

**UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO  
DEPARTAMENTO DE INFORMÁTICA**

**Filipe Gomes Arante de Souza**

**2º Trabalho de Algoritmos Numéricos  
Regressão Linear Múltipla e Polinomial**

**Julho / 2022**

# 1 Impressões de Tela do Editor do Octave

## 1.1 Geração da matriz de variáveis explicativas

Segue abaixo implementação:

```
1 % Gera a matriz de variáveis explicativas de acordo com o modelo especificado.
2 % n: Número de pontos.
3 % v: Quantidade de variáveis explicativas.
4 % p: Quantidade de parâmetros do modelo.
5 % x: Submatriz com variáveis explicativas originais.
6 % Retorna a matriz de variáveis explicativas e a informação sobre erro.
7 function [MatX, info] = matriz_explicativas(n, v, p, x)
8     if p < v % Modelo não permitido.
9         info = 1; % Erro.
10        MatX = []; % Erro.
11        return;
12    endif
13
14    info = 0;
15    if v == 1 % Regressão polinomial.
16        for i = (1: 1: n)
17            MatX(i, 1) = 1;
18        endfor
19
20        for j = (2: 1: p)
21            for i = (1: 1: n)
22                MatX(i, j) = MatX(i, j - 1) * x(i, 1);
23            endfor
24        endfor
25    else % Regressão linear múltipla.
26        for i = (1: 1: n)
27            MatX(i, 1) = 1;
28        endfor
29
30        for j = (2: 1: p)
31            for i = (1: 1: n)
32                MatX(i, j) = x(i, j - 1);
33            endfor
34        endfor
35    endif
36 endfunction
```

Figura 1: Implementação da geração da matriz explicativa.

## 1.2 Regressão linear múltipla e polinomial

Segue abaixo implementação: Obs: Não coloquei foto das funções auxiliares de eliminação de Gauss com pivotação parcial e substituições retroativas pois elas já foram mostradas no trabalho 1. Contudo, os arquivos .m referentes a estes algoritmos foram enviados no classroom.

```

1 % Calcula os parâmetros do modelo de regressão linear.
2 % n: Número de pontos.
3 % v: Quantidade de variáveis explicativas.
4 % p: Quantidade de parâmetros do modelo.
5 % x: Variáveis explicativas originais.
6 % y: Variáveis respostas.
7 % Retorna o vetor solução b com os parâmetros do modelo, o coeficiente de determinação r2
8 % e a informação sobre erro. Caso ocorra erro, convencionei b ser vazio e r2 igual a -1.
9 function [b, r2, info] = reg_linear_mult(n, v, p, x, y)
10     [MatX, info] = matriz_explicativas(n, v, p, x);
11
12     if info != 0 % Erro.
13         b = [] % Erro.
14         r2 = -1 % Erro.
15         return
16     endif
17
18     for i = (1: 1: p)
19         for j = (1: 1: i)
20             soma = 0;
21             for k = (1: 1: n)
22                 soma = soma + MatX(k, i) * MatX(k, j);
23             endfor
24
25             Sxx(i, j) = soma;
26
27             if i != j
28                 Sxx(j, i) = soma;
29             endif
30
31             soma = 0;
32             for k = (1: 1: n)
33                 soma = soma + MatX(k, i) * y(k);
34             endfor
35
36             Sxy(i) = soma; % Vetor dos termos independentes.
37
38         endfor
39     endfor
40
41     [U, d] = elim_gauss_pivotacao(p, Sxx, Sxy);
42     b = subs_retroativas(p, U, d);
43
44     info = 0;
45     S = 0;
46     Sy2 = 0;
47
48     for i = (1: 1: n)
49         u = 0;
50         for j = (1: 1: p)
51             u = u + MatX(i, j) * b(j);
52         endfor
53
54         S = S + (y(i) - u) * (y(i) - u);
55         Sy2 = Sy2 + y(i) * y(i);
56     endfor
57
58     r2 = 1 - S / (Sy2 - Sxy(1) * Sxy(1) / n)
59 endfunction

```

Figura 2: Implementação da resolução de regressão linear múltipla e polinomial.

## 2 Impressões da janela de comandos do Octave

### 2.1 Problema 1: Regressão Linear Múltipla

1)  $\hat{y}(x_1, x_2) = b_0 + b_1x_1 + b_2x_2$  (Campos, 2018; Exercício 4.6, Página 247)

$i$	$x_{i1}$	$x_{i2}$	$y$
1	-1	-2	13
2	0	-1	11
3	1	0	9
4	2	1	4
5	4	1	11
6	5	2	9
7	5	3	1
8	6	4	-1

Figura 3: Primeiro problema proposto.

```

octave:51> n = 8;
octave:52> v = 2;
octave:53> p = 3;
octave:54> x = [-1 -2; 0 -1; 1 0; 2 1; 4 1; 5 2; 5 3; 6 4]
x =

    -1    -2
     0    -1
     1     0
     2     1
     4     1
     5     2
     5     3
     6     4

octave:55> y = [13; 11; 9; 4; 11; 9; 1; -1]
y =

    13
    11
     9
     4
    11
     9
     1
    -1

octave:56> [b, r2, info] = reg_linear_mult(n, v, p, x, y);
MatX =

     1    -1    -2
     1     0    -1
     1     1     0
     1     2     1
     1     4     1
     1     5     2
     1     5     3
     1     6     4

info = 0
b =

    4.2393
    3.4000
   -6.4643

r2 = 0.9771
info = 0
octave:57>

```

Figura 4: Linha de comando do Octave na resolução do primeiro problema.

## 2.2 Problema 2: Regressão Linear Polinomial

2)  $\hat{y}(x) = b_0 + b_1x + b_2x^2$  (Campos, 2018; Exercício 4.7, Página 247)

$i$	$x_1$	$y$
1	-2,0	-30,5
2	-1,5	-20,2
3	0,0	-3,3
4	1,0	8,9
5	2,2	16,8
6	3,1	21,4

Figura 5: Segundo problema proposto.

```

octave:58> n = 6;
octave:59> v = 1;
octave:60> p = 3;
octave:61> x = [-2; -1.5; 0; 1; 2.2; 3.1]
x =

-2.0000
-1.5000
  0
  1.0000
  2.2000
  3.1000

octave:62> y = [-30.5; -20.2; -3.3; 8.9; 16.8; 21.4]
y =

-30.5000
-20.2000
 -3.3000
  8.9000
 16.8000
 21.4000

octave:63> [b, r2, info] = reg_linear_mult(n, v, p, x, y)
MatX =

  1.0000 -2.0000  4.0000
  1.0000 -1.5000  2.2500
  1.0000  0      0
  1.0000  1.0000  1.0000
  1.0000  2.2000  4.8400
  1.0000  3.1000  9.6100

info = 0
b =

-2.0177
11.3315
-1.2222

r2 = 0.9974
info = 0
octave:64>

```

Figura 6: Linha de comando do Octave na resolução do segundo problema.



### 3 Resultados do ajuste de curvas pelo método dos quadrados mínimos para os modelos 1 e 2

Modelo	Parâmetros do Modelo	Coefficiente de Determinação
1	$b_0 = 4.2393$ $b_1 = 3.4$ $b_2 = -6.4643$	$r^2 = 0.9771$
2	$b_0 = -2.0177$ $b_1 = 11.3315$ $b_2 = -1.2222$	$r^2 = 0.9974$

Figura 7: Tabela de resultados dos problemas 1 e 2.