Mínimos quadrados

André Corso Pozzan
Atividade **4** - Iniciação Científica **GICS**

Orientador: Prof. Ph.D. Eduardo Gonçalves de Lima

8 de outubro de 2024

1 Atividade:

Seja o conjunto de 5 medições (x_i, y_i), onde $i = 1, 2, ..., 5$:

- $(x_1, y_1) = (0, 0.2)$
- $(x_2, y_2) = (0.1, 0.3)$
- $(x_3, y_3) = (0.2, 0.45)$
- $(x_4, y_4) = (0.3, 0.7)$
- $(x_5, y_5) = (0.4, 0.8)$

Deseja-se obter a melhor reta que relacione (x_i, y_i). Essa reta terá a seguinte equação $y_i = a.x_i + b$, onde a e b são constantes ainda a serem determinadas. Obter os valores ótimos dos coeficientes ajustáveis a e b, usando mínimos quadrados, de duas formas distintas:

1.1 Fazendo as contas no papel:

1. Defina, para cada par (x_i, y_i), um erro como sendo a diferença entre o valor medido (y_i) e o valor fornecido pela equação da reta $(a.x_i + b)$. Após isso, realizar a soma dos quadrados dos erros. Essa equação, chamada de MSE, dependerá dos coeficientes ajustáveis a e b.

2. O objetivo aqui é escolher coeficientes ajustáveis a e b que minimizem o MSE. Para isso, buscam-se os pontos de mínimos de MSE em relação aos coeficientes ajustáveis a e b. Para isso:

   (a) Derivar a equação do MSE em relação ao coeficiente a e, na sequência, igualar o resultado a zero;
   (b) Derivar a equação do MSE em relação ao coeficiente b e, na sequência, igualar o resultado a zero;
   (c) Resolver o sistema algébrico linear de 2 equações (equações resultantes dos itens 2a e 2b) nas 2 incógnitas que são os coeficientes ajustáveis a e b.

3. Use os coeficientes a e b obtidos em 2c e, para cada x_i, obter a estimativa usando a equação da reta. Em um mesmo gráfico, visualizar $(y_i) \times (x_i)$; $(a.x_i + b) \times (x_i)$.

1.2 Usando mínimos quadrados no MATLAB

(O texto abaixo refere-se a MATLAB, mas se você preferir, você pode usar o Python):

1. Usando notação matricial/vetorial, definir:
   (a) Vetor de saídas desejadas de dimensão 5×1: $Y = [y_1; y_2; \ldots; y_5]$
   (b) Vetor de entradas aplicadas de dimensão 5×1: $X = [x_1; x_2; \ldots; x_5]$
   (c) Matriz de regressão de dimensão 5×2: $XX = [x_1\ 1; x_2\ 1; \ldots; x_5\ 1]$
   (d) Vetor de coeficientes ajustáveis de dimensão 2×1: $COEF = [a; b]$

2. Admitindo que todos os erros sejam nulos, tem-se o seguinte sistema algébrico linear de 5 equações (5 medições) em 2 incógnitas (coeficientes a e b): $Y = XX \cdot COEF$. Em um sistema algébrico linear onde a quantidade de equações é maior que a quantidade de incógnitas, só há solução se houver equações redundantes. Sendo 5 medições diferentes, espera-se que não haja redundância. Dessa forma, não há solução para esse sistema. Contudo, é possível obter a solução que minimize o erro quadrático médio, cuja definição é a mesma do MSE feita anteriormente. No MATLAB, isso é feito através do comando "\". Isso é chamado de mínimos quadrados. No MATLAB, obtém-se o vetor de coeficientes ajustáveis através de $COEF = XX \backslash Y$.

3. Use o vetor de coeficientes ajustáveis obtido em 2 para obter o vetor de saídas estimadas pela reta. Em um mesmo gráfico, visualizar $(Y) \times (X)$; $(XX \cdot COEF) \times (X)$.

2 Resolução

2.1 Cálculos no papel

Primeiramente, iniciei esboçando um gráfico dos os pontos dados para ter uma visualização melhor do problema e então conforme descrito no método utilizei a fórmula da distância de dois pontos onde o Δx não varia para gerar a função que relaciona todos os pontos que será utilizada para obter seu ponto mínimo que é a melhor reta que relaciona os pontos.

Depois de realizar os cálculos utilizei o Geogebra para visualizar o gráfico da função:

$$F(a,b) = (b-0,2)^2 + (0,1a+b-0,3)^2 + (0,2a+b-0,45)^2 + (0,3a+b-0,7)^2 + (0,4a+b-0,8)^2$$

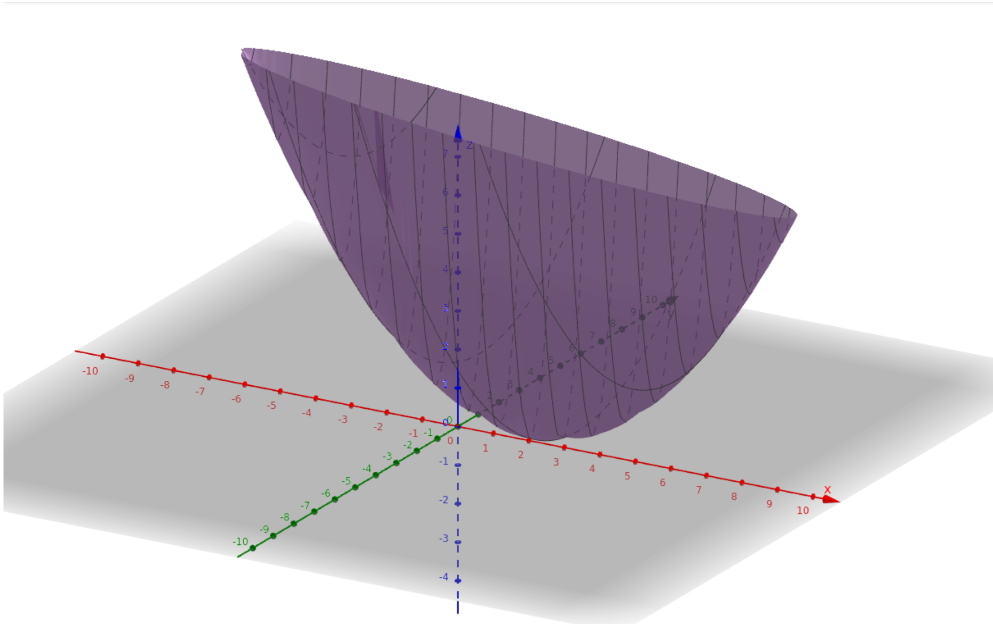


Figura 1: Gráfico no $\mathbb{R}^3$ da função dos mínimos quadrados

O objetivo é encontrar o ponto mínimo, ou seja o ponto mais abaixo na visualização do gráfico assim como demostra o ponto A na imagem:

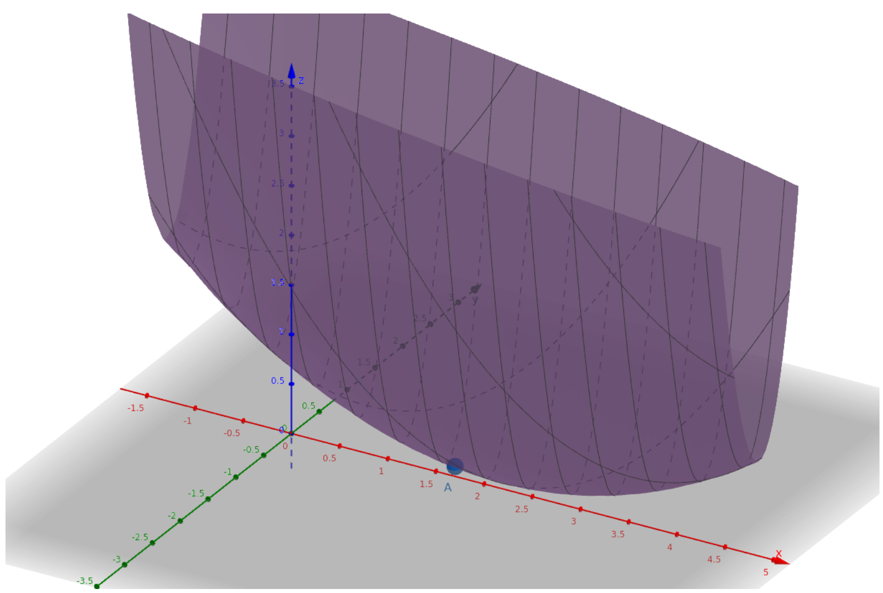
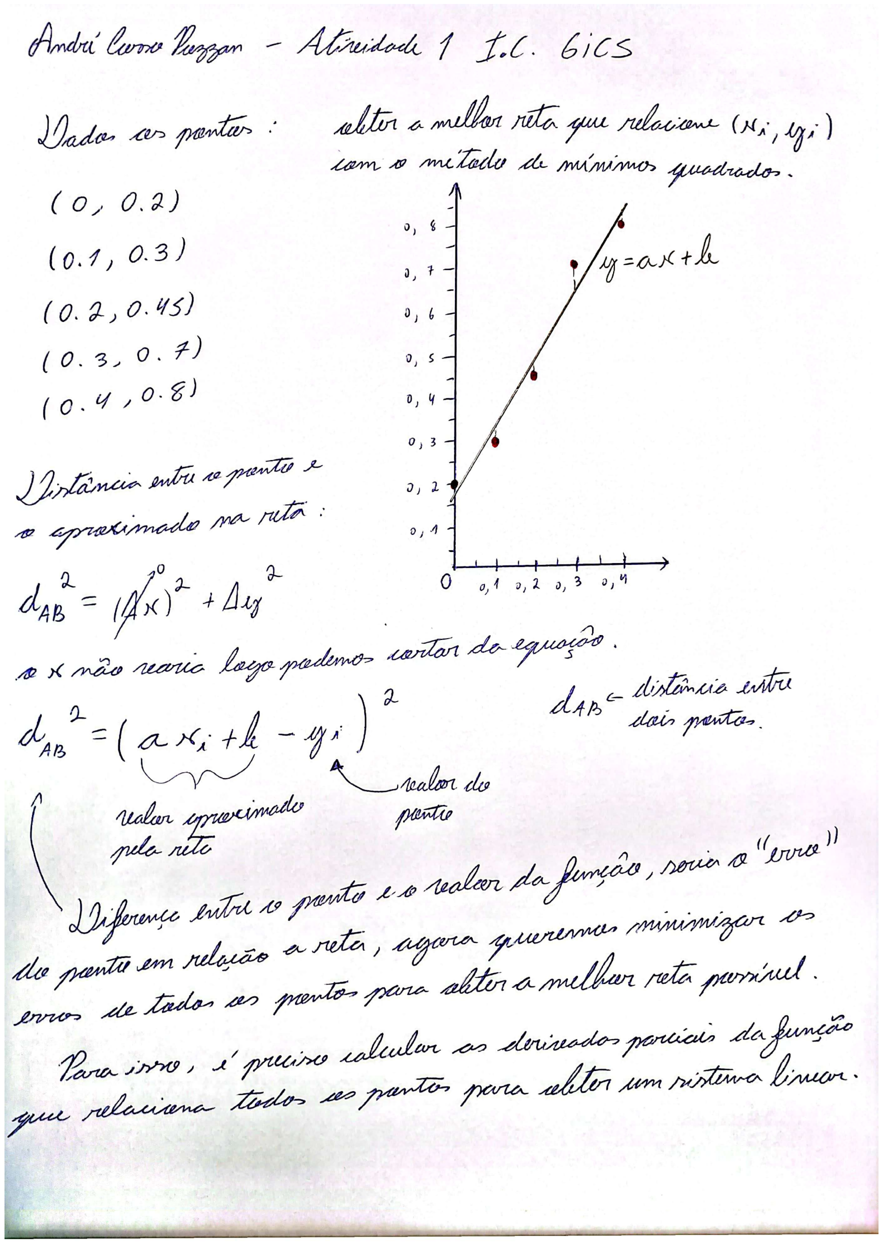


Figura 2: Ponto mínimo



André Corso Pozzan - Atividade 1 I.C. GICS

Dados os pontos: obter a melhor reta que relacione (x_i, y_i) com o método de mínimos quadrados.

(0, 0.2)
(0.1, 0.3)
(0.2, 0.45)
(0.3, 0.7)
(0.4, 0.8)

Distância entre o ponto e o aproximado na reta:

$$d_{AB}^2 = (\cancel{\Delta x})^2 + \Delta y^2$$

o x não varia logo podemos cortar da equação.

$$d_{AB}^2 = (\underbrace{a x_i + b}_{\text{valor aproximado pela reta}} - y_i)^2$$

valor do ponto

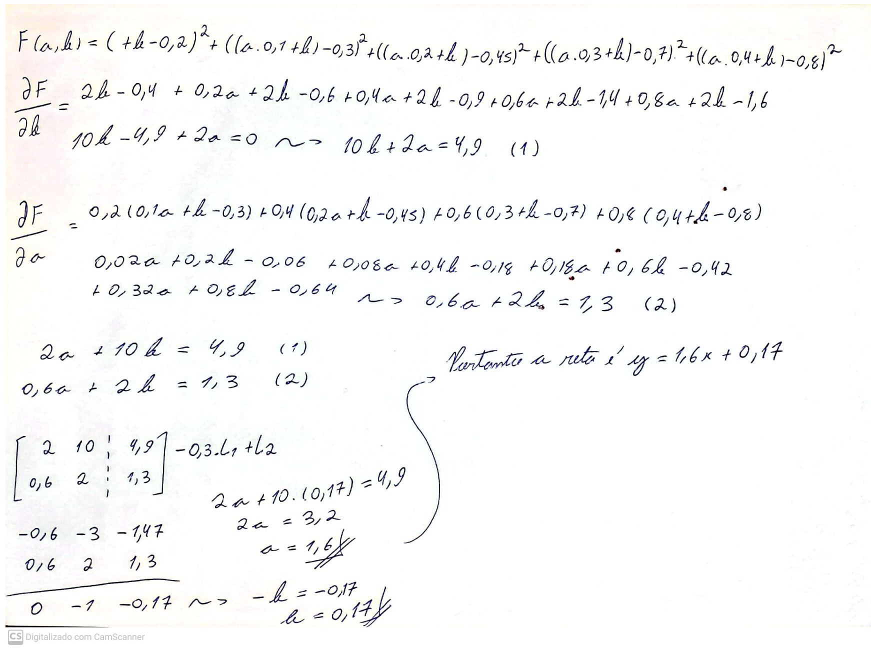
d_{AB} ← distância entre dois pontos.

Diferença entre o ponto e o valor da função, seria o "erro" do ponto em relação a reta, agora queremos minimizar os erros de todos os pontos para obter a melhor reta possível.

Para isso, é preciso calcular as derivadas parciais da função que relaciona todos os pontos para obter um sistema linear.

Figura 3: Apresentação da resolução

Para encontrar os melhores valores para a e b, é necessário calcular as derivadas parciais em a e b, que no fim vão resultar em um sistema linear:



$$F(a,b) = (+b-0,2)^2 + ((a.0,1+b)-0,3)^2 + ((a.0,2+b)-0,45)^2 + ((a.0,3+b)-0,7)^2 + ((a.0,4+b)-0,8)^2$$

$$\frac{\partial F}{\partial b} = 2b - 0,4 + 0,2a + 2b - 0,6 + 0,4a + 2b - 0,9 + 0,6a + 2b - 1,4 + 0,8a + 2b - 1,6$$

$$10b - 4,9 + 2a = 0 \rightsquigarrow 10b + 2a = 4,9 \quad (1)$$

$$\frac{\partial F}{\partial a} = 0,2(0,1a + b - 0,3) + 0,4(0,2a + b - 0,45) + 0,6(0,3 + b - 0,7) + 0,8(0,4 + b - 0,8)$$

$$0,02a + 0,2b - 0,06 + 0,08a + 0,4b - 0,18 + 0,18a + 0,6b - 0,42 + 0,32a + 0,8b - 0,64 \rightsquigarrow 0,6a + 2b = 1,3 \quad (2)$$

$$2a + 10b = 4,9 \quad (1)$$
$$0,6a + 2b = 1,3 \quad (2)$$

Portanto a reta é $y = 1,6x + 0,17$

$$\left[\begin{array}{cc|c} 2 & 10 & 4,9 \\ 0,6 & 2 & 1,3 \end{array}\right] -0,3.L_1 + L_2$$

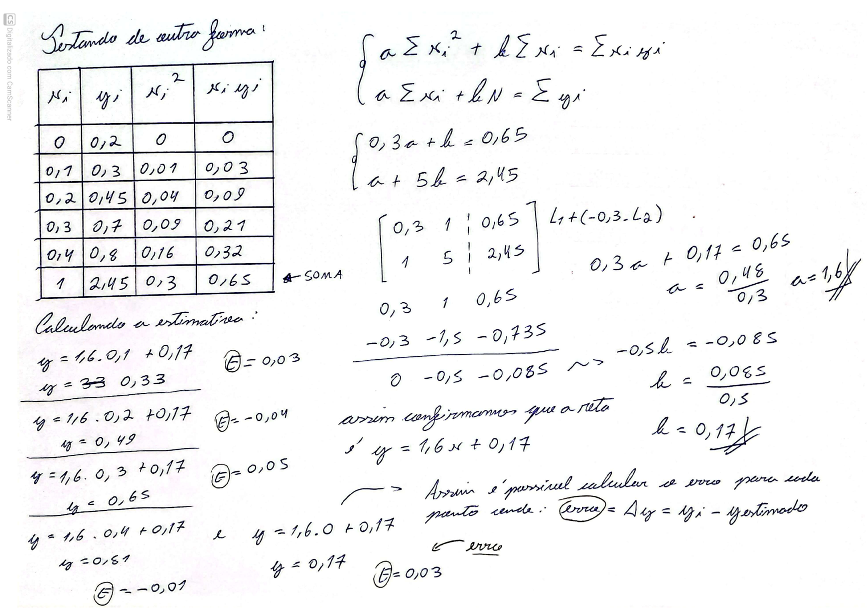
$$\begin{array}{ccc} -0,6 & -3 & -1,47 \\ 0,6 & 2 & 1,3 \\ \hline 0 & -1 & -0,17 \end{array} \rightsquigarrow -b = -0,17 \quad b = 0,17$$

$$2a + 10.(0,17) = 4,9$$
$$2a = 3,2$$
$$a = 1,6$$

Figura 4: Calculando a reta por meio das derivadas parciais

Agora utilizando a fórmula dos somatórios para verificar a resposta:

$$\begin{cases} a\sum x_i^2 + b\sum x_i = \sum x_i y_i \\ a\sum x_i + bN = \sum y_i \end{cases} \tag{1}$$



Testando de outra forma:

x_i	y_i	x_i^2	$x_i y_i$
0	0,2	0	0
0,1	0,3	0,01	0,03
0,2	0,45	0,04	0,09
0,3	0,7	0,09	0,21
0,4	0,8	0,16	0,32
1	2,45	0,3	0,65

← SOMA

$$\begin{cases} a\sum x_i^2 + b\sum x_i = \sum x_i y_i \\ a\sum x_i + bN = \sum y_i \end{cases}$$

$$\begin{cases} 0,3a + b = 0,65 \\ a + 5b = 2,45 \end{cases}$$

$$\left[\begin{array}{cc|c} 0,3 & 1 & 0,65 \\ 1 & 5 & 2,45 \end{array}\right] L_1 + (-0,3.L_2)$$

$$\begin{array}{ccc} 0,3 & 1 & 0,65 \\ -0,3 & -1,5 & -0,735 \\ \hline 0 & -0,5 & -0,085 \end{array} \rightsquigarrow -0,5b = -0,085 \quad b = \frac{0,085}{0,5} \quad b = 0,17$$

$$0,3a + 0,17 = 0,65 \quad a = \frac{0,48}{0,3} \quad a = 1,6$$

assim confirmamos que a reta é $y = 1,6x + 0,17$

Calculando a estimativa:

$y = 1,6.0,1 + 0,17$ $\quad E = 0,03$
$y = \cancel{33}\ 0,33$

$y = 1,6.0,2 + 0,17$ $\quad E = -0,04$
$y = 0,49$

$y = 1,6.0,3 + 0,17$ $\quad E = 0,05$
$y = 0,65$

$y = 1,6.0,4 + 0,17$
$y = 0,81$
$E = -0,01$

e $y = 1,6.0 + 0,17$
$y = 0,17$ $\quad E = 0,03$

Assim é possível calcular o erro para cada ponto onde: erro $= \Delta y = y_i - y_{estimado}$

← erro

Figura 5: Calculando a reta por meio do somatório

Assim é obtido a reta:

$$y = 1,6x + 0,17$$

x_i	y_i	$y_{estimado}$	$erro$
0	0,2	0,17	-0,03
0,1	0,3	0,33	-0,03
0,2	0,45	0,49	+0,04
0,3	0,7	0,65	-0,05
0,4	0,8	0,81	+0,01

Tabela 1: Valores estimados e erros

2.2 Código em Python

Para fazer o cálculo com software utilizei a linguagem de programação Python juntamente com as bibliotecas ”numpy”e ”matplotlib”para realizar as tarefas necessárias. Dessa forma, o código final é o seguinte:

```
import matplotlib.pyplot as plt
import numpy as np

# Definindo o vetor de saídas desejadas (5x1)
vetorSaidasDesejadas = np.array([0.2, 0.3, 0.45, 0.7, 0.8])

# Definindo o vetor de entradas aplicadas (5x1)
vetorEntradasAplicadas = np.array([0, 0.1, 0.2, 0.3, 0.4])

# Criando a matriz de regressão (5x2)
matrizDeRegressao = np.array([[0, 1],
                              [0.1, 1],
                              [0.2, 1],
                              [0.3, 1],
                              [0.4, 1]])

# Usando lstsq para encontrar a solução de mínimos quadrados
resultado = np.linalg.lstsq(matrizDeRegressao, vetorSaidasDesejadas)

# Exibindo os resultados
print("Coeficientes da regressão:", resultado[0])
a = resultado[0][0]
b = resultado[0][1]

# Estimativa dos valores de y usando a equação da reta
yEstimado = [a * x + b  for x in vetorEntradasAplicadas]

# Criar o gráfico
plt.plot(vetorEntradasAplicadas, vetorSaidasDesejadas,
         color='blue', label='Valores reais (yi)', marker='o', linestyle='')
plt.plot(vetorEntradasAplicadas, yEstimado,
         color='red', label='Reta (a*xi + b)', linestyle='-')
plt.plot(vetorEntradasAplicadas, yEstimado,
         color='green', label='Valores estimados', marker='x', linestyle='')

# Títulos e legendas
plt.title('Comparação entre valores reais e estimados')
plt.xlabel('xi')
plt.ylabel('yi')
plt.legend()
plt.grid(True)

# Mostrar o gráfico
plt.show()
```

O código tem como saída no console: *Coeficientes da regressão: [1.6 0.17]* confirmando os valores calculados anteriormente no papel, também é gerado um gráfico que demostra a reta, os pontos e as estimativas:

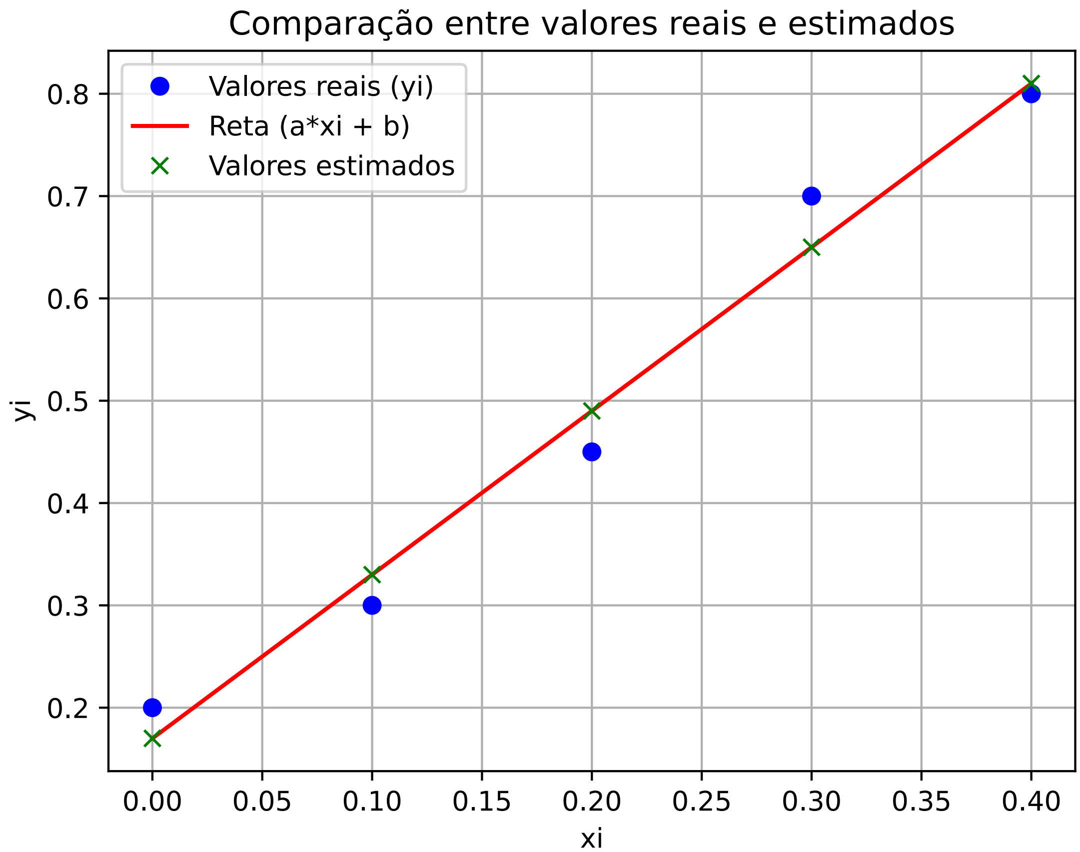


Figura 6: Gráfico gerado pelo código em Python

Para realizar o cálculo com Python utilizei a função np.linalg.lstsq() que assemelha seu funcionamento a ferramenta descrita no MATLAB. A documentação da função descreve:

> ≪Computes the vector x that approximately solves the equation $a \cdot x = b$. The equation may be under-, well-, or over-determined (i.e., the number of linearly independent rows of a can be less than, equal to, or greater than its number of linearly independent columns). If a is square and of full rank, then x (but for round-off error) is the "exact" solution of the equation. Else, x minimizes the Euclidean 2-norm. If there are multiple minimizing solutions, the one with the smallest 2-norm is returned.≫

Texto obtido em NumPy, 2024

Referências

Murakami, M. (2024a). *MÉTODO DOS MÍNIMOS QUADRADOS EP1* [Acessado em: 7 out. 2024]. https://www.youtube.com/watch?v=Gpd14W4vDIQ

Murakami, M. (2024b). *MÉTODO DOS MÍNIMOS QUADRADOS EP2* [Acessado em: 7 out. 2024]. https://www.youtube.com/watch?v=CzQhewuuRWs

NumPy. (2024). *Função "numpy.linalg.lstsq" da biblioteca NumPy* [Acessado em: 7 out. 2024]. https://numpy.org/doc/stable/reference/generated/numpy.linalg.lstsq.html

Ramos, V. (2023). *Como criar gráficos com Matplotlib no Python* [Acessado em: 7 out. 2024]. https://pythonacademy.com.br/blog/como-criar-graficos-utilizando-matplotlib-no-python