Section 2 Interpolating Polynomials in Standard Form

Before we can begin to approximate a function $f(x)$, some information about the function has to be known. In the most common situation, a set of N+1 data points (x_i, y_i), $i = 0, 1, 2, \ldots, N$ is obtained in some fashion, where $y_i = f(x_i)$. An nth order ($n \le N$) polynomial can be selected to approximate $f(x)$. The standard form of the interpolating polynomial is

$$f_n(x) = a_0 + a_1 x + a_2 x^2 + \ldots.. + a_n x^n \quad (2.1)$$

This polynomial is not guaranteed to pass through the entire set of N+1 data points unless the order n is the same as N and the x_i values are discrete. When the polynomial order n is less than N, it will pass through the n + 1 data points used to determine the n + 1 coefficients a_i, $i = 0, 1, 2, \ldots, n$. The n + 1 equations are generated from

$$f_n(x_i) = f(x_i), \quad i = 0, 1, 2, \ldots, n \quad (2.2)$$

Figure 2.1 illustrates the case where two different second order polynomials $f_2(x)$ are used to approximate a function $f(x)$ from which five points are known.

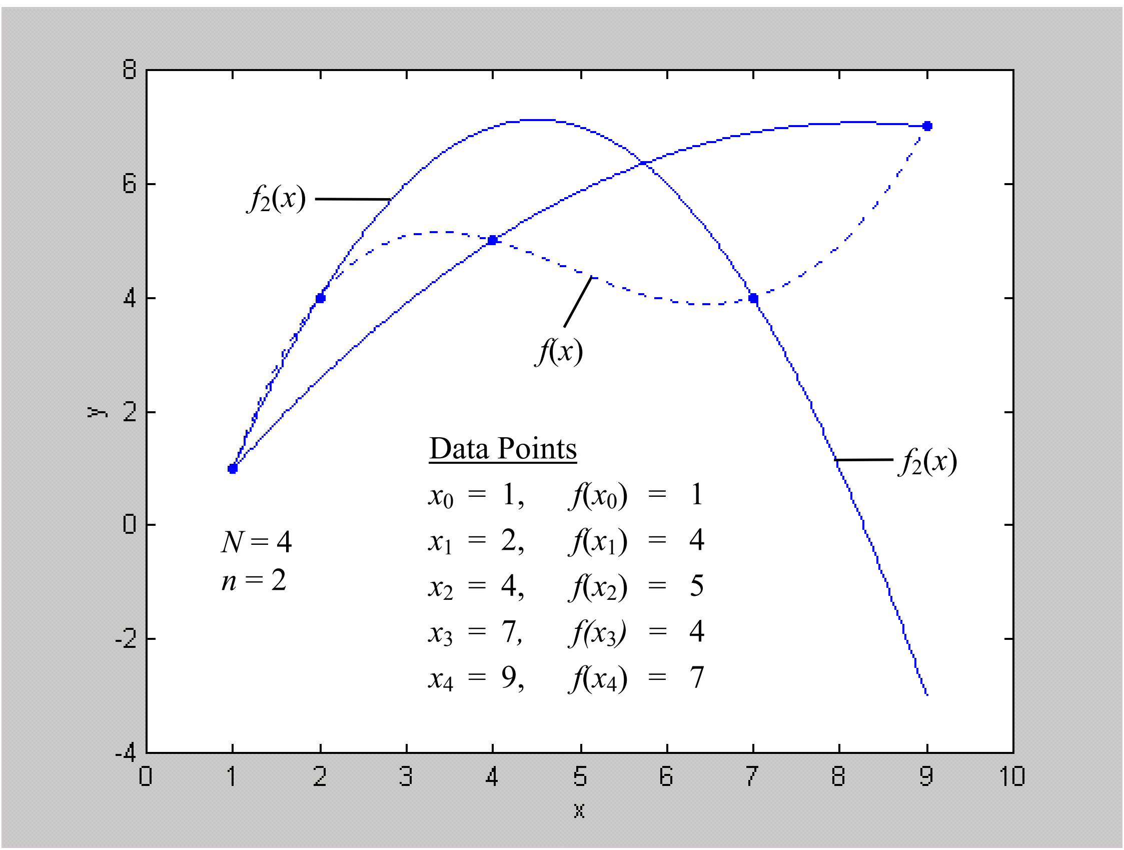


Figure 2.1 Two Second Order Polynomial Approximations to a Function with Five Known Data Points (N = 4, n = 2)

In general, with $n + 1$ data points specified, the leading coefficient $a_n \neq 0$, and the interpolating polynomial is nth order. Under certain conditions, the highest order term(s) may vanish and the resulting polynomial is less than nth order. For example, if three of the data points in Figure 2.1 happen to be collinear, the coefficient a_2 of the quadratic passing through all three points would be zero and the interpolating polynomial reduces to a linear function.

Equation (2.2) results in a system of $n + 1$ simultaneous equations which can be solved for the coefficients a_i, $i = 0, 1, 2, \ldots, n$. The matrix form of the equations are as follows:

$$\begin{pmatrix} 1 & x_0 & x_0^2 & . & . & . & x_0^{n-1} & x_0^n \\ 1 & x_1 & x_1^2 & . & . & . & x_1^{n-1} & x_1^n \\ 1 & x_2 & x_2^2 & . & . & . & x_2^{n-1} & x_2^n \\ . & . & . & & & & . & . \\ . & . & . & & & & . & . \\ . & . & . & & & & . & . \\ 1 & x_{n-1} & x_{n-1}^2 & . & . & . & x_{n-1}^{n-1} & x_{n-1}^n \\ 1 & x_n & x_n^2 & . & . & . & x_n^{n-1} & x_n^n \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ . \\ . \\ . \\ a_{n-1} \\ a_n \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f(x_1) \\ f(x_2) \\ . \\ . \\ . \\ f(x_{n-1}) \\ f(x_n) \end{pmatrix} \tag{2.3}$$

The coefficient matrix above is called the Vandermonde matrix and its nonsingular as long as all the x_i values are different. Since a unique solution exists when $x_i \neq x_j$, $i \neq j$, the resulting polynomial in Equation (2.1) is unique, although as we will see later, it can be represented in different forms.

The case of linear interpolation is considered first. The objective is to find a linear approximation to a function when two or more data points of the function are known. The linear approximation becomes the basis for estimation of function values.

The linear approximation $f_1(x)$, from Equation (2.1) with $n = 1$ is,

$$f_1(x) = a_0 + a_1 x \tag{2.4}$$

where the coefficients a_0 and a_1 come from solution of Equation (2.3) with $n = 1$. That is,

$$\begin{pmatrix} 1 & x_0 \\ 1 & x_1 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f(x_1) \end{pmatrix} \tag{2.5}$$

Equation (2.5) is easily solved for a_0 and a_1. Doing so, and substituting the results into Equation (2.4) yields the familiar equation of a line

$$f_1(x) = f(x_0) + \left[\frac{f(x_1) - f(x_0)}{x_1 - x_0}\right](x - x_0) \tag{2.6}$$

Figure 2.2 shows two different linear approximations to an unknown function $f(x)$. In the top graph, $(x_0, f(x_0)) = (1,4)$ and $(x_1, f(x_1)) = (4,11)$ were used to find $f_1(x)$. In the lower graph, the data points from $f(x)$ are renumbered so that $(x_0, f(x_0)) = (2,9)$ and $(x_1, f(x_1)) = (7,24)$.

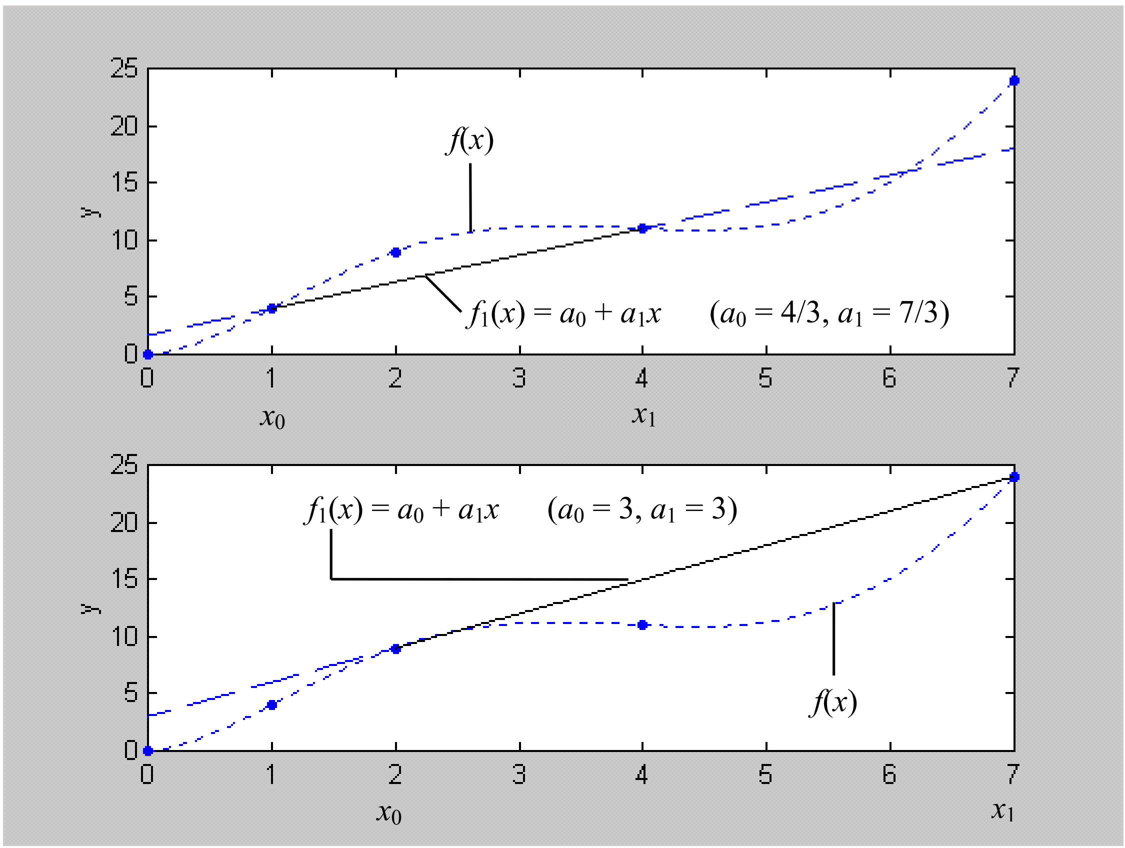


Figure 2.2 Two Different Linear Approximations to an Unknown Function $f(x)$

Keep in mind, interpolation is valid only for x values within the range of the two points used to determine the linear function. Therefore, $f_1(x)$ in the top graph should be used for interpolation only when $1 \le x \le 4$ and $f_1(x)$ in the bottom graph for $2 \le x \le 7$.

Understanding the limitations imposed on the approximating functions by the use of interpolation is important. We are certainly free to use the approximating functions $f_1(x)$ outside the interval $x_0 \le x \le x_1$, however by doing so we may not be using the "best" linear approximation of the function $f(x)$. After all, both linear approximations are based solely on two specific points from $f(x)$ and disregard the remaining (known) data points. In this next chapter we consider methods for obtaining low order polynomial approximating functions that utilize all the known data points. For now we must restrict the domain of approximating functions used for interpolation.

An example of linear interpolation is presented below.

Example 2.1

The table of values from the function $f(x) = \sin x$, Table 1.2, is repeated below. Estimate the value of sin (1.15 rad) using the data point at $x_0 = 1.00$, $y_0 = 0.8415$ and choose the second data point to be each of the four remaining points.

i	x_i	$y_i = f(x_i) = \sin x_i$
0	1.00	0.8415
1	1.25	0.9490
1	1.50	0.9975
1	1.75	0.9840
1	2.00	0.9093

Table 2.1 Several Points From the Function $f(x) = \sin x$

a) $(x_0, y_0) = (1.00, 0.8415)$, $(x_1, y_1) = (1.25, 0.9490)$

$$f_1(x) = f(x_0) + \left[\frac{f(x_1) - f(x_0)}{x_1 - x_0}\right](x - x_0)$$

$$f_1(1.15) = f(1.00) + \left[\frac{f(1.25) - \mathrm{f}(1.00)}{1.25 - 1.00}\right](1.15 - 1.00)$$

$$= 0.8415 + \left[\frac{0.9490 - 0.8415}{1.25 - 1.00}\right](1.15 - 1.00)$$

$$= 0.9060$$

b) $(x_0, y_0) = (1.00, 0.8415)$, $(x_1, y_1) = (1.50, 0.9975)$

$$f_1(1.15) = f(1.00) + \left[\frac{f(1.50) - f(1.00)}{1.50 - 1.00}\right](1.15 - 1.00)$$

$$= 0.8415 + \left[\frac{0.9975 - 0.8415}{1.50 - 1.00}\right](1.15 - 1.00)$$

$$= 0.8883$$

Performing similar calculations for the remaining two end points produces the results,

c) $(x_0, y_0) = (1.00, 0.8415)$, $(x_1, y_1) = (1.75, 0.9840)$

$f_1(1.15) = 0.8700$

d) $(x_0, y_0) = (1.00, 0.8415)$, $(x_1, y_1) = (2.00, 0.9093)$

$f_1(1.15) = 0.8517$

A graphical illustration of the previous calculations is shown in Figure 2.3. It is clear that the accuracy of the interpolated value diminishes as the right end point moves further from the *x* value where the interpolation is performed. Table 2.2 summarizes the results.

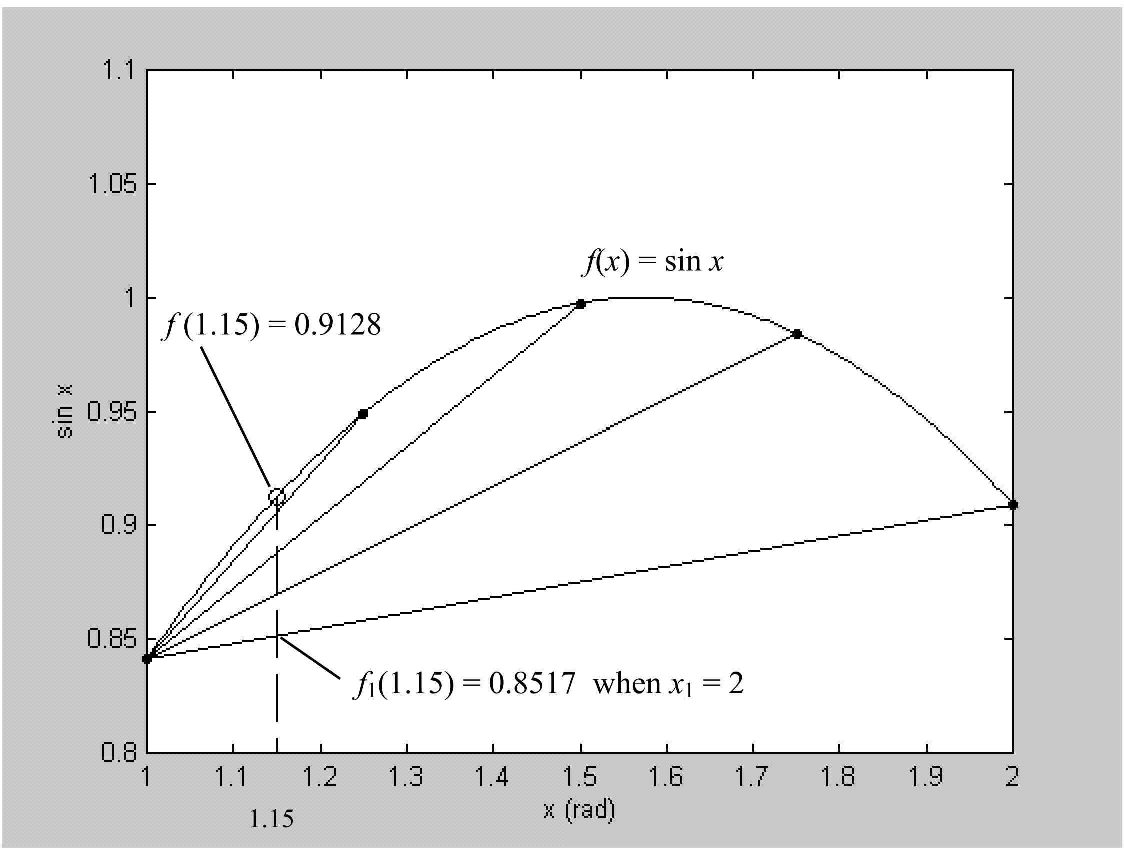


Figure 2.3 Several Linear Interpolating Polynomials for Estimating sin(1.15 rad)

x_0	x_1	$f_1(1.15)$	E_T	e_T, %
1.00	1.25	0.9060	0.0068	0.74
1.00	1.50	0.8883	0.0245	2.68
1.00	1.75	0.8700	0.0428	4.69
1.00	2.00	0.8517	0.0611	6.69

Table 2.2 Accuracy of Linear Interpolation in Example 2.1

Equation (2.6) for linear interpolation is easily implemented with the MATLAB function "interp1". For example, MATLAB statements to generate $f_1(1.15)$ in the first two rows of Table 2.2 are

```
» x0=1;
» f0=sin(x0);
» x1=1.25;
» f1=sin(x1);
» x=[x0 x1];
» f=[f0 f1];
» y=interp1(x,f,1.15)
y =   0.9060
» x1=1.5;
» f1=sin(x1);
» x=[x0 x1];
» f=[f0 f1];
» y=interp1(x,f,1.15)
y =   0.8883
```

The true error, $E_T = f(a) - f_1(a)$ at $x = a$, with linear interpolation will always be zero when the underlying function $f(x)$ is itself a linear function. Of course, it would be pointless to implement linear interpolation to estimate values from a known linear function. In general, linear interpolation produces results which deviate from the true function value, i.e. the true error is ordinarily nonzero. Nonetheless, its possible that the true error could actually be zero or extremely small depending on the actual function $f(x)$ and the location of the point a. Referring to Figure 2.2, if the two end points (0,0) and (7,24) were used to fix $f_1(x)$, there would be two intermediate points where the true error would be zero. Both points would lie at the intersection of the linear interpolating polynomial $f_1(x)$ and the actual function $f(x)$.

Higher order interpolating polynomials are required when the results of linear interpolation may be suspect, either because the spacing between the two end points is too great or the suspected behavior of the function is highly nonlinear in the region of interest. In reality, a combination of both conditions warrants the use of higher order interpolating polynomials.

When 3 data points from a function $f(x)$ are available, the additional data point provides useful information about the curvature of the function in the neighborhood of the points. For example, in Figure 2.4 with only points $P_0(x_0, y_0)$ and $P_1(x_1, y_1)$ obtained from a function $y = f(x)$, it is impossible to account for any nonlinearity inherent in the function. The interpolating polynomial $f_2(x)$ is a quadratic function which passes through all three points thereby capturing some of the nonlinear behavior of the function. Keep in mind that the function $f(x)$ is generally unknown. It is drawn as a dotted curve in Figure 2.4 to emphasize this point.

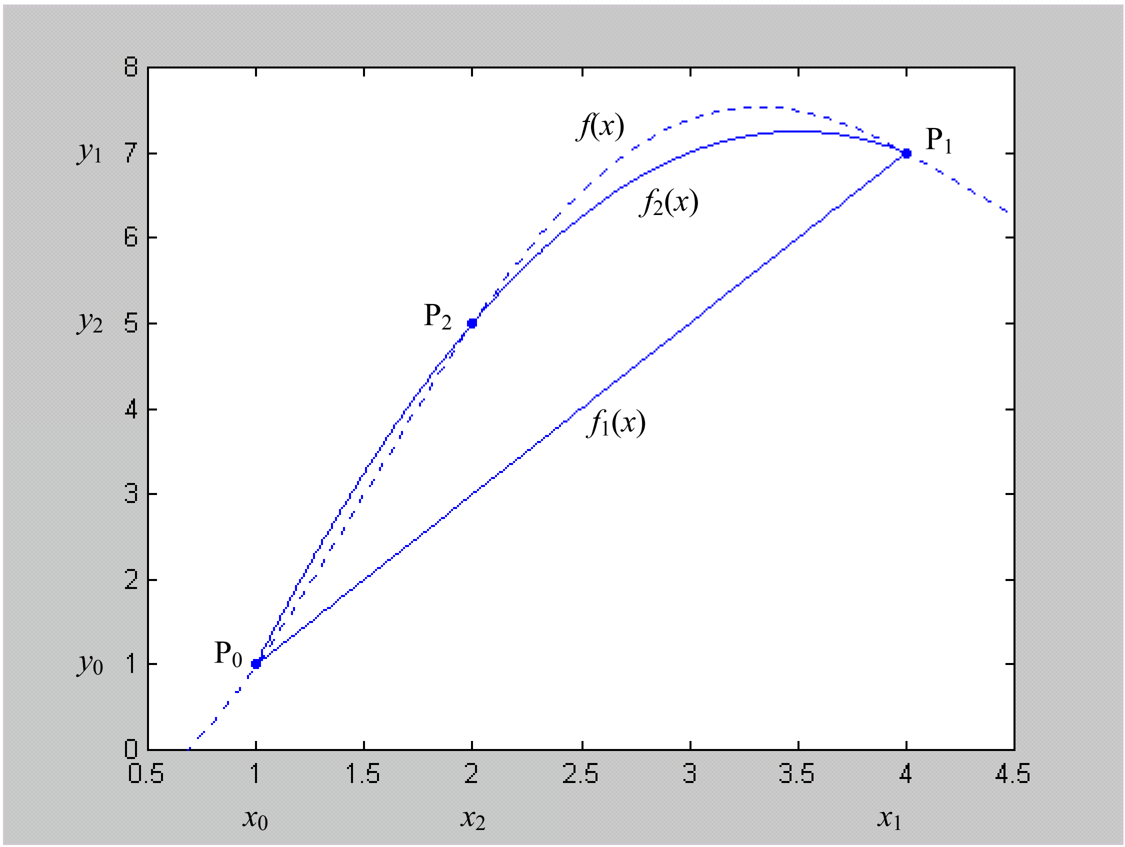


Figure 2.4 First and Second Order Interpolating Polynomials with 3 Data Points

The second order interpolating polynomial $f_2(x)$ is uniquely determined by the three data points $(x_i, f(x_i))$, $i = 0, 1, 2$ which lie on both the function $y = f(x)$ and $f_2(x)$. In standard form $f_2(x)$ is

$$f_2(x) = a_0 + a_1 x + a_2 x^2 \tag{2.7}$$

where the coefficients a_0, a_1, and a_2 are determined from the three data points (x_0, y_0), (x_1, y_1) and (x_2, y_2) which the polynomial must pass through. From Equation (2.7),

$$f_2(x_0) = a_0 + a_1 x_0 + a_2 {x_0}^2 \tag{2.8}$$

$$f_2(x_1) = a_0 + a_1 x_1 + a_2 {x_1}^2 \tag{2.8a}$$

$$f_2(x_2) = a_0 + a_1 x_2 + a_2 x_2^{\ 2} \tag{2.8b}$$

From Equation (2.2) with $n = 2$, the interpolating polynomial $f_2(x)$ and the function $f(x)$ are identical when x is either x_0, x_1 or x_2 (refer to Figure 2.4). Replacing $f_2(x_0)$ with $f(x_0)$, $f_2(x_1)$ with $f(x_1)$, and $f_2(x_2)$ with $f(x_2)$ in Equation (2.8) yields

$$f(x_0) = a_0 + a_1 x_0 + a_2 x_0^{\ 2} \tag{2.9}$$

$$f(x_1) = a_0 + a_1 x_1 + a_2 x_1^{\ 2} \tag{2.9a}$$

$$f(x_2) = a_0 + a_1 x_2 + a_2 x_2^{\ 2} \tag{2.9b}$$

Note, we could have used y_0 in place of $f(x_0)$ and the same for y_1, y_2 instead of $f(x_1)$ and $f(x_2)$ in Equation (2.8). Notation aside, what's important is that the constraints represented by Equations (2.9) allow us to solve for the unknown coefficients a_0, a_1 and a_2. Equations (2.9) in matrix form are

$$\begin{pmatrix} 1 & x_0 & x_0^{\ 2} \\ 1 & x_1 & x_1^{\ 2} \\ 1 & x_2 & x_2^{\ 2} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} \mathrm{f}(x_0) \\ \mathrm{f}(x_1) \\ \mathrm{f}(x_2) \end{pmatrix} \tag{2.10}$$

where the 3 by 3 coefficient matrix is the Vandermonde matrix previously introduced in Equation (2.3).

Example 2.2

Find the second order interpolating polynomial that fits through the last 3 points in Table 1.1 of vehicle crash data. Estimate the damages for a crash at 35 mph and compare the result to the same estimate based on linear interpolation using the data points at 30 and 40 mph.

The last 3 data points are (20, 19750), (30, 43500) and (40, 55000). The second order interpolating polynomial is written

$$f_2(s) = a_0 + a_1 s + a_2 s^2 \tag{2.11}$$

Substituting the data points into Equation (2.11),

$$\begin{pmatrix} 1 & 20 & 20^2 \\ 1 & 30 & 30^2 \\ 1 & 40 & 40^2 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} 19750 \\ 43500 \\ 55000 \end{pmatrix} \tag{2.12}$$

Using MATLAB to obtain the solution,

```
»x0=20;
»x1=30;
»x2=40;
»b=[19750 43500 55000];
»A=[1 x0 x0*x0; 1 x1 x1*x1; 1 x2 x2*x2];
»a=inv(A)*b';
»a'

a' =   1.0e+004 *

  -6.450000  0.543750  -0.006125
```

and the interpolating second order polynomial is

$$f_2(s) = -64500 + 5437.5\,s - 61.25\,s^2 \tag{2.13}$$

which can be verified by substituting in the 3 data points. Note, there is a Vandermonde matrix function in MATLAB 'vander (**x**)' where **x** is the vector of x_i values. The Vandermonde matrix returned however is slightly different than the standard form defined in Equation (2.3). The difference is that the MATLAB Vandermonde matrix columns are in reverse order compared to its definition in Equation (2.3). The MATLAB Vandermonde matrix can still be used as the coefficient matrix if we simply rewrite Equation (2.12) as follows,

$$\begin{pmatrix} 20^2 & 20 & 1 \\ 30^2 & 30 & 1 \\ 40^2 & 40 & 1 \end{pmatrix} \begin{pmatrix} a_2 \\ a_1 \\ a_0 \end{pmatrix} = \begin{pmatrix} 19750 \\ 43500 \\ 55000 \end{pmatrix} \tag{2.14}$$

Hence, when the MATLAB Vandermonde matrix is used, the solution vector **a** is $[a_2\ a_1\ a_0]$ and not the reverse as in the example above. Use of the MATLAB Vandermonde matrix function is illustrated below.

```
»x0=20;
»x1=30;
»x2=40;
»x=[x0 x1 x2];
»b=[19750 43500 55000];
»A=vander(x)
»a=inv(A)*b';
»a'

A =    400          20           1
       900          30           1
      1600          40           1

a' =  1.0e+004 *

  -0.006125  -6.450000   0.543750
```

The first order interpolating polynomial $f_1(s)$ based on the data points (30, 43500) and (40, 55000) and the second order interpolating polynomial f_2(s) using the data points (20, 19750), (30, 43500) and (40, 55000) are shown in Figure 2.5. Using second order polynomial interpolation, the estimated damages for a 35 mph collision is obtained from $f_2(35)$ in Equation (2.13). The result is $50781. The estimated value using linear interpolation is $49250.

MATLAB functions 'polyfit' and 'polyval' will produce the same results. Coefficients for the linear and quadratic interpolating polynomials are computed as well as the interpolated value of damages resulting from a collision at 35 mph.

```
»s=[30 40];
»d=[43500 55000];
»a=polyfit(s,d,1);
»a'
»f1_35=polyval(a,35)

a' = 1150   9000

f1_35 = 49250

»s=[20 30 40];
»d=[19750 43500 55000];
»a=polyfit(s,d,2)
»a'
»f2_35=polyval(a,35)

a' = -61.25   5437.5    -6450

f2_35 = 50781.25
```

The use of interpolating polynomials is straightforward; however caution is necessary when dealing with higher order polynomials. This is because high order polynomials can fluctuate dramatically between the sampled data points. Consequently, estimates of the function, which itself may have exhibit a smooth behavior between sampled points, can be notoriously inaccurate when high order polynomials are used. Fortunately, it is readily apparent when this occurs.

Example 2.3 illustrates the tendency of polynomials to vary significantly between data points.

Example 2.3

The liquid flowrate into a tank is adjustable and measurable. Each time the inflow is changed, the output flowrate eventually reaches a new steady-state or equilibrium value as does the height of liquid in the tank. The relationship between output flowrate and height of liquid in the tank at steady-state is of interest. Table 2.3 contains measurements of the tank output flowrate (same as the input flowrate) and height of liquid for various steady-state operating conditions.

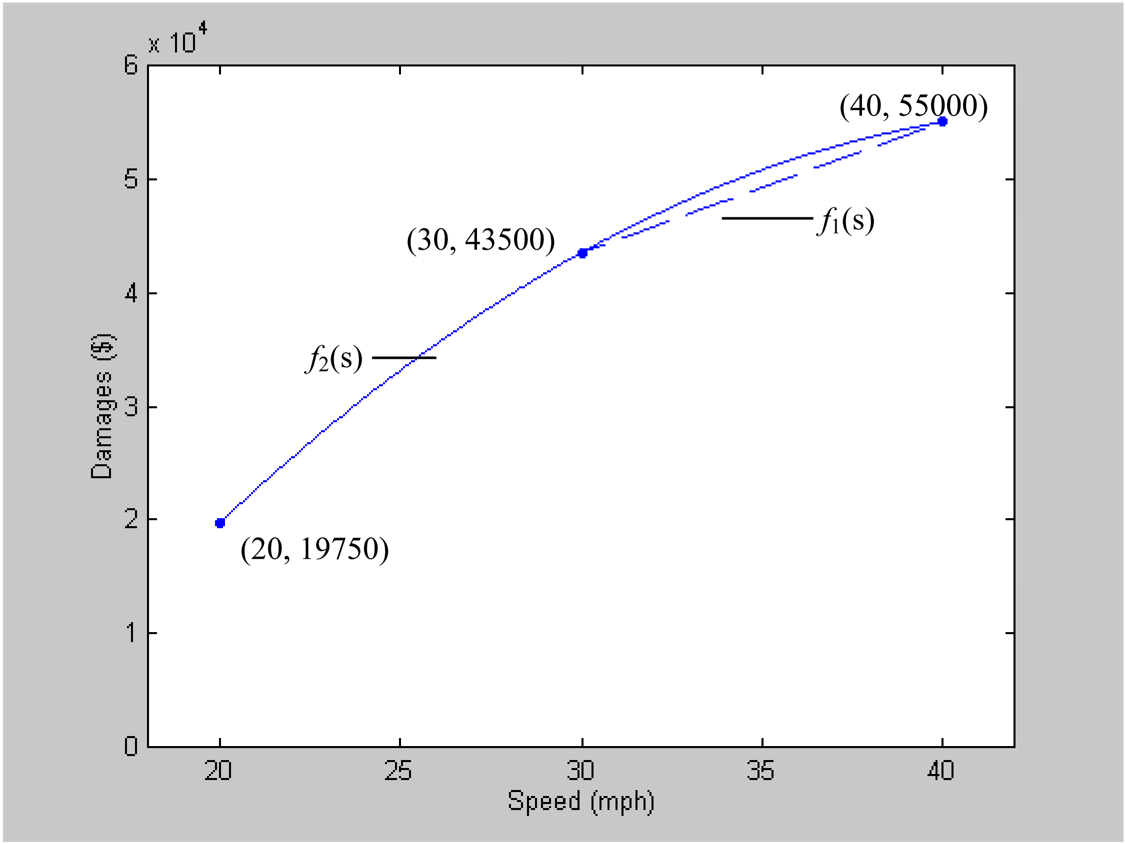


Figure 2.5 Linear and Quadratic Interpolation of Speed Crash Data in Example 3.2

A deterministic relationship exists between the two variables H and F. There is a function $F = f(H)$ that could be evaluated at a specific value of H to determine the corresponding steady-state output flow F. For now, let's assume the actual function $f(H)$ is unknown. The tabulated data is a sampling of points from this function.

Height of Liquid in Tank	Output Flowrate
H (ft)	F (gpm)
0	0
1	50
4	100
9	150
16	200
25	250
36	300

Table 2.3 A Sample of Steady-State Operating Conditions for a Liquid Tank

The 7 data points (H_i, F_i), $i = 0, 1, 2, \ldots, 6$ will uniquely determine a 6th order interpolating polynomial, $f_6(H)$. The MATLAB function 'polyfit' is the quickest way to obtain the vector of coefficients $\mathbf{a} = (a_6, a_5, a_4, a_3, a_2, a_1, a_0)$ that defines the interpolating polynomial $f_6(H)$ below.

$$f_6(H) = a_0 + a_1 H + a_2 H^2 + a_3 H^3 + a_4 H^4 + a_5 H^5 + a_6 H^6 \qquad (2.15)$$

```
H = [0 1 4 9 16 25 36];
F = 0:50:300;
a = polyfit(H,F,6)

a = -5.2609e-005   4.8225e-003 -1.6129e-001       2.4566e+000
    -1.7621e+001   6.5321e+001 -2.9717e-011
```

Figure 2.6 shows the data points and the interpolating polynomial. As expected, all the data points lie on the interpolating polynomial. However, it may be somewhat unsettling to discover the estimated outflow with 20 ft of liquid in the tank is considerably less than the outflow measured when the liquid level was 16 ft. (Refer to Figure 2.6) Even worse, suppose you rely on the interpolating polynomial to predict what the outflow would be when there was 30 ft. of liquid pushing the fluid out. From MATLAB, the results are:

```
»H=[0 1 4 9 16 25 36];
»F=0:50:300;
»a=polyfit(H,F,6)
»f6_20=polyval(a,20)
»f6_30=polyval(a,30)

f6_20  =   168.96

f6_30  =   616.64
```

Of course, one look at the graph of $f_6(H)$ is enough to eliminate its consideration as an interpolating polynomial over the range of fluid levels $0 \le H \le 36$.

In case you're wondering, the dotted function in Figure 2.6 is the correct function $F = f(H)$ for determining the outflow F for any fluid level H. Using some simple principles from Physics, it can be shown that

$$F = f(H) = c\sqrt{H} \qquad (2.16)$$

where the constant of proportionality c is easily computed by substituting in any one of the measured data points (other than $H = 0$, $F = 0$). Using $H = 25$ ft., $F = 250$ gpm, gives $250 = c\sqrt{25}$ from which $c = 50 \text{ gpm} / \text{ft}^{\frac{1}{2}}$. Since we know the true function $f(H)$, let's evaluate the correct outflow for several different fluid levels and compare the results to the estimated outflows using the sixth order interpolating polynomial.

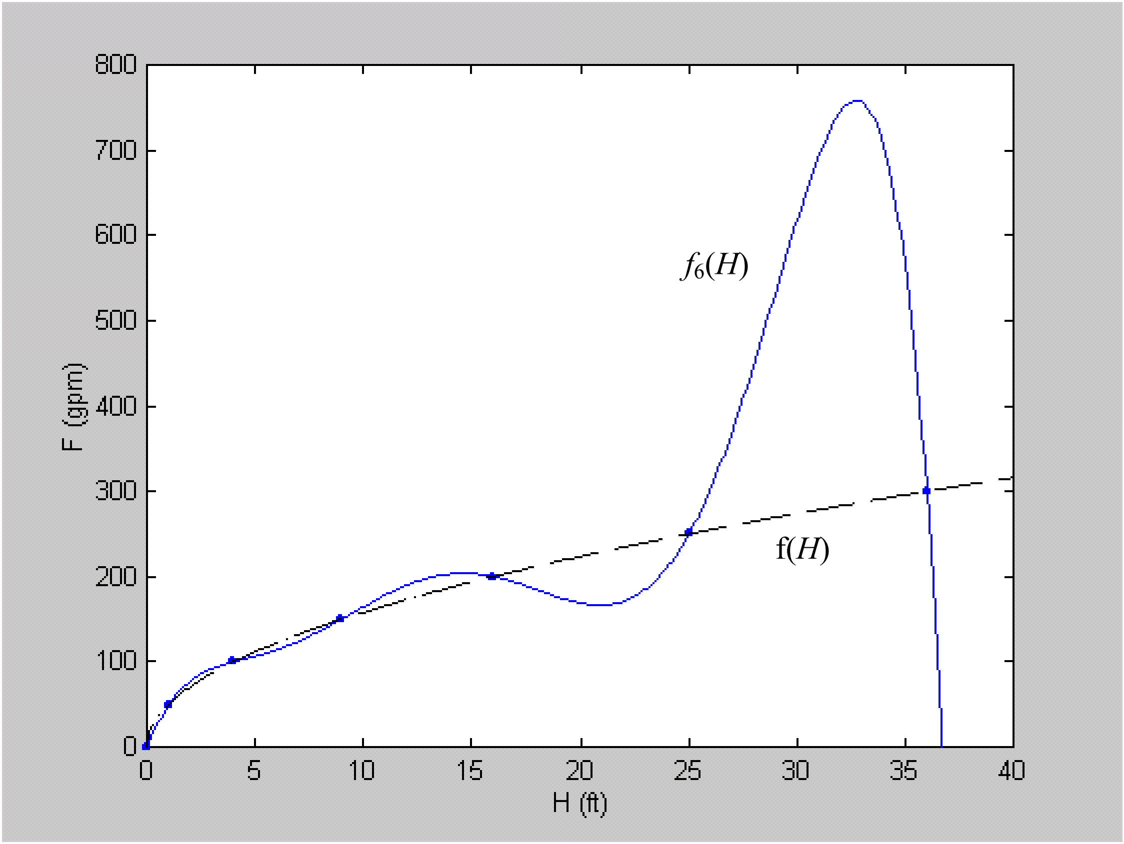


Figure 2.6 Interpolating Polynomial for Data in Table 3.3

Using MATLAB,

```
»for H = 5:5:35
»    f6_H = polyval(a,H)
»    F = 50*H.^0.5;
»    Diff = F - f6_H
»    fprintf('H = %.1f, F(est) = %.1f, F(true) = %.1f,
»    F(true) - F(est) = %.1f\n',H,f6_H, F, Diff)
»end
```

```
H =  5.0, F(est) = 106.6, F(true) = 111.8, F(true) - F(est) = 5.2
H = 10.0, F(est) = 164.4, F(true) = 158.1, F(true) - F(est) = -6.3
H = 15.0, F(est) = 203.5, F(true) = 193.6, F(true) - F(est) = -9.8
H = 20.0, F(est) = 169.0  F(true) = 223.6, F(true) - F(est) = 54.6
H = 25.0, F(est) = 250.0, F(true) = 250.0, F(true) - F(est) = 0.0
H = 30.0, F(est) = 616.6, F(true) = 273.9, F(true) - F(est) = -342.8
H = 35.0, F(est) = 564.9, F(true) = 295.8, F(true) - F(est) = -269.1
```

Notice that the estimated flow from the interpolating polynomial, F(est) is within 5% of the correct flow, F(true), up to roughly 15 ft. of liquid in the tank.

High order polynomials should not be dismissed entirely when it comes to interpolation. Its possible to obtain reliable estimates in certain situations. For example, we might try obtaining additional data points in the interval between H = 16 ft. and H = 36 ft. The order of the interpolating polynomial would increase by one for each additional point, however the resulting polynomial is constrained to pass through the added points leaving less "wiggle room" to behave as it did previously.

The higher order polynomials could still exhibit fluctuations in the original interval where the inaccuracies were most pronounced ($16 \le H \le 36$). To make matters worse, serious errors could be introduced where the original interpolating polynomial $f_6(H)$ was fairly accurate, i.e. from 0 ft. to 16 ft. Figure 2.7 shows what happens when the original set of data points in Table 2.3 is supplemented with three additional points from $f(H)$ in Equation (2.16). The top graph is the same as Figure 2.6 and the lower one shows the dramatic improvement resulting from the use of $f_9(H)$ as the interpolating polynomial.

It should be readily apparent at this point why interpolating functions should not, as a rule, be used for extrapolation. Indeed, the sixth order interpolating polynomial $f_6(H)$ has a zero near H = 37 ft. and the ninth order polynomial is grossly inaccurate as the height increases above H = 36 ft.

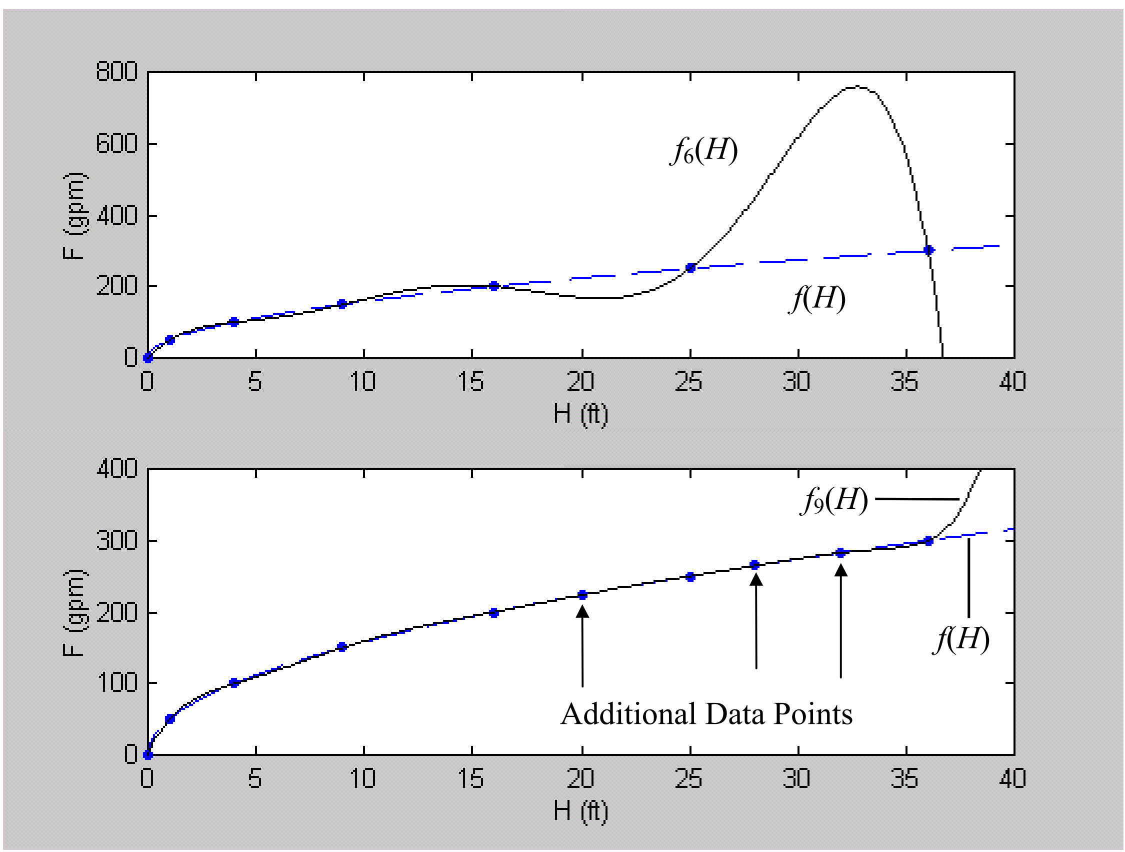


Figure 2.7 Sixth and Ninth Order Interpolating Polynomials for Interpolation of Flowrates Based on Fluid Level

Under the right conditions, interpolating polynomials can be used to optimize functions described by discrete data points. That is, minima and maxima of interpolating polynomials can be used to estimate local or global extreme values of the underlying function $f(x)$ despite the absence of an analytical form. In a specific application, an analyst familiar with the inherent relationship among the variables will ordinarily be able to discriminate between the existence of a true optimal condition and a spurious one resulting from the use of polynomial interpolation. For example, the sixth order polynomial in Figure 2.6 indicates the presence of a local maximum between H = 30 ft and H = 35 ft. Physically this makes no sense and in fact is sufficient reason to reject using $f_6(H)$ over that interval.

The location(s) of extreme points of an interpolating polynomial can be found by looking for critical values, i.e. zeros of the first derivative or by observation of the interpolating function in the region of interest. The following example demonstrates how interpolating polynomials can be used to solve for an optimal condition.

Example 2.4

An Expressway Authority must determine the access toll charge for a road under its jurisdiction. Based on preliminary studies the authority has obtained the following data relating monthly traffic will the toll charged.

x, Toll ($)	$ 0.25	$ 0.50	$ 0.75	$1. 00	$1.25
y, Monthly Traffic	600,000	450,000	200,000	100,000	40,000

Monthly payments to bond holders to pay off the long term bonds used to finance construction of the expressway is $ 150,000. The toll will be a multiple of 5 cents.

a) Use a fourth order interpolating polynomial $y = f_4(x)$ to approximate the true relationship $y = f(x)$ between monthly traffic and the access toll charge. Plot the data points and the interpolating polynomial evaluated at 5 cent toll increments on the same graph.

b) The monthly profit P is the difference between the monthly revenue, $R = xy$, and the fixed bond payment c = $ 150,000. Use the interpolating polynomial $y = f_4(x)$ to generate a second curve to approximate the relationship between P and x.

c) Estimate the maximum profit and the corresponding toll charge to generate the optimum profit.

d) What should the toll be set at if the authority wishes to maximize ridership without losing money each month?

A MATLAB m-file was written to generate the graphs below. From the bottom graph, a toll of $ 0.55 will result in optimized monthly profits of $50,460. Zero profit is expected at tolls of $ 0.30 and $ 0.80 with maximum ridership of nearly 581,000 vehicles per month at the lesser toll.

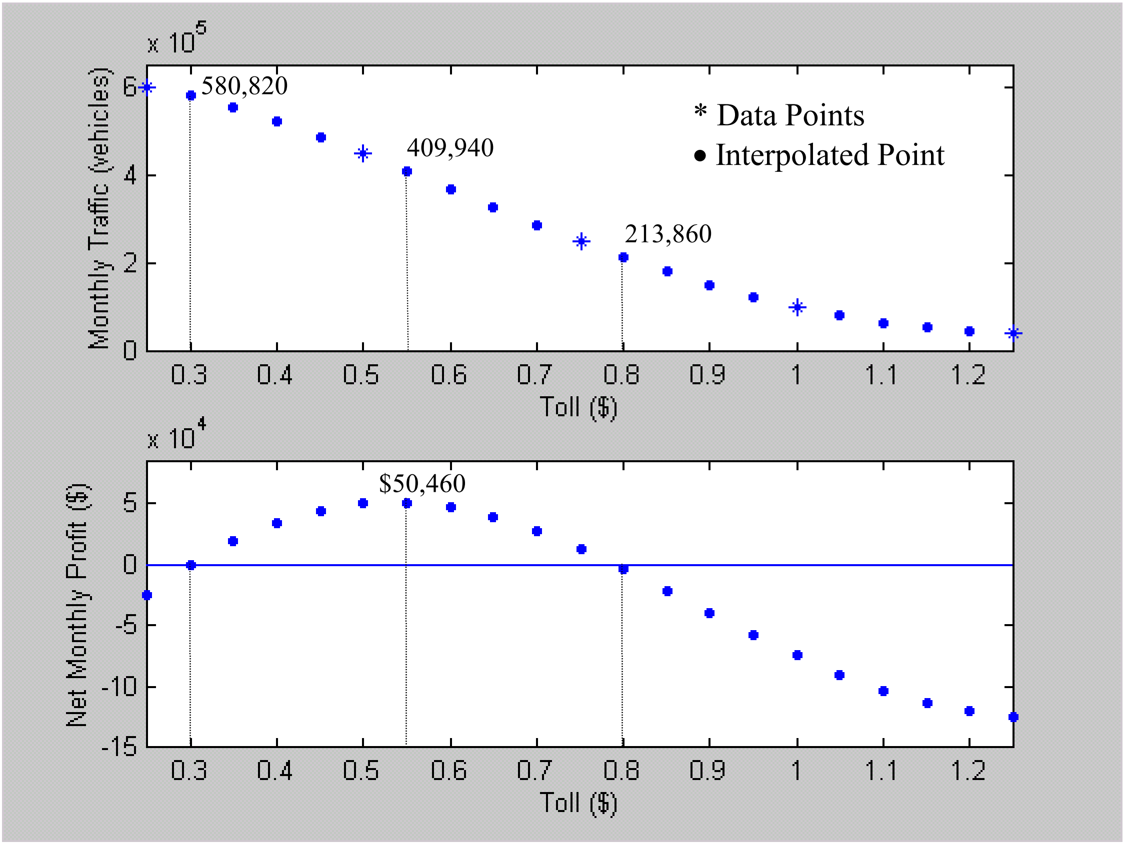


Figure 2.8 Optimum Toll Selection Based on Interpolation of Traffic vs. Toll Data

Exercises

Use MATLAB to approximate the function $f(x) = \sin 2\pi x$ for $0 \le x \le 1$ with the polynomial

$$f_n(x) = \sum_{i=0}^{n} a_i x^i$$

that passes through n+1 equally spaced data points [$(x_i, f(x_i)$, $i = 0,1,2,\ldots,n$] from $x = 0$ to $x = 1$. Start with $n = 1$ and keep incrementing until the maximum error over all data points is less than 0.05, i.e.

$$\underset{i}{Max}\left[f(x_i) - f_n(x_i)\right] \le 0.05 \qquad\qquad i = 0,1,2,\ldots,n$$

Plot the final approximating polynomials along with $f(x)$ on the same graph.

2. Repeat Problem 1 using $f(x) = \tan^{-1} x$ for $0 \le \mathrm{x} \le 10$.

3. Repeat Problem 1 using $f(x) = e^x$ for $0 \le x \le 3$.

4. Graph the data points for the data in the table listing the target heart rate during exercise for different ages. Find and graph the full interpolating polynomial, i.e. the one that passes through all the data points. Is the full order interpolating polynomial the best one to represent the data points? Use the polynomial best suited for interpolation to estimate the target heart rate of a 27 year old and a 62 year old. (Source: The Running Book by Jim Fixx, Beekman House, 1978)

Age	20	25	30	35	40	45	50	55	60	65	70
Target Heart Rate	150	146	142	139	135	131	127	124	120	116	112

Table of Data Points for Problem 4

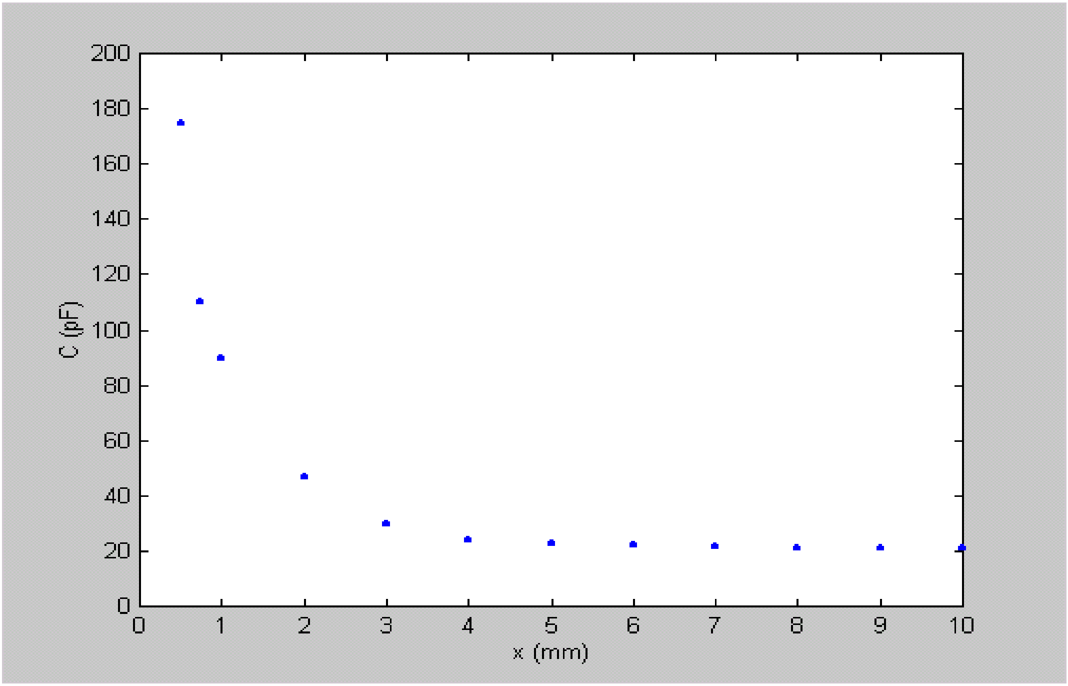
5. Measurements of a capacitive displacement transducer produced the graph shown. Estimate the (x,C) coordinates of each point.

   a) Find a low order polynomial interpolating function, based on a subset of the data points, for estimating the output capacitance C in pf corresponding to an input of x mm separation between the plates of the capacitor.

   b) On the same graph show all the data points, indicate the ones selected for determining the interpolating polynomial, and the interpolating function.

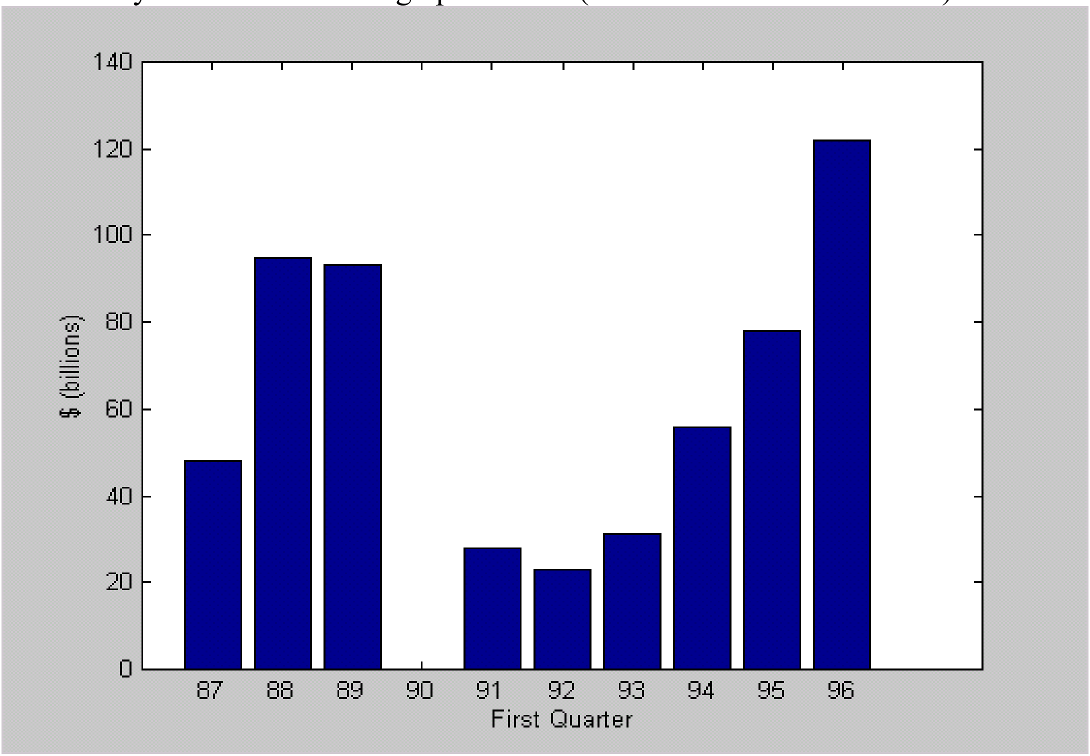
   c) Estimate the output capacitance when x = 1.5 mm, 4.5 mm, and 9.5 mm.

d) Repeat Parts a), b) and c) using the full interpolating polynomial.



Capacitive Displacement Transducer Data Points for Problem 5

6. The total value of U.S. mergers and acquisitions during the first quarter of each calendar year is shown in the graph below. (Source: Securities Data Co.)



Bar Graph for Problem 6

a) There is a missing data point for the year 1990. Use polynomial interpolation to estimate the total value of U.S. mergers and acquisitions for the first quarter of 1990. (The actual value is $40 billion)

b) Extrapolate the interpolating polynomial to estimate the total value of U.S. mergers and acquisitions for the first quarter of 1997. (The actual value is $183.9 billion)

7. Consider the function

$$f(x) = \begin{cases} x & 0 \le x \le 1 \\ 1 & 1 \le x \le 2 \\ 3 - x & 2 \le x \le 3 \end{cases}$$

a) Graph the function for x between 0 and 3.

b) Generate a set of data points from $f(x)$ as follows:

5 equally spaced points between 0 and 1

9 equally spaced points between 1.1 and 1.9

5 equally spaced points between 2 and 3

Plot the data points on the same graph.

c) Plot the interpolating polynomial which passes through all 19 data points.

d) Calculate the interpolation error at x = 0.05, 0.1, 0.4, 0.9, 1.25, 1.5, 1.75, 2.1, 2.6, 2.9 and 2.95 and comment on the results.

8. A graph of the function $f(x) = \dfrac{15x^2}{14 + x^3}$ is an example of an asymmetrical humped curve.

a) Graph the function over the interval $0 \le x \le 14$.

b) Sample the function at 8 equally spaced points, i.e. $[(x_i, f(x_i)]$, $i = 0, 1, \ldots, 7$ where the first point is at $x = 0$. Find the 7^{th} order interpolating polynomial through the data points and plot on the same graph as the function.

c) Choose up to 8 points from the function anywhere in the interval $0 \le x \le 14$ and find the interpolating polynomial that passes through each point. Plot it on the same graph with the function $f(x)$ and the interpolating polynomial from Part b). Compare the results in a quantitative manner.

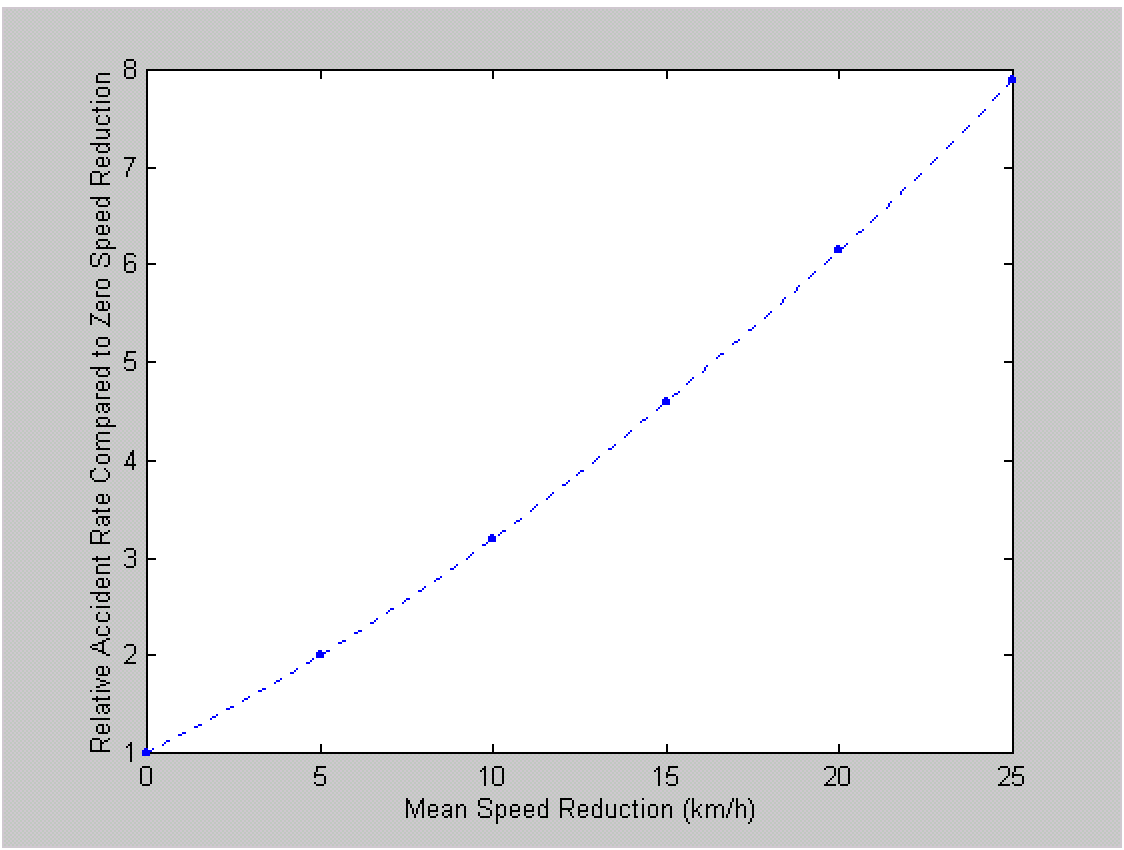
9. A retail store owner is trying to determine the best price for a popular tee-shirt. He has been experimenting with different prices to assess its effect on sales. The following sales records have been accumulated over the past six weeks.

Week No.	1	2	3	4	5	6
Shirt Price	$3.00	$6.00	$5.25	$3.75	$3.50	$4.75
Sales	835	296	403	778	834	512

Sales Figures for Problem 9

Find the optimum shirt price, i.e. the price which maximizes sales revenue based on the interpolated sales.

10. The figure below contains data points from the results of crash studies on curves where speed reduction plays a significant role in the occurrence of accidents. Curves with identical speed limits were compared. In one case drivers were required to reduce speed as they approached from a straight section of road while in the other case the straight section and curve had similar posted speed limits. The relative accident rate on reduced speed curves compared to same speed curves is plotted versus the mean speed reduction. (Source: R. Krammes, Public Roads, Sept/Oct 1997)



Relative Crash Rate Data for Problem 10

a) Find an interpolating polynomial to approximate the underlying function (shown dotted).

b) Estimate the relative accident rate (compared to a zero speed reduction curve) on a curve where the posted speed limits approaching the curve and on the curve are 55 and 40 km/h respectively.
c) Find an interpolating polynomial to approximate the inverse relationship between mean speed reduction and relative accident rate.
d) Estimate the mean speed reduction for the relative accident rate found in Part b).