

O Poderoso SciLab - Parte II

Mário Leite

...

Continuando esta trilogia sobre o SciLab, apresentamos mais um dos recursos importantes que esta ferramenta oferece: a criação e desenho de gráficos estatísticos. O SciLab oferece as mais variadas funções, de diversos tipos, para desenhos de curvas.

Para curvas em 2D:

- **plot(x, y)**: Plota uma curva **f(y)** em função de x.
- **plot2d(x, y)**: Versão mais flexível do plot, com opções adicionais.
- **plot2d2(x, y)**: Semelhante ao plot2d, mas para funções em degraus (constantes por segmentos).
- **plot2d3(x, y)**: Plota gráficos de barras verticais.
- **plot2d4(x, y)**: Plota gráficos com setas.

...

Para curvas em 3D:

- **plot3d(x, y, z)**: Plota gráficos 3D.
- **surf(x, y, z)**: Plota uma superfície 3D.
- **mesh(x, y, z)**: Plota uma malha 3D.
- **champ(x, y, z)**: Plota um campo vetorial 3D.
- **comet(x, y, z)**: Plota um gráfico do tipo cometa.
- **contour3d(z)**: Plota contornos em 3D.

Todos esses tipos de curvas ainda contam com recursos de espessura, cor e padrões. E além de funções para gráficos, também oferece funções de personalização e estilos, tendo na versão atual (2025.1.0) novidades como: recursos de função *lambda*, manipulação de complexos, *handles* de gráficos, LaTeX em gráficos (renderiza textos em gráficos), funções de ajustes em gráficos, etc.

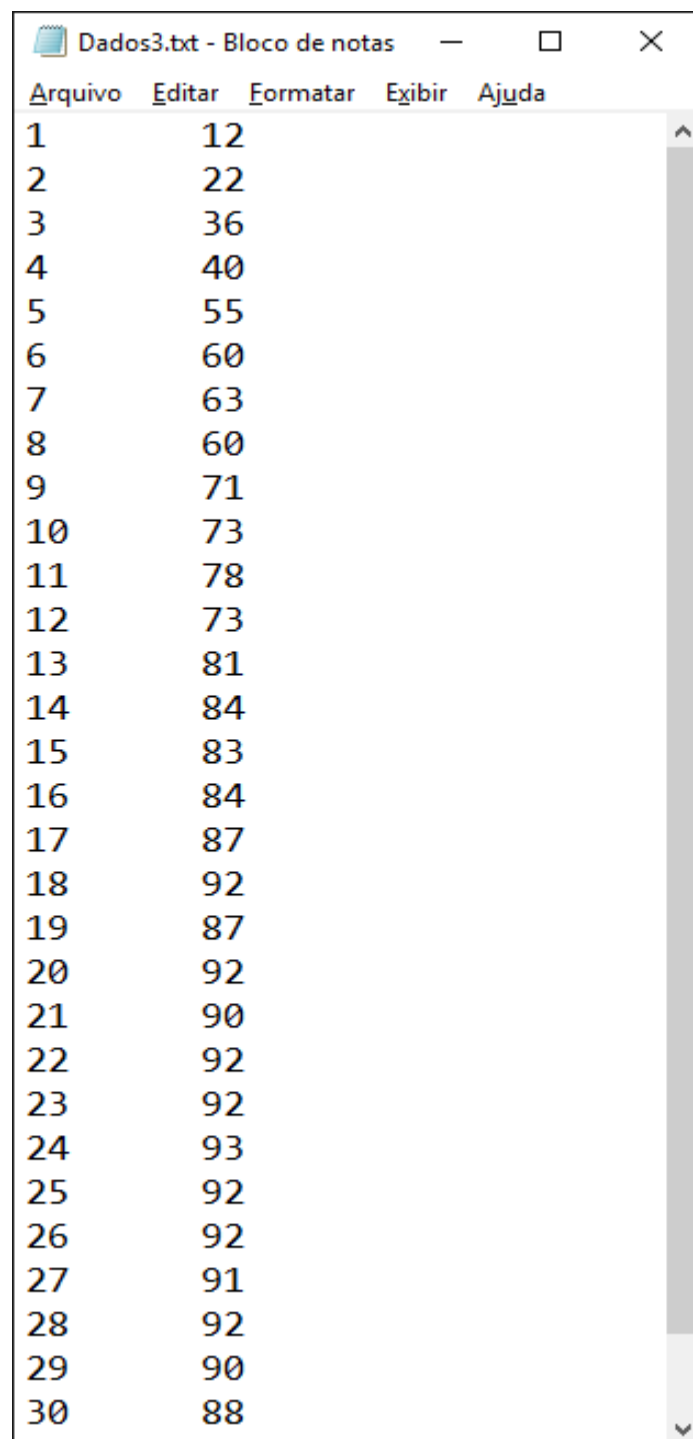
Aqui vamos mostrar um programa simples de ajuste de curva, tirado da leitura de alguns pontos experimentais obtidos. Os pontos em vermelho são os dados obtidos experimentalmente, e os asteriscos em azul forma a curva de ajuste dos pontos experimentais lido no arquivo “Dados3.txt”, que pode ser visto na **tabela 1**. O programa “**Regressao**” mostra o poder do SciLab, fazendo a leitura do arquivo-texto, contendo 30 linhas de dados experimentais **f(x,y)**. Em seguida faz a plotagem desses pontos na cor vermelha, e depois faz a regressão mostrando a curva ajustada desses pontos na cor azul. A **figura 1a** mostra a saída do programa em tela gráfica, exibindo os pontos experimentais (em pontos **vermelhos**) e a curva ajustada (em pontos **azuis**). A **figura 1b** mostra a saída do programa no terminal, exibindo os valores dos coeficientes de *correlação* e de *determinação*, indicando que o ajuste da curva está ótimo (valores maiores que 0.9)

Continua na Parte III

O livro “**SciLab: Uma Abordagem Prática e Didática**”, do autor, pode ser uma grande ajuda para os programadores iniciantes que querem aprender sobre esta ferramenta.

Veja mais detalhes deste livro no *link*:

<https://www.amazon.com.br/SciLab-Uma-Abordagem-Pr%C3%A1tica-Did%C3%A1tica/dp/8539906570>



A screenshot of a Notepad window titled "Dados3.txt - Bloco de notas". The window contains a table with 30 rows of experimental data. The table has two columns: an index from 1 to 30 and corresponding numerical values. The menu bar includes "Arquivo", "Editar", "Formatar", "Exibir", and "Ajuda". A vertical scrollbar is visible on the right side of the text area.

1	12
2	22
3	36
4	40
5	55
6	60
7	63
8	60
9	71
10	73
11	78
12	73
13	81
14	84
15	83
16	84
17	87
18	92
19	87
20	92
21	90
22	92
23	92
24	93
25	92
26	92
27	91
28	92
29	90
30	88

Tabela 1 - Dados Experimentais: arquivo "Dados3.txt"

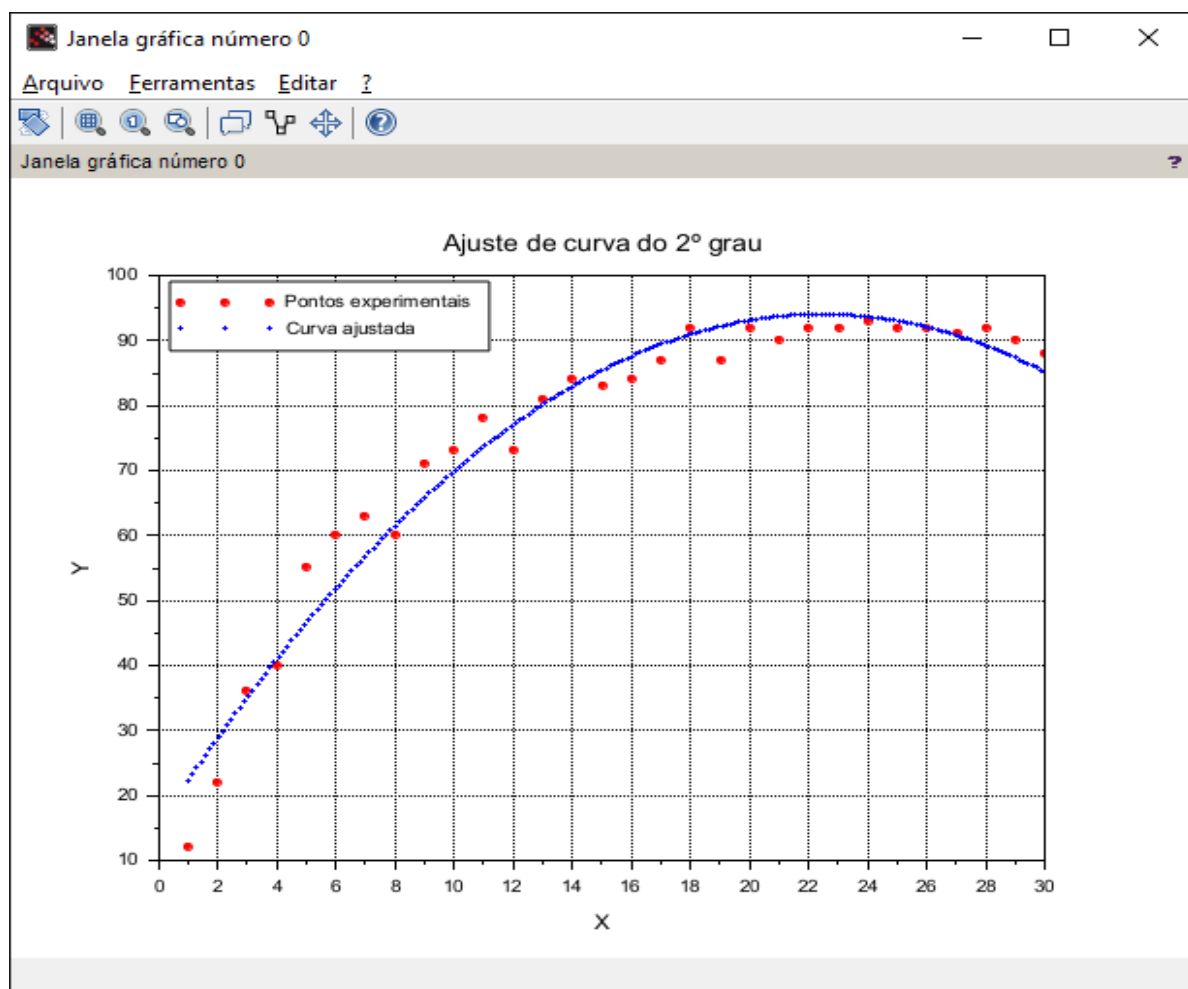


Figura 1a - Saída do programa em tela gráfica: os pontos experimentais e a curva de ajuste

```

Scilab 2025.1.0 Console

Arquivo  Editar  Controle  Aplicativos  ?

Scilab 2025.1.0 Console

Execução de iniciação:
  carregando o ambiente inicial

--> exec('D:\Cantinho da Programação\Códigos\SciLab\Regressao2.sce', -1)
"Primeiras linhas dos dados:"
1.  12.
2.  22.
3.  36.
4.  40.
5.  55.
"Coeficientes do ajuste:"
"a (x²) = -0.1560712"
"b (x) = 7.0043928"
"c (constante) = 15.453695"
"Coeficiente de correlação (R) = 0.9821766"
"Coeficiente de determinação (R²) = 0.964671"

--> |

```

Figura 1b - Saída do programa no termina; Coeficientes de Correlação e de Determinação

```

//Regressao.sce
//Faz a regressão dos pontos lidos num arquivo de texto
//-----

// Lê os dados do arquivo no caminho determinado
caminho = "D:/Livros/Livro5/Exemplos/Dados3.txt";
MD = read(caminho, -1, 2); // -1 para ler todas as linhas, 2 colunas

// Extrai colunas X e Y
xDados = MD(:,1);
yDados = MD(:,2);

// Mostra as primeiras linhas para verificação
disp("Primeiras linhas dos dados:");
disp(MD(1:5,:));

// Ajuste polinomial de grau 2 usando polyfit
[p, S] = polyfit(xDados, yDados, 2);

// Extrai os coeficientes
a = p(1); // coeficiente x²
b = p(2); // coeficiente x
c = p(3); // termo constante

disp("Coeficientes do ajuste:");
disp("a (x²) = " + string(a));
disp("b (x) = " + string(b));
disp("c (constante) = " + string(c));

// Cria pontos para a curva ajustada
xCurva = linspace(min(xDados), max(xDados), 200);

// Avalia o polinômio nos pontos xCurva
yCurva = a*xCurva.^2 + b*xCurva + c;

// Plot dos pontos experimentais (BOLINHAS VERMELHAS)
plot(xDados, yDados, "ro", "markersize", 4, "markerfacecolor", "red");

// Plotagem da curva ajustada (ASTERISCOS AZUIS)
plot(xCurva, yCurva, "b*", "markersize", 2);

// Configurações do gráfico
xtitle("Ajuste de curva do 2º grau", "X", "Y");
xlabel("X");
ylabel("Y");
legend(["Pontos experimentais"; "Curva ajustada"], "in_upper_left");

// Ajusta os limites dos eixos para melhor visualização
axeHandle = gca(); // renomeado para não sobrescrever 'a'
axeHandle.data_bounds = [min(xDados)-1, min(yDados)-10; max(xDados)+1,
    max(yDados)+10];
axeHandle.grid = [1 1]; // adiciona grid

// Atualiza o gráfico
replot();

```

```

// -----
// Cálculo do Coeficiente de Correlação (R) e Coeficiente de Determinação (R²)
// -----

// Avalia o polinômio nos pontos originais
yAjustado = a*xDados.^2 + b*xDados + c;

// Coeficiente de correlação (R) usando fórmula manual
xMean = mean(yDados);
yMean = mean(yAjustado);
R = sum((yDados - xMean).*(yAjustado - yMean)) / sqrt(sum((yDados - xMean).^2) *
    sum((yAjustado - yMean).^2));

// Coeficiente de determinação (R²)
R2 = R^2;

// Mostra no terminal
disp("Coeficiente de correlação (R) = " + string(R));
disp("Coeficiente de determinação (R²) = " + string(R2));
//Fim do programa "Regressao" -----

```