# SME0305 - 2016

**Roberto Ausas/Gustavo Buscaglia**

---

**Lista 5 - Métodos diretos para resolver sistemas lineares**

1. Ler o capitulo 5 do livro de Quarteroni, paginas 137 até 158.

2. Testar em Octave criando matrizes, p.e. de $5 \times 5$ e verificar as propriédades:

   (a) A inversa de uma matriz triangular inferior unitaria (i.e., $l_{ii} = 1 \; \forall i, l_{ij} = 0 \; \forall i < j$) também é triangular inferior unitaria (Pode usar a função `inv(L)` de Octave para verificar).

   (b) Considere por exemplo uma matriz triangular inferior unitaria da forma

   $$L = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ l_{21} & 1 & 0 & 0 & 0 \\ l_{31} & 0 & 1 & 0 & 0 \\ l_{41} & 0 & 0 & 1 & 0 \\ l_{51} & 0 & 0 & 0 & 1 \end{pmatrix}$$

   ou outras similares, em que aparecem elementos que poderiam ser não nulos $l_{ij}$ embaixo da diagonal apenas numa das colunas (no exemplo seria $l_{ij} \neq 0 \; \forall \, i = 2, \ldots, 5, \; j = 1$.

   - A inversa tem a mesma forma, mas com os $l_{ij}$ trocados de sinal.
   - Como é o produto de duas delas quando os $l_{ij}$ aparecem em colunas diferentes?.
   - Pegue uma matriz $A$ e multiplique a esquerda pela inversa de uma dessas $L$.

   (c) Considere uma matriz de permutação $P$ (i.e., uma matriz obtida aplicando permutações de filas da matriz identidade).

   - A sua inversa é a trasposta.
   - As mesmas permutações podem ser aplicadas numa matriz $A$ premultiplicando pela $P$.

3. Sejam $A, L_i, P_i \in \mathbb{R}^{n\times n}$, $i = 1, 2$. Provar que $L_2P_2L_1P_1A = L_2(P_2L_1P_2^{-1})(P_2P_1)A$.

4. Considere matrizes quadradas de $n \times n$.

   (a) Como se calcula o determinante de matrizes triangulares inferiores ou triangulares superiores?

   (b) Tendo a decomposição $LU$ de uma matriz $A$ e conhecendo as propriedades dos determinantes, como calcularia o determinante de $A$?

   (c) Que acontece no caso anterior se o que temos é a decomposição $LU$ de $PA$ .

5. Programar em Octave algum dos algoritmos de *Forward substitution* ou *Backward substitution* para resolver sistemas triangulares.

6. Programar em Octave o algoritmo de decomposição $LU$ visto na aula para matrizes em que não seja necessário aplicar permutações, p.e., as matrizes simétricas e definidas positivas, que podem ser geradas com as intruções na sequência:

   ```
   > R = rand(n);
   > A = R'*R;
   ```

   Use a mesma matriz $A$ para guardar os fatores $L$ e $U$. Comparar o resultado com Octave, fazendo `[L, U, P] = lu(A)`.

7. Considerar as redes hidráulicas de um bairro de $n \times n$ nós. Fazer um programa de Octave para medir o tempo necessário para resolver o sistema de equações como função do número de vertices na rede. Plotar o resultado utilizando a função `loglog`. Fazer redes desde $20 \times 20$ até $100 \times 100$ pulando de a 20 por exemplo.

   Repetir o anterior, mas convertindo a matriz para o formato esparso, i.e., na hora de resolver colocar:

   ```
   > p = sparse(Atilde) \ b;
   ```

---

8. Responder com verdadeiro ou falso:

   (a) Para um sistema de equações $A\mathbf{x} = \mathbf{b}$, $A \in \mathbb{R}^{n\times n}$ que admite solução única se verifica que as linhas de $A$ são linearmente dependentes.

   (b) Para um sistema de equações $A\mathbf{x} = \mathbf{b}$, $A \in \mathbb{R}^{n\times n}$ que admite solução única se verifica que $A\mathbf{x} = 0$ só se $\mathbf{x} = 0$.

   (c) Se premultiplicar (ou seja, multiplicar a esquerda) uma matriz $A$ por uma matriz de permutação $P$, então o resultado será uma matriz com as mesmas linhas que $A$, mas numa ordem diferente.

   (d) Uma matriz de permutação, obtida trocando a ordem apenas de duas linhas da matriz identidade, é simétrica.

   (e) Conhecida a fatoração $LU$ de $A$, podemos calcular o deteteminante dela como $\det(A) = \det(L)$.

   (f) Uma vez calculada a fatoração $LU$ de uma matriz $A \in \mathbb{R}^{n\times n}$, o determinante dela pode ser calculado fazendo

   ```
   for i=1:n
     det=1;
     det=det*U(i,i);
   end
   ```

   (g) O custo computacional de calcular o produto escalar de dois vetores é da ordem $\mathcal{O}(n^2)$.

   (h) O custo computacional de calcular o produto de uma matriz por um vetor é da ordem $\mathcal{O}(n^3)$.

   (i) O custo computacional de calcular a fatoração $LU$ de uma matriz, $A \in \mathbb{R}^{n\times n}$ é proporcional $n^3$.

(j) O custo computacional de aplicar o método de substituição progressiva (Forward substitution) para resolver um sistema triangular inferior de $n \times n$ é proporcional a $n$.

(k) Uma vez obtida a fatoração LU de uma matriz $A \in \mathbb{R}^{n \times n}$, para resolver qualquer sistema da forma

$$A\,\mathbf{x} = \mathbf{b}$$

são necessárias $\mathcal{O}(n)$ operações.

(l) O pivotamento ajuda na precisão minimizando o erro de arredondamento.

(m) Sabemos que na fatoração LU com pivotamento, são calculadas matrizes $P$, $L$ e $U$ tais que $P\,A = L\,U$. Ainda, a matriz de permutação $P$ é sempre simétrica.

(n) Sabemos que na fatoração $LU$ com pivotamento, são calculadas matrizes $P$, $L$ e $U$ tais que $P\,A = L\,U$. Ainda, os elementos da matriz de permutação $P$ são 0 ou 1.

(o) Se uma matriz tem determinante negativo, então ela não pode ser fatorada da forma

$$A = H\;H^T$$

(p) O custo computacional de calcular a fatoração de Cholesky para uma matriz $A \in \mathbb{R}^{n \times n}$ é proporcional a $n^3$ (i.e., #flops $\sim Cn^3$). Essa constante $C$ é menor que no caso da fatoração $LU$.

(q) A decomposição de Cholesky pode ser aplicada em qualquer matriz simétrica.