Universidade Federal do Espírito Santo
Departamento de Informática / Programa de Pós Graduação em Informática
1º **Exercício Computacional de Algoritmos Numéricos II / Computação Científica - 2020/1**
Sistemas Lineares usando o Octave

Objetivos

- Observar o comportamento dos métodos diretos e iterativos quanto as características da matriz dos coeficientes.

Conceitos/comandos importantes:

- A coleção de matrizes esparsas *SuiteSparse Matrix Collection*[1] disponibiliza uma variedade de matrizes esparsas oriundas das mais deversas áreas do conhecimento. Um dos formatos disponíveis para as matrizes é <nome>.mat. Arquivo binário que armazena as informações para gerar uma matriz esparsa no formato *Compressed Column Sparse*(CCR). Os principais comandos do Octave relativos a obtenção da matriz esparsa `A` são:
  - `load <nome>.mat` (carrega dados da matriz em uma estrutura auxiliar `A`
  - `AA = Problem.A` (Armazena os dados da estrutura `A` na matriz esparsa `AA` no formato CCR)
  - `nnz(A)` (obtem o número de coeficientes não nulo da matriz `A`)
  - `n = rows(A);` (obtem o número de linhas da matriz `A`)
- Os métodos diretos são exatos a menos de erros de ponto flutuante cometidos no processo de transformar o sistema original em um sistema trivial. A seguir alguns importantes comando do Octave na solução de sistemas lineares via Métodos Diretos:
  - $\mathtt{x} = \mathtt{A}\backslash\mathtt{b}$ (resolve o sistema linear por Eliminação de Gauss)
  - $\mathtt{r} = \mathtt{b} - \mathtt{A} * \mathtt{x}$ (calcula o resíduo da solução aproximada encontrada)
  - Métricas importantes para avaliação da solução aproximada: $||x_e - x||_*$ e $||r||_*$ (x_e solução exata) $\mathtt{norm}(\mathtt{x}, *)$ (obtem a norma $*$ do vetor x - por exemplo, pode ser a norma euclidiana $* = 2$ ou a norma do máximo $* = \mathtt{inf}$)
- Os métodos diretos são bem eficientes para matrizes de pequeno porte. Entretanto, o processo de solução prevê preenchimento de posições originalmente nulas (fill-in), aumentando assim o número de operações de ponto flutuante. Principais comandos do Octave:
  - $[\mathtt{L}, \mathtt{U}, \mathtt{P}] = \mathtt{lu}(\mathtt{A})$ (obtem os fatores L, U e P)
  - $\mathtt{spy}(\mathtt{A})$ (obtem a esparsidade da matriz A)
- Uma matriz é dita mal-condicionada se:

$$cond(A) = \|A\|_* \|A^{-1}\|_* \quad \text{for um valor expressivamente elevado}$$

  Comandos do octave:
  - $\mathtt{cond}(\mathtt{A})$
  - $\mathtt{norm}(\mathtt{A}, *)$ (obtem a norma $*$ da matriz A - por exemplo, pode ser a norma por coluna $* = 1$ ou a norma por linha $* = \mathtt{inf}$)

[1]https://sparse.tamu.edu/

- Os métodos iterativos dependem de critérios de convergência (critério necessário e suficiente):

$$x^{(k+1)} = M\,x^{(k)} + c \text{ converge } \Leftrightarrow \rho(M) < 1$$

onde $\rho(M)$ é o maior módulo dos autovalores de M.

- Dado $Ax = b$, os métodos iterativos estacionários convergem se a matriz dos coeficientes A for diagonal dominante (ou satisfazer o Critério das linhas - critério suficiente):

$$|a_{ii}| > \sum_{j=1, j\neq i}^{n} |a_{ij}| \quad \forall i = 1, \ldots, n$$

- `tril(A,-1)` (obtem a submatriz estritamente inferior de `A`)
- `triu(A,1)` (obtem a submatriz estritamente superior de `A`)
- `diag(diag(A))` (obtem a matriz diagonal de `A`)
- `[V lambda]=eig(A)` (obtem os autovetores `V` e os autovalores `lambda` de `A`)
- `max(abs(diag(lambda)))` (obtem o maior valor em módulo dos elementos da diagonal de `lambda`)
- `plot(x,y)` (plota o grafico dos pontos de $y_i = f(x_i)$)
- `x = [1:n]` (gera o vetor $x = [1\ 2\ 3\ \ldots\ n]$)
- `log(x)` (calcula o logarítmo de x)

Exercícios propostos

1. **Métodos Diretos**:

   O objetivo dos execícios a seguir é observar o comportamento de matrizes esparsas na solução de sistemas lineares via métodos diretos e iterativos estacionários. Faça download de matrizes esparsas de ordem $n = 10^2, 10^3, 10^4$ na *SuiteSparse Matrix Collection* que sejam quadradas e inversíveis. Para cada uma das matrizes:

   (a) Recupere as matrizes esparsas a partir do arquivo `.mat`
   (b) Obtenha os fatores L, U e P utilizando a função `[L,U,P]=lu(A);`
   (c) Observe a configuração de esparsidade das matrizes A, L e U.
   (d) Calcule a taxa de preenchimento: $\mathtt{100} - \left(\frac{\mathtt{nnz(A)}}{\mathtt{nnz(L)+nnz(U)}}\right) * \mathtt{100}$.
   (e) Calcule a solução do sistema linear onde $\mathtt{b} = \mathtt{A} * \mathtt{ones}(\mathtt{n}, 1)$, através de $\bar{\mathtt{x}} = \mathtt{A}\backslash\mathtt{b}$.
   (f) Calcule a distância relativa entre a solução exata e a solução aproximada $\frac{||\delta\mathtt{x}||_\infty}{||\bar{\mathtt{x}}||_\infty}$
   (g) Calcule a distância relativa entre a matriz original e a matriz resultante da decomposição LU: $\frac{||\delta\mathtt{A}||_\infty}{||\mathtt{A}||_\infty}$; $\delta A = A - P * L * U$.
   (h) Calcule a distância relativa entre o vetor dos termos independentes original e o vetor resultante da decomposição LU: $\frac{||\delta\mathtt{b}||_\infty}{||\bar{\mathtt{b}}||_\infty}$
   (i) Calcule a norma do resíduo: através de $||\mathtt{r}||_\infty = \mathtt{norm}(\mathtt{b} - \mathtt{A} * \mathtt{x}, \mathtt{inf})$.
   (j) Calcule o número de condicionamento da matriz: $\mathtt{K} = \mathtt{cond}(\mathtt{A})$.

Monte uma tabela contendo as informações observadas e calculadas de cada matriz. Faça um estudo sobre o comportamento de cada matriz com relação ao preenchimento no processo de decomposição e seu condicionamento.

2. **Métodos Iterativos**:

   O objetivo dos execícios a seguir é observar o comportamento de matrizes esparsas na solução de sistemas lineares via métodos iterativos. Faça download de matrizes esparsas de ordem $n = 10^2, 10^3, 10^4$ na *SuiteSparse Matrix Collection* que sejam quadradas e inversíveis. Procure obter matrizes capazes de explorar as variadas características de convergência dos métodos iterativos. Para cada uma das matrizes:

   (a) Recupere a matriz esparsa a partir do arquivo `.mat`
   (b) Defina o vetor dos termos independentes `b = A*ones(n,1)`;
   (c) Verifique se a matriz é diagonal dominante;
   (d) Calcule o raio espectral da matriz de iteração dos métodos Jacobi, Seidel e SOR(w) para cada uma das matrizes obtidas.
   (e) Obtenha a solução pelos métodos Jacobi, Seidel e SOR se o raio espectral da matriz de iteração do método for menor que 1.0 para um conjunto de parâmetros `w` do método SOR. Estabeleça uma tolerância adequada para cada matriz.
   (f) Escolha o `w` que obtem o melhor comportamento para o SOR e faça o gráfico $iter \times log(er)$ dos três métodos no mesmo sistema de eixos.
   (g) O que podemos dizer sobre a convergência dos métodos Jacobi, Seidel e SOR(w) para cada uma das matrizes?

   Considere as seguintes funções já implementadas no material de apoio.

   - `[x,er,iter]=jacobi(A,b,tol,nmaxiter)`, sendo `tol` a tolerância pré estabelecida; `nmaxiter` o número máximo de iterações; `er` vetor contendo o erro relativo em cada iteração; `iter` número de iterações necessárias para convergir.
   - `[x,er,iter]=sor(A,b,tol,nmaxiter,w)`, sendo `tol` a tolerância pré estabelecida; `nmaxiter` o número máximo de iterações; `er` vetor contendo o erro relativo em cada iteração; `iter` número de iterações necessárias para convergir; `w` parâmetro de relaxação (`w = 1`, método Seidel).
   - `[MJ,MS,MSOR]=fatora(A,w)`, sendo `MJ`, `MS`, `MSOR`, matrizes dos métodos Jacobi, Seidel e SOR, respectivamente.
   - `diagonal_dominante(A)`, retorna `TRUE` or `FALSE` caso a matriz seja ou não diagonal dominante

Relatório

Escreva um relatório com suas conclusões sobre os objetivos listados acima. Entregar uma cópia em pdf via email (luciac@inf.ufes.br)até 26/03/2020. O título do email deve ser AN2201-EXE1-<Nome1Ultimosobrenome1-Nome2Ultimosobrenome2>, para os alunos de Algoritmos Numéricos II; e CC201-EXE1-<Nome1Ultimosobrenome1-Nome2Ultimosobrenome2> para os alunos de Computação Científica.