Introdução ao Cálculo Numérico

Fatoração LUP

Instituto de MATEMÁTICA E ESTATÍSTICA UFRGS

Prof. Rudnei Dias da Cunha

- Motivação
  - Existem situações em que é necessário resolver vários sistemas de equações lineares nos quais a matriz de coeficientes é a mesma, porém o termo independente muda a cada sistema
  - Tal situação surge, por exemplo, ao se calcular a inversa de uma matriz (por exemplo, em cálculos envolvendo autovalores e autovetores)
  - A inversa deve ser obtida valendo-se da igualdade

$$A^{-1}A = AA^{-1} = I$$

- Motivação
  - Escrevendo

$$AX = I,$$

onde $X = A^{-1}$, obteremos as colunas x_i da matriz A^{-1} resolvendo n sistemas de equações lineares da forma

$$A(x_i) = (e_i)$$

onde a i-ésima coluna da matriz identidade é o vetor canônico de ordem n,

$$e_i = (\underbrace{0, \dots, 0,}_{i-1} \quad 1, \quad 0, \dots, 0)^T$$

para $i = 1,2, \dots, n$.

- Motivação
  - Ora, se utilizássemos o método da eliminação de Gauss *para cada sistema*, visto anteriormente – e que funcionaria, assumindo que A admite inversa – o custo computacional para se obter A^{-1} seria da ordem de n^4, o que tornaria proibitivo tal processo mesmo para n relativamente pequeno

- Motivação
  - Além disso, parece ser um certo desperdício de operações o fato de reduzirmos a matriz A à forma triangular superior n vezes, através das operações elementares efetuadas na eliminação de Gauss, já que A é constante nesse caso – apenas os termos independentes é que mudam
  - Assim, para tornarmos aceitável o custo computacional para problemas desse tipo, vamos transformar A apenas uma vez, de tal forma que possamos facilmente resolver os n sistemas triangulares obtidos

- A fatoração LUP
  - O método da eliminação de Gauss, visto anteriormente, pode ser modificado de tal forma que a matriz A passa a ser expressa como o produto entre duas matrizes triangulares (uma, inferior e a outra, superior) e uma matriz de permutação:

$$PA = LU$$

onde L é uma matriz triangular inferior, U é uma matriz triangular superior e P é uma matriz de permutação

- A fatoração LUP
  - Cabe ressaltar que essa fatoração não tem uma forma única, pois depende de como são formados os fatores L e U
  - Vamos adotar aqui a chamada *forma de Doolittle*, na qual a matriz L tem diagonal unitária ($L_{i,i} = 1$, para $1 \leq i \leq n$)
  - A matriz de permutação P também deve ser analisada:

- A fatoração LUP
  - Uma matriz de permutação P é uma matriz cujas linhas são permutações das linhas da matriz identidade I
  - Evidentemente, $P^{-1}P = PP^{-1} = I$
  - Uma característica importante exibida por tais matrizes é que $P^{-1} = P^T$, de onde
$$P^T P = PP^T = I$$
  - Essas relações vão ser utilizadas para obter a solução de $Ax = b$ usando a fatoração LU como sendo a **solução de dois sistemas triangulares de equações**

- A fatoração LUP
  - Dado que A tenha sido fatorada como acima, podemos resolver o sistema $Ax = b$ escrevendo:

$$PA = LU$$

como $P^{-1} = P^T$, multiplicando por P^T, vem

$$(P^T P)A = P^T LU \therefore P^T P = I \therefore$$

$$A = P^T LU.$$

Substituindo na equação do sistema, vem

$$(P^T LU)x = b$$

como $PP^T = I$, multiplicando-a por P, obtemos

$$(PP^T LU)x = Pb$$

$$(PP^T)LUx = Pb$$

$$\boxed{LUx = Pb}$$

- A fatoração LUP
  - Dessa forma, com a fatoração $PA = LU$, obtemos o sistema $LUx = Pb$, o qual é equivalente ao sistema $Ax = b$
  - Para resolvê-lo, chamamos de y o produto Ux, de onde a solução do sistema $Ax = b$ é obtida resolvendo-se os dois sistemas triangulares

$$\boxed{\begin{array}{c} Ly = Pb \\ Ux = y \end{array}}$$

- Implementação da fatoração LUP
  - A matriz A é sobrescrita com os fatores L e U durante a fatoração
  - Os elementos de U são os elementos $A(p(i), j)$ obtidos no método da eliminação de Gauss, os quais são armazenados na porção **triangular superior** de A

- Implementação da fatoração LUP
  - Os elementos de L são os *multiplicadores* $z = A(p(i),k)/A(p(k),k)$ calculados durante a eliminação, e são armazenados na porção triangular inferior de A, i.e., os elementos $A(p(i),k)$
  - Como a diagonal de L é unitária (na forma de Doolittle), ela não precisa ser armazenada!

- Implementação da fatoração LUP
  - A matriz de permutação P é armazenada de forma compacta no vetor de permutação p, de tal forma que se $p(i) = j$, então a linha i foi trocada com a linha j
  - Esse vetor de permutação será usado depois na solução dos sistemas triangulares $Ly = Pb$ e $Ux = b$
  - Um algoritmo para a fatoração LU pode ser descrito como segue:

```
proc fatoracao_lup(entrada: n,A,b; saida: A,b,p,resultado)
  resultado:= 0
  ! Inicializa o arranjo de permutação e os valores de escalonamento
  para i:=1 até n faça
    p(i):= i
    s(i):= -MAXR
    para j:=1 até n faça
      s(i):= max(s(i),abs(A(i,j)))
    fim para
  fim para
  para k:=1 até n-1 faça
    ! Localiza o pivô, indicado pelo valor de j
    j:= k
    para i:=k+1 até n faça
      se abs(A(p(i),k))/s(p(i))>sabs(A(p(j),k))/s(p(j)) então
        j:= i
        para
      fim se
    fim para
    ! Indica a troca entre as linhas k e j
    q:= p(k) ; p(k):= p(j) ; p(j) = q
```

```
    se A(p(k),k))/=0.0 então
      para i:=k+1 até n faça
        z:= A(p(i),k)/A(p(k),k)
        A(p(i),k):= z
        para j:=k+1 até n faça
          A(p(i),j):= A(p(i),j)-z*A(p(k),j)
        fim para
      fim para
    senão
      resultado:= -p(k)
      retorna
    fim se
  fim para
  se A(p(n),n)==0.0 então
    resultado:= -p(n)
    retorna
  fim se
end proc
```

Armazenamento dos elementos de L (i.e. multiplicadores z) na porção estritamente triangular inferior de A

Armazenamento dos elementos de U na porção triangular superior de A

- Implementação da fatoração LUP
  - Observe que dois sistemas triangulares devem ser resolvidos após a fatoração:
    - Para o sistema $Ly = Pb$, utiliza-se o método da substituição direta, sem a necessidade de se fazer divisões pelos elementos da diagonal de L
    - O sistema $Ux = y$ é resolvido utilizando-se o método da retrossubstituição

•Implementação da fatoração LUP

```
proc resolve_sistema_lup(entrada: n,A,b,p; saida: x)
  // Resolve Ly=Pb
  para i:=1 até n faça
    s:= 0.0
    para j:=1 até i-1 faça
      s:= s+A(p(i),j)*y(j)
    fim para
    y(i):= b(p(i))-s
  fim para
  // Resolve Ux=y
  para i:=n até 1 por -1 faça
    s:= 0.0
    para j:=n até i+1 por -1 faça
      s:= s+A(p(i),j)*x(j)
    fim para
    x(i):= (y(i)-s)/A(p(i),i)
  fim para
fim proc
```

- Análise da quantidade de multiplicações realizadas
  - Observe que o algoritmo da fatoração LUP é igual ao da eliminação gaussiana, sem a resolução do sistema triangular de equações, logo,

$$c_{\times}^{(LU)} = \frac{1}{3}(n^3 - n)$$

  - A quantidade de multiplicações realizadas para resolver o sistema $Lx = Pb$ é igual à do sistema $Ux = y$,

$$c_{\times}^{(t)} = \frac{1}{2}(n^2 - n)$$

- Análise da quantidade de multiplicações realizadas
  - Retomando o nosso problema motivador – calcular a inversa de A – vemos que a fatoração LUP pode ser calculada uma vez apenas e, após, são resolvidos $2n$ sistemas triangulares, de onde a quantidade de multiplicações para calcular A^{-1} é dada por

$$c_{\times}{}^{(A^{-1})} = c_{\times}{}^{(LU)} + 2nc_{\times}{}^{(t)}$$
$$= \frac{1}{3}(n^3 - n) + n^3 - n^2 \cong \frac{4}{3}n^3 \ll n^4$$

para n grande.

- Refinamento iterativo
  - O processo de *refinamento iterativo* consiste em adicionar ao vetor solução do sistema $Ax = b$, obtido através da resolução dos sistemas $Ly = Pb$ e $Ux = y$, um fator de correção, obtido com o *dobro da precisão numérica* com a qual foi calculado x
  - Como qualquer operação aritmética em ponto-flutuante é feita na presença de erros de arredondamento, a solução obtida não é x, mas uma aproximação numérica para ela, $\tilde{x}$

- Refinamento iterativo
  - O erro obtido em se calcular $\tilde{x}$ pode ser medido pelo *resíduo*

$$r = b - A\tilde{x}$$

  - Isolando $A\tilde{x}$ na equação acima e subtraindo de $Ax = b$, vem

$$Ax - A\tilde{x} = b - (b - r)$$

ou

$$Ae = r$$

onde $e = x - \tilde{x}$ é o vetor *erro*.

- Refinamento iterativo
  - A solução de $Ae = r$ é obtida usando-se a fatoração LUP de A e a correção e é adicionada a $\tilde{x}$, obtendo $x = \tilde{x} + e$
  - Esse processo de correção pode ser repetido tantas vezes quantas forem necessárias até que o resíduo seja pequeno o suficiente, levando a um processo iterativo de melhoria da solução do sistema com um custo de ordem n^2, como mostrado a seguir:

- Refinamento iterativo

1. Calcula a fatoração LUP de A
2. Calcula a solução inicial $\tilde{x}$ através de $Ly = Pb, U\tilde{x} = y$
3. Calcula o resíduo $r = b - A\tilde{x}$ com o **dobro de precisão numérica**
4. Calcula a correção e através de $Ly = Pr, Ue = y$
5. Calcula $x = \tilde{x} + e$
6. Armazena $\tilde{x}$ em x
7. Repete os passos 3 a 6 até que $||r|| < \epsilon$, onde $\epsilon \ll 1$ é a tolerância especificada