Chapter Three

INTERPOLATION

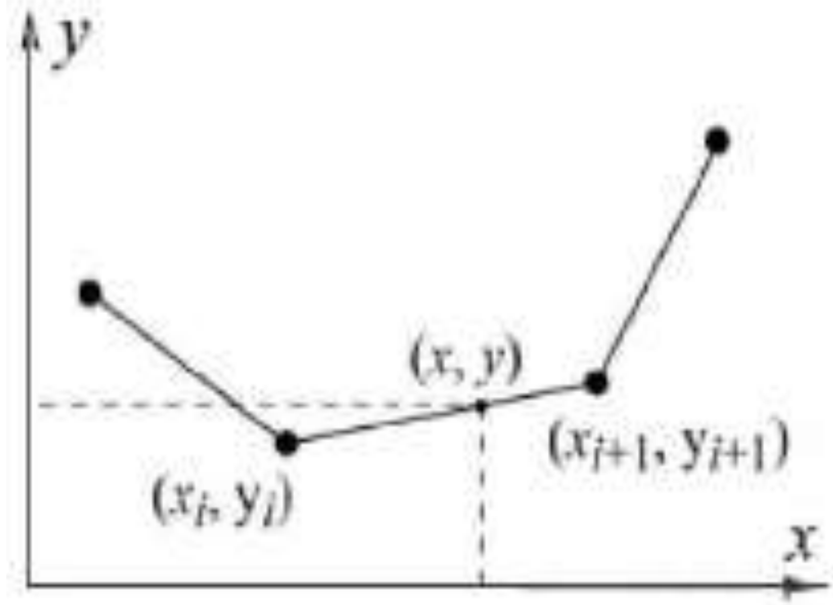
Introduction

- Interpolation is the estimation of values between data points.
- MATLAB has interpolation functions that are based on polynomials.
- In one-dimensional interpolation, each point has one independent variable (x) and one dependent variable (y).
- In two-dimensional interpolation, each point has two independent variables (x and y) and one dependent variable (z).

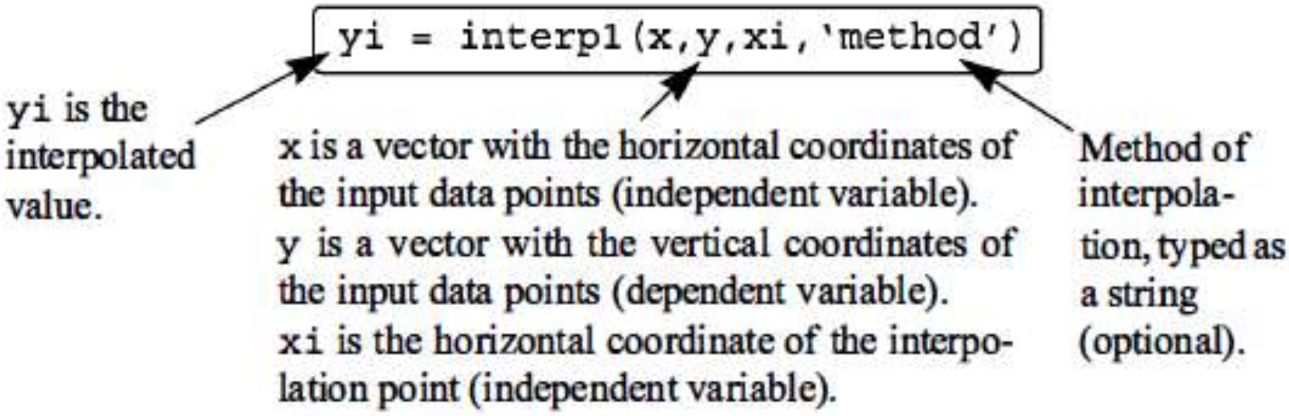
One-dimensional interpolation:

- If only two data points exist, the points can be connected with a straight line and a linear equation can be used to estimate values between the points.
- If three (or four) data points exist, a second- (or a third-) order polynomial that passes through the points can be determined and then be used to estimate values between the points.
- As the number of points increases, a higher-order polynomial is required for the polynomial to pass through all the points.
- A more accurate interpolation can be obtained, only a few data points in the neighborhood where the interpolation is needed are considered, this method, called spline interpolation.
- The simplest method of spline interpolation is called linear spline interpolation.
- In this method, every two adjacent points are connected with a straight line.
- The equation of a straight line that passes through two adjacent points (x_i, y_i) and (x_{i+1}, y_{j+1}) is given by:

$$y = \frac{y_{i+1}-y_i}{x_{i+1}-x_i}x + \frac{y_i x_{i+1}-y_{i+1}x_i}{x_{i+1}-x_i}$$



- In a linear interpolation, the line between two data points has a constant slope, and there is a change in the slope at every point. A smoother interpolation curve can be obtained by using quadratic or cubic polynomials.
- In these methods, called quadratic splines and cubic splines, a second, or third-order polynomial is used to interpolate between every two points.
- One-dimensional interpolation in MATLAB is done with the interp1 function, which has the form:



- The vector x must be with elements in ascending or descending order.
- xi can be a scalar (interpolation of one point) or a vector (interpolation of many points). yi is a scalar or a vector with the corresponding interpolated values.
- MATLAB can do the interpolation using one of several methods that can be specified. These methods include:
  - **nearest** – nearest neighbor interpolation
  - **linear** – linear interpolation; this is the default interpolation
  - **spline** – cubic spline interpolation; this does also extrapolation
  - **cubic** – cubic interpolation; this requires equidistant values of x
- When the 'nearest' and the 'linear' methods are used, the value(s) of xi must be within the domain of x. If the 'spline' or the 'cubic' pchip methods are used, xi can have values outside the domain of x and the function interp1 performs extrapolation.
- The 'spline' method can give large errors if the input data points are non-uniform such that some points are much closer together than others.
- Specification of the method is optional. If no method is specified, the default is 'linear‘.

Ex/ The following data points, which are points of the function $f(x) = 1.5^x \cos(2x)$, are given. Use linear, spline, and pchip interpolation methods to calculate the value of y between the points. Make a figure for each of the interpolation methods.

SOLUTION

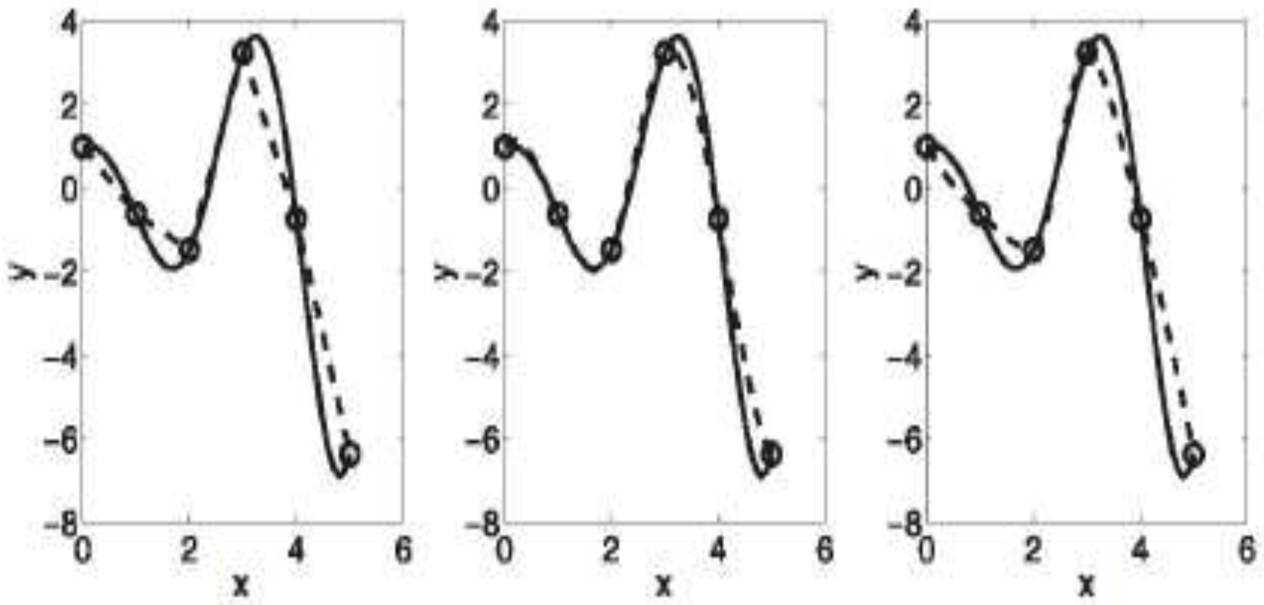
x	0	1	2	3	4	5
y	1.0	−0.6242	−1.4707	3.2406	−0.7366	−6.3717

Solution

The following is a program written in a script file that solves the problem:

```
x=0:1.0:5;                    Create vectors x and y with coordinates of the data points.
y=[1.0  -0.6242  -1.4707  3.2406  -0.7366  -6.3717];
xi=0:0.1:5;                   Create vector xi with points for interpolation.
yilin=interp1(x,y,xi,'linear');   Calculate y points from linear interpolation.
yispl=interp1(x,y,xi,'spline');   Calculate y points from spline interpolation.
yipch=interp1(x,y,xi,'pchip');    Calculate y points from pchip interpolation.
yfun=1.5.^xi.*cos(2*xi);          Calculate y points from the function.
subplot(1,3,1)
plot(x,y,'o',xi,yfun,xi,yilin,'--');
subplot(1,3,2)
plot(x,y,'o',xi,yfun,xi,yispl,'--');
subplot(1,3,3)
plot(x,y,'o',xi,yfun,xi,yipch,'--');
```

The three figures generated by the program are shown below. The data points are marked with circles, the interpolation curves are plotted with dashed lines, and the function is shown with a solid line. The left figure shows the linear interpolation, the middle is the spline, and the figure on the right shows the pchip interpolation.



Other built in Interpolation Functions

1. interp2(x,y,z,xi,yi) is similar to **interp1(x,y,xi)** but performs two dimensional interpolation.

2. interp2(x,y,z,xi,yi,'method') is similar to **interp1(x,y,xi,'method')** but performs two dimensional interpolation. The default is **linear**. The **spline** method does not apply to two dimensional interpolation.

Example

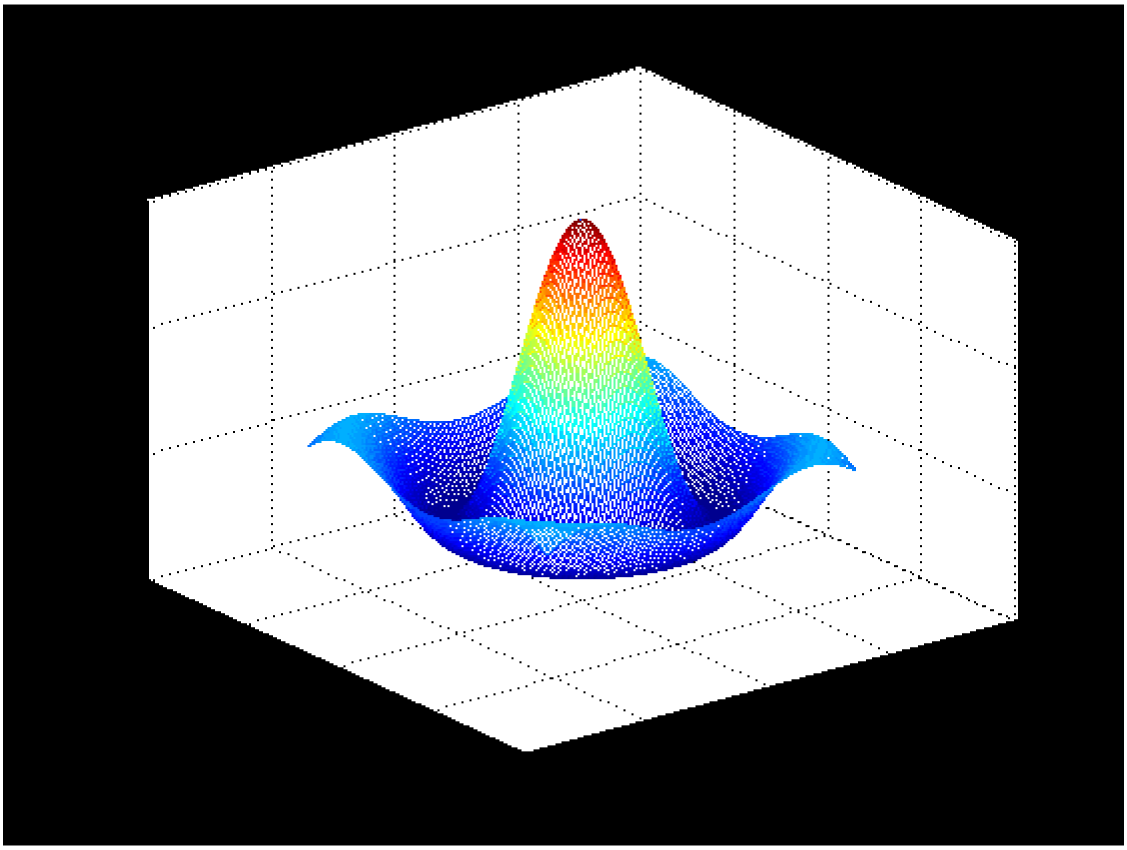
Generate the plot of the function

$$\boldsymbol{Z} = \frac{\sin \boldsymbol{R}}{\boldsymbol{R}}$$

in three dimensions. Here, R is a matrix that contains the distances from the origin to each point in the pair of [X,Y] matrices that form a rectangular grid of points in the x-y plane.
Solution:
The matrix R that contains the distances from the origin to each point in the pair of [X, Y] matrices, is

$$\boldsymbol{R} = \sqrt{\boldsymbol{X}^2 + \boldsymbol{Y}^2}$$

```
x=-2*pi: pi/24: 2*pi;          % Define interval in increments of pi/24
y=x;                           % y must have same number of points as x
[X,Y]=meshgrid(x,y);           % Create X and Y matrices
R=sqrt(X.^ 2 + Y.^ 2);         % Compute distances from origin (0,0) to x&y points
Z=sin(R)./ (R+eps);            % eps prevents division by zero
mesh(X,Y,Z);                   % Generate mesh plot for Z=sin(R)/R
xlabel('x'); ylabel('y'); zlabel('z');
title('Plot for the Three dimensional sin(R) / R Function')
```



Example

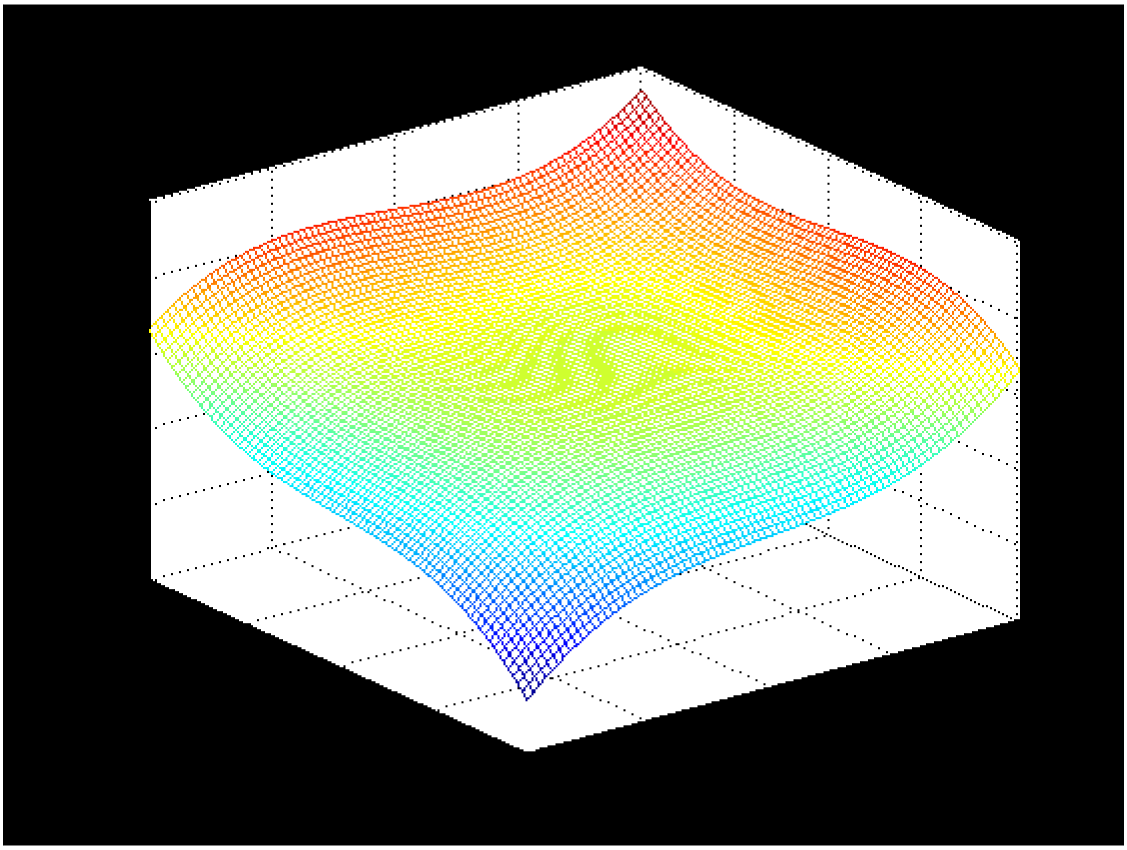
Generate the plot of the function

$$z=x^3+y^3-3xy$$

in three dimensions , and use the cubic method to interpolate the value of z at x=-1 and y=2.

Solution:

```
x=-10: 0.25: 10;                    % Define interval in increments of 0.25
y=x;                               % y must have same number of points as x
[X,Y]=meshgrid(x,y);               % Create X and Y matrices
 Z=X.^3+Y.^3-3.*X.*Y;
mesh(X,Y,Z);                        % Generate mesh plot
xlabel('x'); ylabel('y'); zlabel('z');
title('Plot for the Function of Example 7.14');
z_int=interp2(X,Y,Z, -1,2,'cubic');
fprintf(' \n')
fprintf('Interpolated Value of z at x = -1 and y = 2 is z = %4.2f \n',z_int)
fprintf(' \n')
```



Problems

1.

When rubber is stretched, its elongation is initially proportional to the applied force, but as it reaches about twice its original length, the force required to stretch the rubber increases rapidly. The force, as a function of elongation, that was required to stretch a rubber specimen that was initially 3 in. long is displayed in the following table.
(*a*) Curve-fit the data with a forth-order polynomial. Make a plot of the data points and the polynomial. Use the polynomial to estimate the force when the rubber specimen was 11.5 in. long.
(*b*) Fit the data with spline interpolation (use MATLAB's built-in function `interp1`). Make a plot that shows the data points and a curve made by interpolation. Use interpolation to estimate the force when the rubber specimen was 11.5 in. long.

Force (lb)	0	0.6	0.9	1.16	1.18	1.19	1.24	1.48
Elongation (in.)	0	1.2	2.4	3.6	4.8	6.0	7.2	8.4
Force (lb)	1.92	3.12	4.14	5.34	6.22	7.12	7.86	8.42
Elongation (in.)	9.6	10.8	12.0	13.2	14.4	15.6	16.8	18

2.

The ideal gas equation relates the volume, pressure, temperature, and the quantity of a gas by:

$$V = \frac{nRT}{P}$$

where V is the volume in liters, P is the pressure in atm, T is the temperature in kelvins, n is the number of moles, and R is the gas constant.

An experiment is conducted for determining the value of the gas constant R. In the experiment, 0.05 mol of gas is compressed to different volumes by applying pressure to the gas. At each volume, the pressure and temperature of the gas are recorded. Using the data given below, determine R by plotting V versus T/P and fitting the data points with a linear equation.

V (L)	0.75	0.65	0.55	0.45	0.35
T (°C)	25	37	45	56	65
P (atm)	1.63	1.96	2.37	3.00	3.96