جامعة المستقبل
AL MUSTAQBAL UNIVERSITY
كلية العلوم

2^{nd} class

2024- 2025

Numerical Analysis

Practical

MATLAB

Lecture 3

Asst. Lect. Mohammed Jabbar

Mohammed.Jabbar.Obaid@uomus.edu.iq

الرياضيات المتقدمة: المرحلة الثانية

مادة التحليل العددي عملي

ماتلاب

المحاضرة الثالثة

استاذ المادة: م.م محمد جبار

Cybersecurity Department

قسم الأمن السيبراني

Contents

1 Interpolation and Curve Fitting in Matlab

Interpolation and curve fitting are both methods to estimate new data points based on a known set of data. They are widely used in data analysis, scientific research, and engineering. MATLAB provides various functions for performing interpolation and curve fitting.

1.1 Interpolation

Interpolation estimates intermediate values between known data points. The interpolated curve passes exactly through the known data points.

It is a numerical method to obtain additional data from existing data. It is either (Linear) or (Spline = Nonlinear) It is performed in Matlab vie the command (`interp1`) for a vector:

- `yn = interp1(x, y, xn,'linear')`: Linear interpolation (default).
- `yn = interp1(x, y, xn,'Spline')`: Cubic spline interpolation.

Other Interpolation Methods Available in `interp1`: where:

- x, y: Known data points.
- x_n: Finer grid for interpolation.
- `'linear'`, `'Spline'`: Interpolation Methods.
- y_n: Required data.

Other Interpolation Methods Available in

- `'nearest'`: Nearest neighbor interpolation.
- `'pchip'`: Piecewise cubic Hermite interpolation.

```
x=0:2:6
y = [1, 4, 9, 16]

xn=0:6
ynl=interp1(x,y,xn); % or interp1 (x, y, xn,'linear')
yns=interp1(x,y,xn, 'spline')
ynp=interp1(x,y,xn, 'pchip')
ynn=interp1(x,y,xn, 'nearest')

%%

% Plotting the original data and interpolated points
figure;
plot(x, y,'bo', 'MarkerFaceColor', 'b'); % 'Original Data'
hold on;
plot(xn, ynl, 'r-'); % 'Linear Interpolation'
plot(x, y,'s', 'MarkerFaceColor', 'g'); % 'spline'
plot(xn, ynp, 'g--'); %  Piecewise cubic Hermite interpolation.
plot(xn, ynn, 'k*'); %  Nearest neighbor interpolation.
hold off;
```

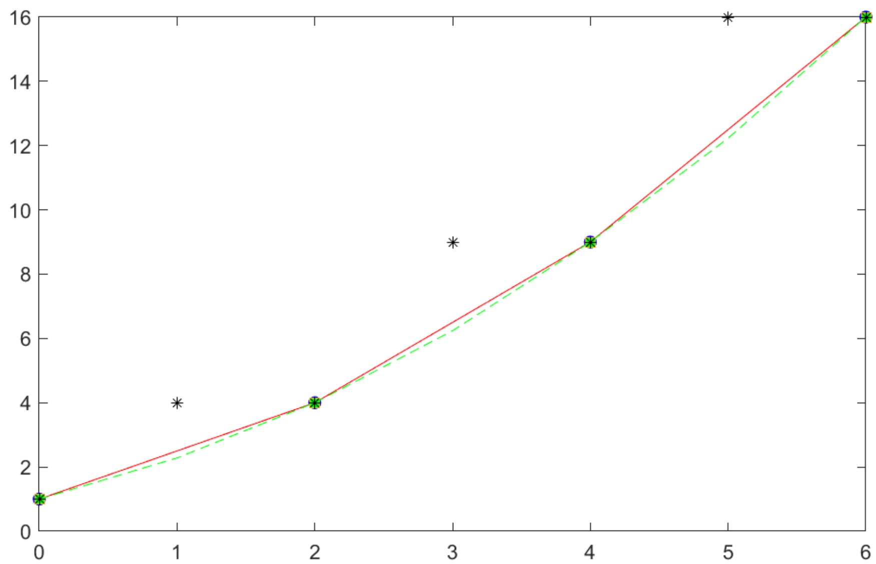

Figure 1: Interpolation of y

x	x_n	y	y 'linear'	y 'Spline'	y 'nearest'	y 'pchip'
0	0	1	1	1	1	1
	1		2.5	2.25	4	2.2812
2	2	4	4	4	4	4
	3		6.5	6.25	9	6.2396
4	4	9	9	9	9	9
	5		12.5	12.2292	16	12.2292
6	6	16	16	16	16	16

Table 1: Interpolation of y

Homework: Interpolation in MATLAB

You are given the following data points:

$x = [0, 1, 2, 3, 4, 5]\,, \quad y = [2, 8, 14, 28, 39, 62]$

1. Use linear interpolation to estimate the value of y at $x = 2.5$, $x = 3.5$, and $x = 4.5$.
2. Use spline interpolation to estimate the value of y at $x = 2.5$, $x = 3.5$, and $x = 4.5$.
3. Plot the original data points and the interpolated values on the same graph. Use a different color for each interpolation method and mark the interpolated points.

1.2 Curve Fitting

Curve fitting involves finding a curve that best fits a series of data points.

Data Fitting: It is a numerical method by which a polynomial equations are sought to fit a given data points. The polynomial equations may be linear (order one) or non-linear (order two onward). The order of the polynomial must be less than the number of points by one. The command (polyfit) is used to get the polynomial coefficients. If the order of the polynomial is (n) then the number of coefficients is ($n + 1$).

```
z = polyfit (x, y, n)
```

If $n = 3$ then: $z = z_1 + z_2 + z_3 + z_4$

In general: $z = z_1 + z_2 + z_3 \cdots z_{n+1}$

The command (polyval) can be used to get polynomial y_n

$$y_n = z_1x^n + z_2x^{n-1} + z_3x^{n-2} \cdots + z_{n-1}x^2 + z_nx + z_{n+1}$$

yn = polyval (z,x),

```
%% Known data points
x = [1, 2, 3, 4, 5];
y = [5, 9, 7, 8, 10];

% Fit a polynomial of degree 7
z = polyfit(x, y, 7);

% Evaluate the polynomial at the original points
yn= polyval(z, x);

% Plotting the original data and polynomial fit
figure;
plot(x, y, 'bo', 'MarkerFaceColor', 'b', 'DisplayName', ...
    'Original Data');
hold on;
plot(x, yn, 'r-', 'DisplayName', 'Fit');
xlabel('x');
ylabel('y');
legend;
title('Polynomial Fitting (Degree 7)');
hold off;
```

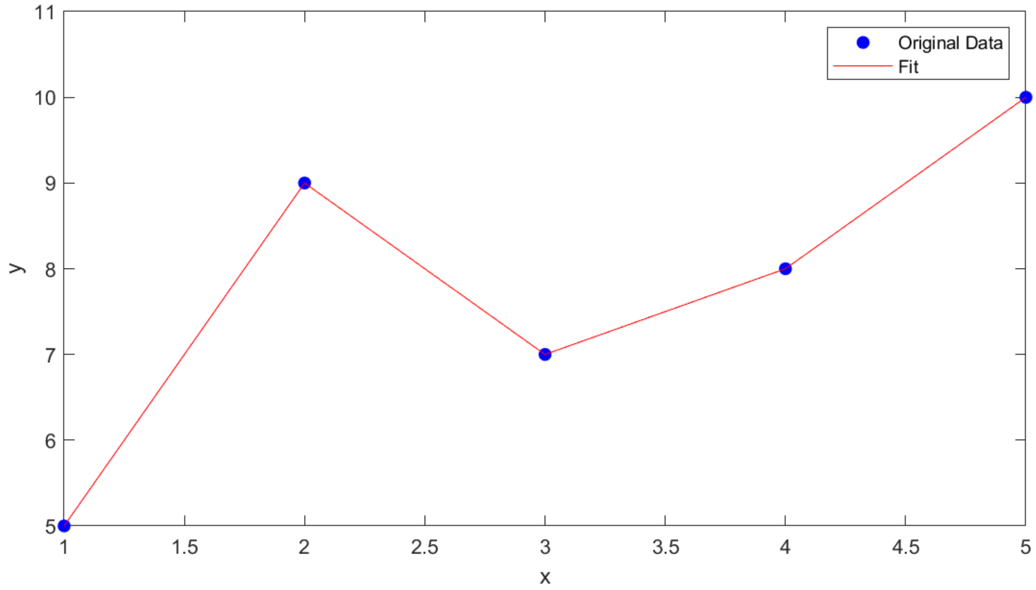

Figure 2: Polynomial Fitting (Degree 7)

Homework: Curve Fitting in MATLAB

You are given the following data points representing the height of an object as a function of time:

$Time\,(seconds): \; T = [0, 1, 2, 3, 4, 5]\,, \quad Height\,(meters): \; H = [0, 2, 5, 10, 17, 26, 37]$

1. Use polyfit and polyval to fit a quadratic polynomial to the data. Display the polynomial coefficients.
2. Estimate the height of the object at $t = 2.5$ and $t = 4.5$ seconds using the fitted quadratic model.
3. Plot the original data points and the interpolated values on the same graph. Use a different color for each method.