad A_{blue}]$$

Assim, no processo de compressão de imagens coloridas, devemos aplicar a decomposição SVD a cada uma das matrizes A_{red} , A_{green} e A_{blue} separadamente. As imagens podem então ser comprimidas e, por fim, recompostas para formar a imagem comprimida final.



Figura 4: Imagem original

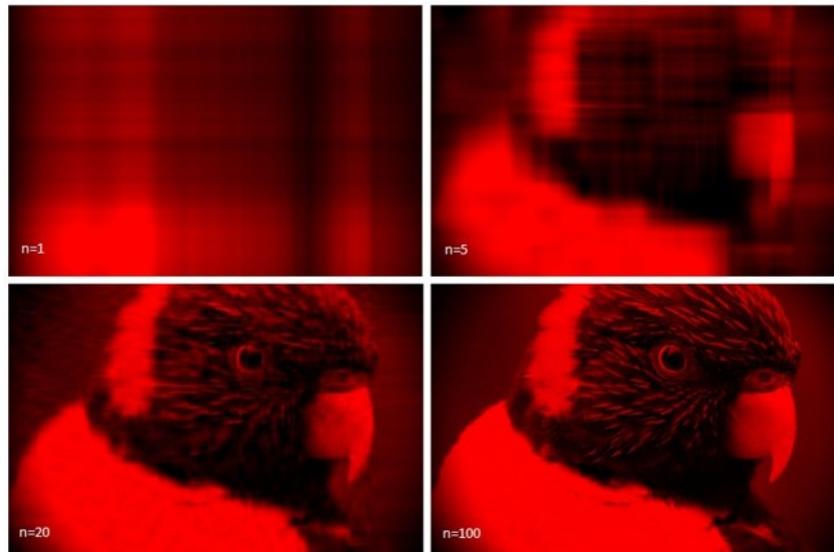


Figura 5: Decomposição SVD do padrão *red* com utilização 1%, 5%, 20% e 100% dos valores singulares totais, respectivamente

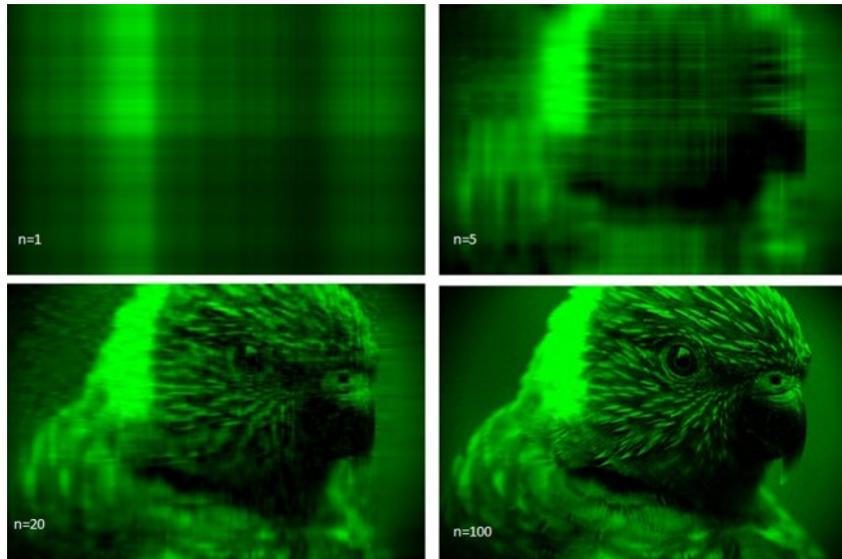


Figura 6: Decomposição SVD do padrão *green* com utilização 1%, 5%, 20% e 100% dos valores singulares totais, respectivamente

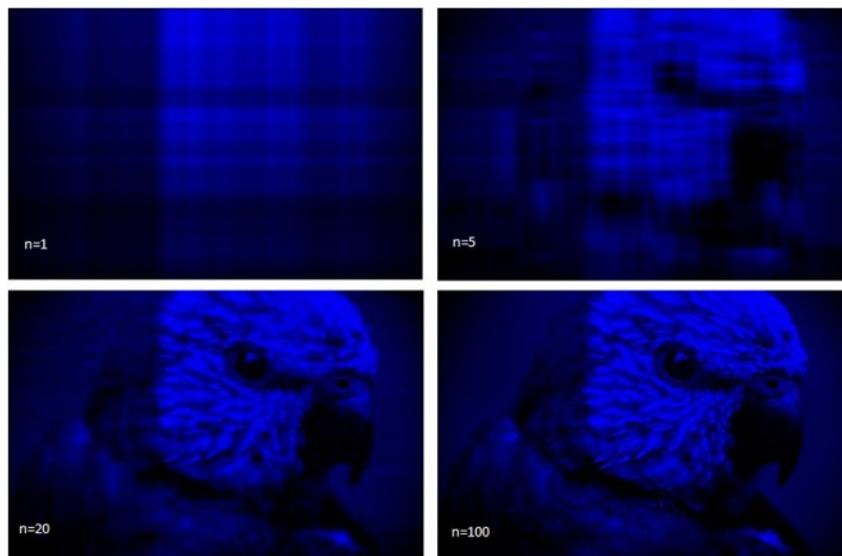


Figura 7: Decomposição SVD do padrão *blue* com utilização 1%, 5%, 20% e 100% dos valores singulares totais, respectivamente

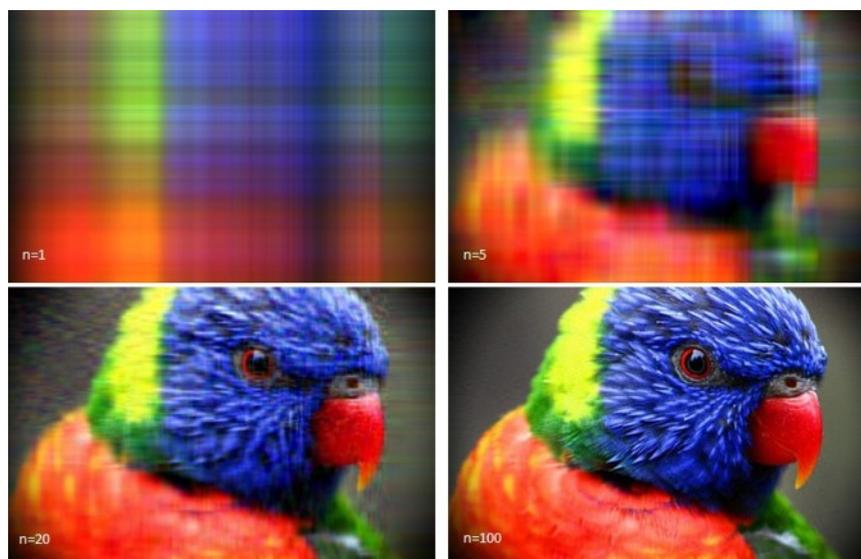


Figura 8: Decomposição SVD (após composição dos padrões RGB) com utilização 1%, 5%, 20% e 100% dos valores singulares totais, respectivamente

Uma figura colorida toma maior espaço na memória, como era de se esperar, uma vez que é necessário maior informação acerca das cores, ao contrário do que ocorre com imagens em escalas de cinza, as quais só exigem informações de brilho.

A imagem mostrada na figura (4) foi comprimida em padrões de *red* na figura (5), padrões de *green* na figura (6) e padrões de *blue* na figura (7). Finalmente, na figura (8), a figura foi recomposta fornecendo a figura final comprimida.

As imagens mostram que a decomposição SVD é bastante eficiente quanto ao armazenamento de informações, pois observamos que com apenas 50% valores singulares totais já obtemos uma boa aproximação da figura original. Também se observa que quanto menos valores singulares tivermos, menos nítida é a imagem comprimida resultante.

Nas próximas páginas, serão mostradas as imagens comprimidas resultantes de várias figuras. O número de valores singulares adotados também foram variados, para efeito de análise das informações obtidas nas respectivas compressões.



Figura 9: Figura original

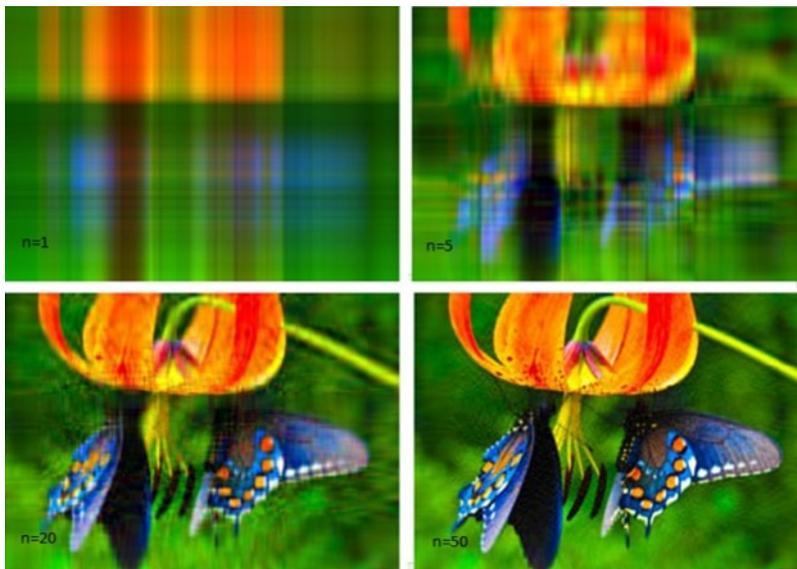


Figura 10: Decomposição SVD (após composição dos padrões RGB) com utilização 1%, 5%, 20% e 50% dos valores singulares totais, respectivamente



Figura 11: Figura original



Figura 12: Decomposição SVD (após composição dos padrões RGB) com utilização 1%, 5%, 20% e 100% dos valores singulares totais



Figura 13: Figura original



Figura 14: Decomposição SVD (após composição dos padrões RGB) com utilização 1%, 5%, 20% e 100% dos valores singulares totais, respectivamente

2 Teorema da Decomposição de Schur e Teorema Espectral

2.1 Decomposição de Schur

Dada uma matriz quadrada $A \in \mathbb{C}^{n \times n}$, a decomposição de Schur garante que podemos encontrar uma matriz triangular superior T similar a A através de transformações unitárias, obtendo assim seus autovalores.

Teorema 2.1. *Seja $A \in \mathbb{C}^{n \times n}$. Então, existem uma matriz unitária $U \in \mathbb{C}^{n \times n}$ e uma matriz triangular superior $T \in \mathbb{C}^{n \times n}$ tais que $U^*AU = T$.*

OBS: M^* é uma notação para matrizes complexas transpostas e conjugadas.

Prova: A prova é feita por indução sobre a ordem da matriz A . Para $n = 1$ o resultado é imediato. Suponhamos que o resultado seja válido para $n = k - 1$ e vamos mostrar que é válido para $n = k$. Seja A uma matriz de ordem k e (λ, v) um autopar da matriz A , com $\langle v, v \rangle = 1$. Consideremos U_1 uma matriz unitária que possui v como sua primeira coluna, e as outras colunas consideramos como sendo um complemento para uma base ortonormal do espaço vetorial complexo \mathbb{C}^k . Seja W uma submatriz de ordem $k \times (k - 1)$ de U_1 , de maneira que $U_1 = [v \ W]$. Percebe-se também que $W^*v = 0$. Consideremos agora uma matriz $A_1 = U_1^*AU_1$, escrita da seguinte maneira:

$$A_1 = \begin{bmatrix} v^* \\ W^* \end{bmatrix} A \begin{bmatrix} v & W \end{bmatrix} = \begin{bmatrix} v^*Av & v^*AW \\ W^*Av & W^*AW \end{bmatrix}$$

Como $vA = \lambda v$, temos que $v^*Av = \lambda$ e $W^*Av = \lambda W^*v = 0$. Tomemos agora a matriz $\hat{A} = W^*AW$, de ordem $(k - 1)$, e o elemento $w = W^*Av$. A matriz A_1 terá, então, a seguinte forma:

$$A_1 = \begin{bmatrix} \lambda & w^* \\ 0 & \hat{A} \end{bmatrix}$$

Pela hipótese de indução temos que existem uma matriz unitária \widehat{U}_2 e uma matriz triangular superior \widehat{T} , de ordem $(k - 1)$, tal que $\widehat{T} = \widehat{U}_2^* \widehat{A} \widehat{U}_2$. Consideremos uma matriz U_2 de ordem k da seguinte forma:

$$U_2 = \begin{bmatrix} 1 & 0^T \\ 0 & \widehat{U}_2 \widehat{A} \end{bmatrix}$$

Assim, temos que U_2 é uma matriz unitária e $U_2^* A_1 U_2 = T$ é uma matriz triangular superior representada por:

$$T = \begin{bmatrix} \lambda & w^* \widehat{U}_2 \\ 0 & \widehat{T} \end{bmatrix}$$

Fazendo $U = U_1 U_2$ e utilizando a representação da matriz A_1 , temos:

$$T = U_2^* A_1 U_2 = (U_1 U_2)^* A (U_1 U_2) = U^* A U.$$

□

2.2 Teorema Espectral

Definição 2.1: Uma matriz Hermitiana é uma matriz quadrada de elementos complexos que tem a característica de ser igual a sua própria transposta conjugada.

Teorema 2.2. *Seja $A \in \mathbb{R}^{n \times n}$ uma matriz que possui um conjunto linearmente independente de autovetores v_1, v_2, \dots, v_n associados aos autovalores $\lambda_1, \lambda_2, \dots, \lambda_n$. Vamos definir uma matriz diagonal $D = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$ e uma matriz $V = [v_1 \ v_2 \ \dots \ v_n]$, invertível. Então, $V^{-1} A V = D$. Da mesma forma se $V^{-1} A V = D$, onde D é uma matriz diagonal e V uma matriz invertível, então, as colunas da matriz V formam um conjunto linearmente independente de autovetores de A e os elementos da diagonal principal de D são os autovalores de A .*

Prova: (\Rightarrow) Temos que $Av_i = \lambda_i v_i$, para $i = 1 : n$. Na forma matricial temos que $AV = VD$. Como V é invertível temos que $V^{-1}AV = D$.
 (\Leftarrow) Inversão do argumento da ida. \square

Teorema 2.3. (*Decomposição Espectral*) Seja $A \in \mathbb{R}^{n \times n}$ uma matriz Hermitiana. Então existem uma matriz unitária $U \in \mathbb{R}^{n \times n}$ e uma matriz diagonal $D \in \mathbb{R}^{n \times n}$ tais que $U^*AU = D$. Ademais, as colunas da matriz U são os autovetores de A e os elementos da diagonal de D são os autovalores de A .

Prova: A prova segue do teorema de Schur. Sabemos que existem uma matriz unitária U e uma matriz triangular superior T tais que $T = U^*AU$. Como A é uma matriz hermitiana temos que $T^* = U^*AU$. Segue que $T^* = T$, ou seja, T é uma matriz diagonal real. O restante da demonstração segue do Teorema (2.2). \square

2.3 SVD \times Decomposição Espectral

Existem pelo menos três diferenças fundamentais entre a decomposição espectral e a SVD. A primeira delas é que a decomposição espectral baseia-se em apenas uma base de autovetores, enquanto que a SVD baseia-se em duas bases, uma de vetores singulares à esquerda (colunas de U) pertencentes ao \mathbb{R}^m e outra de vetores singulares à direita (colunas de V) pertencente ao \mathbb{R}^n . Por exemplo, uma matriz $A \in \mathbb{R}^{m \times m}$ não degenerada pode ser expressa por meio de uma matriz diagonal de autovalores D , se tanto a imagem como o domínio da transformação linear envolvida forem representados pela mesma base de autovetores. Ou seja, dada a decomposição $A = VDV^{-1}$, os vetores x e b pertencentes ao \mathbb{R}^m que satisfazem:

$$Ax = b, \tag{2}$$

quando representados pela mesma base de autovetores onde $b' = V^{-1}b$ e $x' = V^{-1}x$ permitem que a relação (2) seja expressa em termos de b' e x' . Portanto esta transformação linear se reduz a $b' = Dx'$ ao se multiplicar ambos os lados pela mesma base de vetores (V^{-1}) e substituir a matriz A pela decomposição $V^{-1}DV$, como mostrado abaixo:

$$Ax = b \rightarrow V^{-1}b = V^{-1}(V^{-1}DV)x \rightarrow (V^{-1}b) = D(V^{-1}x) \rightarrow b' = Dx' \quad (3)$$

Exemplo 1. *Mudança de base via decomposição espectral* ($b = Ax \rightarrow b' = Dx'$)

```
A=[1 2 3; 1 3 4; 1 4 5];
[V, D]=eig(A);
x=[1; 4; 5];
b=(A*x)';
b =

24    33    42
b1=inv(A)*b;
x1=inv(V)*x;
b1' =
ans =

58.721406   -0.7228259   -7.105e-14
(D*x1)' =
ans =

58.721406   -0.7228259   -6.911e-16
```

□

Por outro lado, a mudança de bases realizada pela SVD consiste no fato de que qualquer vetor $b \in \mathbb{R}^m$ pode ser expresso pela base de vetores singulares à esquerda de A (colunas de U) e qualquer vetor $x \in \mathbb{R}^n$ pode ser expresso pela base de vetores singulares à direita de A (colunas de V). Isto é, $b' = U^T b$ e $x' = V^T x$. Dessa forma, pela decomposição $A = UDV^T$, a relação (2) pode ser expressa em termos de b' e x' , como visto abaixo:

$$Ax = b \rightarrow U^T b = U^T U D V^T x \rightarrow U^T b = D V^T x \rightarrow b' = Dx' \quad (4)$$

Portanto, a transformação linear (2) se reduz à transformação linear $b' = Dx'$ quando a imagem ($b \in \mathbb{R}^m$) é expressa por meio da base formada

pelas colunas de $U \in \mathbb{R}^{m \times m}$ e o domínio ($x \in \mathbb{R}^n$) é expresso por meio da base formada pelas colunas de $V \in \mathbb{R}^{n \times n}$.

Exemplo 2. *Mudança de base da SVD ($b = Ax \rightarrow b' = Sx'$)*

A=[1 2; 1 3; 1 4]

A =

1 2

1 3

1 4

[U, S, V]=svd(A,0)

U =

-0.39134 0.82474

-0.56057 0.13818

-0.72981 -0.54838

S =

Diagonal Matrix

5.64016 0

0 0.43429

V =

-0.29817 0.95451

-0.95451 -0.29817

x=[1;4];

b=(A*x)'

b =

9 13 17

b1=U'*b';

x1=V1*x;

x1=V'*x;

```
b1'
ans =

    -23.21614    -0.10343
(S*x1)'
ans =

    -23.21614    -0.10343
```

□

A segunda diferença entre as duas decomposições é que a SVD sempre utiliza bases ortonormais, enquanto que na decomposição espectral, nem sempre a base de vetores gerada é sequer ortogonal.

Finalmente, a terceira e última principal diferença é que somente matrizes quadradas tem decomposição espectral, enquanto que a SVD pode ser aplicada à qualquer matriz, até mesmo às retangulares.

3 A Equação de Sylvester

A equação de Sylvester consiste da seguinte equação linear com matrizes:

$$AX + XB = C \quad (5)$$

onde A é uma matriz de dimensão $n \times n$, B uma matriz de dimensão $m \times m$ e X uma matriz arbitrária de dimensão $m \times n$.

Como a equação é linear em X somos capazes de descrevê-la na forma $Ax = b$. De fato, se considerarmos como $vec(x)$ o vetor que engloba todas as colunas de X uma em cima da outra, da primeira à última, então a equação (5) pode ser escrita como:

$$(I_n \otimes A + B^T \otimes I_m)vec(X) = vec(C), \quad (6)$$

onde a dimensão de $(I_n \otimes A + B^T \otimes I_m)$ é $nm \times nm$.

O coeficiente $(I_n \otimes A + B^T \otimes I_m)$ da matriz apresentada em (6) é muito bem estruturado, mas, esta estrutura não proporcionou vantagens para a resolução da equação (5). Ainda há muita pesquisa direcionada à procura de uma forma de resolver a equação (5) de maneira direta. Em uma das fórmulas encontradas, temos que se a integral $\int_a^\infty e^{At} C e^{Bt} dt$ tem solução, o resultado desta integral multiplicado por -1 é a solução da equação de Sylvester.

3.1 Algumas Aplicações

A equação de Sylvester aparece em problemas bastante interessantes da Álgebra Linear. Podemos destacar:

- Sua relação com o Tensor Produto de Kronecker, que foi explorado no Projeto Computacional 1, no estudo de matrizes esparsas;
- A diagonalização por blocos de uma matriz A se reduz à solução da equação de Sylvester;

- O cálculo da derivada de Fréchet;
- A solução da equação de Sylvester para os casos em que as matrizes A e B são esparsas e C tem posto pequeno, tem grande aplicação na teoria de controle e na redução de ordem de modelos. Para o último, é possível encontrar métodos iterativos para aproximação de X baseados nos subespaços de Krylov;

3.2 Lei da Inércia

A inércia de uma matriz Hermitiana consiste numa tripla de inteiros (ν, ζ, π) onde ν é o número de autovalores negativos, ζ o número de autovalores nulos e π o número de autovalores positivos.

Lei da Inércia de Sylvester: *Para qualquer matriz hermitiana A e qualquer matriz X não singular, a inércia de A é a mesma de X^*AX . Uma transformação do tipo X^*AX é chamada de congruência, então, a Lei da Inércia de Sylvester, mostra-nos que o número de autovalores negativos, positivos e nulos não mudam durante uma transformação congruente.*

Dentre diversas aplicações, a Lei da Inércia pode ser utilizada para o cálculo de autovetores de matrizes hermitianas tridiagonais. O cômputo dos autovalores de uma matriz tridiagonal através da Lei de Inércia de Sylvester, tem custo computacional reduzido. No contexto deste curso, esta ferramenta pode ser importante para o cálculo de autovalores de matrizes de banda, por exemplo, bastante explorada no Projeto Computacional 1.

4 A Fatoração QR

Dada uma matriz A de ordem $m \times n$, $m \geq n$, a fatoração QR desta matriz consiste em representá-la por meio do produto de matrizes Q e R , sendo Q uma matriz de ordem $m \times m$ ortogonal e R uma matriz de ordem $n \times n$ triangular superior:

$$A=Q \begin{bmatrix} R \\ 0 \end{bmatrix}$$

É comum empregarem-se transformações ortogonais para o cálculo da decomposição QR. Dentre estas transformações, destacam-se as reflexões de Householder e as rotações de Givens que consistem em aplicar sucessivas transformações com o intuito de construir uma matriz ortogonal Q de ordem $m \times m$ composta pelos vetores ortonormais geradores da imagem de A , e, ao mesmo tempo, reduzir a matriz A para forma triangular superior.

4.1 Exemplos Computacionais

Modificando os códigos oferecidos `runQRGivensRot.m` e `runQRHouseholderRef.m`, pudemos verificar a cada iteração as respectivas matrizes de transformações ortogonais de Givens e Householder.

Exemplo 3. *Exemplo de uma aplicação da transformação de Givens (rotações) em uma matriz A de ordem 3×3 gerada aleatoriamente.*

A =

0.2259	0.4357	0.4302
0.1707	0.3111	0.1848
0.2277	0.9234	0.9049

G =

0.7978	-0.6029
0.6029	0.7978

A =

0.2832	0.5352	0.4547
0	-0.0145	-0.1119
0.2277	0.9234	0.9049

G =

0.7793	-0.6266
0.6266	0.7793

A =

0.3633	0.9957	0.9213
0	-0.0145	-0.1119
0	0.3843	0.4203

G =

0.0376	0.9993
-0.9993	0.0376

A =

0.3633	0.9957	0.9213
0	-0.3846	-0.4242
0	0	-0.0960

R =

0.3633	0.9957	0.9213
0	0.3846	0.4242
0	0	0.0960

Exemplo 4. *Exemplo de uma aplicação da transformação ortogonal de Householder (reflexões) em uma matriz A de ordem 3×3 gerada aleatoriamente.*

A =

0.0292	0.4886	0.4588
0.9289	0.5785	0.9631
0.7303	0.2373	0.5468

u =

0.7158
0.5489
0.4316

P =

-0.0247	-0.7859	-0.6179
-0.7859	0.3973	-0.4739
-0.6179	-0.4739	0.6274

A =

-1.1820	-0.6133	-1.1061
-0.0000	-0.2666	-0.2371
-0.0000	-0.4272	-0.3968

u =

0.8745
0.4851

P =

-0.5294 -0.8484
-0.8484 0.5294

A =

-1.1820 -0.6133 -1.1061
-0.0000 0.5035 0.4622
-0.0000 0 -0.0090

u =

1

P =

-1

A =

-1.1820 -0.6133 -1.1061
-0.0000 0.5035 0.4622
-0.0000 0 0.0090

R =

1.1820	0.6133	1.1061
0	0.5035	0.4622
0	0	0.0090

5 O Algoritmo QR

O algoritmo QR é um dos mais utilizados na Álgebra Linear Computacional. Desde o início de seu desenvolvimento, nos anos 60, o algoritmo tem gerado muito estudo e análise, sendo que melhorias em sua implementação computacional tem sido propostas ao longo do tempo.

Nos dois textos analisados (ref. [7] e [8]) percebemos a preocupação do autor em investigar o algoritmo QR de forma bastante minuciosa, demonstrando de forma mais rigorosa sua equivalência com o Método das Iterações Simultâneas, uma vez que em argumentos prévios, o algoritmo QR era mais comumente relacionado ao Método das Iterações Simultâneas Inverso.

Vale ainda ressaltar a utilização do algoritmo em métodos oriundos de conceitos dos subespaços de Krylov, dos quais derivam muitos métodos iterativos interessantes, tal como o Método dos Gradientes Conjugados.

5.1 Algoritmo QR sem mudança de escala

O algoritmo QR, em sua forma mais simples, pode ser visto como um procedimento estável para o cômputo de fatorações QR das matrizes de potências A , A^2 , A^3 ...

Algoritmo 1: Algoritmo QR

```
A(0) = A
for k = 1, 2 ...
    Q(k)R(k) = A(k)
    A(k) = R(k)Q(k)
```

O que fazemos é tomar uma decomposição QR de uma matriz real $A \in \mathbb{R}^{(m \times n)}$, multiplicar os fatores Q e R em ordem reversa e repetir o procedimento. Aqui, assumiremos que A é simétrica com autovalores reais e autovetores ortonormais. Nosso interesse é que as matrizes $A^{(k)}$ convirjam para

uma matriz diagonal.

Para que a convergência para uma matriz diagonal seja eficiente para o cálculo dos autovalores, as operações envolvidas devem ser similares. Com efeito: QR triangulariza $A^{(k)} \rightarrow R^{(k)} = (Q^{(k)})^T A^{(k-1)}$. Multiplicando $Q^{(k)}$ à direita temos que $A^{(k)} = (Q^{(k)})^T Q^{(k)} A^{(k-1)} Q^{(k)}$.

5.2 Algoritmo QR com mudança de escala

O Método QR tem complexidade computacional da ordem de n^3 . Para torná-lo mais robusto, podemos introduzir mudanças de escala em cada passo. Desta maneira ao invés de fatorarmos $A^{(k)}$, efetuamos uma mudança de escala $A^{(k)} - \mu^{(k)}I$ a cada passo, sendo $\mu^{(k)}$ uma estimativa para um autovalor. Logo, toda vez que se encontra um autovalor o problema é reduzido pois a matriz $A^{(k)}$ é 'quebrada' em submatrizes.

Algoritmo 2: Método QR com mudança de escala

$$(Q^{(0)})^T A^{(0)} Q^{(0)} = A$$

for $k = 1, 2 \dots$

Escolha uma escala $\mu^{(k)}$

$$Q^{(k)} R^{(k)} = A^{(k)} - \mu^{(k)} I$$

$$A^{(k)} = R^{(k)} Q^{(k)} + \mu^{(k)} I$$

Se qualquer elemento fora da diagonal de $A_{j,j+1}^{(k)}$ é suficientemente próximo de zero,

façamos $A_{j,j+1} = A_{j+1,j}$, obtendo,

$$\begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} = A^{(k)}$$

aplicamos, então, o algoritmo QR para A_1 e A_2 .

5.3 Iterações Simultâneas não normalizadas

Queremos relacionar o algoritmo QR ao método das Iterações Simultâneas. A ideia deste último método consiste em aplicar o método das potências a vários vetores de uma só vez.

Suponhamos que comecemos com um conjunto linearmente independente $v_1^{(0)}, v_2^{(0)}, \dots, v_n^{(0)}$. Parece plausível dizer que para $k \rightarrow \infty$ temos que $A^{(k)}v_1^{(0)}$ converge ao autovetor correspondente ao maior autovalor de A , em valor absoluto. Assim, o espaço $\langle A^{(k)}v_1^{(0)}, \dots, A^{(k)}v_n^{(0)} \rangle$ deve convergir ao espaço $\langle q_1, \dots, q_n \rangle$ gerado pelos autovetores de A .

Definamos $V^{(0)}$ como uma matriz $m \times n$ da seguinte forma:

$$V^{(0)} = [v_1^{(0)} \mid \dots \mid v_n^{(0)}] \quad (7)$$

Após k aplicações de A , temos que:

$$V^{(k)} = A^{(k)}V^{(0)} = [v_1^{(k)} \mid \dots \mid v_n^{(k)}] \quad (8)$$

Como temos interesse na coluna $V^{(k)}$, tomemos uma base para este espaço aplicando a fatoração QR a $V^{(k)}$:

$$\widehat{Q}^{(k)}\widehat{R}^{(k)} = V^{(k)} \quad (9)$$

Sendo $\widehat{Q}^{(k)}$ uma matriz de ordem $m \times n$ e $\widehat{R}^{(k)}$ uma matriz de ordem $n \times n$. Para $k \rightarrow \infty$ as colunas sucessivas de $\widehat{Q}^{(k)}$ devem convergir para os autovetores $\pm q_1, \dots, \pm q_n$.

5.4 Iterações Simultâneas Normalizadas

Os vetores coluna da matriz da equação (7) convergem para o autovalor dominante q_1 de A quando $k \rightarrow \infty$. Embora o espaço $\langle v_1, \dots, v_n \rangle$ tenha uma convergência útil, estes vetores formam uma base mal condicionada deste espaço. Para melhorar este aspecto do algoritmo, devemos realizar a orthonormalização a cada passo e não de uma só vez.

Assim, não devemos construir $V^{(k)}$ como definida na equação (7), mas uma sequência $Z^{(k)}$ diferente com os mesmos espaços coluna.

Algoritmo 3: Método das Iterações Simultâneas

$\widehat{Q}^{(0)} \in \mathbb{R}^{(m \times n)}$ com colunas ortonormais

for $k = 1, 2, \dots$
 $Z = A\widehat{Q}^{(k-1)}$
 $\widehat{Q}^{(k)}\widehat{R}^{(k)} = Z$

5.5 Iterações Simultâneas \Leftrightarrow Algoritmo QR

O algoritmo QR é equivalente ao método das iterações simultâneas quando aplicado à um conjunto de $n = m$ vetores iniciais, onde denotamos a identidade $\widehat{Q}^{(0)} = I$. Como as matrizes $\widehat{Q}^{(k)}$ e $\widehat{R}^{(k)}$ são agora quadradas, estamos lidando com fatorações QR completas, de maneira que $\widehat{Q}^{(k)} = \underline{Q}^{(k)}$ e $\widehat{R}^{(k)} = \underline{R}^{(k)}$. A nova notação foi adotada para diferenciarmos a fatoração QR de cada um dos métodos.

Seguindo o algoritmo (3), temos abaixo três fórmulas que definem o método das iterações simultâneas para $Q^{(0)} = I$ e uma quarta fórmula que tomaremos como definição para uma matriz $A^{(k)}$ de ordem $m \times m$

$$\underline{Q}^{(0)} = I, \quad (10)$$

$$Z = A\underline{Q}^{(k-1)}, \quad (11)$$

$$Z = \underline{R}^{(k)}\underline{Q}^{(k)}, \quad (12)$$

$$A^{(k)} = (\underline{Q}^{(k)})^T A \underline{Q}^{(k)} \quad (13)$$

A seguir as três fórmulas que definem o algoritmo QR e uma quarta fórmula que tomaremos como definição para uma matriz $\underline{Q}^{(k)}$ de ordem $m \times m$:

$$A^{(0)} = A, \quad (14)$$

$$A^{(k-1)} = \underline{Q}^{(k)} \underline{R}^{(k)}, \quad (15)$$

$$A^{(k)} = \underline{R}^{(k)} \underline{Q}^{(k)}, \quad (16)$$

$$\underline{Q}^{(k)} = Q^{(1)}Q^{(2)} \dots Q^{(k)} \quad (17)$$

Definamos também uma matriz de ordem $m \times m$ $\underline{R}^{(k)}$, tal que:

$$A^{(k)} = \underline{Q}^{(k)} \underline{R}^{(k)} \quad (18)$$

O seguinte teorema prova a equivalência dos dois métodos:

Teorema 5.1. *Os processos (10)-(13) e (14)-(17) geram sequências idênticas de matrizes $\underline{Q}^{(k)}$, $\underline{R}^{(k)}$ e $A^{(k)}$, ou seja, as mesmas sequências definidas pela fatoração QR da k -ésima potência de A :*

$$A^{(k)} = \underline{Q}^{(k)} \underline{R}^{(k)}, \quad (19)$$

juntamente com a projeção,

$$A^{(k)} = (\underline{Q}^{(k)})^T A \underline{Q}^{(k)}, \quad (20)$$

Prova: A demonstração segue por indução em k .

5.6 Convergência do Algoritmo QR

Definição 5.1 (*Coefficiente de Rayleigh*): *Dada uma matriz complexa e Hermitiana M e um vetor x não nulo. o coeficiente de Rayleigh $R(M, x)$ é dado por:*

$$R(M, x) = \frac{x^* M x}{x^* x} \quad (21)$$

Para o caso de matrizes reais, a condição de matriz Hermitiana reduz-se à exigência de matriz simétrica. O coeficiente de Rayleigh atinge seu mínimo em λ_{\min} , o menor autovalor de M , quando x é v_{\min} , o correspondente autovetor. Similarmente, $R(M, x) \leq \lambda_{\max}$ e $R(M, v_{\max}) = \lambda_{\max}$.

Investigando-se a equação (19), percebe-se que o algoritmo QR encontra autovalores pois constroi bases para sucessivas potências A^k . Já a equação (20) nos mostra que os elementos da diagonal de $A^{(k)}$ são coeficientes de Rayleigh de A correspondentes às colunas de $\underline{Q}^{(k)}$. À medida que estas colunas convergem para autovetores, os coeficientes de Rayleigh convergem para os respectivos autovalores duas vezes mais rapidamente. Ao mesmo tempo, os elementos que estão fora da diagonal principal de $A^{(k)}$ são correspondentes aos coeficientes gerais de Rayleigh, envolvendo diferentes aproximações de autovetores distintos de A à direita e à esquerda. Como as aproximações devem ser ortogonais à medida em que convergem para autovetores distintos, os elementos fora da diagonal principal de $A^{(k)}$ devem convergir para zero.

6 Resolução de Sistemas de Equações não Lineares

Equações não lineares aparecem em uma infinidade de problemas. Modelos físicos, que são expressos como equações diferenciais parciais não homogêneas, tornam-se grandes sistemas de equações não homogêneas quando discretizados.

Resolver um sistema de equações não homogêneas é um problema que deve ser evitado quando possível. Usualmente, é feita uma tentativa para aproximar o problema de equações não lineares para um sistema de equações lineares. Quando não é possível realizar tal aproximação, o problema deve ser resolvido diretamente.

6.1 Método do Ponto Fixo

Um sistema de equações não lineares de n equações e n variáveis, pode ser representado como uma função $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$, definida por:

$$F(x_1, \dots, x_n) = (f_1(x_1, \dots, x_n), f_2(x_1, \dots, x_n), \dots, f_n(x_1, \dots, x_n)) \quad (22)$$

Em notação vetorial:

$$F(x) = 0 \quad (23)$$

onde f_1, \dots, f_n são funções coordenadas da função F .

Definição 6.1 *Seja uma função $F : D \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$ da forma:*

$$F(x) = (f_1(x_1, \dots, x_n), f_2(x_1, \dots, x_n), \dots, f_n(x_1, \dots, x_n))^T,$$

onde $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$, $i = 1 : n$. Definimos:

$$\lim_{x \rightarrow x_0} F(x) = L = (L_1, \dots, L_n)^T,$$

se e somente se $\lim_{x \rightarrow x_0} f_i(x) = L_i$, para cada $i = 1 : n$.

A função F é contínua em $x_0 \in D$ se $\lim_{x \rightarrow x_0} F(x)$ existir e $\lim_{x \rightarrow x_0} F(x) = F(x_0)$.

Teorema 6.1. *Seja $f : D \subset \mathbb{R}^n \rightarrow \mathbb{R}$ e $x_0 \in D$. Suponhamos que todas as derivadas parciais de f e constantes $\delta > 0$ e $K > 0$ existam para qualquer que sejam $\|x - x_0\| < \delta$ e $x \in D$, temos que se:*

$$\left| \frac{\partial f(x)}{\partial x_j} \right| \leq K$$

para cada $j = 1 : n$, então f é contínua em x_0 .

Definição 6.2 *Uma função $G : D \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$ tem ponto fixo em $p \in D$ se $G(p) = p$.*

Teorema 6.2. *Seja $D = \{(x_1, \dots, x_n)^T \mid a_i \leq b_i, i = 1 : n\}$ para alguma coleção de a_1, \dots, a_n e b_1, \dots, b_n . Suponhamos que G seja uma função constante de $D \subset \mathbb{R}^n$ no \mathbb{R}^n com a propriedade $G(x) \in D$ para qualquer $x \in D$. Então, G tem ponto fixo em D . Ainda mais, suponhamos que todas as funções componentes de G tenham derivadas parciais contínuas e que exista uma constante $K < 1$, tal que:*

$$\left| \frac{\partial g_i(x)}{\partial x_j} \right| \leq \frac{K}{n}$$

para todo $x \in D$, para cada $j \in 1 : n$ e cada função componente g_i . Então, a sequência $\{x^{(k)}\}_{k=0}^{\infty}$ definida por um $x^{(0)}$ arbitrário em D gerado por:

$$x^{(k)} = G(x^{(k-1)}),$$

para cada $k \geq 1$ converge para o único ponto fixo $p \in D$ e:

$$\|x^{(k)} - p\|_{\infty} \leq \frac{K}{1 - K} \|x^{(1)} - x^{(0)}\|_{\infty}.$$

Assim, se tivermos um sistema de equações não lineares $F(x) = 0$, podemos aplicar o método do ponto fixo para resolvê-lo. Primeiramente, transformamos o sistema de equações em um sistema equivalente:

$$x = G(x)$$

Uma maneira de fazer isto, é resolver a i -ésima equação do sistema original para x_i . Isto é análogo à derivação do método de Jacobi para resolução de sistemas lineares. Daí, escolhemos uma aproximação inicial x_0 . Então, computamos as seguintes iterações sucessivas:

$$x^{(k+1)} = G(x^{(k)}), \quad k = 1 : n.$$

É interessante também ressaltar que a convergência do método do ponto fixo para várias variáveis pode ser realizada utilizando uma aproximação semelhante àquela utilizada no método para resolução de equações lineares de Gauss-Seidel. A saber, quando computamos $x_i^{(k+1)}$ com a avaliação de $g_i(x^{(k)})$, podemos substituir $x_j^{(k)}$, com $j < i$, por $x_j^{(k+1)}$, uma vez que já foi anteriormente calculado. Logo, como em Gauss-Seidel, estamos utilizando a última informação disponível para o cálculo da próxima iteração.

6.2 Método de Newton

Consideremos a matriz:

$$A(x) = \begin{bmatrix} a_{11}(x) & a_{12}(x) & \dots & a_{1n}(x) \\ a_{21}(x) & a_{22}(x) & \dots & a_{2n}(x) \\ \vdots & \ddots & \vdots & \vdots \\ a_{n1}(x) & a_{n2}(x) & \dots & a_{nn}(x) \end{bmatrix}$$

onde cada uma das entradas $a_{ij}(x)$ é uma função de \mathbb{R}^n em \mathbb{R} . Desta forma, $A(x)$ deve ser tal que:

$$G(x) = x - A(x)^{-1}F(x) \quad (24)$$

ofereça uma convergência quadrática para a solução $F(x) = 0$, assumindo que $A(x)$ é não singular no ponto fixo p de G .

Teorema 6.3. *Seja p uma solução de $G(x) = x$. Suponhamos que exista $\delta > 0$ tal que:*

- $\frac{\partial g_i}{\partial x_j}$ em $N_\delta = \{x \mid \|x - p\| < \delta\}$ para cada $i = 1 : n$ e $j = 1 : n$;
- $\frac{\partial^2 g_i}{\partial x_j \partial x_k}$ é contínua e $\left| \frac{\partial^2 g_i}{\partial x_j \partial x_k} \right| \leq M$ para alguma constante M , para qualquer $x \in N_\delta$ e cada $i = 1 : n$, $j = 1 : n$ e $k = 1 : n$;
- $\frac{\partial g_i(p)}{\partial x_k} = 0$ para cada $i = 1 : n$ e $k = 1 : n$.

Então, existe $\hat{\delta} < \delta$ tal que a sequência gerada por $x^{(k)} = G(x^{(k-1)})$ converge quadraticamente para p para qualquer escolha de $x^{(0)}$ e se $\|x - x^{(0)}\| < \hat{\delta}$. Ademais,

$$\|x^{(k)} - p\|_\infty \leq \frac{n^2 M}{2} \|x^{(k-1)} - p\|_\infty^2, \quad k \geq 1. \quad (25)$$

Consideremos que a matriz $A(x)$ seja não singular próxima à uma solução p de $F(x) = 0$. Sejam $b_{ij}(x)$ as entradas de $A(x)^{-1}$ na i -ésima linha e j -ésima coluna.

Para $G(x) = x - A(x)^{-1}F(x)$, temos $g_i(x) = x_i - \sum_{j=1}^n b_{ij}(x)f_j(x)$. Do teorema (6.3), temos que:

$$\sum_{j=1}^n b_{ij}(p) \frac{\partial f_j(p)}{\partial x_i} = 1, \quad k = i \quad (26)$$

$$\sum_{j=1}^n b_{ij}(p) \frac{\partial f_j(p)}{\partial x_k} = 0, \quad k \neq i \quad (27)$$

Seja $J(x)$ o jacobiano de F . Das condições (26) e (27), temos que:

$$A(p)^{-1}J(p) = I \Rightarrow A(p) = J(p) \quad (28)$$

Assim, podemos rescrever (24) da seguinte maneira:

$$G(x) = x - J(x)^{-1}F(x) \quad (29)$$

Desta forma, definimos o método de Newton para resolução de equações não lineares, com uma aproximação inicial $x^{(0)}$ apropriada, como segue:

$$x^{(k)} = G(x^{(k-1)}) = x^{(k-1)} - J(x^{(k-1)})^{-1}F(x^{(k-1)}) \quad (30)$$

O método de Newton fornece uma convergência quadrática desde que a aproximação inicial seja apropriada e o inverso do jacobiano seja não singular em p . Um dos grandes problemas deste método, no entanto, reside no fato de que a cada passo é necessário calcular $J(x)^{-1}$, o que representa um gasto computacional significativo. Assim, o custo computacional de uma iteração do método de Newton é de $n^2 + n$ (n avaliações da função F e n^2 derivadas parciais calculadas) mais operações aritméticas da ordem de n^3 para resolução do sistema linear. Além disto, se $J(x)^{-1}$ é singular, o método apresentará problemas na convergência.

6.3 Método de Broyden

O método de Broyden (um dos métodos da classe quase-Newton) propõe uma melhora no método de Newton. O método exige somente n avaliações de funções por iteração e o número de operações aritméticas é reduzida à ordem de n^2 . A desvantagem do método de Broyden, e dos métodos quase-Newton em geral, consiste na perda de convergência quadrática do método

de Newton, apresentando convergência superlinear.

O método da secante aproxima $f'(x_n)$ por:

$$b_n = \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}} \quad (31)$$

e, então, dá o passo:

$$x_{n+1} = x_n - b_n^{-1} f(x_n) \quad (32)$$

Para dimensões maiores, podemos realizar operações similares a (31) e (32), carregando uma aproximação para o jacobiano juntamente com uma aproximação para x^* e atualizando a aproximação para o jacobiano à medida que progredam as iterações. Desta maneira, chamemos de B_n a aproximação de $F'(x_n)$ que satisfazem a equação da secante:

$$B_n(x_n - x_{n-1}) = F(x_n) - F(x_{n-1}) \quad (33)$$

No método de Broyden, se x_n e B_n são as aproximações para a solução e para o jacobiano, respectivamente, então:

$$x_{n+1} = x_n - \lambda_n B_n^{-1} F(x_n) \quad (34)$$

onde λ_n é o tamanho do passo para a direção de Broyden:

$$d_n = -B_n^{-1} F(x_n) \quad (35)$$

Após o cálculo de x_{n+1} , B_n é atualizado para:

$$B_{n+1} = B_n + \frac{(y - B_n^{-1}s)s^T}{s^T s} \quad (36)$$

onde $y = F(x_{n+1}) - F(x_n)$ e $s = x_{n+1} - x_n = \lambda_n d_n$.

6.4 Exercícios

Exercícios referentes ao capítulo 10, seção 10.3, da ref. [11].

Exercício 7

Sistema não linear de equações a ser resolvido:

$$\begin{cases} 3x_1 - \cos(x_2x_3) - \frac{1}{2} = 0 \\ x_1^2 - 625x_2^2 - \frac{1}{4} = 0 \\ e^{-x_1x_2} + 20x_3 + \frac{10\pi - 3}{3} = 0 \end{cases}$$

Aplicando o método de Broyden obtemos:

$x_1 = 5.000000e - 01$, $x_2 = 3.050732e - 06$ e $x_3 = -5.235987e - 01$ com erro de $5.816597e - 09$ em 27 iterações.

Já com o método de Newton, os resultados obtidos foram:

$x_1 = 5.000000e - 01$, $x_2 = 7.451308e - 09$ e $x_3 = -5.235988e - 01$ com erro de $3.469447e - 14$ em 27 iterações.

De fato, o número de iterações é significativamente alto para um problema de apenas três dimensões. A convergência é lenta para ambos os métodos, por conta da singularidade da matriz jacobiana.

7 Conclusão

Neste projeto, foram explorados várias aplicações físicas da decomposição SVD, em especial, na compressão de imagens coloridas. Vimos que para comprimir este tipo de imagem, precisamos decompor a figura original nos três padrões RGB e aplicar a SVD a cada um deles. Para recompor novamente a imagem juntamos cada uma das informações obtidas na decomposição dos padrões RGB, resultando em uma figura final comprimida. Observamos também, que quanto maior o número de valores singulares adotados durante a compressão melhor será a qualidade da imagem final. Analisamos também, a relação entre a SVD e a decomposição espectral para casos em que as decomposições são equivalentes, quando temos matrizes simétricas e positivas definidas e para casos em que divergem. Para matrizes retangulares, por exemplo, a decomposição espectral não existe, enquanto que a SVD é para qualquer tipo de matriz.

Com o sumário realizado acerca da Equação de Sylvester, foram analisadas algumas de suas aplicações na Álgebra Linear Computacional. Neste contexto, destacamos a Lei de Inércia de Sylvester para o cálculo de autovetores de matrizes tridiagonais.

Com base nas referências [4], [7] e [8] foi possível provar a equivalência entre os métodos de iterações simultâneas e o algoritmo QR, demonstramos ainda a convergência deste último por meio da análise do comportamento dos coeficientes de Rayleigh.

Finalmente, foram abordados alguns métodos para resolução de equações homogêneas não lineares: o Método do Ponto Fixo, o Método de Newton e o Método de Broyden. Neste contexto, foram analisados o desenvolvimento destes métodos, suas respectivas complexidades computacionais e aplicações em alguns problemas propostos da ref. [11]. Percebeu-se que a grande desvantagem do método de Newton reside na necessidade do cálculo da jacobiana a cada iteração, bem como na possibilidade da inversa desta matriz ser singular. Já para o método de Broyden, ocorre a perda da convergência quadrática do método de Newton, que é substituída por uma convergência superlinear.

8 Referências Bibliográficas

- [1] DAVID S. WATKINS. Fundamentals of matrix computations - 3rd ed. – NY, John Wiley, 2010.
- [2] P. PULINO, Álgebra Linear e suas Aplicações: Notas de Aula, IMECC, UNICAMP, Janeiro de 2015, <http://www.ime.unicamp.br/~pulino/ALESA/>.
- [3] G.H.GOLUB and C.F. van LOAM, Matrix Computations - 3rd ed. - The Johns Hopkins University Press.
- [4] L. N. TREFETHEN and D. BAU III – Numerical Linear Algebra - SIAM, 1997.
- [5] Singular Value Decomposition, Investigation in Mathematics, Singular Value Decomposition, REBEKAH WHEELER, 10214927, 28th March 2013.
- [6] NICHOLAS J. HIGHAM, Sylvester’s Influence on Applied Mathematics (ISSN 1361-2042). Mathematics Today, 50(4). pp. 202-206 (2014).
- [7] DAVID S. WATKINS, Understanding the QR algorithm, SIAM REVIEW, Vol. 24, No. 4, pp. 427-440, 1982.
- [8] DAVID S. WATKINS, The QR Algorithm Revisited, SIAM REVIEW, Vol. 50, No. 1, pp. 133–145, 2008.
- [9] C. T. KELLEY, Iterative Methods for Linear and Nonlinear Equations, Philadelphia, PA - SIAM, 1995. Série (Frontiers in Applied Mathematics; Vol. 16).
- [10] C. T. KELLEY. Fundamentals of Algorithms Solving Nonlinear Equations with Newton’s Methods, Philadelphia - SIAM, 2003.
- [11] L. BURDEN e J. DOUGLAS FAIRES, Análise Numérica, revisão técnica Helena Maria Ávila de Castro. São Paulo, SP : Cengage Learning, 2013.