

# Matemática numérica

Pedro H A Konzen

24 de maio de 2018

# Licença

Este trabalho está licenciado sob a Licença Atribuição-CompartilhaIgual 4.0 Internacional Creative Commons. Para visualizar uma cópia desta licença, visite <http://creativecommons.org/licenses/by-sa/4.0/> ou mande uma carta para Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.

# Prefácio

# Sumário

Capa	i
Licença	ii
Prefácio	iii
Sumário	iv
1 Introdução	1
2 Aproximação por mínimos quadrados	2
2.1 Problemas lineares . . . . .	2
2.1.1 Equações normais . . . . .	2
Referências Bibliográficas	7

# Capítulo 1

## Introdução

Em construção ...

## Capítulo 2

# Aproximação por mínimos quadrados

### 2.1 Problemas lineares

Dado um conjunto de  $n$  pontos  $\{(x_i, y_i)\}_{i=1}^n$ ,  $x_i \neq x_j$  para  $i \neq j$ , e uma família de  $m \leq n$  funções  $\{f_i(x)\}_{i=1}^m$ , o problema linear de aproximação por mínimos quadrados consiste em determinar os  $m$  coeficientes  $\{c_i\}_{i=1}^m$  tal que a função

$$f(x) = \sum_{i=1}^m c_i f_i(x) = c_1 f_1(x) + c_2 f_2(x) + c_3 f_3(x) + \cdots + c_m f_m(x) \quad (2.1)$$

aproxime o conjunto de pontos dados no sentido de mínimos quadrados, i.e. os coeficientes que são solução do seguinte problema linear de minimização

$$\min_{\{c_1, c_2, \dots, c_m\}} \sum_{i=1}^n (y_i - f_i(x_i))^2. \quad (2.2)$$

#### 2.1.1 Equações normais

Afim de resolver o problema de mínimos quadrados (2.2), observamos que o erro quadrático

$$E := \sum_{i=1}^n (y_i - f_i(x_i))^2 = \sum_{i=1}^n \left( y_i - \sum_{j=1}^m c_j f_j(x_i) \right)^2 \quad (2.3)$$

tem seu mínimo para os coeficientes  $c_k$  tais que

$$\frac{\partial E}{\partial c_k} = 0, \quad k = 1, 2, 3, \dots, m. \quad (2.4)$$

Calculando estas derivadas parciais, temos

$$\begin{aligned}
\frac{\partial E}{\partial c_k} &= \frac{\partial}{\partial c_k} \sum_{i=1}^n \left( y_i - \sum_{j=1}^m c_j f_j(x_i) \right)^2 \\
&= 2 \sum_{i=1}^n \left( y_i - \sum_{j=1}^m c_j f_j(x_i) \right) f(x_k) \\
&= 2 \sum_{i=1}^n y_i f(x_k) - 2 \sum_{i=1}^n \sum_{k=1}^m c_j f_j(x_i) f(x_k), \quad k = 1, 2, 3, \dots, m.
\end{aligned} \tag{2.5}$$

Desta forma, a equação (2.4) é equivalente a resolver o seguinte sistema linear

$$\sum_{i=1}^n \sum_{k=1}^m c_j f_j(x_i) f(x_k) = \sum_{i=1}^n y_i f(x_k), \quad k = 1, 2, 3, \dots, m. \tag{2.6}$$

Agora, denotando

$$A := \begin{bmatrix} f_1(x_1) & f_2(x_1) & \cdots & f_m(x_1) \\ f_1(x_2) & f_2(x_2) & \cdots & f_m(x_2) \\ \vdots & \vdots & \vdots & \vdots \\ f_1(x_n) & f_2(x_n) & \cdots & f_m(x_n) \end{bmatrix} \tag{2.7}$$

o sistema linear (2.6) pode ser escrito na seguinte forma matricial

$$A^T A c = A^T y, \tag{2.8}$$

onde  $c = (c_1, c_2, \dots, c_m)$  é o vetor dos coeficientes e  $y = (y_1, y_2, \dots, y_n)$  é o vetor das abscissas dos pontos dados.

**Exemplo 2.1.1.** (Ajuste de polinômios) Considere o problema de ajustar o conjunto de pontos

hline $i$	1	2	3	4
$x_i$	-1	0	1	1,5
$y_i$	1,2	-0,1	0,7	2,4

por um polinômio quadrático da forma

$$p(x) = p_1 x^2 + p_2 x + p_n \tag{2.9}$$

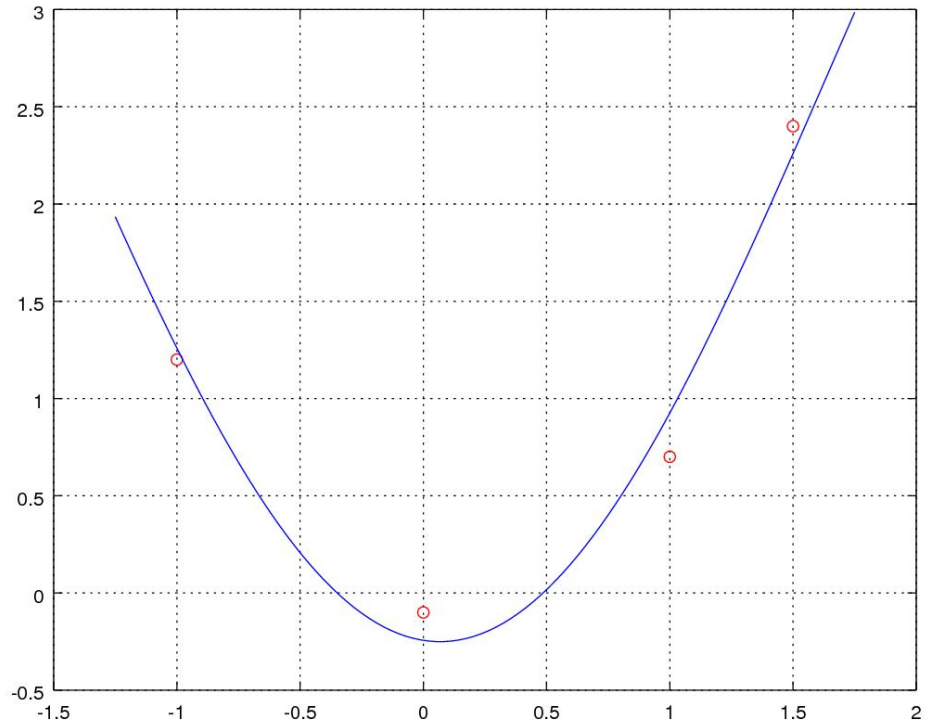


Figura 2.1: Esboço da curva ajustada no Exemplo 2.1.2.

no sentido de mínimos quadrados.

Neste caso, a família de funções do problema de mínimos quadrados é  $f_1(x) = x^2$ ,  $f_2(x) = x$  e  $f_3(x) = 1$ . Assim sendo, os coeficientes  $p = (p_1, p_2, p_3)$  são solução do seguinte sistema linear

$$A^T A p = A^T y, \quad (2.10)$$

onde  $y = (y_1, y_2, y_3)$  e

$$A := \begin{bmatrix} x_1^2 & x_1 & 1 \\ x_2^2 & x_2 & 1 \\ x_3^2 & x_3 & 1 \\ x_4^2 & x_4 & 1 \end{bmatrix}. \quad (2.11)$$

Emfim, resolvendo as equações normais (2.10), obtemos

$$p(x) = 1,25x^2 - 0,188x - 0,203. \quad (2.12)$$



A Figura 2.2 mostra um esboço dos pontos (em vermelho) e do polinômio ajustado (em azul).

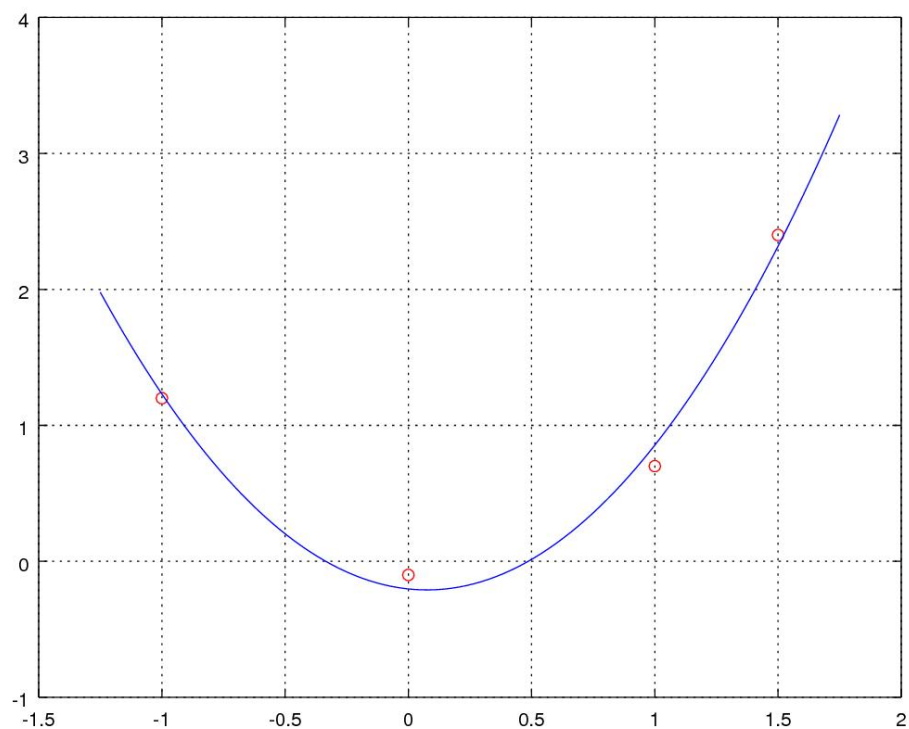


Figura 2.2: Esboço do polinômio ajustado no Exemplo 2.1.1.

Os coeficientes e um esboço do polinômio ajustado podem ser obtidos no GNU Octave com o seguinte código:

```
#pontos
x = [-1 0 1 1.5]';
y = [1.2, -0.1, 0.7, 2.4]';

#resol. as eqs. normais
A = [x.^2 x.^1 x.^0];
p = inv(A'*A)*A'*y

#esboco do pol. ajustado
xx = linspace(-1.25,1.75);
plot(x,y,'ro',...
```

```
xx,polyval(p,xx));grid
```

**Exemplo 2.1.2.** (Ajuste de curvas) Consideremos o mesmo conjunto de pontos do exemplo anterior (Exemplo 2.1.1). Aqui, vamos ajustar uma curva da forma

$$f(x) = c_1 \sin(x) + c_2 \cos(x) + c_3 \quad (2.13)$$

no sentido de mínimos quadrados. Para tanto, formamos a matrix

$$A := \begin{bmatrix} \sin(x_1) & \cos(x_1) & 1 \\ \sin(x_2) & \cos(x_2) & 1 \\ \sin(x_3) & \cos(x_3) & 1 \\ \sin(x_4) & \cos(x_4) & 1 \end{bmatrix} \quad (2.14)$$

e, então, resolvemos as equações normais  $A^T A c = A^T y$  para o vetor de coeficientes  $c = (c_1, c_2)$ . Fazendo isso, obtemos  $c_1 = -0,198$ ,  $c_2 = -2,906$  e  $c_3 = 2,662$ . A Figura ?? mostra um esboço da curva ajustada (linha azul) aos pontos dados (círculos vermelhos).

Os coeficientes e um esboço do polinômio ajustado podem ser obtidos no GNU Octave com o seguinte código:

```
#pontos
x = [-1 0 1 1.5]';
y = [1.2, -0.1, 0.7, 2.4]';

#resol. as eqs. normais
A = [sin(x) cos(x) ones(4,1)];
c = inv(A'*A)*A'*y

#curva ajustada
f = @(x) c(1)*sin(x) + c(2)*cos(x) + c(3)

#esboco do pol. ajustado
xx = linspace(-1.25,1.75);
plot(x,y,'ro',...
      xx,f(xx));grid
```

# Referências Bibliográficas

- [1] R.L. Burden, J.D. Faires, and A.M. Burden. *Análise Numérica*. CENGAGE Learning, 10. ed. edition, 2015.