Chapter 2  Interpolation

Reference

James F. Epperson (Revised Edition, 2007)
- Chapter 4

Content

2.1 Introduction

2.2 Linear Interpolation

2.3 Piecewise Linear Interpolation

2.4 Lagrange Interpolation

2.5 Least-Squares Regression (Least Squares Data Fitting)

2.6 Spline Interpolation
2.1 Introduction

• Typical sample problems:
  1. Given \( \{x_i, y_i \mid 0 \leq i \leq n\} \), nodes \( x_i \) distinct, find simple \( p_n(x) \) such that \( p_n(x_i) = y_i, 0 \leq i \leq n \).
  2. Given \( \{x_i, 0 \leq i \leq n\} \) and \( f(x) \), replace complicated \( f(x) \) with \( p_n(x) \) such that \( p_n(x_i) \approx f(x_i), 0 \leq i \leq n \).
  3. Assume data includes (experimental) errors, we want to “smooth” the data \( \rightarrow \) least squares data fitting (first year lab).

• Motivation: we will learn how to construct an approximating polynomial to a given set of data points or function.

• Find a polynomial \( p_n(x_i) \) of degree \( \leq n \) that approximates a given data \( y_i \) or function \( f(x_i) \) for a set of nodes \( \{x_i, 0 \leq i \leq n\} \) i.e.

\[
 p_n(x_i) = y_i \quad \text{or} \quad p_n(x_i) = f(x_i) \quad \text{for } 0 \leq i \leq n. \tag{2.1}
\]

• Also \( y_i = f(x_i) \) is a special case.

2.2 Linear Interpolation

• A straight line approximates \( f(x) \).

• Given two points \( (x_0, f(x_0)) \) and \( (x_1, f(x_1)) \). Linear polynomial (interpolant) with degree \( n = 1 \):

\[
 p_1(x) = \frac{x_1 - x}{x_1 - x_0} f(x_0) + \frac{x - x_0}{x_1 - x_0} f(x_1). \tag{2.2}
\]

• Obey interpolatory condition \( p_i(x_i) = f(x_i) \) for \( i = 0, 1 \).

• Error:

\[
 |f(x) - p(x)| \leq \frac{1}{8} (x_1 - x_0)^2 \max_{x_0 \leq x \leq x_1} |f''(x)|. \tag{2.3}
\]

Example 2.1 (JE Chapter 2 Section 2.4 Example 2.3)

Approximate the error function \( erf(x) \) at \( x = 0.14 \) given the data

\( x = \{x_0, x_1\} = \{0.1, 0.2\} \)

and

\( f(x) = \{f(x_0), f(x_1)\} = \{0.112, 0.223\} \).
Solution:

**Step 1**

From Equation (2.2),

\[ p_1(x) = \frac{0.2 - x}{0.2 - 0.1} (0.112) + \frac{x - 0.1}{0.2 - 0.1} (0.223) = \frac{0.2 - x}{0.1} (0.112) + \frac{x - 0.1}{0.1} (0.223). \]

**Step 2**

Therefore at \( x = 0.14 \),

\[ p_1(x = 0.14) = \frac{0.2 - 0.14}{0.1} (0.112) + \frac{0.14 - 0.1}{0.1} (0.223) = 0.157 \approx \text{erf}(0.14) \]

**Step 3**

Since \( \text{erf}'(x) = \frac{2}{\sqrt{\pi}} e^{-x^2} \) then error bound from (2.3),

\[ |\text{erf}(0.14) - p_1(0.14)| \leq \frac{1}{8} (0.2 - 0.1)^2 (0.447) = 5.586 \times 10^{-4}. \]
2.3 Piecewise Linear Interpolation

- Linear interpolation is good for (very) close nodes.
- For \( n > 1 \), polynomial introduces **undesirable oscillations**.

![Figure 2.1](image)

**Figure 2.1** \( f(x) = \frac{1}{1 + 0.5x^2} \).

- Add more data points but use lower-order polynomial on subintervals ⇒ **piecewise linear interpolation**.
- Approximate a curve by breaking the given interval into subintervals and use linear interpolation on each subinterval.
- Net effect is the curve will be approximated by a set of connected straight lines (still linear interpolation).
- Use error bound Equation (2.3) separately on each subinterval.
- Definition:

Let \( f \) be a function defined on interval \([a, b]\) with \((k + 1)\) distinct nodes/points:

\[
a = x_0 < x_1 < x_2 < \cdots < x_{k-1} < x_k = b
\]

at which \( f \) is to be interpolated is a function \( p_i(x) \) with degree \( n = 1 \) that satisfies:

1. \( p_i \) is continuous on \([a, b]\);
2. at each subinterval \([x_i, x_{i+1}]\), \( i = 0, 1, 2, \ldots, k - 1 \), \( p_i \) is described by a linear interpolant:
\[ p_i(x) = q_i(x) = c_i + d_i (x - x_i), \quad i = 0, 1, 2, \ldots, k - 1; \]

3. \( p_i \) interpolates \( f \) exactly at the nodes \( x_0, x_1, \ldots, x_k \) (interpolation condition).

- What are the coefficients \( c_i \) and \( d_i \)?
- From condition 2 and condition 3, at each node \( x = x_j \):
  \[ f(x_j) = p_i(x_j) = q_i(x_j) = c_i, \quad i = 0, 1, 2, \ldots, k - 1. \]  \hspace{1cm} (2.4)
- From condition 2, continuity requires that \( q_i(x_{i+1}) = q_{i+1}(x_{i+1}) \). Thus,
  \[ d_i = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}, \quad i = 0, 1, 2, \ldots, k - 1. \]  \hspace{1cm} (2.5)

**Example 2.2** (JE Chapter 2 Section 2.4 Example 2.4)

Construct a piecewise linear approximation \( q(x) \) to \( f(x) = \log_2 x \) using nodes \( x = \{x_0, x_1, x_2\} = \{\frac{1}{4}, \frac{1}{2}, 1\} \).

**Solution:**

**Step 1**

From (2.4) at nodes \( x_0 \) and \( x_1 \),

\[ c_0 = f(x_0) = \log_2(x_0) = \log_2\left(\frac{1}{4}\right) = -2 \]
\[ c_1 = f(x_1) = \log_2(x_1) = \log_2\left(\frac{1}{2}\right) = -1 \]

**Step 2**

From (2.5) at nodes \( x_0 \) and \( x_1 \),

\[ d_0 = \frac{f(x_1) - f(x_0)}{x_1 - x_0} = \frac{-1 + 2}{\frac{1}{2} - \frac{1}{4}} = 4 \]

From (2.5) at nodes \( x_1 \) and \( x_2 \),

\[ d_1 = \frac{f(x_2) - f(x_1)}{x_2 - x_1} = \frac{0 + 1}{1 - \frac{1}{2}} = 2 \]
Step 3

At nodes $x_0$ and $x_1$,

$$q_1(x) = 4x - 3.$$  

At nodes $x_1$ and $x_2$,

$$q_2(x) = 2x - 2.$$  

Step 4

The piecewise linear polynomial to $f(x) = \log_2 x$ is

$$p_1(x) = \begin{cases} 
q_1(x) = 4x - 3, & \frac{1}{2} \leq x \leq \frac{1}{2} \\
q_2(x) = 2x - 2, & \frac{1}{2} \leq x \leq 1 
\end{cases}.$$  

Step 5

Error bounds from (2.3),

$$|\log_2 x - q_1(x)| \leq 0.1083, \quad \frac{1}{2} \leq x \leq \frac{1}{2}$$

$$|\log_2 x - q_2(x)| \leq 0.1083, \quad \frac{1}{2} \leq x \leq 1$$

$$\therefore |\log_2 x - p_1(x)| \leq 0.1083, \quad \frac{1}{2} \leq x \leq 1$$

Example 2.3

Given $f(x) = e^x$.

(a) Construct a linear approximation to $f(x) = e^x$ using nodes $x = \{x_0, x_1\} = \{0, 1\}$.

(b) Construct a piecewise linear interpolant/approximation $q(x)$ to $f(x) = e^x$ using nodes $x = \{x_0, x_1, x_2\} = \{0, \frac{1}{2}, 1\}$.

Solution:

(a) Follow exactly Example 2.1.
(b) Follow exactly Example 2.2.
2.4 Lagrange Interpolation

- Define the Lagrange polynomials or functions of degree \( n \) as

\[
L_i^{(n)}(x) = \prod_{k=0}^{n} \frac{x - x_k}{x_i - x_k} 
\]

(2.6)

with property

\[
L_i^{(n)}(x_j) = \delta_{ij} = \begin{cases} 
1 & i = j \\
0 & i \neq j 
\end{cases} 
\]

(2.7)

- Then the interpolating polynomial of degree \( n \) is given as

\[
p_n(x) = \sum_{k=0}^{n} y_k L_k^{(n)}(x) 
\]

(2.8)

with \( p_n(x) = \sum_{k=0}^{n} y_k L_k^{(n)} \) is the interpolated value at \( x_i \).

- Suitable for hand calculations.
- Some interpolation can be improved by using bigger \( n \).

Example 2.4

Given \( f(x) = e^x = y(x) \).

Find \( p_2(x) \) for \( x_i = \{x_0, x_1, x_2\} = \{-1, 0, 1\} \) in the interval \( x \in [-1, 1] \).

Solution:

We know \( n = 2 \).

Step 1: Construct all Lagrange polynomials/functions

From Equation (2.4),

\[
L_0^{(2)}(x) = \prod_{k=0}^{2} \frac{x - x_k}{x_i - x_k} = \frac{(x - x_0)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} \quad \text{for } i = 0 
\]

\[
= \frac{(x - 0)(x - 1)}{(-1 - 0)(-1 - 1)} 
\]

\[
= \frac{1}{2} x(x - 1) 
\]
Chapter 2  Interpolation

Step 2: Construct the interpolating polynomial

From Equation (2.6),

\[ p_2(x) = \sum_{k=0}^{2} y_k L_k^{(2)}(x) \]
\[ = y_0 L_0^{(2)}(x) + y_1 L_1^{(2)}(x) + y_2 L_2^{(2)}(x) \]
\[ = e^{-1} L_0^{(2)} + e^0 L_1^{(2)} + e^1 L_2^{(2)} \]
\[ = -1.0 + 3.175x + 0.543x^2 \]

We can try \( n = 4 \) (JE Chapter 4, Section 4.1, Example 4.1) i.e find \( p_4(x) \) for \( x = \{x_0, x_1, x_2, x_3, x_4\} = \{-1, -\frac{1}{2}, 0, \frac{1}{2}, 1\} \) in the interval \( x \in [-1,1] \). The solution is exactly the same as above but we need to find \( L_0^{(4)}, L_1^{(4)}, L_2^{(4)}, L_3^{(4)} \) and \( L_4^{(4)} \). Thus we find

\[ p_4(x) = \sum_{k=0}^{4} y_k L_k^{(4)}(x) \]
\[ = y_0 L_0^{(4)}(x) + y_1 L_1^{(4)}(x) + y_2 L_2^{(4)}(x) + y_3 L_3^{(4)}(x) + y_4 L_4^{(4)}(x) \]
\[ = e^{-1} L_0^{(4)} + e^{-2} L_1^{(4)} + e^0 L_2^{(4)} + e^2 L_3^{(4)} + e^4 L_4^{(4)} \]
\[ = 1.0 + 0.998x + 0.500x^2 + 0.177x^3 + 0.043x^4 \]

Comparison between \( p_2, p_4 \) and the exact function \( e^x \) is shown in Figure 2.2. There is practically no difference between \( p_4 \) and \( e^x \), hence \( p_4 \) gives good estimate of \( e^x \). As we might expect \( p_2 \) gives much more error than \( p_4 \) and it worsens at lower and higher \( x \).
Example 2.4

Figure 2.2 Example 2.4
2.5 Least-Squares Regression (Least Squares Data Fitting)

- Regression (data fitting) is the method of obtaining the **best fit** to a **given set of data**.
- Linear regression (least squares data fitting) is the method of obtaining the **best straight line (linear equation)** to a **given set of data**.
- We want to fit a straight line fit to experimental data in the same way that you have done in the First Year Laboratory (SMES1271 Physics Practical).
- The **straight line** is the one that **minimises** the sum of the squares of the distances between the data points and the line.
- Let the experimental data (true values) be \((x_k, y_k)\), \(1 \leq k \leq n\).
- The task is to find the gradient \(m\) and intercept \(b\) of the straight line

\[ y = mx + b \]  \hspace{1cm} (2.9)

- Since Equation (2.9) is an approximation, there will be an error between the model given by Equation (2.9) and the experimental data (true values).
- To fit the best straight line is by the method of least squares where we minimize the sum of the squares of the distances between the data points and the straight line (2.9) and is given by

\[ F(m, b) = \sum_{k=1}^{n} (y_k - (mx_k + b))^2 \]  \hspace{1cm} (2.10)

- Minimise \(F(m, b)\) by differentiating (2.10) and equate to zero to find the global minimum,

\[ \frac{\partial F}{\partial m} = 0 \Rightarrow \left( \sum_{k=1}^{n} x_k^2 \right) m + \left( \sum_{k=1}^{n} x_k \right) b = \sum_{k=1}^{n} x_k y_k \]  \hspace{1cm} (2.11)

\[ \frac{\partial F}{\partial b} = 0 \Rightarrow \left( \sum_{k=1}^{n} x_k \right) m + \left( \sum_{k=1}^{n} 1 \right) b = \sum_{k=1}^{n} y_k \]  \hspace{1cm} (2.12)

- Equations (2.11) and (2.12) are two simultaneous equations in \(m\) and \(b\) that can be written in a compact form:

\[ a_{11}m + a_{12}b = B_1 \]
\[ a_{21}m + a_{22}b = B_2 \]  \hspace{1cm} (2.13)

with the coefficients given in Equations (2.11) and (2.12).

- The solutions are:

\[ m = \frac{n \sum_{k=1}^{n} x_k y_k - \sum_{k=1}^{n} x_k \sum_{i=1}^{n} y_i}{n \sum_{k=1}^{n} x_k^2 - \left( \sum_{k=1}^{n} x_k \right)^2} \]  \hspace{1cm} (2.14)
\[b = \frac{\sum_{k=1}^{n} x_k^2 \sum_{k=1}^{n} y_k - \sum_{k=1}^{n} x_k \sum_{k=1}^{n} x_k y_k}{n \sum_{k=1}^{n} x_k^2 - \left(\sum_{k=1}^{n} x_k\right)^2}. \] (2.15)

- The least squares method can be generalized to higher-dimensional data for example three dimensional data in \((x, y, z)\) coordinates and higher-degree polynomials for instance quadratic polynomials of degree two.

**Example 2.5 JE Chapter 4, Section 4.10, Example 4.10**

Given the experimental data below.

<table>
<thead>
<tr>
<th>(x)</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>(y)</td>
<td>10</td>
<td>25</td>
<td>51</td>
<td>66</td>
<td>97</td>
<td>118</td>
</tr>
</tbody>
</table>

**Solution:**

The data represents a linear trend when plotted (Figure 2.3). Hence we can use Equations (2.14) and (2.15) directly. If you have forgotten please refer to your first year lab reports.

![Figure 2.3 Example 2.5](image)

**Step 1**

From (2.14) and (2.15),

\[m = 22.0286 \quad \text{and} \quad b = 6.0952.\]
Step 2

From (2.9), the straight line fit to the data is

\[ y = 22.0286x + 6.0952. \]

See Figure 2.3.

Example 2.6 JE Chapter 4, Section 4.10, Example 4.11

Given the experimental data below.

<table>
<thead>
<tr>
<th>x</th>
<th>0</th>
<th>10</th>
<th>20</th>
<th>30</th>
<th>40</th>
<th>50</th>
<th>60</th>
<th>70</th>
<th>80</th>
<th>90</th>
<th>100</th>
<th>110</th>
<th>120</th>
</tr>
</thead>
<tbody>
<tr>
<td>y</td>
<td>0.716</td>
<td>0.893</td>
<td>1.055</td>
<td>1.134</td>
<td>1.281</td>
<td>1.994</td>
<td>2.500</td>
<td>3.151</td>
<td>4.300</td>
<td>5.308</td>
<td>4.966</td>
<td>11.919</td>
<td></td>
</tr>
</tbody>
</table>

Solution:

The data represents an exponential growth (as oppose to exponential decay) curve when plotted (Figure 2.4).

\[ y = 0.0502x + 0.462 \]

Figure 2.4 Example 2.6

Step 1

Write a general equation that can represent the data. In our case it is \( y \propto e^{ax} = Ae^{ax} \) where the constants \( a \) and \( A \) are to be calculated by using the least squares method Equations (2.14) and (2.15).
**Step 2**

Transform the exponential equation in **Step 1** into a linear equation:

\[ \ln y = ax + \ln A. \]

Use Equations (2.14) and (2.15) directly i.e. we get

\[ a = m = 0.0209 \quad \text{and} \quad \ln A = b = -0.4842. \]

The straight line equation is

\[ \ln y = 0.0209x - 0.4842. \]

**Step 3**

Convert back the linear equation in Step 2 into its original form:

\[ y = e^{0.0209x - 0.4842}. \]

See Figure 2.4.

**Example 2.7 The Sun**

This is an interesting application of least squares data fitting in solar modeling in the Theoretical Physics Group. The density profile of the latest Standard Solar Model is given below. If you are interested in the details please refer to the paper, Kassim et al., Astrophysics and Space Science (2010).

<table>
<thead>
<tr>
<th>Normalised Solar Radius, ( R/R_s )</th>
<th>Solar Density, ( \rho )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.05121</td>
<td>1.27E+02</td>
</tr>
<tr>
<td>0.10168</td>
<td>8.61E+01</td>
</tr>
<tr>
<td>0.20092</td>
<td>3.47E+01</td>
</tr>
<tr>
<td>0.30111</td>
<td>1.19E+01</td>
</tr>
<tr>
<td>0.50157</td>
<td>1.32E+00</td>
</tr>
<tr>
<td>0.71389</td>
<td>1.85E-01</td>
</tr>
</tbody>
</table>

**Solution:**

The data represents an exponential decay (as oppose to exponential growth) curve when plotted (Figure 2.5).
Example 2.7 Solar Density

Figure 2.5 Example 2.7

**Step 1**

Write a general equation that can represent the data. In our case it is $y \propto e^{ax} = Ae^{ax}$ where the constants $a$ and $A$ are to be calculated by using the least squares method Equations (2.14) and (2.15).

**Step 2**

Transform the exponential equation in Step 1 into a linear equation:

$$\ln y = ax + \ln A.$$ 

Use Equations (2.14) and (2.15) directly i.e. we get

$$a = m = -10.0691 \quad \text{and} \quad \ln A = b = 5.46.$$ 

The straight line equation is

$$\ln y = -10.0691x + 5.46.$$
Example 2.7 Linear Solar Density

\[ y = -10.069x + 5.4587 \]
\[ R^2 = 0.9987 \]

Figure 2.6 Example 2.7

**Step 3**

Convert back the linear equation in Step 2 into its original form:

\[ y = e^{-10.069x + 5.46} \]

See Figure 2.6.

- How good is the fitting?
- The goodness of fit is determined by the **coefficient of determination** \( r^2 \) where

\[
\begin{align*}
    r^2 &= \frac{S_0 - S_r}{S_0} \quad \text{ (2.16)}
\end{align*}
\]

- \( S_0 \) is the sum of the squares of the deviation of the true (original) data around the mean/average of the true data, \( y_{\text{mean}} \), before the application of the linear regression:

\[
\begin{align*}
    S_0 &= \sum_{k=1}^{n} (y_{\text{original}} - y_{\text{mean}})^2 \\
    &= \sum_{k=1}^{n} (y_k - y_{\text{mean}})^2 \quad \text{ (2.17)}
\end{align*}
\]
and the mean/average is given as

\[ y_{\text{mean}} = \frac{\sum_{k=1}^{n} y_k}{n} \]  

(2.18)

- \( S_r \) is the square of the sum of the deviation/residual (or error) about the straight line between the true (original) and fitted data (after application of the linear regression):

\[ S_r = \sum_{k=1}^{n} (y_{\text{original}} - y_{\text{fitted}})^2 \]

\[ = \sum_{k=1}^{n} (y_k - (mx_k + b))^2 . \]  

(2.19)

- Equation (2.19) is the same as given by \( F(m,b) = \sum_{k=1}^{n} (y_k - (mx_k + b))^2 \).

- Then the correlation coefficient \( r \) is given as

\[
 r = \frac{n \sum_{k=1}^{n} x_k y_k - \sum_{k=1}^{n} x_k \sum_{k=1}^{n} y_k}{\sqrt{n \sum_{k=1}^{n} x_k^2 - \left( \sum_{k=1}^{n} x_k \right)^2} \sqrt{n \sum_{k=1}^{n} y_k^2 - \left( \sum_{k=1}^{n} y_k \right)^2}} .
\]  

(2.20)

- In the case of perfect fit, \( r = r^2 = 1 \) otherwise \( r = r^2 = 0 \).

- What is the accuracy of the linear regression?

- Standard error (standard deviation) in least-squares method is given by

\[
 S_{n-2} = \sqrt{\frac{S_r}{n-2}}
\]  

(2.21)

where \( n \) is the number of data points.

- Least square fitting method can easily be extended to multidimensional fitting.

- Lets say \( y \) is a function of two variables \( x_1 \) and \( x_2 \) (2-D):

\[ y = m_1 x_1 + m_2 x_2 + b . \]  

(2.22)

- Then Equation (2.22) is a plane with the squares of the sum of the residual (or error) between the original and fitted data

\[
 S_r = \sum_{k=1}^{n} (y_{\text{original}} - y_{\text{fitted}})^2 \]

\[ = \sum_{k=1}^{n} (y_k - (m_1 x_{1k} + m_2 x_{2k} + b))^2 . \]  

(2.23)
• As in the one dimensional case the residuals are minimized by differentiating with respect to the unknown coefficients:

$$\frac{\partial S_r}{\partial m_1} = 0, \frac{\partial S_r}{\partial m_2} = 0$$ and $$\frac{\partial S_r}{\partial b} = 0.$$  \hspace{1cm} (2.24)

• The unknown coefficients are solved by setting Equation (2.24) in a matrix form:

$$\begin{bmatrix}
\sum x_{1k} & \sum x_{1k}^2 & \sum x_{1k}x_{2k} \\
\sum x_{2k} & \sum x_{1k}x_{2k} & \sum x_{2k}^2
\end{bmatrix}
\begin{bmatrix}
m_1 \\
m_2 \\
b
\end{bmatrix} =
\begin{bmatrix}
\sum x_{1k}y_k \\
\sum x_{2k}y_k
\end{bmatrix}.$$  \hspace{1cm} (2.25)

2.6 Spline Interpolation

• Higher order interpolating polynomials can lead to erroneous results due to round-off error and oscillatory behaviour.
• As an alternative, use lower-order polynomials to subintervals of the given data points.
• Then the connecting polynomials are spline functions.
• This section describes three spline interpolations:

  1. First-order splines i.e. linear splines.
  2. Second-order splines i.e. quadratic splines.
  3. Third-order splines i.e. cubic splines.

2.6.1 First-Order/Linear Splines

• Represented by a set of linear functions/straight lines:

$$f_i(x) = f(x_i) + m_i(x - x_i), x_i \leq x \leq x_{i+1}, i = 0,1,2...n-1.$$  \hspace{1cm} (2.26)

• Gradient of the straight line is given by

$$m_i = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}.$$  \hspace{1cm} (2.27)
To evaluate \( f(x) \) at say, \( x_k \) \((x_0 \leq x_k \leq x_n)\):

1. Locate the interval within which \( x_k \) lies.
2. Use the appropriate equation \( f(x) = f_k(x_k) \) at \( x_k \).

Identical to the piecewise linear interpolation.

**Example 2.8**

You are given the experimental data below.

<table>
<thead>
<tr>
<th>( x )</th>
<th>( y )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.2</td>
</tr>
<tr>
<td>2.5</td>
<td>0.3</td>
</tr>
<tr>
<td>3.0</td>
<td>0.9</td>
</tr>
</tbody>
</table>

(a) Fit the data with first-order spline.
(b) Evaluate the function at \( x = 1.5 \)
Solution:

**Step 1**

<table>
<thead>
<tr>
<th>((x, y)) Pairs</th>
<th>(m_i)</th>
<th>(f(x))</th>
</tr>
</thead>
<tbody>
<tr>
<td>((0,0.2) - (1.0,0.3))</td>
<td>(m_1 = \frac{0.3 - 0.2}{1.0 - 0} = 0.1)</td>
<td>(0.3 + 0.1(x - 1.0))</td>
</tr>
<tr>
<td>((1.0,0.3) - (2.5,0.6))</td>
<td>(m_2 = \frac{0.6 - 0.3}{2.5 - 1.0} = 0.2)</td>
<td>(0.6 + 0.2(x - 2.5))</td>
</tr>
<tr>
<td>((2.5,0.6) - (3.0,0.9))</td>
<td>(m_3 = \frac{0.9 - 0.6}{3.0 - 2.5} = 0.6)</td>
<td>(0.9 + 0.6(x - 3.0))</td>
</tr>
</tbody>
</table>

**Step 2**

At \(x = 1.5\) choose the second spline with slope \(m_2\),

\[ f(x = 1.5) = 0.6 + 0.2(1.5 - 2.5) = 0.4. \]

**Figure 2.8 Example 2.8**

- From Figure 2.8, first-order splines are not smooth at knots where two splines meet.
- They exhibit discontinuous \(f'\) at the knots.
- The remedy is to use higher order splines.
2.6.2 Second-Order/Quadratic Splines

- The objective is to derive a second-order polynomial for each interval between data points in which the first-order derivative is continuous at the knots.
- Each interval is represented by

\[ f(x) = a_i x^2 + b_i x + c_i, \quad i = 1, 2 \ldots n \]  \hspace{1cm} (2.28)

![Figure 2.9](image)

- There are \( n + 1 \) points/nodes, \( n \) intervals and thus 3\( n \) unknown constants \( a_i, b_i \) and \( c_i \).
- Must have 3\( n \) equations:
  1. The functions \( f(x) \) of adjacent polynomials must be the same at interior knots.

\[
\begin{align*}
   f_i(x) &= a_i x^2 + b_i x + c_i \\
   f_{i-1}(x) &= a_{i-1} x^2 + b_{i-1} x + c_{i-1}, \\
   f_{i-1}(x) &= a_i x_{i-1}^2 + b_i x_{i-1} + c_i, \quad i = 2, 3 \ldots n 
\end{align*}
\]  \hspace{1cm} (2.29) \hspace{1cm} (2.30)

Each equation contributes \( n - 1 \) conditions and thus the total is 2\( n - 2 \) conditions.
2. The first function \( f(x_0) \) and last function \( f(x_n) \) must pass through the first and last points.

\[
f(x_0) = a_1 x_0^2 + b_1 x_0 + c_1,
\]

\[
f(x_n) = a_n x_n^2 + b_n x_n + c_n
\]

(2.31)

(2.32)

There are \((2n - 2) + 2 = 2n\) conditions.

3. The first-order derivatives \( f' \) at interior knots are equal. From (2.28),

\[
2a_i x_i + b_i = 2a_{i-1} x_{i-1} + b_{i-1} \quad i = 2, 3 \ldots n
\]

(2.33)

There are \(2n + (n - 1) = 3n - 1\) conditions.

4. Last factor is we assume second-order derivative \( f'' = 0 \) at \( x_0 \). From

\[
f''(x) = 2a_i = 0.
\]

(2.34)

- Grouping all the equations, all we can form a \(9 \times 9\) matrix:

\[
\begin{bmatrix}
  x_1^2 & x_1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
  0 & 0 & 0 & x_1^2 & x_1 & 1 & 0 & 0 & 0 \\
  0 & 0 & 0 & x_2^2 & x_2 & 1 & 0 & 0 & 0 \\
  0 & 0 & 0 & 0 & 0 & x_2^2 & x_2 & 1 & a_2 \\
  x_0^2 & x_0 & 1 & 0 & 0 & 0 & 0 & 0 & b_2 \\
  0 & 0 & 0 & 0 & 0 & x_3^2 & x_3 & 1 & c_2 \\
  2x_1 & 1 & 0 & -2x_1 & 1 & 0 & 0 & 0 & a_3 \\
  0 & 0 & 0 & 2x_2 & 1 & 0 & -2x_2 & -1 & 0 \\
  1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_3
\end{bmatrix}
\begin{bmatrix}
a_1 \\
b_1 \\
c_1 \\
a_2 \\
b_2 \\
c_2 \\
a_3 \\
b_3 \\
c_3
\end{bmatrix}
= \begin{bmatrix}
f(x_1) \\
f(x_1) \\
f(x_2) \\
f(x_2) \\
f(x_0) \\
f(x_3) \\
f(x_3) \\
f(x_3)
\end{bmatrix}.
\]

(2.35)

- Knowing the first solution, \( a_1 = 0 \) due to Eq. (2.34), the matrix reduces to a \(8 \times 8\) matrix:
We solve for \( b_i, c_1, a_2, b_2, c_2, a_3, b_3 \) and \( c_3 \).

**Example 2.9**

You are given the experimental data below.

<table>
<thead>
<tr>
<th>( i )</th>
<th>( x )</th>
<th>( y = f(x) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>

(a) Fit the data with second-order spline.
(b) Evaluate the function at \( x = 2.5 \).

\[
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & b_1 \\
0 & x_1 & x_1 & 0 & 0 & 0 & c_1 \\
0 & x_2 & x_2 & 0 & 0 & 0 & a_2 \\
0 & 0 & 0 & x_2 & x_2 & 1 & b_2 \\
0 & 0 & 0 & 0 & x_2 & x_2 & c_2 \\
1 & 0 & x_3 & 0 & 0 & 0 & a_3 \\
0 & 0 & 2x_3 & 1 & 1 & 0 & b_3 \\
0 & 0 & 2x_3 & 1 & 1 & 0 & c_3
\end{bmatrix}
= \begin{bmatrix}
f(x_1) \\
f(x_1) \\
f(x_2) \\
f(x_2) \\
f(x_0) \\
f(x_0) \\
f(x_0) \\
f(x_0)
\end{bmatrix}. \quad (2.36)
\]

\[
f_2(x) = a_2 x^2 + b_2 x + c_2
\]

\[
f_1(x) = a_1 x^2 + b_1 x + c_1
\]

\[
f_3(x) = a_3 x^2 + b_3 x + c_3
\]
Solution:

**Step 1**

We know $n = 3$. Therefore there are $3n = 9$ unknowns. Since we set $a_i = 0$ then what remains are $3n - 1 = 8$ unknowns. Thus we have to write down a $8 \times 8$ matrix.

**Step 2**

From Equation (2.29) and Equation (2.30), for $i = 2, 3$:

When $i = 2$,

\[4a_i + 2b_i + c_i = 3\]
\[4a_2 + 2b_2 + c_2 = 3\]

When $i = 3$,

\[9a_2 + 3b_2 + c_2 = 0\]
\[9a_3 + 3b_3 + c_3 = 0\]

**Step 3**

From Equation (2.31) and Equation (2.32), for $i = 0$ and $3$:

When $i = 0$,

\[a_i + b_i + c_i = 0\]

When $i = 3$,

\[16a_3 + 4b_3 + c_3 = 4\]

**Step 4**

From Equation (2.33), for $i = 2$ and $3$:

When $i = 2$,

\[4a_i + b_i - 4a_2 - b_2 = 0\]

When $i = 3$,

\[6a_2 + b_2 - 6a_3 - b_3 = 0\]
**Step 5**

From Equation (2.34), \( a_i = 0 \).

**Step 6**

Collect all the equations in **Step 1** to **Step 5** in a matrix equation:

\[
\begin{bmatrix}
2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
b_1 \\
c_1 \\
a_2 \\
b_2 \\
c_2 \\
a_3 \\
b_3 \\
c_3
\end{bmatrix}
= \begin{bmatrix}
3 \\
0 \\
0 \\
0 \\
0 \\
4 \\
0 \\
0
\end{bmatrix}
\]

**Step 7**

Solve the matrix equation using techniques described later in Chapter 5.

**Step 8**

Insert into Equation (2.28) for each interval:

\[
f_1(x) = a_1x^2 + b_1x + c_1 \quad 0 \leq x \leq 1
\]

\[
f_2(x) = a_2x^2 + b_2x + c_2 \quad 1 \leq x \leq 2
\]

\[
f_3(x) = a_3x^2 + b_3x + c_3 \quad 2 \leq x \leq 3
\]

**Step 9**

At \( x = 2.5 \), \( f_3(2.5) = a_3 \times 2.5^2 + b_3 \times 2.5 + c_3 \).

2.6.3 Third-Order Splines/Cubic Splines

- The objective is to derive a third-order polynomial for each interval between the knots.
- Each interval is represented by

\[
f(x) = a_i x^3 + b_i x^2 + c_i x + d_i, \quad i = 1, 2, \ldots n
\]

(2.37)
• There are $n + 1$ points/nodes, $n$ intervals and thus $4n$ unknown constants $a_i, b_i, c_i$ and $d_i$.
• Must have $4n$ equations.
• The conditions to be met are similar to quadratic splines.

(Last page of Chapter 2)