$1^{\underline{o}}$ **Trabalho de Algoritmos Numéricos I - 19/1**
Sistemas Lineares e Ajuste de Curvas usando o Octave - DI/UFES

Objetivos:

- Observar o comportamento dos métodos diretos e iterativos estacionários quanto as características da matriz dos coeficientes.
- Observar o comportamento do método dos Quadrados Mínimos.

Relatório:

Para cada exercício, faça um *script*, colocando os enunciados, observações e conclusões como saídas no terminal, seguindo o modelo fornecido no repositório. Entregar os arquivos via email (avalli@inf.ufes.br) até 21/05/2019. O título do email deve ser Trab1<nome1><nome2>. Para os alunos da física e da matemática, os grupos podem ser de no máximo três alunos. Para as outras turmas, grupos de no máximo dois alunos.

Conceitos/comandos importantes:

- Uma matriz é dita mal-condicionada se $\|A\|\|A^{-1}\|$ for um valor expressivamente elevado. Observe que no Octave as contas são feitas usando precisão dupla (comando *eps* fornece a precisão). Comandos:
  - $\texttt{cond(A)} = \|\texttt{A}\|\|\texttt{A}^{-1}\|$
  - $\texttt{norm}(\texttt{x}, *)$ = norma $*$ do vetor x (a norma euclidiana $* = 2$ ou a norma do máximo $* = \texttt{inf}$)
  - $\texttt{n} = \texttt{rows(A)}$; $\texttt{b} = \texttt{A} * \texttt{ones(n, 1)}$ (cria um vetor de tamanho `n` contendo a soma das linhas de `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. Comandos:
  - $\texttt{x} = \texttt{A}\backslash\texttt{b}$ (resolve o sistema linear por Eliminação de Gauss)
  - $\texttt{r} = \texttt{b} - \texttt{A} * \texttt{x}$ (calcula o resíduo da solução aproximada encontrada)
- 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. Comandos:
  - $[\texttt{L}, \texttt{U}, \texttt{P}] = \texttt{lu(A)}$ (obtem os fatores L, U e P)
  - `spy(A)` (obtém a esparsidade da matriz A)
  - $nnz(A)$ (retorna o número de elementos não nulos da matriz A)
- Os métodos iterativos dependem de critérios de convergência:

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

  onde $\rho(M)$ é o maior módulo dos autovalores de M. Comandos do Octave:
  - `[V lambda]=eig(A)` (obtem os autovetores `V` e os autovalores `lambda` de `A`)
  - `max(abs(diag(lambda)))` (obtém o maior valor em módulo dos elementos da diagonal de `lambda`)
- Ajuste de curvas. Comandos do Octave:
  - `p = polyfit(x,y,m)` (ajustar um polinômio de grau m a tabela de dados (x, y))
  - `y = polyval(p,x))` (avaliar o polinômio p em um ponto x)

- Modelo apresentação em *https : //inf.ufes.br/ avalli/download/algoritmos_numericosI/trab1/exemplo/*
- Algumas das funções em *https : //inf.ufes.br/ avalli/download/algoritmos_numericosI/trab1/codigos/*
  - `[MJ, MS, MSOR] = fatora(A, w)`
  - `[x, iter, res] = jacobi(A, b, tol, nmaxiter)`
  - `[x, iter, res] = sor(A, b, tol, nmaxiter, w)`
  - `[r2, var] = qualidade_ajuste(x, y, m, pol)`
  - `r2 = coef_determinacao(yData, yfit)`
- A coleção de matrizes esparsas *SuiteSparse Matrix Collection*[1] disponibiliza uma variedade de matrizes esparsas. 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) para o Octave. Comandos:
  - `load` <nome>`.mat` (carrega dados da matriz em uma estrutura auxiliar `A`)
  - `A = Problem.A` (armazena os dados da estrutura `A` na matriz esparsa `A` no formato CCR)

Sistemas Lineares

- Faça download dos arquivos `bcsstk01.mat`, `fs_183_3.mat`, `hor_131.mat`, `rail_5177.mat`, `orsirr_1.mat`, `plat362.mat` disponíveis na *SuiteSparse Matrix Collection* (*https : //sparse.tamu.edu/*) e recupere as matrizes esparsas a partir do arquivo `.mat`, utilizando os comandos `load` <nome>`.mat` e `A = Problem.A`.
- Resolva os exercícios a seguir para três matrizes escolhidas entre as sugeridas. Para cada exercício, faça um *script*, colocando os enunciados, observações e conclusões como saídas no terminal, seguindo o modelo fornecido no repositório. Pode usar as funções que disponibilizei no repositório, modificar ou fazer novas funções, se preferir.

1. O objetivo deste exercício é observar o comportamento dos **métodos diretos**, para cada uma das matrizes escolhidas.
   (a) Obtenha os fatores L, U e P utilizando a função `[L,U,P]=lu(A)` e observe a configuração de esparsidade das matrizes A, L e U. O que podemos observar com relação ao preenchimento no processo de decomposição LU?
   (b) Calcule a solução do sistema linear onde `b = A*ones(n, 1)`, através de `x = A\b`, e a norma do máximo relativa do resíduo usando `norm((b − A * x)/b, inf)`.
   (c) Calcule o número de condicionamento das matrizes escolhidas, utilizando a função `cond(A)`. O que podemos dizer sobre a qualidade da solução encontrada?

2. O objetivo desse exercício é observar o comportamento dos **métodos iterativos** estudados, para cada uma das matrizes escolhidas. Considere nos testes tolerância $\epsilon = 0.00001$, número máximo de iterações $niter = 10000$, parâmetro $\omega \in (0, 2)$ e `b = A * ones(n, 1)`. Para as três matrizes escolhidas, faça:
   (a) Avalie o comportamento dos processos iterativos de cada um dos métodos, Jacobi, Seidel (função SOR com $w = 1$) e SOR utilizando a função `fatora`. Faça algumas escolhas do parâmetro ω com o objetivo de diminuir o número de iterações do método SOR.
   (b) Calcule a solução de cada sistema pelos métodos Jacobi, Seidel (função SOR com $w = 1$) e SOR para o melhor valor de w encontrado.

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

(c) Faça o gráfico da norma do máximo relativa do resíduo ($res(k) = ||b - Ax^k||_\infty / ||b||_\infty$) com relação as iterações, utilizando os dados calculados nas funções `jacobi.m` e `sor.m`. No mesmo sistema de eixos plote o gráfico dos resíduos relativos para os métodos que convergiram.

(d) Comente suas conclusões sobre o comportamento das matrizes para métodos diretos e iterativos.

Ajuste de Curvas

O objetivo desse exercício é usar regressão polinomial e ajuste não-linear, pelo **método dos quadrados mínimos**, para ajustar polinômios no Octave e resolver aplicações. Para cada exercício, faça um *script*, colocando os enunciados, observações e conclusões como saídas no terminal, seguindo o modelo fornecido no repositório. Pode usar as funções que disponibilizei no repositório, modificar ou fazer novas funções, se preferir.

1. Os dados a seguir representam representam o tempo (T), em segundos, de congelamento para um certo volume (V) de uma substância. Use a regressão para determinar um modelo para prever T como uma função de V. Tente várias possibilidades - linear, parabólica, etc. Estime o tempo de congelamento usando 2.8 volumes. Mostre o estudo feito para a regressão polinomial, imprimindo a tabela contendo r^2 e σ. Mostre, em um mesmo gráfico, os pontos da tabela e a curva do melhor ajuste.

V	2.65	2.65	2.7	2.7	2.75	2.75	2.85	2.85	2.90	2.90	2.95	2.95	3.00	3.00
T	6.85	6.80	6.70	6.30	6.33	6.20	5.90	5.82	5.80	5.80	6.15	6.00	6.30	6.15

2. Vamos utilizar dois modelos de Michaelis-Menten [1] para analisar o crescimento de uma bactéria v como uma função da concentração de oxigênio $[S]$, descritos pelas equações:

$$Caso\,1: \quad v = \frac{v_m[S]}{k_s + [S]} \tag{1}$$

$$Caso\,2: \quad v = \frac{v_m[S]^2}{k_s^2 + [S]^2} \tag{2}$$

onde v_m é o crescimento máximo da bactéria e k_s é a constante representando a metade do crescimento máximo, como mostrado no gráfico abaixo. As equações descrevem uma relação que se estabiliza com o aumento de $[S]$, onde a equação (2) representa um modelo de segunda ordem.

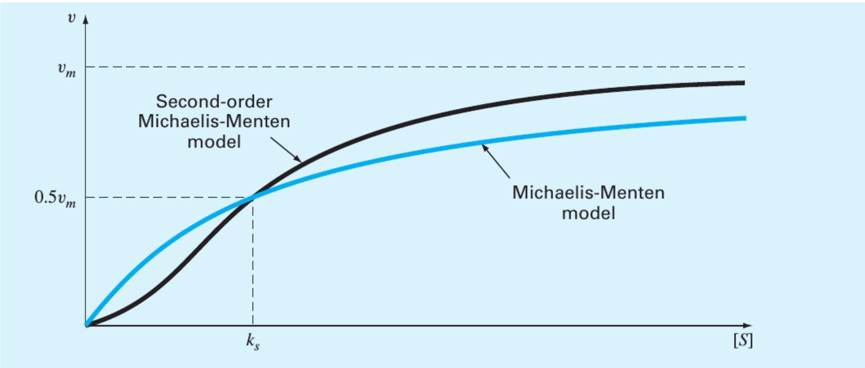

Figura 1: Duas versões do modelo de Michaelis-Menten para cinética enzimática.

Use o método dos quadrados mínimos para ajustar os dados da tabela abaixo com versões linearizadas das Equações (1) e (2). Além de estimar os parâmetros dos modelos, avalie a qualidade dos ajustes através de medidas estatísticas e gráficos dos ajustes.

$[S]$	1.3	1.8	3	4.5	6	8	9
v	0.07	0.13	0.22	0.275	0.335	0.35	0.36

Para isso, faça:

(a) Determine os coeficientes dos ajustes e recupere as equações dos modelos originais. Estime a taxa de crescimento em $[S] = 7$.

(b) Calcule r^2 para os modelos linearizados e originais.

(c) Faça os gráficos das soluções linearizadas ($Caso\,1:\ 1/[S] \times 1/v$ e $Caso\,2:\ 1/[S]^2 \times 1/v$) e originais ($[S] \times v$). Mostre em um mesmo gráfico a curva do ajuste junto com os pontos tabelados.

(d) Analise qual caso fornceceu um ajuste mais adequado, baseado nos valores estimados para $[S] = 7$, medidas estatísticas e gráficos dos ajustes.

Bibliografia

[1] Steven C. Chapra, Applied Numerical Methods with MATLAB for Engineers and Scientists, 2012.