Lab 5: Método dos Mínimos Quadrados

INF1608 – Análise Numérica

Leonardo Quatrin Campagnolo
lquatrin@tecgraf.puc-rio.br
Departamento de Informática, PUC-Rio

24 de Abril de 2025

Para este exercício, considere a representação de matrizes por vetor de ponteiros do Lab 0 e o método de solução de sistemas lineares do Lab 3. **Se usar os códigos dos laboratórios anteriores, envie suas implementações junto com a solução deste laboratório para a correção.** Se preferir, você pode copiar as funções necessárias já existentes para o código deste exercício, dentro do arquivo "mmq.c".

Problema: Podemos resolver um sistema inconsistente na forma $A_{m\times n}x_n = b_m$, com $m > n$, através do Método dos Mínimos Quadrados (MMQ). Na sua forma mais direta, a solução do MMQ é feita resolvendo o sistema linear $n \times n$ definido pelo sistema de equações normais:

$$A^T A\, \bar{x} = A^T b$$

onde A^T representa a matriz transposta de A e $\bar{x}$ a solução aproximada do problema. O erro do método pode ser avaliado pelo vetor residual $r = b - A\bar{x}$. Como métrica de erro, iremos usar a norma-2 desse vetor:

$$e = \|r\|_2 = \sqrt{\sum_{i=1}^{m} r_i^2}$$

1. A partir disso, pede-se para que realize as seguintes implementações no arquivo "mmq.c":

   (a) Implemente uma função que resolva o sistema $A_{m\times n}x_n = b_m$ pelo método dos mínimos quadrados. A função também recebe como parâmetro o vetor $\bar{x}$, já alocado com dimensão n, que deve ser preenchido com a solução aproximada. A função deve retornar a norma-2 do vetor residual.

   ```
   double mmq (int m, int n, double** A, double* b, double* x);
   ```

   Sua solução não deve mudar o conteúdo da matriz A. Para isso, crie duas matrizes dentro do método *mmq*, uma para guardar a transposta de A^T, e outra para guardar a multiplicação $A^T A$.

   (b) Usando a função do item anterior, implemente uma função que ajuste uma parábola, $y = a + bx + cx^2$, a um conjunto de n pontos (px_i, py_i) fornecido. A função deve determinar os coeficientes a, b, e c, preenchendo os endereços respectivos recebidos, e retornar a norma-2 do vetor residual.

```
double ajuste_parabola (int n, double* px, double* py,
                        double* a, double* b, double* c);
```

(c) Similar ao item anterior, implemente uma função que ajuste uma cúbica, $y = a + bx + cx^2 + dx^3$, a um conjunto de n pontos (px_i, py_i) fornecido.

```
double ajuste_cubica (int n, double* px, double* py,
                      double* a, double* b, double* c, double* d);
```

(d) Por fim, implemente uma função que ajuste uma exponencial, $y = ae^{bx}$, a um conjunto de n pontos (px_i, py_i) fornecido. Nesse caso, deve ser aplicada a linearização no modelo antes de calcular o ajuste. Neste caso, lembre-se que o erro é calculado no espaço ln.

```
double ajuste_exponencial_exp (int n, double* px, double* py,
                               double* a, double* b);
```

2. Para testar, complete o arquivo main com os seguintes experimentos:

(a) Para a implementação feita na questão 1(a), avalie os seguintes sistemas a partir do Método de Mínimos Quadrados:

i. $$\begin{bmatrix} 3 & -1 & 2 \\ 4 & 1 & 0 \\ -3 & 2 & 1 \\ 1 & 1 & 5 \\ -2 & 0 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 10 \\ 10 \\ -5 \\ 15 \\ 0 \end{bmatrix}$$

ii. $$\begin{bmatrix} 4 & 2 & 3 & 0 \\ -2 & 3 & -1 & 1 \\ 1 & 3 & -4 & 2 \\ 1 & 0 & 1 & -1 \\ 3 & 1 & 3 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 10 \\ 0 \\ 2 \\ 0 \\ 5 \end{bmatrix}$$

(b) Para as implementações feitas nas questões 1(b) e 1(c), ajuste os conjuntos de pontos por uma parábola e por uma cúbica. Você pode verificar a diferença de erro gerado entre as duas abordagens.

i. $(-1, 1),\ (0, 0),\ (1, 0),\ (2, -2)$
ii. $(0, 0),\ (1, 3),\ (2, 3),\ (5, 6)$
iii. $(1, 2),\ (3, 2),\ (4, 1),\ (6, 3)$

(c) Para a questão 1(d), calcule o ajuste do modelo exponencial (Ex 4.8 do livro). Você pode testar de duas formas: primeiro considerando o ano começando em 1950, e depois aplicar o deslocamento de -1950 para a coluna *year*.

year	cars ($\times 10^6$)
1950	53.05
1955	73.04
1960	98.31
1965	139.78
1970	193.48
1975	260.20
1980	320.39

Agrupe os protótipos das funções pedidas em um módulo "mmq.h" e as implementações em um módulo "mmq.c". Escreva um outro módulo "main.c" para o código de teste da sua implementação.

Entrega: O código fonte deste trabalho (isto é, os arquivos "mmq.h", "mmq.c" e "main.c", *e eventuais códigos de laboratórios passados usados na solução*, **não** zipados) devem ser enviados via página da disciplina no EAD. Como este laboratório será remoto, o prazo de entrega é até terça feira da semana que vem, dia 29 de abril.