

Ajuste de Curvas

Prof. Americo Cunha

Universidade do Estado do Rio de Janeiro – UERJ

americo.cunha@uerj.br

www.americocunha.org



 @AmericoCunhaJr

 @AmericoCunhaJr

 @AmericoCunhaJr

 @AmericoCunhaJr



Primórdios da ciência de dados



Johannes Kepler (1571-1630)
Astronomia nova (1609)



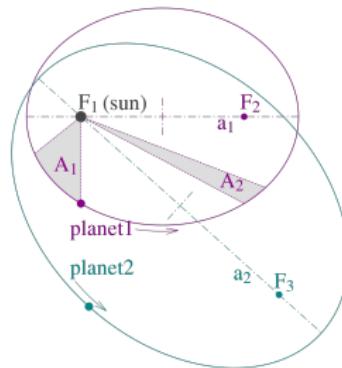
Tycho Brahe (1546-1601)
Astronomiae Instauratae (1598)

Leis do movimento planetário:

$$1. \frac{\alpha}{r} = 1 + \epsilon \cos \theta$$

$$2. \frac{dA}{dt} = \frac{L}{2m}$$

$$3. \frac{a^3}{T^2} \approx \frac{GM}{4\pi^2}$$



Ciência de dados hoje

- A cada segundo cerca de 1,7 megabytes de novas informações são criados para cada ser humano no planeta;
 - O mundo digital hoje tem a ordem de 44 zettabytes em dados.



Fonte: <https://techjury.net/blog/how-much-data-is-created-every-day/>

Ciência de dados hoje

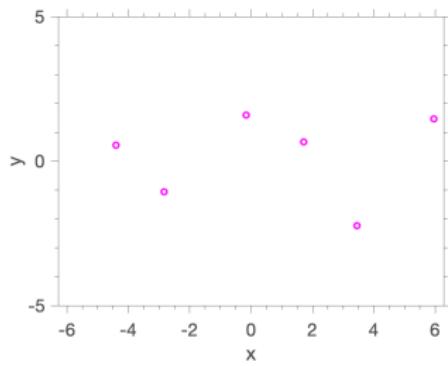
- A cada segundo cerca de 1,7 megabytes de novas informações são criados para cada ser humano no planeta;
 - O mundo digital hoje tem a ordem de 44 zettabytes em dados.



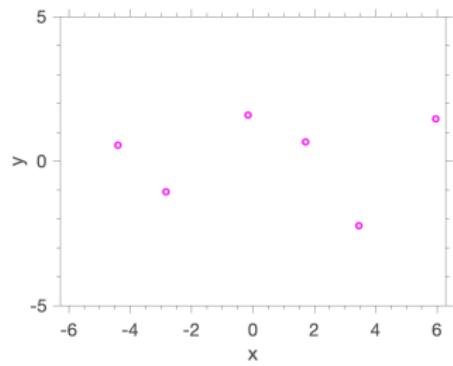
**Precisamos de ferramentas para
extrair informações úteis dessa
grande massa de dados!**

Interpolação × Regressão

Interpolação

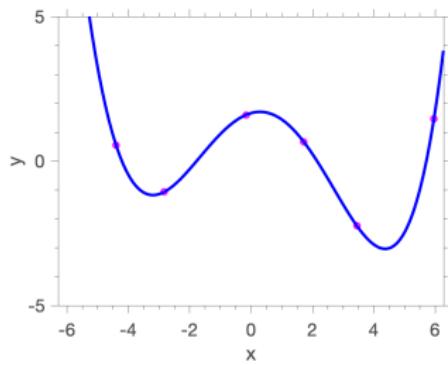


Regressão



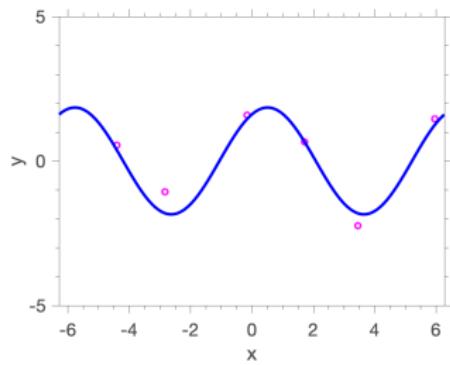
Interpolação × Regressão

Interpolação



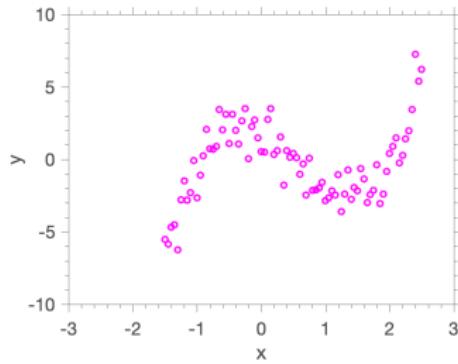
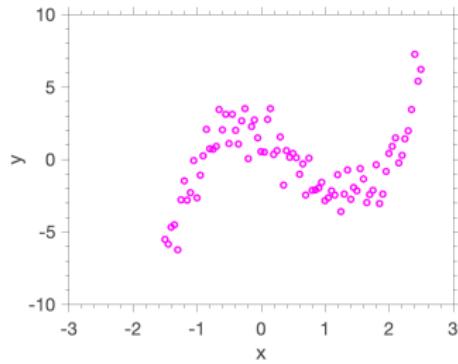
A curva interpolante passa
por todos os pontos.

Regressão

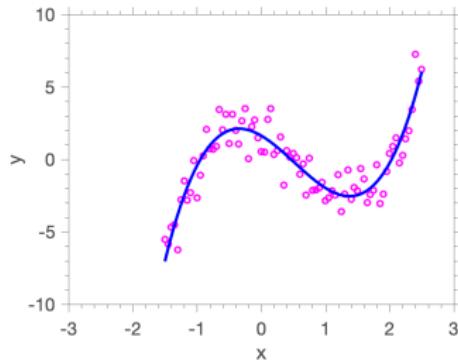
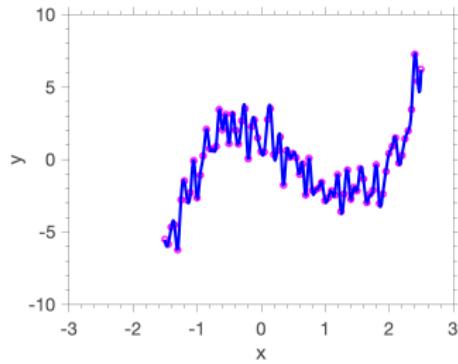


A curva regressora passa
“próxima” de todos os pontos.

Para aproximar a tendência de muitos dados (com ruído) ...



Para aproximar a tendência de muitos dados (com ruído) ...



... um processo de regressão é muito mais robusto!



O problema de regressão (ajuste de uma curva)

Imagine que você tenha m observações obtidas por um processo de (medição) aquisição de dados:

$$(x_1, y_1), \dots, (x_m, y_m)$$

Cada ponto dessa amostra de dados tem dois valores organizados como par ordenado (x, y) , onde x representa uma grandeza independente, e y representa uma grandeza que dependente de x , i.e. $y = y(x)$.



O problema de regressão (ajuste de uma curva)

Imagine que você tenha m observações obtidas por um processo de (medição) aquisição de dados:

$$(x_1, y_1), \dots, (x_m, y_m)$$

Cada ponto dessa amostra de dados tem dois valores organizados como par ordenado (x, y) , onde x representa uma grandeza independente, e y representa uma grandeza que dependente de x , i.e. $y = y(x)$.

**Qual a curva que “melhor aproxima”
a tendência desses pontos?**



Regressão

Hipótese:

Os ***m*** pontos têm todas as abscissas distintas dois a dois, i.e.,

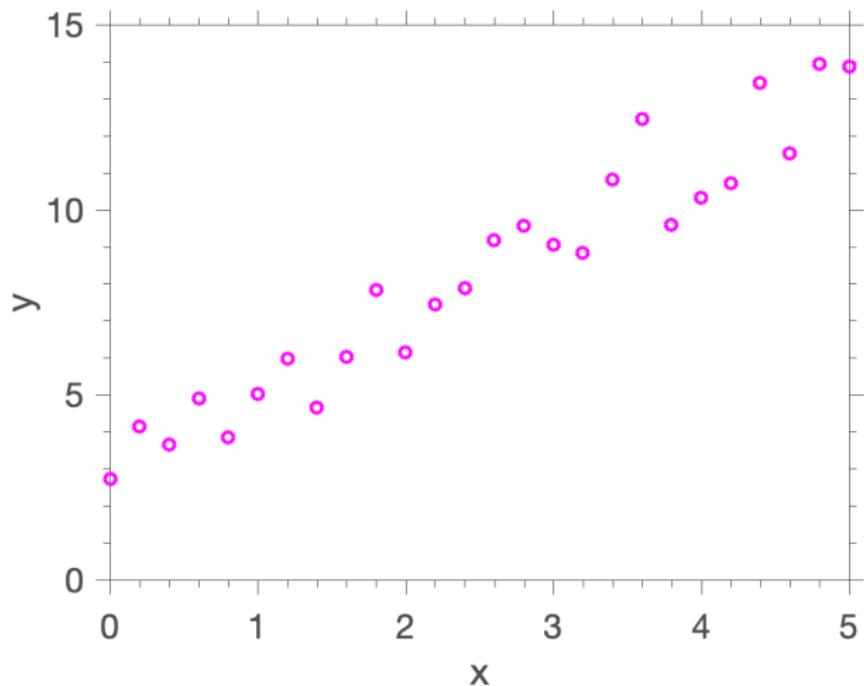
$$x_i \neq x_j, \text{ para qualquer par } (i, j).$$

Ideia do método:

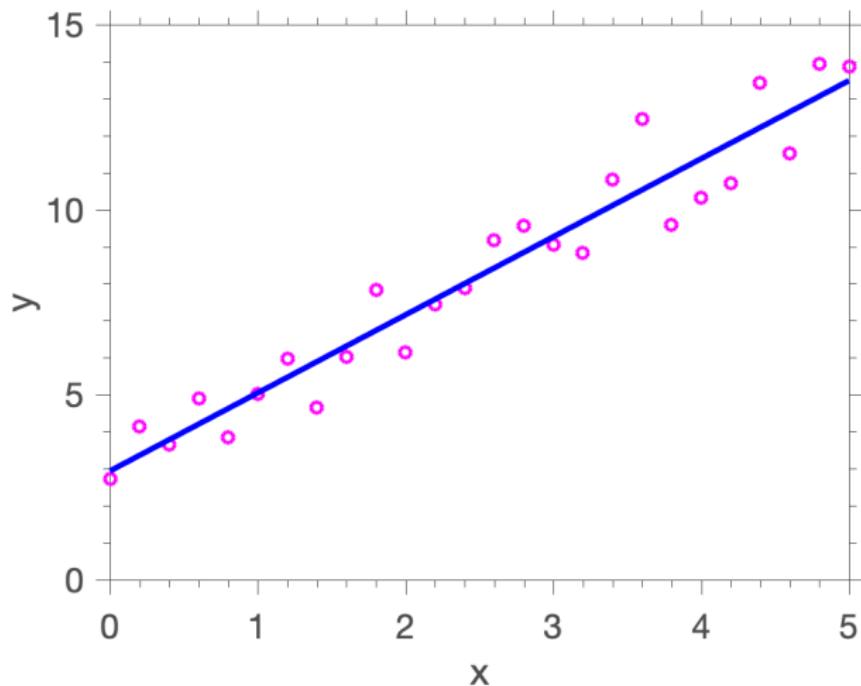
- Aproximar a relação entre y e x por uma função $f(x)$, i.e., $y \approx f(x)$;
- Escolher f de modo que $y = f(x)$ seja a curva que minimize uma medida de erro E , que quantifica o quanto distante os ***m*** pontos estão da curva aproximante.



Escolhendo o regressor



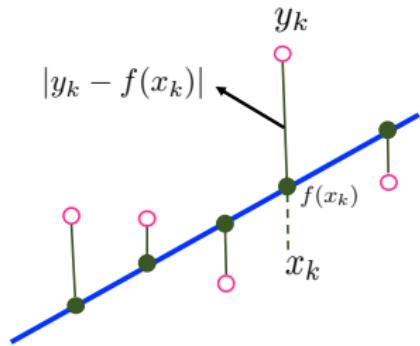
Escolhendo o regressor



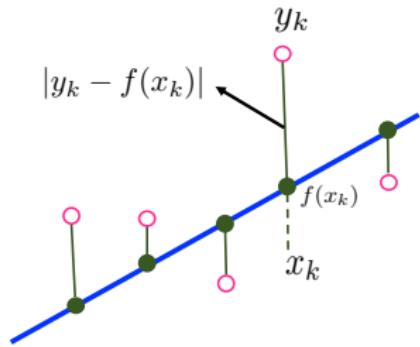
$$y = f(x) = \alpha x + \beta$$



Definindo uma medida de erro

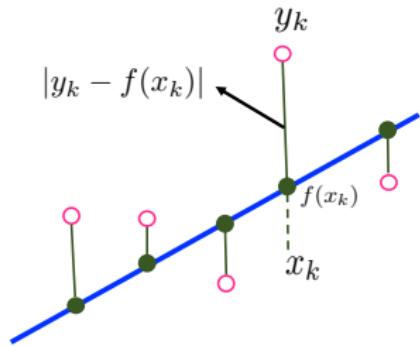


Definindo uma medida de erro



Erro quadrático:

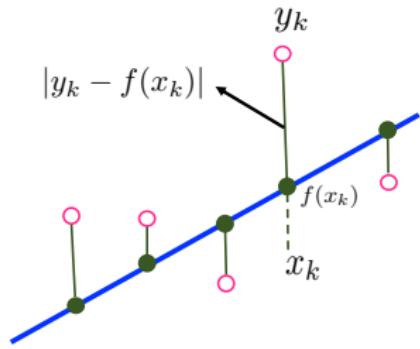
Definindo uma medida de erro



Erro quadrático:

$$E_2 = (y_1 - f(x_1))^2 + \cdots + (y_m - f(x_m))^2$$

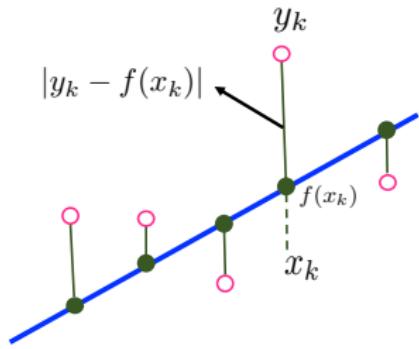
Definindo uma medida de erro



Erro quadrático:

$$\begin{aligned}E_2 &= (y_1 - f(x_1))^2 + \cdots + (y_m - f(x_m))^2 \\&= (y_1 - (\alpha x_1 + \beta))^2 + \cdots + (y_m - (\alpha x_m + \beta))^2\end{aligned}$$

Definindo uma medida de erro



Erro quadrático:

$$\begin{aligned}E_2 &= (y_1 - f(x_1))^2 + \cdots + (y_m - f(x_m))^2 \\&= (y_1 - (\alpha x_1 + \beta))^2 + \cdots + (y_m - (\alpha x_m + \beta))^2\end{aligned}$$

Note que a medida de erro é uma função dos parâmetros α e β , i.e., $E_2 = E_2(\alpha, \beta)$.

Encontrando a curva que minimiza o erro

$$\frac{\partial E_2}{\partial \alpha} = 0 \quad \text{e} \quad \frac{\partial E_2}{\partial \beta} = 0$$



Encontrando a curva que minimiza o erro

$$\frac{\partial E_2}{\partial \alpha} = 0 \quad \text{e} \quad \frac{\partial E_2}{\partial \beta} = 0$$

\iff



Encontrando a curva que minimiza o erro

$$\frac{\partial E_2}{\partial \alpha} = 0 \quad \text{e} \quad \frac{\partial E_2}{\partial \beta} = 0$$

\iff

$$\begin{cases} (x_1^2 + \cdots + x_m^2) \alpha + (x_1 + \cdots + x_m) \beta = x_1 y_1 + \cdots + x_m y_m \\ (x_1 + \cdots + x_m) \alpha + m \beta = y_1 + \cdots + y_m \end{cases}$$

Encontrando a curva que minimiza o erro

$$\frac{\partial E_2}{\partial \alpha} = 0 \quad \text{e} \quad \frac{\partial E_2}{\partial \beta} = 0$$

\iff

$$\begin{cases} (x_1^2 + \cdots + x_m^2) \alpha + (x_1 + \cdots + x_m) \beta = x_1 y_1 + \cdots + x_m y_m \\ (x_1 + \cdots + x_m) \alpha + m \beta = y_1 + \cdots + y_m \end{cases}$$

\iff



Encontrando a curva que minimiza o erro

$$\frac{\partial E_2}{\partial \alpha} = 0 \quad \text{e} \quad \frac{\partial E_2}{\partial \beta} = 0$$

\iff

$$\begin{cases} (x_1^2 + \cdots + x_m^2) \alpha + (x_1 + \cdots + x_m) \beta = x_1 y_1 + \cdots + x_m y_m \\ (x_1 + \cdots + x_m) \alpha + m \beta = y_1 + \cdots + y_m \end{cases}$$

\iff

$$\begin{bmatrix} (x_1^2 + \cdots + x_m^2) & (x_1 + \cdots + x_m) \\ (x_1 + \cdots + x_m) & m \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} x_1 y_1 + \cdots + x_m y_m \\ y_1 + \cdots + y_m \end{bmatrix}$$



Encontrando a curva que minimiza o erro

$$\frac{\partial E_2}{\partial \alpha} = 0 \quad \text{e} \quad \frac{\partial E_2}{\partial \beta} = 0$$

\iff

$$\begin{cases} (x_1^2 + \cdots + x_m^2) \alpha + (x_1 + \cdots + x_m) \beta = x_1 y_1 + \cdots + x_m y_m \\ (x_1 + \cdots + x_m) \alpha + m \beta = y_1 + \cdots + y_m \end{cases}$$

\iff

$$\begin{bmatrix} (x_1^2 + \cdots + x_m^2) & (x_1 + \cdots + x_m) \\ (x_1 + \cdots + x_m) & m \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} x_1 y_1 + \cdots + x_m y_m \\ y_1 + \cdots + y_m \end{bmatrix}$$

A solução é um mínimo global (E_2 não tem máximo)!

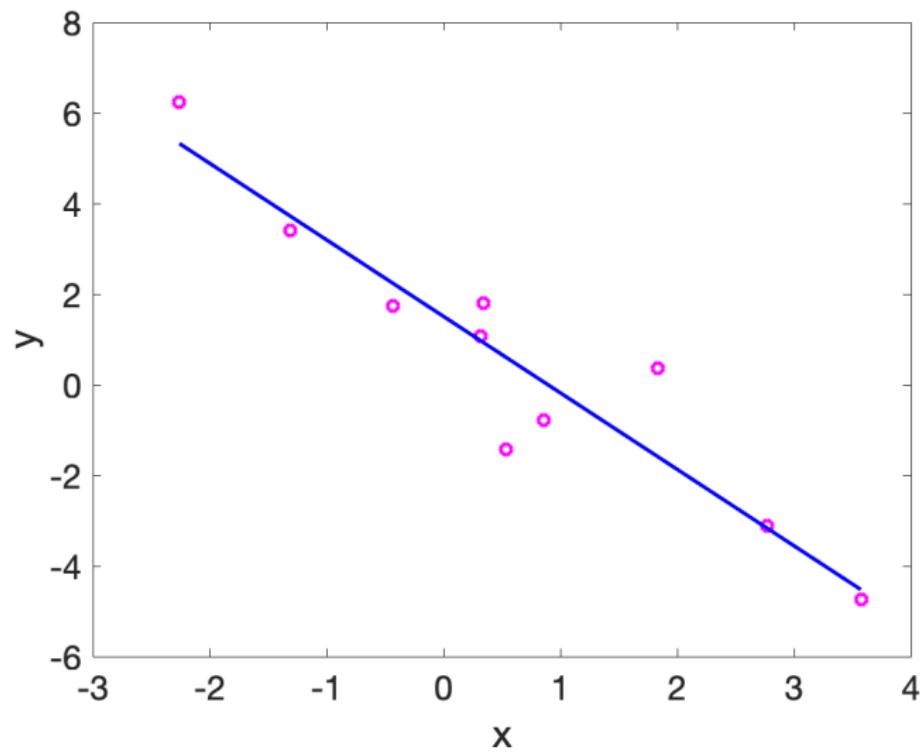


Regressão linear simples

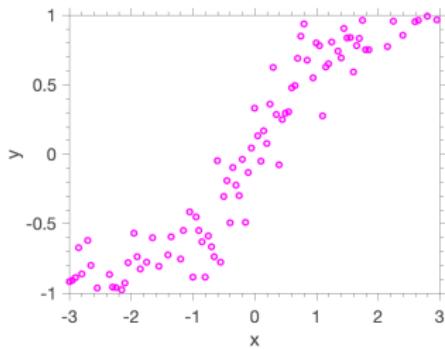
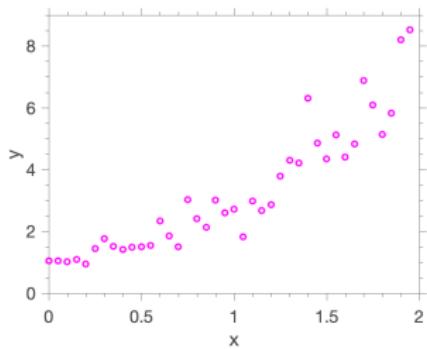
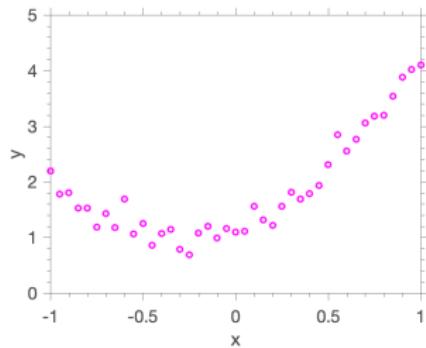
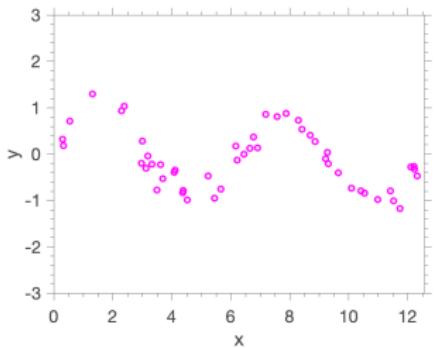
```
clc  
clear  
  
m = 10;  
xdata = randn(m,1);  
ydata = -2*xdata + 1 + randn(size(xdata));  
  
A = [sum(xdata.^2) sum(xdata); sum(xdata) m];  
b = [sum(xdata.*ydata); sum(ydata)];  
x = A\b;  
  
xfit = min(xdata):0.01:max(xdata);  
yfit = x(1)*xfit + x(2);  
  
plot(xdata,ydata,'o', xfit,yfit, '-b', 'LineWidth', 2);  
xlabel('x')  
ylabel('y')  
set(gca, 'FontSize', 18);
```



Regressão linear simples



E se a tendência não for uma reta?



Regressão linear via mínimos quadrados (caso geral)

1. Escolha uma forma para a função de regressão

$$f(x) = \alpha_1 \phi_1(x) + \cdots + \alpha_n \phi_n(x)$$

- ϕ_k - funções linearmente independentes (predefinidas)
- α_k - parâmetros de ajuste (a serem determinados)

2. Defina o erro quadrático

$$E_2 = \|A\mathbf{x} - \mathbf{b}\|^2 = \sum_{k=1}^m (y_k - f(x_k))^2$$

3. Encontre os parâmetros de ajuste que minimizam o erro quadrático, i.e.,

$$\mathbf{x}^\dagger = \arg \min \|A\mathbf{x} - \mathbf{b}\|$$



Função aproximante

Função aproximante

$$f(x) = \alpha_1 \phi_1(x) + \cdots + \alpha_n \phi_n(x)$$

Função aproximante

$$f(x) = \alpha_1 \phi_1(x) + \cdots + \alpha_n \phi_n(x)$$

$$f(x_1) = \alpha_1 \phi_1(x_1) + \cdots + \alpha_n \phi_n(x_1) \approx y_1$$

 \vdots \vdots

$$f(x_m) = \alpha_1 \phi_1(x_m) + \cdots + \alpha_n \phi_n(x_m) \approx y_m$$



Função aproximante

$$f(x) = \alpha_1 \phi_1(x) + \cdots + \alpha_n \phi_n(x)$$

$$\underbrace{\begin{bmatrix} \phi_1(x_1) & \cdots & \phi_n(x_1) \\ \vdots & \ddots & \vdots \\ \phi_1(x_m) & \cdots & \phi_n(x_m) \end{bmatrix}}_A \underbrace{\begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix}}_x \approx \underbrace{\begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix}}_b \iff A \mathbf{x} \approx \mathbf{b}$$



Função aproximante

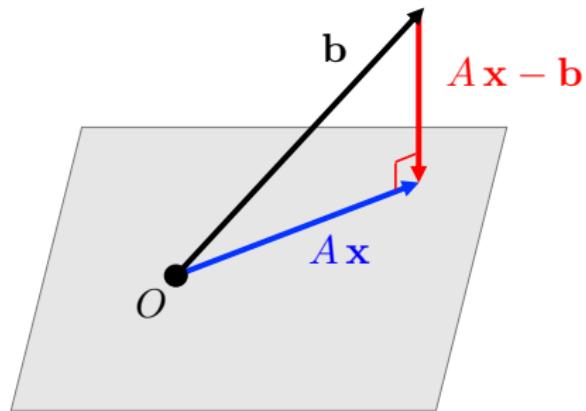
$$f(x) = \alpha_1 \phi_1(x) + \cdots + \alpha_n \phi_n(x)$$

$$\underbrace{\begin{bmatrix} \phi_1(x_1) & \cdots & \phi_n(x_1) \\ \vdots & \ddots & \vdots \\ \phi_1(x_m) & \cdots & \phi_n(x_m) \end{bmatrix}}_A \underbrace{\begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix}}_x \approx \underbrace{\begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix}}_b \iff A \mathbf{x} \approx \mathbf{b}$$

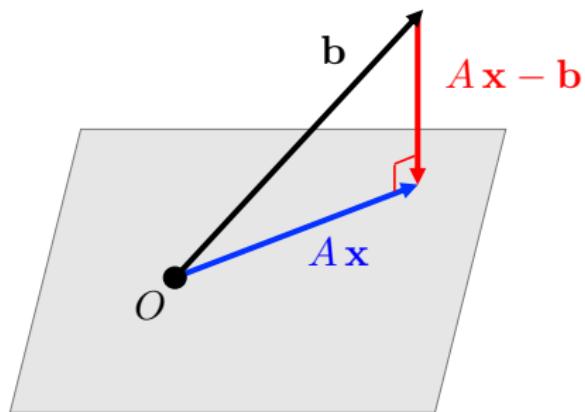
Em geral, esse sistema não tem solução!



Interpretação geométrica

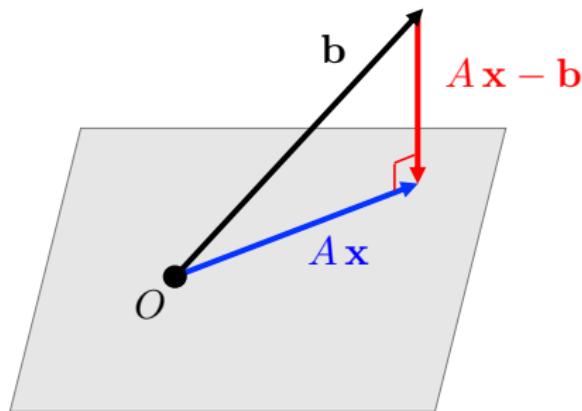


Interpretação geométrica



$\mathbf{b} \notin$ sub-espaco $A\mathbf{x}$

Interpretação geométrica

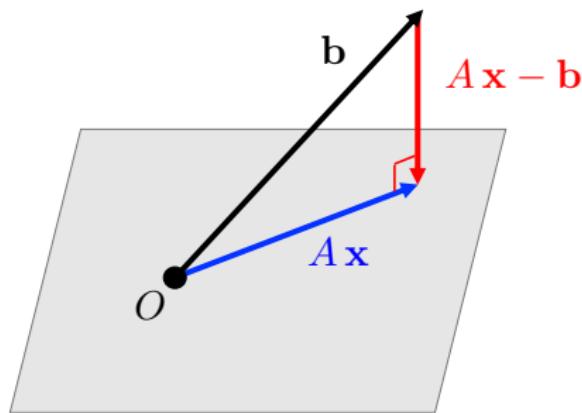


$\mathbf{b} \notin \text{sub-espaco } Ax$

\iff

$Ax - b \perp \text{sub-espaco } Ax$

Interpretação geométrica



$\mathbf{b} \notin \text{sub-espaco } Ax$

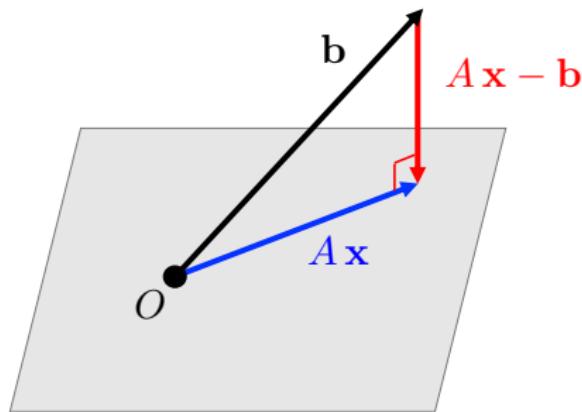
\iff

$Ax - \mathbf{b} \perp \text{sub-espaco } Ax$

\iff

$$A^T (Ax - \mathbf{b}) = \mathbf{0}$$

Interpretação geométrica



$\mathbf{b} \notin \text{sub-espaco } Ax$

\iff

$Ax - b \perp \text{sub-espaco } Ax$

\iff

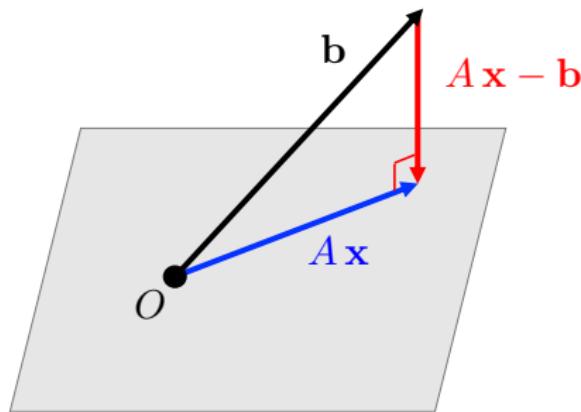
$$A^T (Ax - b) = \mathbf{0}$$

\iff

$$A^T Ax = A^T b$$

(equação normal)

Interpretação geométrica



$\mathbf{b} \notin \text{sub-espaco } A\mathbf{x}$

\iff

$A\mathbf{x} - \mathbf{b} \perp \text{sub-espaco } A\mathbf{x}$

\iff

$$A^T (A\mathbf{x} - \mathbf{b}) = \mathbf{0}$$

\iff

$$A^T A\mathbf{x} = A^T \mathbf{b}$$

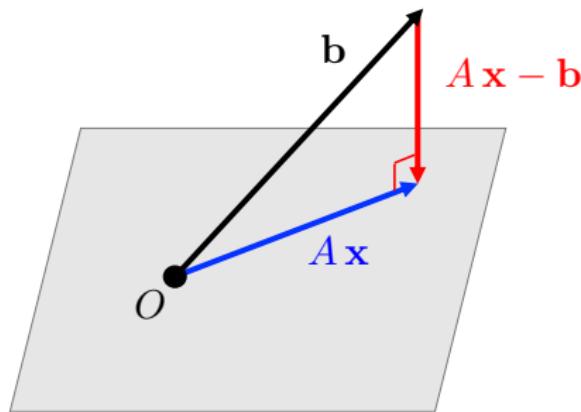
(equação normal)

A solução da equação normal é o vetor que minimiza o tamanho do vetor resíduo $A\mathbf{x} - \mathbf{b}$, i.e.,

$$\mathbf{x}^\dagger = \arg \min ||A\mathbf{x} - \mathbf{b}||$$



Interpretação geométrica



$\mathbf{b} \notin \text{sub-espaco } A\mathbf{x}$

\iff

$A\mathbf{x} - \mathbf{b} \perp \text{sub-espaco } A\mathbf{x}$

\iff

$$A^T (A\mathbf{x} - \mathbf{b}) = \mathbf{0}$$

\iff

$$A^T A\mathbf{x} = A^T \mathbf{b}$$

(equação normal)

A solução da equação normal é o vetor que minimiza o tamanho do vetor resíduo $A\mathbf{x} - \mathbf{b}$, i.e.,

$$\mathbf{x}^\dagger = \arg \min ||A\mathbf{x} - \mathbf{b}||$$

“projeção ortogonal de \mathbf{b} no sub-espaco $A\mathbf{x}$ ”



Uma visão mais analítica

Encontre o vetor que minimiza a função resíduo

$$\psi(\mathbf{x}) = \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2$$



Uma visão mais analítica

Encontre o vetor que minimiza a função resíduo

$$\psi(\mathbf{x}) = \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2$$

$$\psi(\mathbf{x}) = \frac{1}{2} (\mathbf{A}\mathbf{x} - \mathbf{b})^T (\mathbf{A}\mathbf{x} - \mathbf{b})$$



Uma visão mais analítica

Encontre o vetor que minimiza a função resíduo

$$\psi(\mathbf{x}) = \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2$$

$$\begin{aligned}\psi(\mathbf{x}) &= \frac{1}{2} (\mathbf{A}\mathbf{x} - \mathbf{b})^T (\mathbf{A}\mathbf{x} - \mathbf{b}) \\ &= \frac{1}{2} \left(\mathbf{x}^T \mathbf{A}^T \mathbf{A}\mathbf{x} - \mathbf{x}^T \mathbf{A}^T \mathbf{b} - \mathbf{b}^T \mathbf{A}\mathbf{x} + \mathbf{b}^T \mathbf{b} \right)\end{aligned}$$

Uma visão mais analítica

Encontre o vetor que minimiza a função resíduo

$$\psi(\mathbf{x}) = \frac{1}{2} \|\mathbf{Ax} - \mathbf{b}\|^2$$

$$\begin{aligned}\psi(\mathbf{x}) &= \frac{1}{2} (\mathbf{Ax} - \mathbf{b})^T (\mathbf{Ax} - \mathbf{b}) \\ &= \frac{1}{2} \left(\mathbf{x}^T \mathbf{A}^T \mathbf{Ax} - \mathbf{x}^T \mathbf{A}^T \mathbf{b} - \mathbf{b}^T \mathbf{Ax} + \mathbf{b}^T \mathbf{b} \right) \\ &= \frac{1}{2} \left(\mathbf{x}^T \mathbf{A}^T \mathbf{Ax} \right) - \left(\mathbf{A}^T \mathbf{b} \right)^T \mathbf{x} + \frac{1}{2} \mathbf{b}^T \mathbf{b}\end{aligned}$$

Uma visão mais analítica

Encontre o vetor que minimiza a função resíduo

$$\psi(\mathbf{x}) = \frac{1}{2} \|\mathbf{Ax} - \mathbf{b}\|^2$$

$$\begin{aligned}\psi(\mathbf{x}) &= \frac{1}{2} (\mathbf{Ax} - \mathbf{b})^T (\mathbf{Ax} - \mathbf{b}) \\ &= \frac{1}{2} \left(\mathbf{x}^T \mathbf{A}^T \mathbf{Ax} - \mathbf{x}^T \mathbf{A}^T \mathbf{b} - \mathbf{b}^T \mathbf{Ax} + \mathbf{b}^T \mathbf{b} \right) \\ &= \frac{1}{2} \left(\mathbf{x}^T \mathbf{A}^T \mathbf{Ax} \right) - \left(\mathbf{A}^T \mathbf{b} \right)^T \mathbf{x} + \frac{1}{2} \mathbf{b}^T \mathbf{b}\end{aligned}$$

$$\nabla \psi(\mathbf{x}) = \mathbf{0} \iff \mathbf{A}^T \mathbf{Ax} - \mathbf{A}^T \mathbf{b} = \mathbf{0}$$



Uma visão mais analítica

Encontre o vetor que minimiza a função resíduo

$$\psi(\mathbf{x}) = \frac{1}{2} \|\mathbf{Ax} - \mathbf{b}\|^2$$

$$\begin{aligned}\psi(\mathbf{x}) &= \frac{1}{2} (\mathbf{Ax} - \mathbf{b})^T (\mathbf{Ax} - \mathbf{b}) \\ &= \frac{1}{2} \left(\mathbf{x}^T \mathbf{A}^T \mathbf{Ax} - \mathbf{x}^T \mathbf{A}^T \mathbf{b} - \mathbf{b}^T \mathbf{Ax} + \mathbf{b}^T \mathbf{b} \right) \\ &= \frac{1}{2} \left(\mathbf{x}^T \mathbf{A}^T \mathbf{Ax} \right) - \left(\mathbf{A}^T \mathbf{b} \right)^T \mathbf{x} + \frac{1}{2} \mathbf{b}^T \mathbf{b}\end{aligned}$$

$$\nabla \psi(\mathbf{x}) = \mathbf{0} \iff \mathbf{A}^T \mathbf{Ax} - \mathbf{A}^T \mathbf{b} = \mathbf{0}$$

$$\iff \mathbf{A}^T \mathbf{Ax} = \mathbf{A}^T \mathbf{b}$$

(equação normal)



Uma visão mais analítica

Encontre o vetor que minimiza a função resíduo

$$\psi(\mathbf{x}) = \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2$$

$$\begin{aligned}\psi(\mathbf{x}) &= \frac{1}{2} (\mathbf{A}\mathbf{x} - \mathbf{b})^T (\mathbf{A}\mathbf{x} - \mathbf{b}) \\ &= \frac{1}{2} \left(\mathbf{x}^T \mathbf{A}^T \mathbf{A}\mathbf{x} - \mathbf{x}^T \mathbf{A}^T \mathbf{b} - \mathbf{b}^T \mathbf{A}\mathbf{x} + \mathbf{b}^T \mathbf{b} \right) \\ &= \frac{1}{2} \left(\mathbf{x}^T \mathbf{A}^T \mathbf{A}\mathbf{x} \right) - \left(\mathbf{A}^T \mathbf{b} \right)^T \mathbf{x} + \frac{1}{2} \mathbf{b}^T \mathbf{b}\end{aligned}$$

$$\nabla \psi(\mathbf{x}) = \mathbf{0} \iff \mathbf{A}^T \mathbf{A}\mathbf{x} - \mathbf{A}^T \mathbf{b} = \mathbf{0}$$

$$\iff \mathbf{A}^T \mathbf{A}\mathbf{x} = \mathbf{A}^T \mathbf{b}$$

(equação normal)

$\psi(\mathbf{x})$ é convexa, então a solução é um mínimo global!



Fundamentação teórica

Teorema (Método dos Mínimos Quadrados)

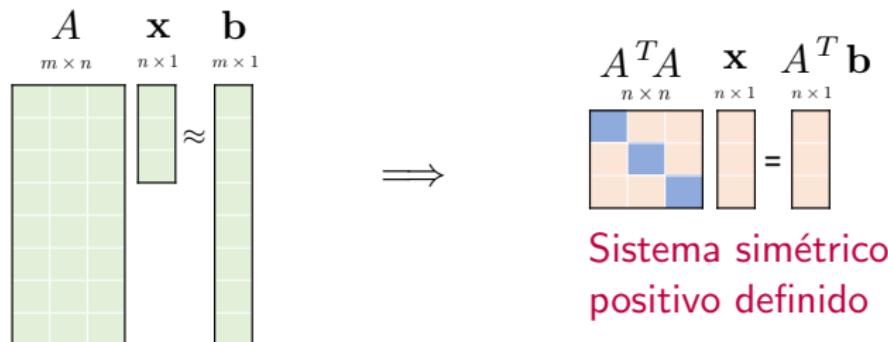
O problema de mínimos quadrados

$$\mathbf{x}^{\dagger} = \arg \min ||A \mathbf{x} - \mathbf{b}||,$$

onde $A \in \mathbb{R}^{m \times n}$ é uma matriz de posto n , tem uma única solução $\mathbf{x}^{\dagger} = (A^T A)^{-1} A^T \mathbf{b}$ que satisfaz a equação normal

$$A^T A \mathbf{x} = A^T \mathbf{b}.$$

A matriz $(A^T A)^{-1} A^T$ é chamada de pseudo-inversa de Moore-Penrose.



Mínimos quadrados via Cholesky

$$\begin{cases} A^T A \mathbf{x} = A^T \mathbf{b} \\ A^T A = G G^T \end{cases} \iff \begin{cases} G \mathbf{y} = A^T \mathbf{b} \\ G^T \mathbf{x} = \mathbf{y} \end{cases}$$

Receita computacional

1. Construir $A^T A$ e $A^T \mathbf{b}$
2. Calcular $A^T A = G G^T$
3. Resolver $G \mathbf{y} = A^T \mathbf{b}$
4. Resolver $G^T \mathbf{x} = \mathbf{y}$

$$\begin{matrix} G_{n \times n} & \mathbf{y}_{n \times 1} & A^T \mathbf{b}_{n \times 1} \\ \text{---} & \text{---} & \text{---} \\ \text{---} & \text{---} & \text{---} \end{matrix} = \begin{matrix} \text{---} & \text{---} & \text{---} \\ \text{---} & \text{---} & \text{---} \\ \text{---} & \text{---} & \text{---} \end{matrix}$$

$$\begin{matrix} G^T_{n \times n} & \mathbf{x}_{n \times 1} & \mathbf{y}_{n \times 1} \\ \text{---} & \text{---} & \text{---} \\ \text{---} & \text{---} & \text{---} \end{matrix} = \begin{matrix} \text{---} & \text{---} & \text{---} \\ \text{---} & \text{---} & \text{---} \\ \text{---} & \text{---} & \text{---} \end{matrix}$$

$$\text{flops (MQ Cholesky)} \sim m n^2 + \frac{1}{3} n^3$$



Mínimos quadrados via fatoração QR

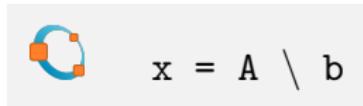
$$\begin{cases} \mathbf{x} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b} \\ \mathbf{A} = \mathbf{Q} \mathbf{R} \end{cases} \iff \mathbf{R} \mathbf{x} = \mathbf{Q}^T \mathbf{b}$$

Receita computacional

1. Calcular $\mathbf{A} = \mathbf{Q} \mathbf{R}$
2. Calcular $\mathbf{Q}^T \mathbf{b}$
3. Resolver $\mathbf{R} \mathbf{x} = \mathbf{Q}^T \mathbf{b}$

$$\begin{matrix} \mathbf{R} & \mathbf{x} \\ n \times n & n \times 1 \end{matrix} = \begin{matrix} \mathbf{Q}^T & \mathbf{b} \\ n \times n & n \times 1 \end{matrix}$$

$$\text{flops}(\mathbf{M} \mathbf{Q} \text{ QR}) \sim 2m n^2 - \frac{2}{3} n^3$$



“backslash” usa fatoração QR para mínimos quadrados



Mínimos quadrados via fatoração SVD

$$\left\{ \begin{array}{lcl} A^T A \mathbf{x} & = & A^T \mathbf{b} \\ A & = & U \Sigma V^T \end{array} \right. \iff \left\{ \begin{array}{lcl} \Sigma \mathbf{w} & = & U^T \mathbf{b} \\ \mathbf{x} & = & V \mathbf{w} \end{array} \right.$$

Receita computacional

1. Calcular $A = U \Sigma V^T$
2. Calcular $U^T \mathbf{b}$
3. Resolver $\Sigma \mathbf{w} = U^T \mathbf{b}$
4. Calcular $\mathbf{x} = V \mathbf{w}$

$$\begin{matrix} \Sigma & & \mathbf{w} & & U^T & & \mathbf{b} \\ n \times n & & n \times 1 & & n \times n & & n \times 1 \\ \boxed{\text{---}} & & \boxed{\text{---}} & = & \boxed{\text{---}} & & \boxed{\text{---}} \end{matrix}$$
$$\begin{matrix} \mathbf{x} & & V & & \mathbf{w} & & \\ n \times 1 & & n \times n & & n \times 1 & & \\ \boxed{\text{---}} & & \boxed{\text{---}} & = & \boxed{\text{---}} & & \boxed{\text{---}} \end{matrix}$$

$$\text{flops (MQ SVD)} \sim 2 m n^2 + 11 n^3$$



Custo do problema de mínimos quadrados ($m \gg n$)

fatoração	flops	estabilidade	custo
Cholesky	$\sim m n^2 + \frac{1}{3} n^3$	***	$1 \times \$$
QR	$\sim 2 m n^2 - \frac{2}{3} n^3$	****	$(3m/n-2) \times \$$
SVD	$\sim 2 m n^2 + 11n^3$	*****	$(3m/n+33) \times \$$



Queda livre perto da superfície da Terra



t (s)	s (cm)
0,1	5,49
0,2	-9,45
0,3	-31,39
0,4	-75,59
0,5	-113,69

$$s(t) = \frac{1}{2} g t^2 + v_0 t + s_0$$

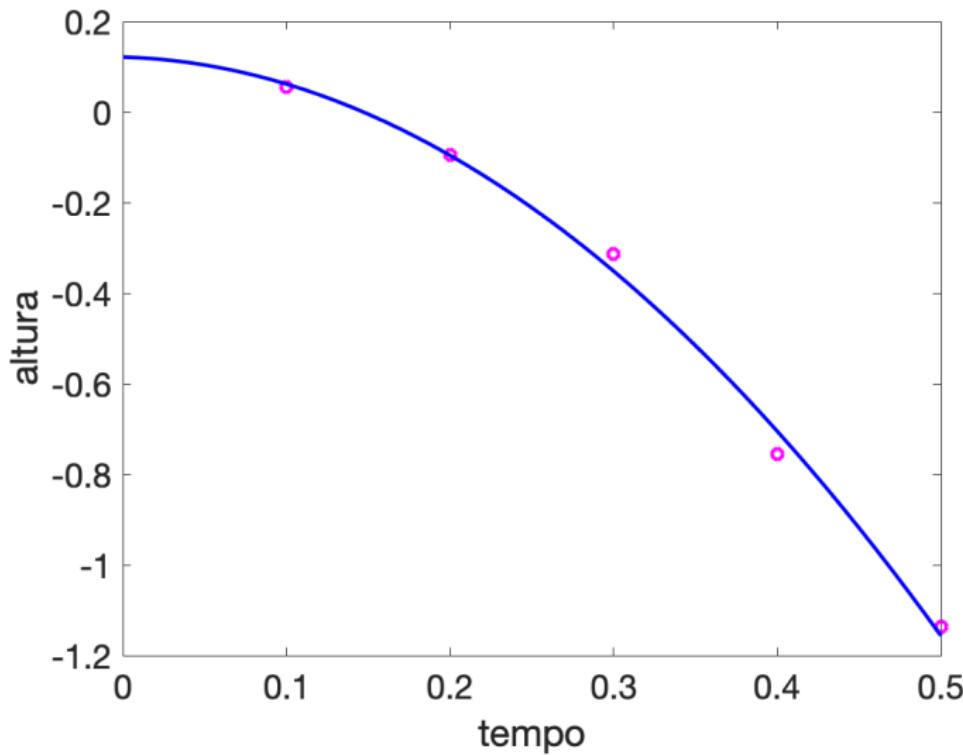


Queda livre perto da superfície da Terra

```
clc  
clear  
  
time = [0.1 0.2 0.3 0.4 0.5];  
height = [5.49 -9.45 -31.39 -75.59 -113.69]*1e-2;  
  
A = [0.5*time.^2; time; ones(size(time))]';  
b = height';  
x = A\b;  
  
xfit = 0:0.01:0.5;  
yfit = 0.5*x(1)*xfit.^2 + x(2)*xfit + x(3);  
  
plot(time,height,'om',xfit,yfit,'-b','LineWidth',2);  
xlabel('tempo')  
ylabel('altura')  
set(gca,'FontSize',18);
```



Queda livre perto da superfície da Terra



Ajuste de curva robusto

Outras medidas de erro são possíveis (e pertinentes):

- Erro médio quadrático

$$E_2 = \left(\frac{1}{m} \sum_{k=1}^m (y_k - f(x_k))^2 \right)^{1/2}$$

- Erro médio

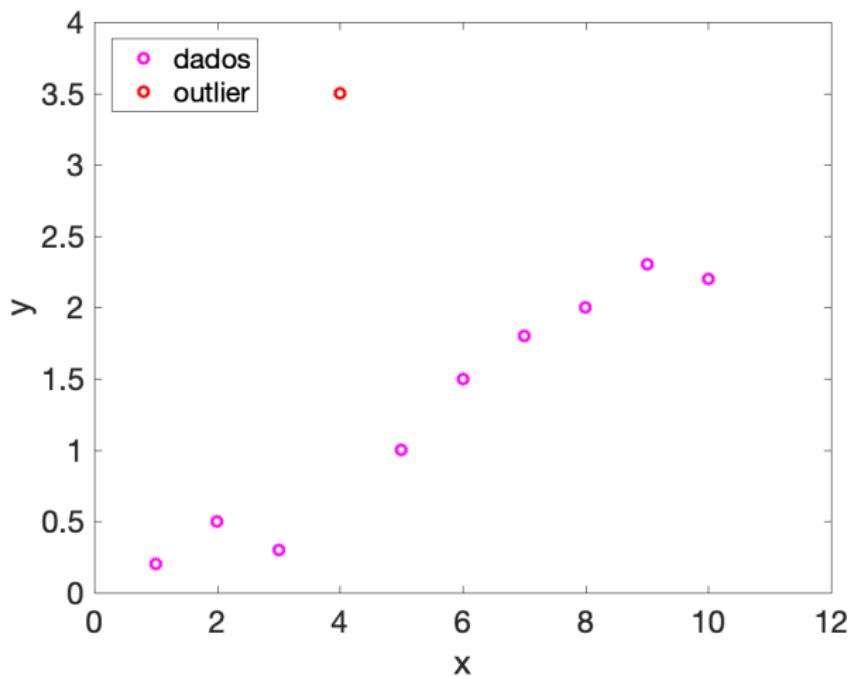
$$E_1 = \frac{1}{m} \sum_{k=1}^m |y_k - f(x_k)|$$

- Erro máximo

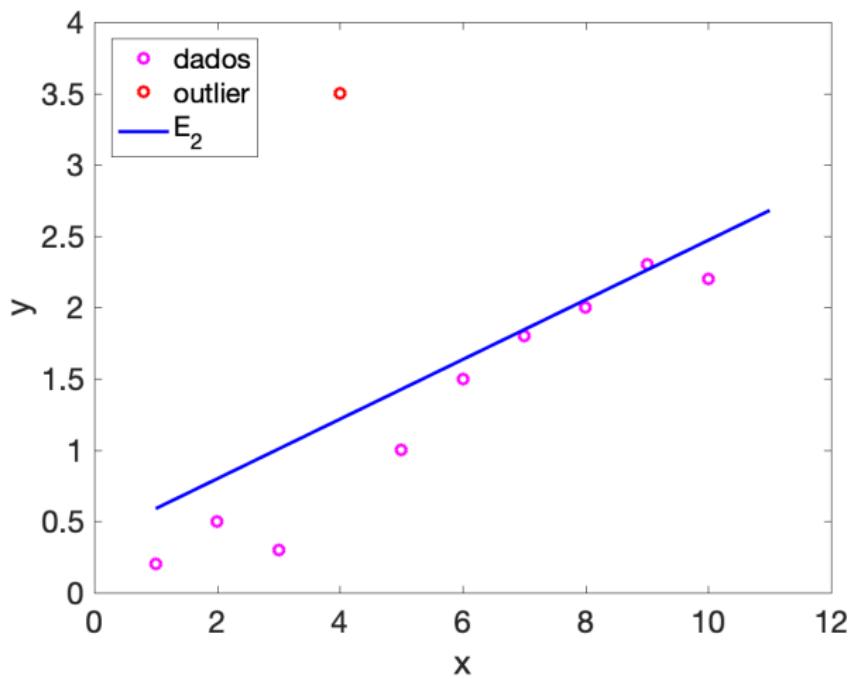
$$E_\infty = \max_{1 \leq k \leq m} |y_k - f(x_k)|$$



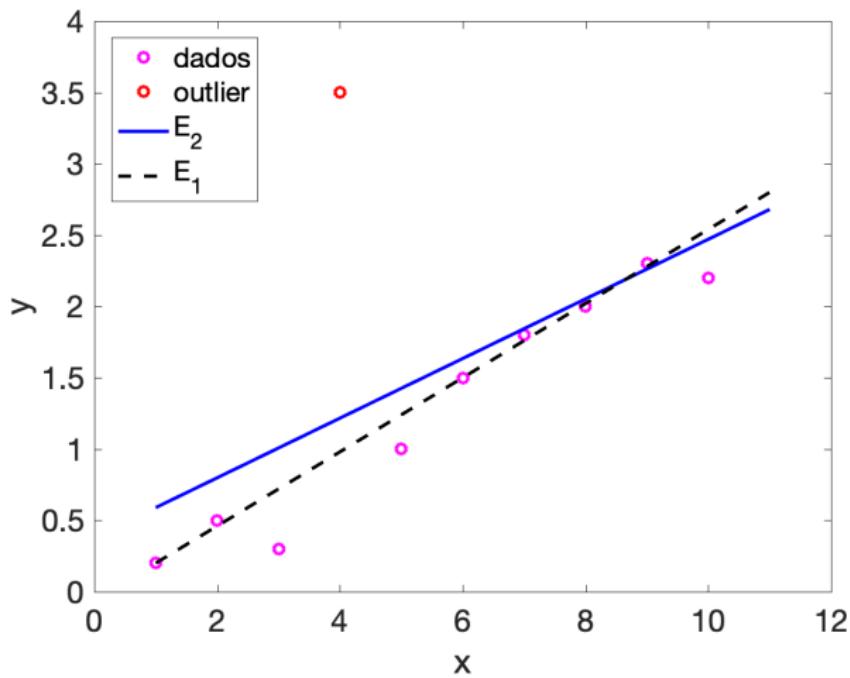
Ajuste de curva robusto



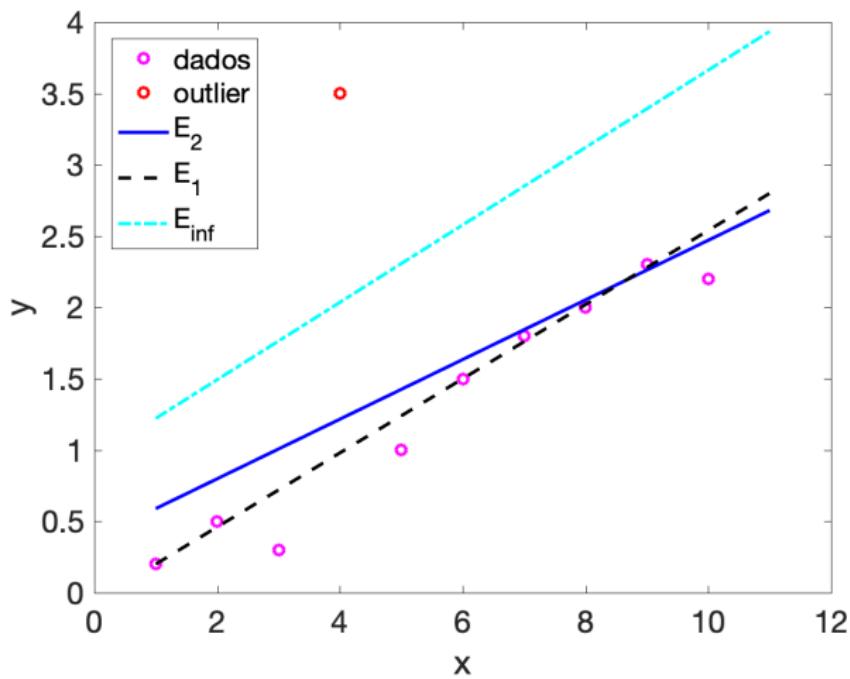
Ajuste de curva robusto



Ajuste de curva robusto



Ajuste de curva robusto



Ajuste de curva robusto

```
clc; clear; close all

m      = 10;
xdata = [1 2 3 4 5 6 7 8 9 10];
ydata = [0.2 0.5 0.3 3.5 1.0 1.5 1.8 2.0 2.3 2.2];

E_2    = @(p) sqrt(sum(abs(p(1)*xdata + p(2) - ydata).^2)/m);
E_1    = @(p) sum(abs(p(1)*xdata + p(2) - ydata)) / m;
E_inf = @(p) max(abs(p(1)*xdata + p(2) - ydata));

p2    = fminsearch(E_2 , [1 1]);
p1    = fminsearch(E_1 , [1 1]);
pinf = fminsearch(E_inf, [1 1]);

xfit = 1:0.01:11;
y_2   = polyval( p2,xfit);
y_1   = polyval( p1,xfit);
y_inf = polyval(pinf,xfit);
```

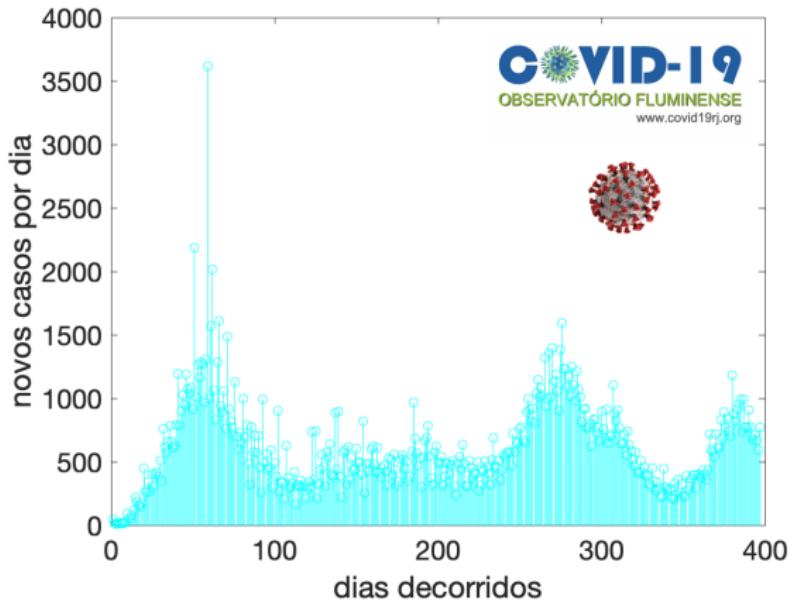


Ajuste de curva robusto

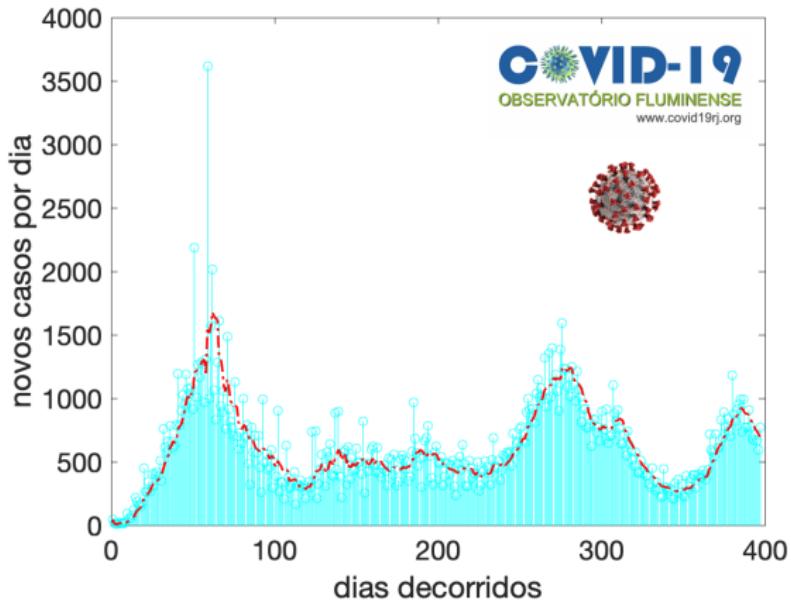
```
plot(xdata,ydata,'om',xdata(4),ydata(4),'or','LineWidth',2);
hold on
plot(xfit,y_2 ,'-b','LineWidth',2);
plot(xfit,y_1 ,'--k','LineWidth',2);
plot(xfit,y_inf,'-.c','LineWidth',2);
hold off
xlabel('x')
ylabel('y')
set(gca,'FontSize',18);
legend('dados','outlier','E_2','E_1','E_{inf}')
```



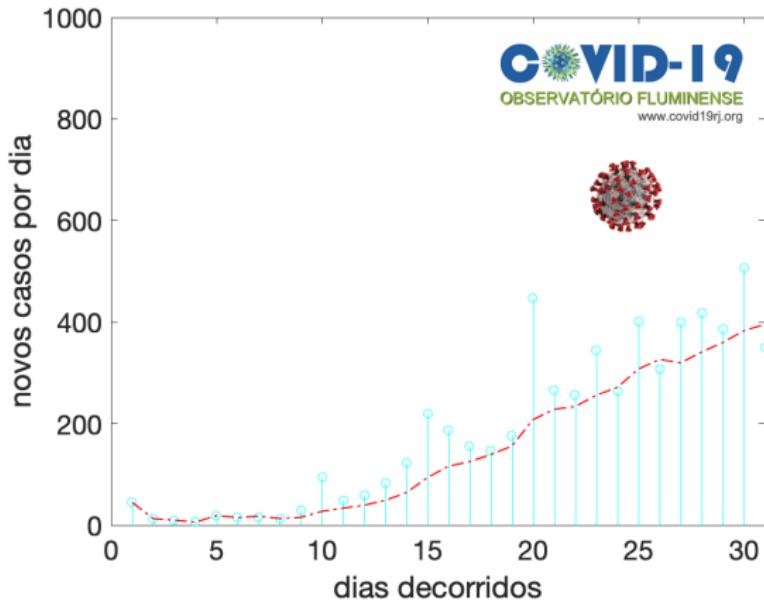
COVID-19 na cidade do Rio de Janeiro



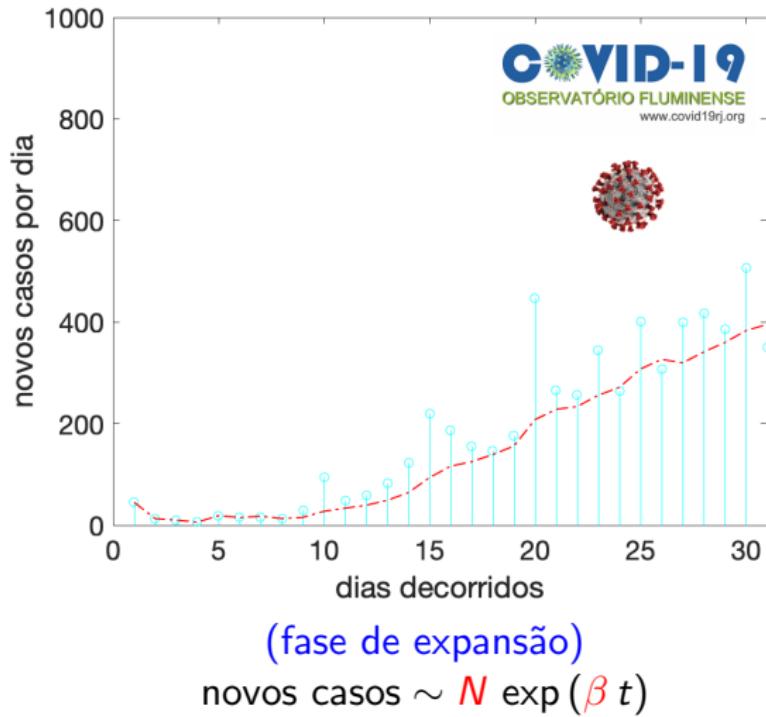
COVID-19 na cidade do Rio de Janeiro



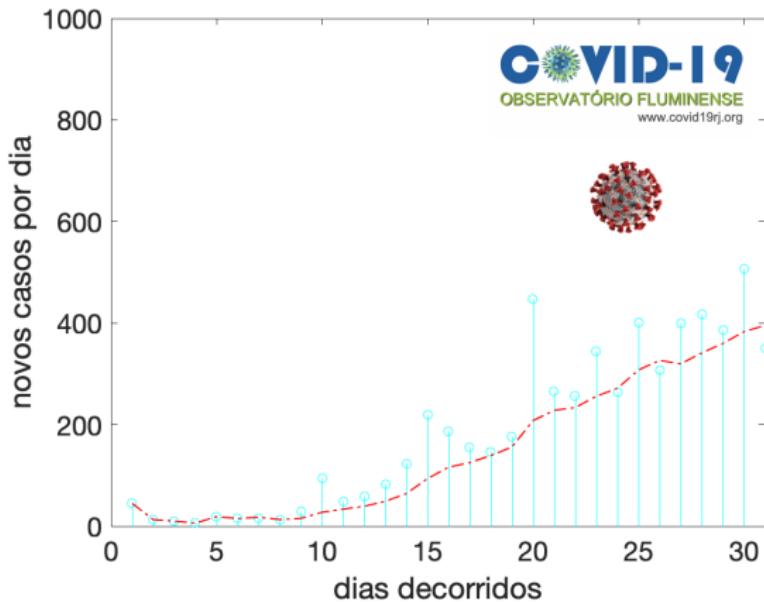
COVID-19 na cidade do Rio de Janeiro



COVID-19 na cidade do Rio de Janeiro



COVID-19 na cidade do Rio de Janeiro



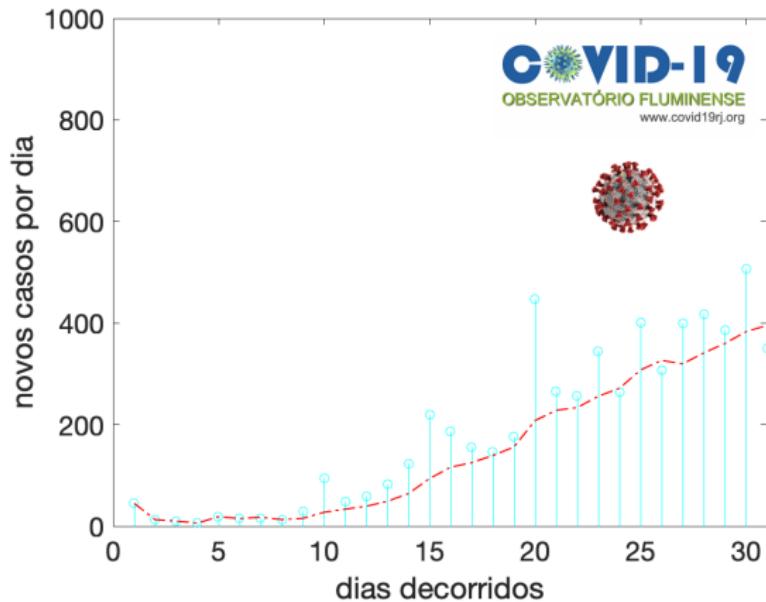
(fase de expansão)

novos casos $\sim N \exp(\beta t)$

$$y = N \exp(\beta t)$$



COVID-19 na cidade do Rio de Janeiro



(fase de expansão)

$$\text{novos casos} \sim N \exp(\beta t)$$

$$y = N \exp(\beta t) \implies \ln y = \beta t + \ln N$$

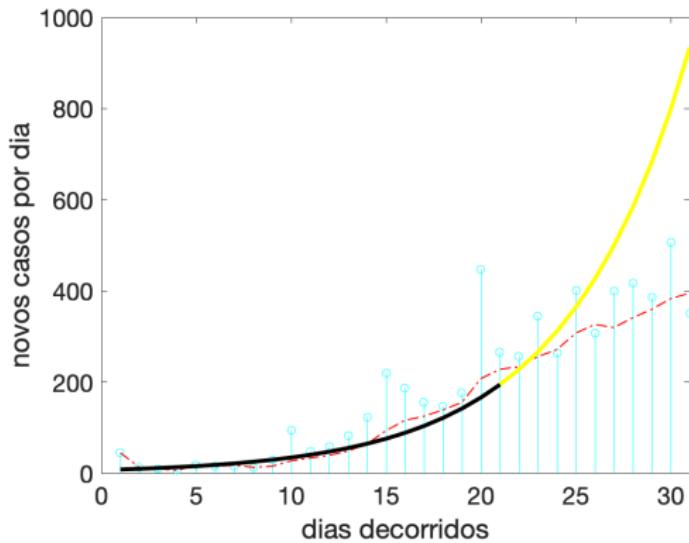


COVID-19 na cidade do Rio de Janeiro

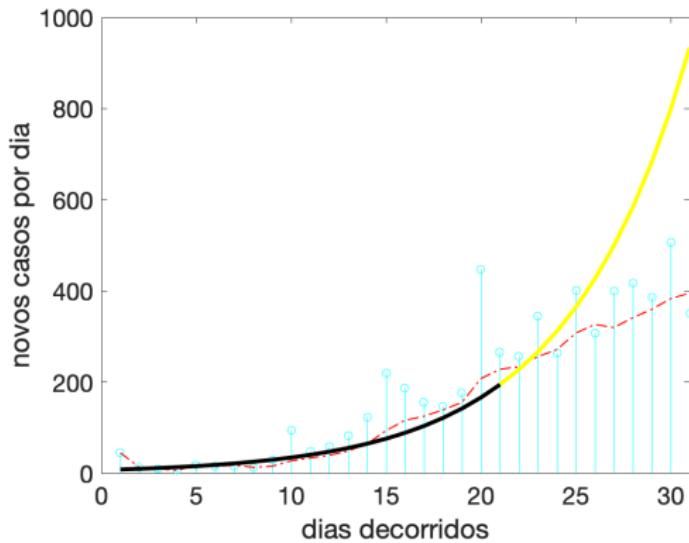
```
clc; clear; close all
dados = load('covid19_casos_rio_2020_2021.csv');
mm7 = dados;
for j=7:length(dados)
    mm7(j) = sum(dados(j-6:j))/7;
end
xdata = 1:21; ydata = mm7(xdata)';
A = [xdata; ones(size(xdata))]';
b = log(ydata)';
x = A\b;
yfit = @(z) exp(x(2))*exp(x(1)*z);
stem(dados(1:31), 'oc', 'LineWidth', 0.5)
hold on
plot(1:31, mm7(1:31), '-r', 'LineWidth', 1)
plot(yfit(1:31), '-y', 'LineWidth', 3)
plot(yfit(1:21), '-k', 'LineWidth', 3)
hold off
xlabel('dias decorridos')
ylabel('novos casos por dia')
set(gca, 'FontSize', 18)
```



COVID-19 na cidade do Rio de Janeiro



COVID-19 na cidade do Rio de Janeiro



Outra possibilidade, curva logística (via regressão não linear):

$$I(x) = \frac{\alpha L e^{-\alpha(t-t_m)}}{(1 + e^{-\alpha(t-t_m)})^2}$$

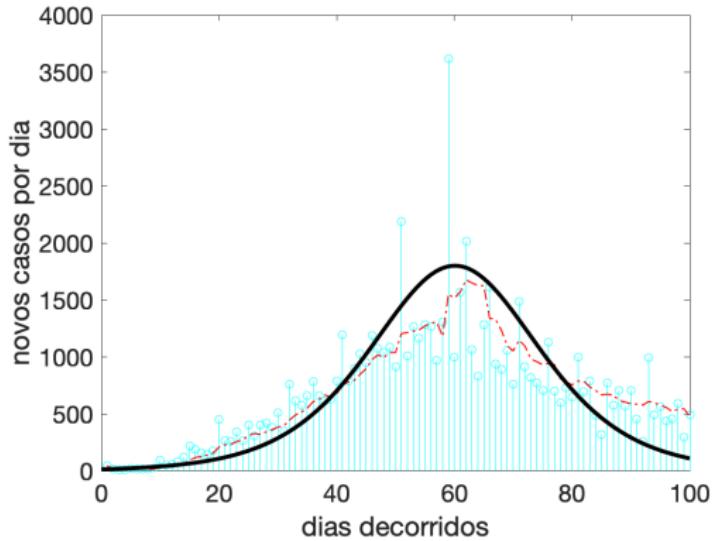


COVID-19 na cidade do Rio de Janeiro

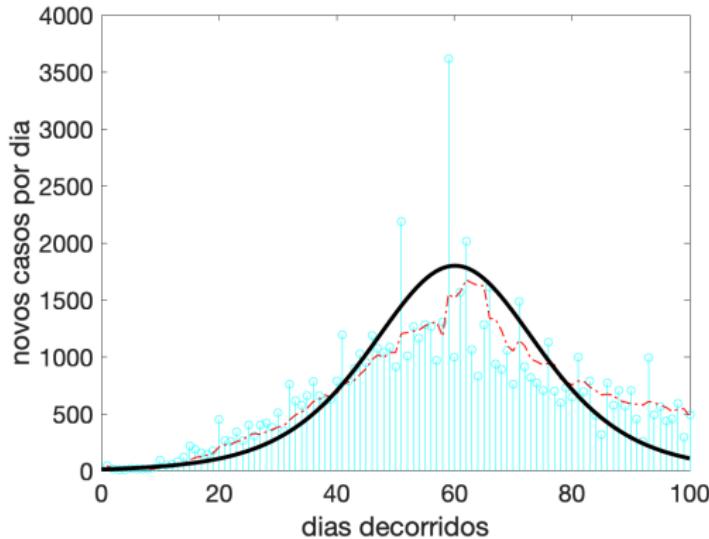
```
clc; clear; close all
dados = load('covid19_casos_rio_2020_2021.csv');
mm7 = dados;
for j=7:length(dados)
    mm7(j) = sum(dados(j-6:j))/7;
end
xdata = 1:100; ydata = mm7(xdata)';
N = @(x,p) p(1)*p(2)*exp(-p(1)*(x-p(3)));
D = @(x,p) (1+exp(-p(1)*(x-p(3))).^2;
I = @(x,p) N(x,p)./D(x,p);
E_2 = @(p) sqrt(sum(I(xdata,p) - ydata).^2/length(ydata));
p2 = fminsearch(E_2,[0.1 8e4 55]);
yfit = I(0:0.1:100,p2);
stem(dados(1:100), 'oc', 'LineWidth', 0.5)
hold on
plot(1:100 , mm7(1:100), '-.r', 'LineWidth', 1)
plot(0:0.1:100, yfit, '-k' , 'LineWidth', 3)
hold off
xlabel('dias decorridos')
ylabel('novos casos por dia')
set(gca, 'FontSize', 18)
```



COVID-19 na cidade do Rio de Janeiro



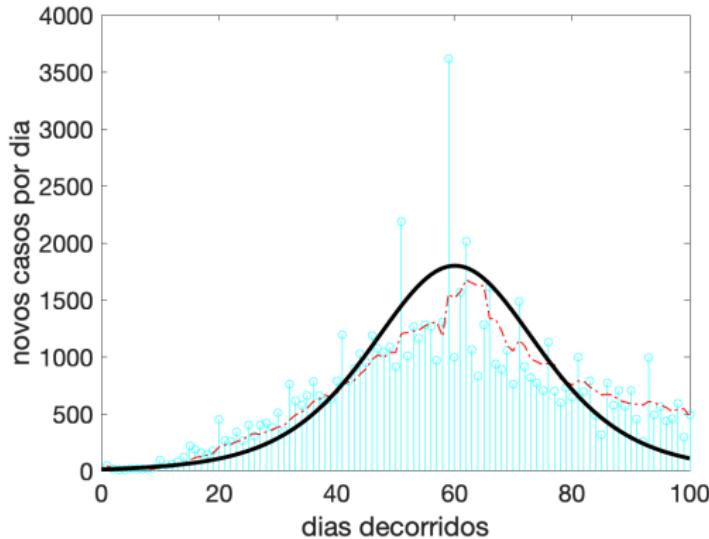
COVID-19 na cidade do Rio de Janeiro



E para ajustar várias ondas?



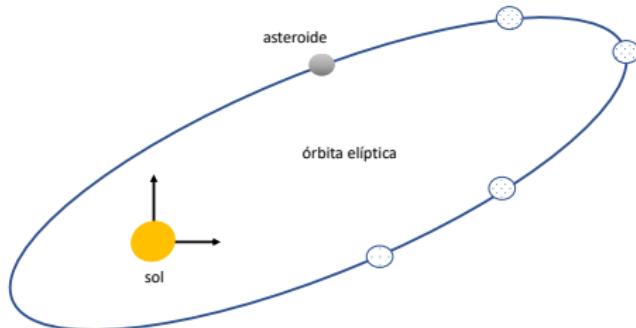
COVID-19 na cidade do Rio de Janeiro



E para ajustar várias ondas?
Tente somar várias curvas logísticas!



Órbita de um asteroide



5 observações estão disponíveis



x (ua)	y (ua)
8,025	8,310
10,170	6,355
11,202	3,212
10,736	0,375
9,092	-2,267

ua = unidade astronômica

$$Ax^2 + Bxy + Cy^2 + Dx + Ey + 1 = 0$$

Determine a equação da órbita!

Algumas bases de dados



Pegue alguns dados nessas bases e construa regressores!

Como citar esse material?

A. Cunha, *Ajuste de Curvas*,

Universidade do Estado do Rio de Janeiro – UERJ, 2021.



@AmericoCunhaJr



@AmericoCunhaJr



@AmericoCunhaJr



@AmericoCunhaJr

Essas notas de aula podem ser compartilhadas nos termos da licença Creative Commons BY-NC-ND 3.0, com propósitos exclusivamente educacionais.

