ENGR30003
Numerical Programming for Engineers

Semester 2, 2020

Coordinator and lecturer:
Dr Aman G. Kidanemariam
aman.kidanemariam@unimelb.edu.au

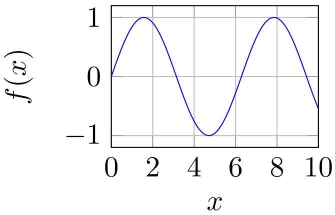
Slides developed by Prof Richard Sandberg

This week

LECTURE 17/18

Interpolation

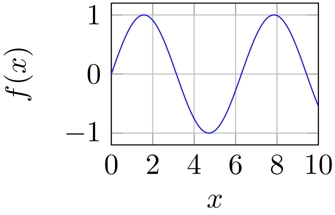
Interpolation - Motivation

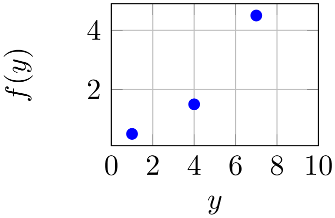


Continuous Function

$$f : x \mapsto \sin(x)$$

Interpolation - Motivation





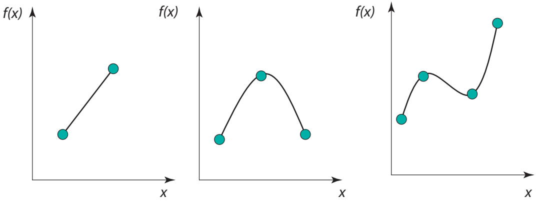
Continuous Function

$$f : x \mapsto \sin(x)$$

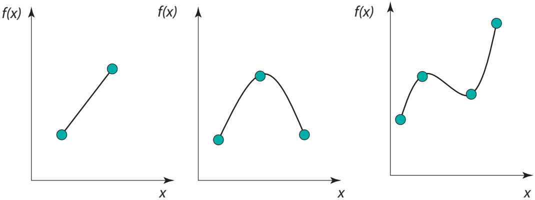
Function at discrete points

y	$f(y)$
1	.5
4	1.5
7	4.5

Polynomial Interpolation



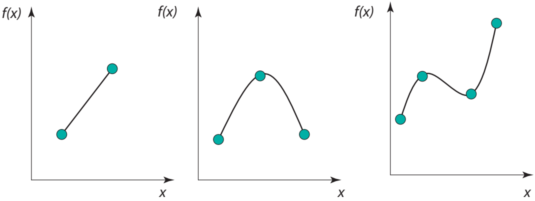
Polynomial Interpolation



We define a polynomial of the form

$$f(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n \tag{1}$$

Polynomial Interpolation

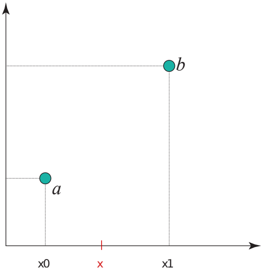


We define a polynomial of the form

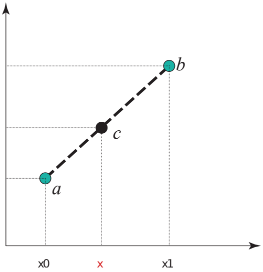
$$f(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n \tag{1}$$

- require polynomial to pass through known points
- compute coefficient a_i
- evaluate polynomial to obtain interpolated values

Newton Interpolation - Linear



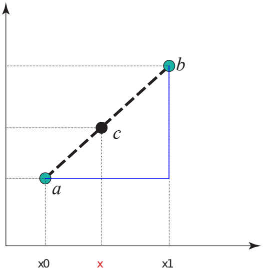
Newton Interpolation - Linear



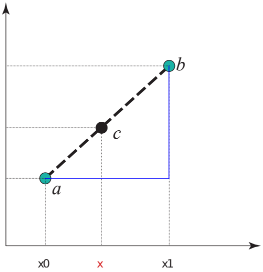
Newton Interpolation - Linear

Using similar triangles:

$$\frac{f(x_1) - f(x_0)}{x_1 - x_0} = \frac{f_1(x) - f(x_0)}{x - x_0} \tag{2}$$



Newton Interpolation - Linear



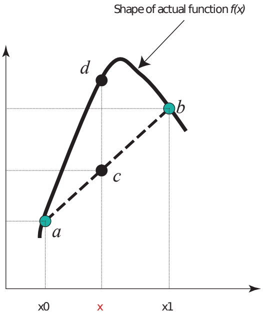
Using similar triangles:

$$\frac{f(x_1) - f(x_0)}{x_1 - x_0} = \frac{f_1(x) - f(x_0)}{x - x_0} \quad (2)$$

Rearrange Eq. (2) to give

$$f_1(x) = f(x_0) + \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_0) \quad (3)$$

Newton Interpolation - Linear



Using similar triangles:

$$\frac{f(x_1) - f(x_0)}{x_1 - x_0} = \frac{f_1(x) - f(x_0)}{x - x_0} \tag{2}$$

Rearrange Eq. (2) to give

$$f_1(x) = f(x_0) + \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_0) \tag{3}$$

Linear Interpolation Properties

$$f_1(x) = f(x_0)\left(\frac{x_1 - x}{x_1 - x_0}\right) + f(x_1)\left(\frac{x - x_0}{x_1 - x_0}\right) \tag{4}$$

Linear Interpolation Properties

$$f_1(x) = f(x_0)\left(\frac{x_1 - x}{x_1 - x_0}\right) + f(x_1)\left(\frac{x - x_0}{x_1 - x_0}\right) \tag{4}$$

A few things worth noting about the linear interpolation

- Equation (4) is called the linear-interpolation formula
- In general, the smaller the interval between x_0 and x_1, the better the approximation at point x.
- The subscript 1 for $f_1(x)$ denotes a first order interpolating polynomial: $f_1(x) = a_0 + a_1 x$.

Linear Interpolation – Example 1

Use linear interpolation to estimate the value of the function

$$f(x) = 1 - e^{-x}$$

at $x = 1.0$. Use the interval $x_0 = 0$ and $x_1 = 5.0$. Repeat the calculation with $x_1 = 4.0$, $x_1 = 3.0$ and $x_1 = 2.0$. Illustrate on a graph how you are approaching the exact value of $f(1) = 0.6321$.

Linear Interpolation – Example 1

Use

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

$$\begin{aligned}
f(x_0 = 0) &= 1 - 1 = 0 \\
f(x_1 = 5) &= 1 - e^{-5} = 0.993269053 \\
f(x_1 = 4) &= 1 - e^{-4} = 0.981684361 \\
f(x_1 = 3) &= 1 - e^{-3} = 0.950212932
\end{aligned}$$

Linear Interpolation – Example 1

Use

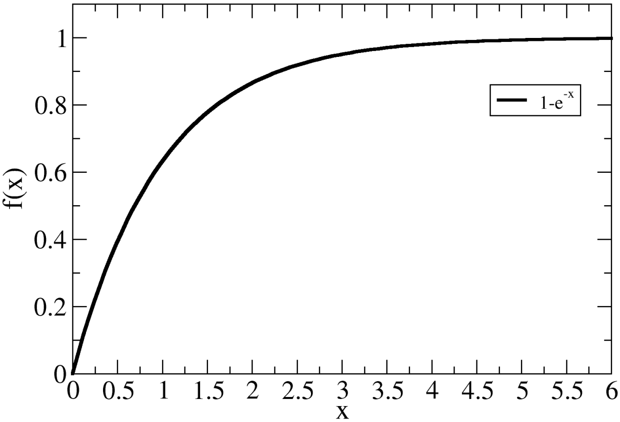
$$f_1(x) = f(x_0)\left(\frac{x_1 - x}{x_1 - x_0}\right) + f(x_1)\left(\frac{x - x_0}{x_1 - x_0}\right)$$

$$x_1 = 5 \quad \rightarrow \quad f_1(1) \quad = \quad 0.993269053\left(\frac{1-0}{5-0}\right) = 0.19966538$$

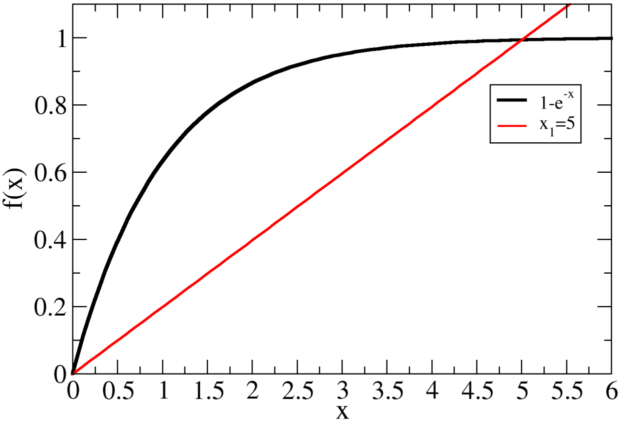
$$x_1 = 4 \quad \rightarrow \quad f_1(1) \quad = \quad 0.981684361\left(\frac{1-0}{4-0}\right) = 0.24542109$$

$$x_1 = 3 \quad \rightarrow \quad f_1(1) \quad = \quad 0.950212932\left(\frac{1-0}{3-0}\right) = 0.31673764$$

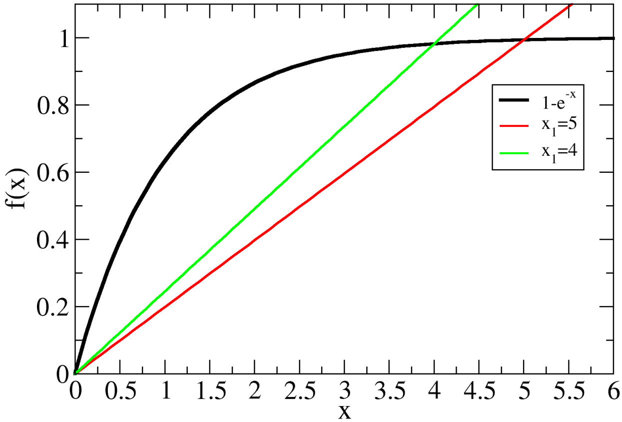
Linear Interpolation – Example 1



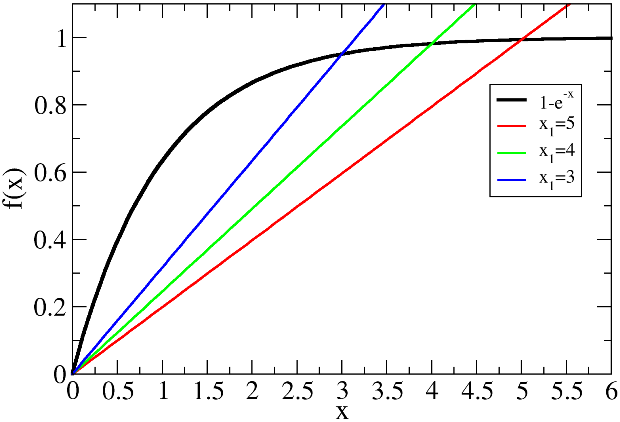
Linear Interpolation – Example 1



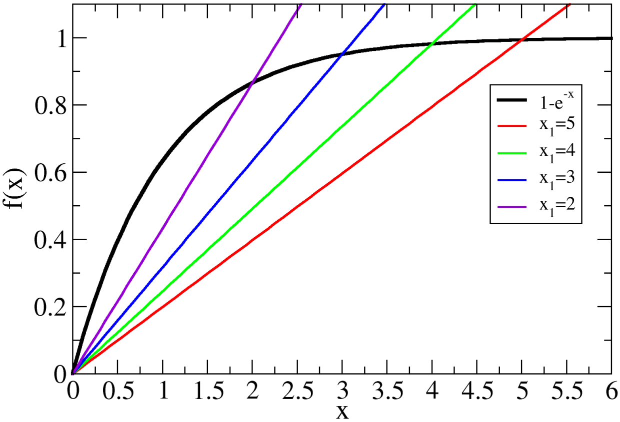
Linear Interpolation – Example 1



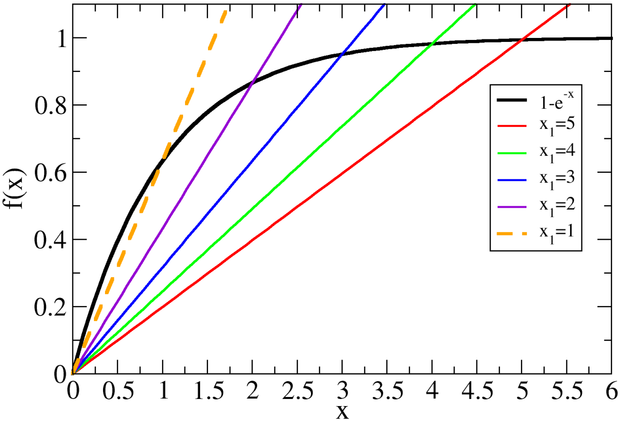
Linear Interpolation – Example 1



Linear Interpolation – Example 1

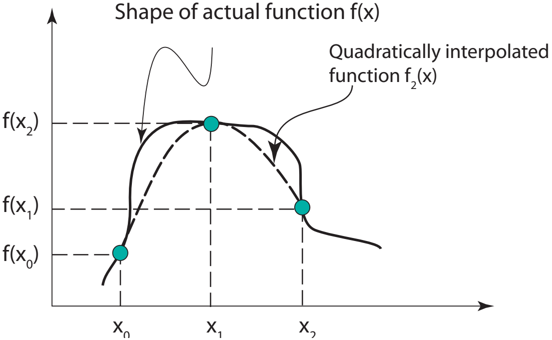


Linear Interpolation – Example 1



Newton Interpolation - Quadratic Polynomial I

$$f_2(x) = a_0 + a_1 x + a_2 x^2 \tag{5}$$



Newton Interpolation - Quadratic Polynomial II

The interpolated function is required to have the same values as the actual function at the data points, hence

- $f_2(x_0) = f(x_0)$
- $f_2(x_1) = f(x_1)$
- $f_2(x_2) = f(x_2)$

Note that the subscript 2 in $f_2(x)$ denotes a second order interpolating polynomial.
Eq. (5) is just another form of the polynomial function defined in Eq. (1). Alternatively, it can be written as

$$f_2(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) \tag{6}$$

where

$$\begin{aligned} a_0 &= b_0 - b_1x_0 + b_2x_0x_1 \\ a_1 &= b_1 - b_2x_0 - b_2x_1 \\ a_2 &= b_2 \end{aligned}$$

Newton Interpolation - Quadratic Polynomial III

Hence, in order to do quadratic interpolation, we need to find all the b's in Eq. (6).

(1) Set $x = x_0$ in Eq. (6). Therefore

$$b_0 = f(x_0) \tag{7}$$

(2) If we now let $x = x_1$ in Eq. (6) and use the result from step (1) above, we get

$$b_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0} \tag{8}$$

(3) Since $f_2(x_2) = f(x_2)$, we can use Eqs. (7) and (8) together with Eq. (6) to obtain

$$b_2 = \frac{\frac{f(x_2)-f(x_1)}{x_2-x_1} - \frac{f(x_1)-f(x_0)}{x_1-x_0}}{x_2 - x_0} \tag{9}$$

Newton Interpolation - Quadratic Polynomial IV

Equations (8) and (9) can be simplified by introducing the following notation

$$b_1 = f[x_1, x_0] = \frac{f(x_1) - f(x_0)}{x_1 - x_0} \tag{10}$$

$$b_2 = f[x_2, x_1, x_0] = \frac{f[x_2, x_1] - f[x_1, x_0]}{x_2 - x_0} \tag{11}$$

Quadratic Interpolation – Example 2

Use quadratic interpolation to estimate the value of the function

$$f(x) = 1 - e^{-x}$$

at $x = 1.0$. Use the intervals $x_0 = 0$, $x_1 = 2.0$ and $x_2 = 6.0$. Repeat the calculation with $x_0 = 0$, $x_1 = 2.0$, $x_2 = 4.0$, then $x_0 = 0$, $x_1 = 2.0$, $x_2 = 3.0$, and finally $x_0 = 0$, $x_1 = 0.5$ and $x_2 = 2.0$. Illustrate on a graph how you are approaching the exact value of $f(1) = 0.6321$.

Quadratic Interpolation – Example 2

Use

$$f_2(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1)$$

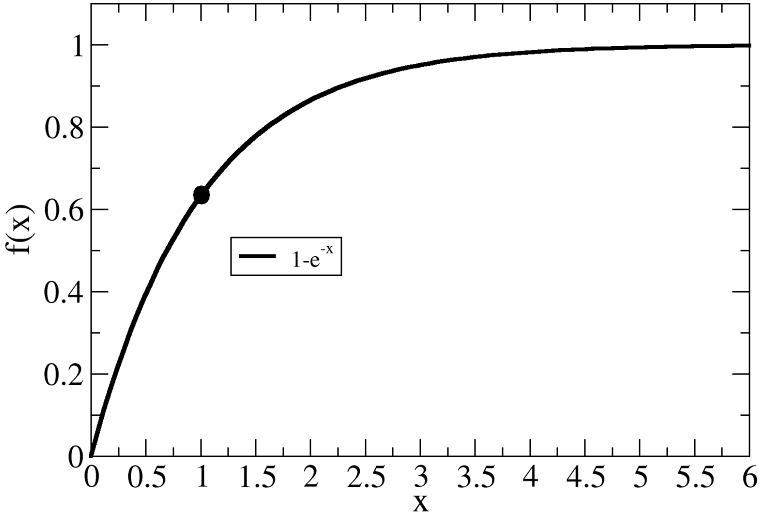
$$b_0 = f(x_0 = 0) = 0$$

$$b_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0} = \frac{f(x_1) - b_0}{x_1 - x_0}$$

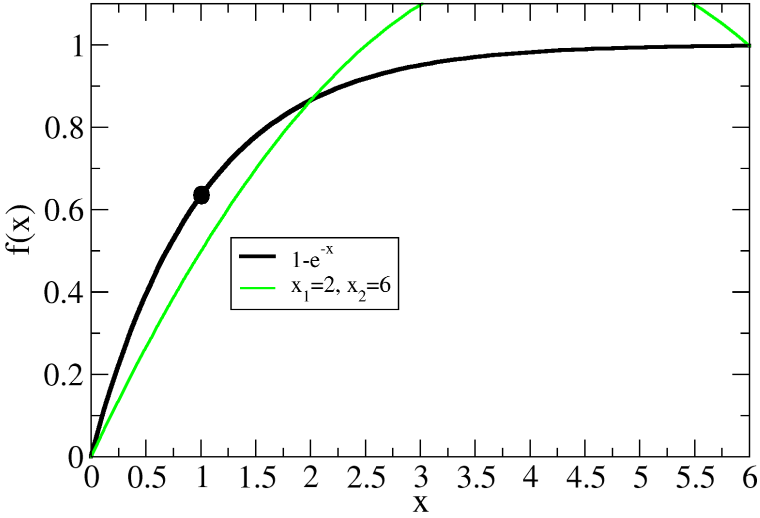
$$b_2 = \frac{\frac{f(x_2)-f(x_1)}{x_2-x_1} - \frac{f(x_1)-f(x_0)}{x_1-x_0}}{x_2 - x_0}$$

$$b_2 = \frac{\frac{f(x_2)-f(x_1)}{x_2-x_1} - b_1}{x_2 - x_0}$$

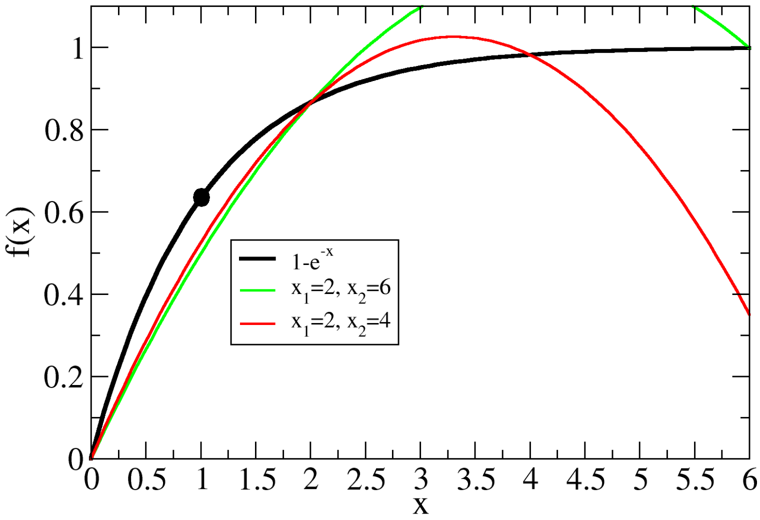
Quadratic Interpolation – Example 2



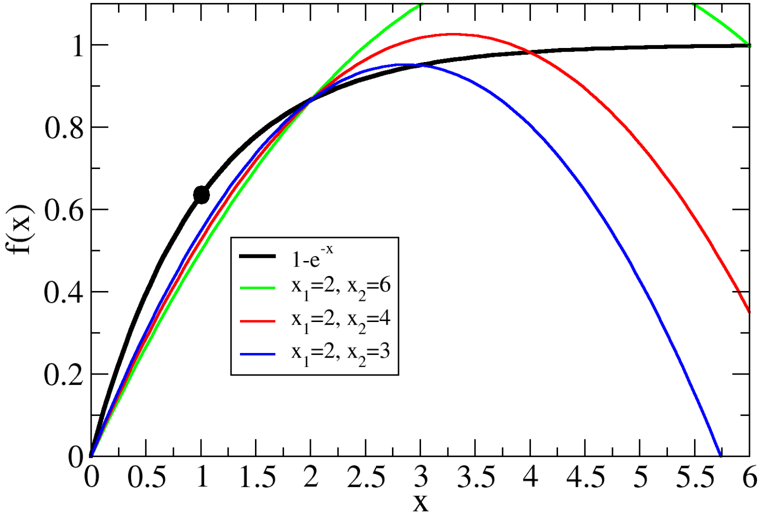
Quadratic Interpolation – Example 2



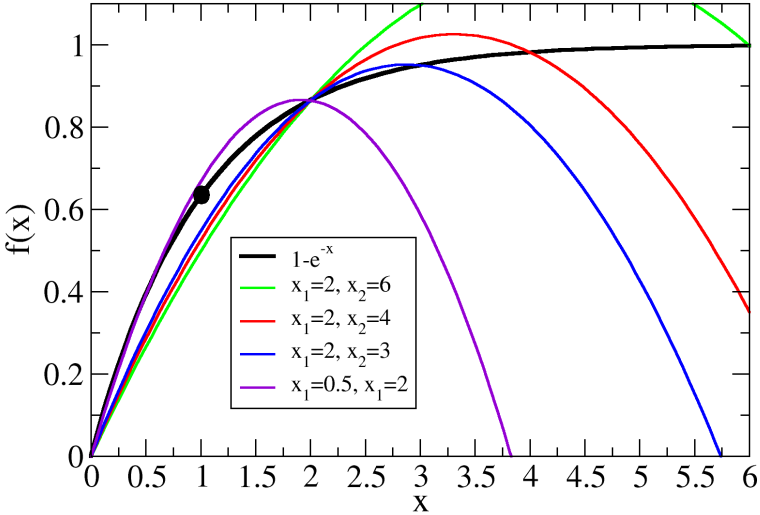
Quadratic Interpolation – Example 2



Quadratic Interpolation – Example 2



Quadratic Interpolation – Example 2



Newton Interpolation - General I

In general, if you have n data points, you will fit a polynomial of order $n-1$ through all the n data points.

$$f_n(x) = b_0 + b_1(x-x_0) + b_2(x-x_0)(x-x_1) + \cdots + \\ + b_n(x-x_0)(x-x_1)\cdots(x-x_{n-1})$$

where all the b's are defined as

$$\begin{aligned} b_0 &= f(x_0) \\ b_1 &= f[x_1, x_0] \\ b_2 &= f[x_2, x_1, x_0] \\ \vdots & \quad \vdots \quad \vdots \\ b_n &= f[x_n, x_{n-1}, \cdots, x_2, x_1, x_0] \end{aligned}$$

Newton Interpolation - General II

$$f[x_i, x_j] = \frac{f(x_i) - f(x_j)}{x_i - x_j}$$

is called the first divided difference,

$$f[x_i, x_j, x_k] = \frac{f[x_i, x_j] - f[x_j, x_k]}{x_i - x_k}$$

is called the second divided difference

$$f[x_i, x_j, x_k, x_l] = \frac{f[x_i, x_j, x_k] - f[x_j, x_k, x_l]}{x_i - x_l}$$

is called the third divided difference and

$$f[x_n, \ldots, x_0] = \frac{f[x_n, x_{n-1}, \ldots, x_1] - f[x_{n-1}, x_{n-2}, \ldots, x_0]}{x_n - x_0}$$

is called the n'th divided difference.

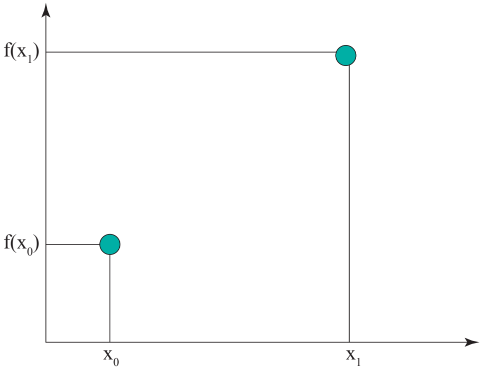
Newton Interpolation - General III

Note that the second divided difference is calculated from the first divided difference. The third divided difference is calculated from the second divided difference and so on.

i	x_i	$f(x_i)$	First	Second	Third
0	x_0	$f(x_0)$	$f[x_1, x_0]$	$f[x_2, x_1, x_0]$	$f[x_3, x_2, x_1, x_0]$
1	x_1	$f(x_1)$	$f[x_2, x_1]$	$f[x_3, x_2, x_1]$	
2	x_2	$f(x_2)$	$f[x_3, x_2]$		
3	x_3	$f(x_3)$			

Lagrange Interpolating Polynomials I

Suppose we want to interpolate between two data points, $(x_0, f(x_0))$, $(x_1, f(x_1))$



Lagrange Interpolating Polynomials II

Assume that we have two functions, $L_0(x)$ and $L_1(x)$. They are defined such that

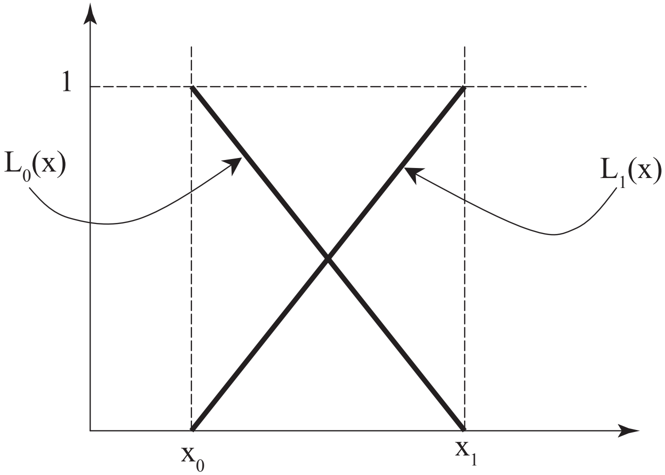
$$L_0(x) = \begin{cases} 1 & \text{if} \quad x = x_0 \\ 0 & \text{if} \quad x = x_1 \end{cases}$$

$$L_1(x) = \begin{cases} 0 & \text{if} \quad x = x_0 \\ 1 & \text{if} \quad x = x_1 \end{cases}$$

$$L_0(x) = \frac{(x - x_1)}{(x_0 - x_1)}, \qquad L_1(x) = \frac{(x - x_0)}{(x_1 - x_0)}$$

$$L_0(x) = \frac{(x - x_1)}{(x_0 - x_1)}, \qquad L_1(x) = \frac{(x - x_0)}{(x_1 - x_0)}$$

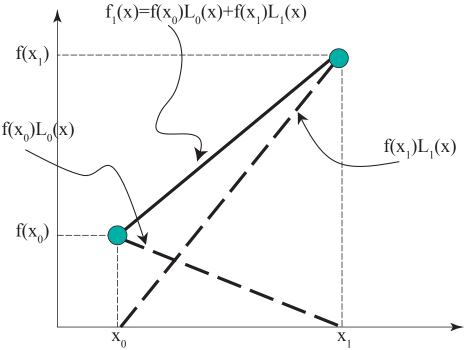
Lagrange Interpolating Polynomials III



Lagrange Interpolating Polynomials IV

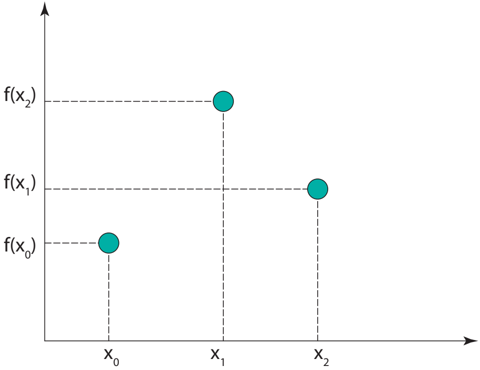
The first order Lagrange interpolating polynomial is obtained by a linear combination of $L_0(x)$ and $L_1(x)$.

$$f_1(x) = f(x_0)L_0(x) + f(x_1)L_1(x) \tag{12}$$



Lagrange Interpolating Polynomials V

Suppose we have three data points, $(x_0, f(x_0)), (x_1, f(x_1)), (x_2, f(x_2))$ and we want to fit a polynomial



Lagrange Interpolating Polynomials VI

Assume that we have three functions, $L_0(x), L_1(x)$ and $L_2(x)$ defined such that

$$L_0(x) = \begin{cases} 1 & \text{if} \quad x = x_0 \\ 0 & \text{if} \quad x = x_1 \\ 0 & \text{if} \quad x = x_2 \end{cases} \tag{13}$$

$$L_1(x) = \begin{cases} 0 & \text{if} \quad x = x_0 \\ 1 & \text{if} \quad x = x_1 \\ 0 & \text{if} \quad x = x_2 \end{cases} \tag{14}$$

$$L_2(x) = \begin{cases} 0 & \text{if} \quad x = x_0 \\ 0 & \text{if} \quad x = x_1 \\ 1 & \text{if} \quad x = x_2 \end{cases} \tag{15}$$

Lagrange Interpolating Polynomials VII

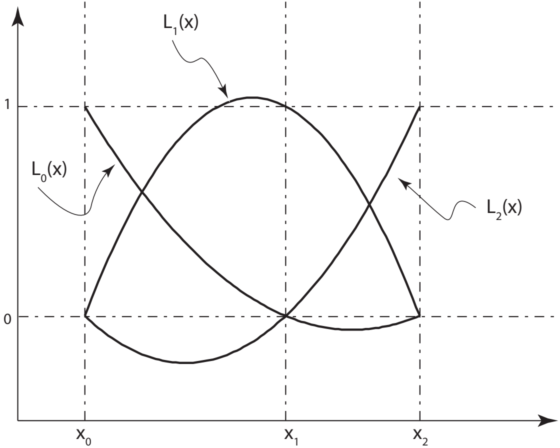
It should be easy to convince yourself that the following expressions for $L_0(x), L_1(x)$ and $L_2(x)$ satisfy the conditions listed in Eqs. (13) - (15).

$$L_0(x) = \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} \tag{16}$$

$$L_1(x) = \frac{(x - x_2)(x - x_0)}{(x_1 - x_2)(x_1 - x_0)} \tag{17}$$

$$L_2(x) = \frac{(x - x_1)(x - x_0)}{(x_2 - x_1)(x_2 - x_0)} \tag{18}$$

Lagrange Interpolating Polynomials VIII



Lagrange Interpolating Polynomials IX

The second order Lagrange interpolating polynomial is obtained by a linear combination of $L_0(x), L_1(x)$ and $L_2(x)$.

$$f_2(x) = f(x_0)L_0(x) + f(x_1)L_1(x) + f(x_2)L_2(x) \tag{19}$$

In general, the nth order Lagrange polynomial (i.e. the Lagrange polynomial that can be fit through $n+1$ data points) can be represented concisely as

$$f_n(x) = \sum_{i=0}^{n} L_i(x) f(x_i) \tag{20}$$

where

$$L_i(x) = \prod_{j=0, j\neq i}^{n} \frac{x - x_j}{x_i - x_j} \tag{21}$$

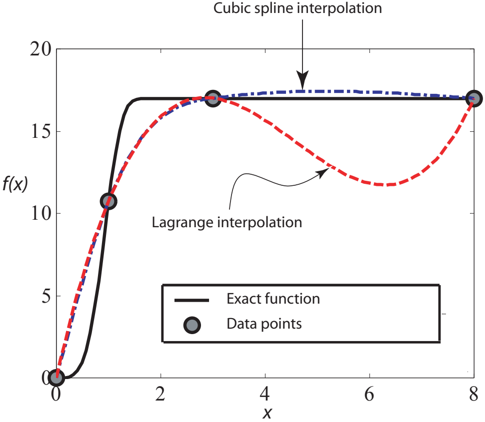
Lagrange Interpolating Polynomials X

Note that:

- The Lagrange interpolating polynomials are just a different form of the Newton interpolating polynomials.
- The main advantage is that they are slightly easier to program.
- They are slightly slower to compute than the Newton Interpolating polynomials.

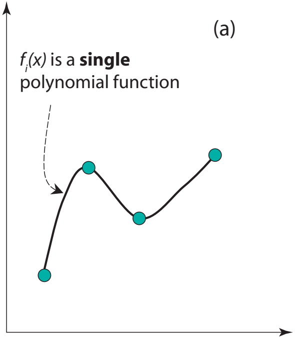
Spline Interpolation I

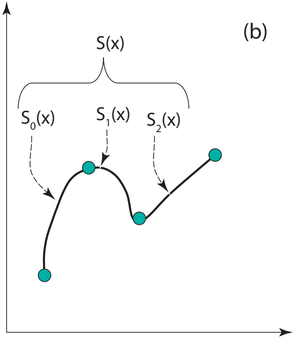
Newton or Lagrange interpolation can lead to erroneous results when there is an abrupt change in data



Spline Interpolation II

Splines are made up of piecewise polynomials connecting only two data points. This is different to Newton or Lagrange polynomials where you have only one polynomial connecting all data points.

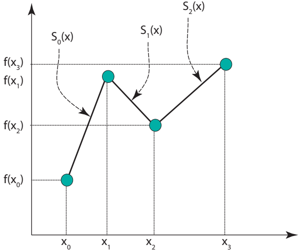




Spline Interpolation III

- The spline function,$S(x)$, is made up of several polynomial functions, $S_i(x)$.
- $S(x) = S_i(x)$ for $x_i < x < x_{i+1}$.
- $S_i(x)$ is ONLY defined between x_{i+1} and x_i. Outside of the interval, $S_i(x)$ is not defined and has got NO meaning.
- If $S_i(x)$ are linear functions, then $S(x)$ is called a linear spline.
  $S_i(x)$ quadratic $\rightarrow$ $S(x)$ quadratic spline.
  $S_i(x)$ cubic $\rightarrow$ $S(x)$ cubic spline

Linear Spline Interpolation I



- $S_i(x) = a_i + b_i(x - x_i)$
- $S(x) = S_i(x)$ for $x_i < x < x_{i+1}$.
- Note that $S(x)$ is continuous but the derivative of $S(x)$, $S'(x)$, is not continuous at $x = x_i$.
- If there are $n + 1$ data points, there are only n intervals and hence, n number of defined $S_i(x)$.

Linear Spline Interpolation II

Need to find all the a_i's and b_i's in order to find $S(x)$. For this case, there are $2n$ number of unknowns. In order to find all the unknowns, we need to impose the following requirements

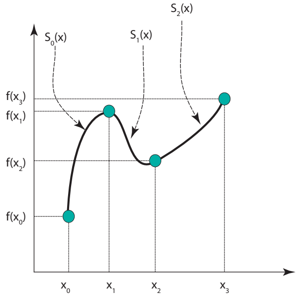
1. Require $S(x)$ to have the values of $f(x_i)$ at $x = x_i$, hence,

$$S_i(x_i) = a_i = f(x_i) \tag{22}$$

2. Require that the function values at adjacent polynomials must be equal at the interior nodes

$$\begin{aligned} S_i(x_{i+1}) &= S_{i+1}(x_{i+1}) \\ a_i + b_i(x_{i+1} - x_i) &= a_{i+1} \\ b_i &= \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i} \end{aligned} \tag{23}$$

Quadratic Spline Interpolation I



- $S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2$
- $S(x) = S_i(x)$ for $x_i < x < x_{i+1}$.
- $S(x)$ is continuous and the derivative of $S(x)$, $S'(x)$, is also continuous at $x = x_i$.
- For $n + 1$ data points, there are n number of defined $S_i(x)$.

Quadratic Spline Interpolation II

Need to find all a_i's, b_i's and c_i's to completely define $S(x)$, hence we have to impose $3n$ conditions. To do this we will require that

1. $S(x)$ to have the values of $f(x_i)$ at $x = x_i$, hence,

$$S_i(x_i) = a_i = f(x_i) \tag{24}$$

2. Function values at adjacent polynomials must be equal at interior nodes

$$S_i(x_{i+1}) = S_{i+1}(x_{i+1}) \tag{25}$$

$$a_i + b_i(x_{i+1} - x_i) + c_i(x_{i+1} - x_i)^2 = a_{i+1} \tag{26}$$

For the sake of conciseness, let $h_i = x_{i+1} - x_i$, so

$$a_i + b_i h_i + c_i h_i^2 = a_{i+1} \tag{27}$$

Quadratic Spline Interpolation III

3. Derivative of $S_i(x)$, $S_i^{'}(x)$, to be continuous at the interior nodes

$$S_i^{'}(x_{i+1}) = S_{i+1}^{'}(x_{i+1}) \tag{28}$$

Equation (28) leads to

$$b_i + 2c_i h_i = b_{i+1} \tag{29}$$

From Eq. (27) we can get

$$b_i = \frac{a_{i+1} - a_i}{h_i} - c_i h_i \tag{30}$$

Substitute Eq. (30) into (29) and rearrange to get

$$c_i = \frac{1}{h_i}\left(\frac{a_{i+1}}{h_i} - a_i\left[\frac{1}{h_{i-1}} + \frac{1}{h_i}\right] + \frac{a_{i-1}}{h_{i-1}} - c_{i-1}h_{i-1}\right) \tag{31}$$

Quadratic Spline Interpolation IV

Hence, to construct quadratic splines, do the following

1. Make use of Eq. (24) and set $a_i = f(x_i)$.
2. Assume $c_0 = 0$. This effectively makes the first spline segment linear!
3. Use Eq. (31) to obtain all other values of c_i's.
4. Use Eq. (30) to obtain the b_i's.

After steps 1-4, we can evaluate the function values at the point you are interested from the following formula

$$S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2. \tag{32}$$

The only thing left is to figure out which interval your x value belongs to.

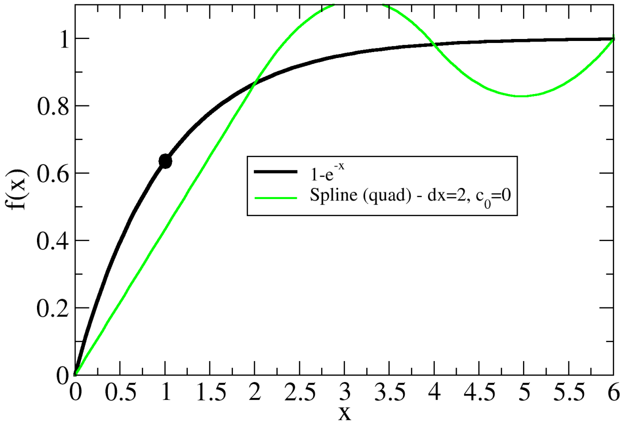
Piecewise quadratic Interpolation – Example 3

Use quadratic splines to estimate the value of the function

$$f(x) = 1 - e^{-x}$$

at $x = 1.0$. Use the intervals $x_0 = 0$, $x_1 = 2.0$, $x_2 = 4.0$ and $x_3 = 6.0$. Plot the spline for the interval $0 \leq x \leq 6$.

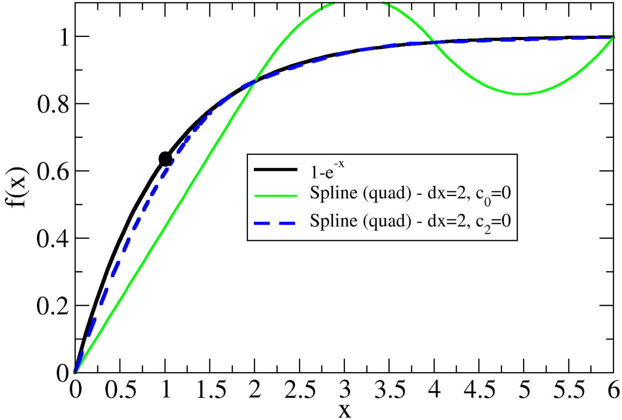
Piecewise quadratic Interpolation – Example 3



Notice, spline solution linear for first segment. For this function, this is probably not where we want it to be linear (and then quadratic for larger x).

Replace condition $c_0 = 0$ by $c_2 = 0$ and see what happens.

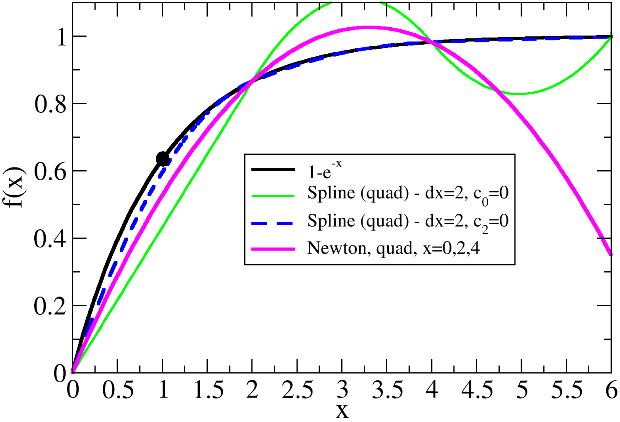
Piecewise quadratic Interpolation – Example 3



Notice, spline solution linear for first segment. For this function, this is probably not where we want it to be linear (and then quadratic for larger x).

Replace condition $c_0 = 0$ by $c_2 = 0$ and see what happens.

Piecewise quadratic Interpolation – Example 3

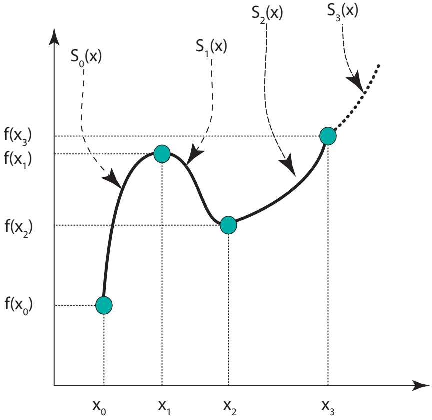


Notice, spline solution linear for first segment. For this function, this is probably not where we want it to be linear (and then quadratic for larger x).

Replace condition $c_0 = 0$ by $c_2 = 0$ and see what happens.

Spline Interpolation - Alternative I

Fit a quadratic spline through four data points



Spline Interpolation - Alternative II

Use three 'real' quadratic polynomial $S_0(x)$, $S_1(x)$, $S_2(x)$ and one 'imaginary' polynomial $S_3(x)$ as

$$S_0(x) = a_0 + b_0(x - x_0) + c_0(x - x_0)^2 \tag{33}$$

$$S_1(x) = a_1 + b_1(x - x_1) + c_1(x - x_1)^2 \tag{34}$$

$$S_2(x) = a_2 + b_2(x - x_2) + c_2(x - x_2)^2 \tag{35}$$

$$S_3(x) = a_3 + b_3(x - x_3) + c_3(x - x_3)^2 \tag{36}$$

In the first instance, we know that

$$a_i = f(x_i) \qquad \text{for } i = 0,\ 1,\ 2,\ 3\,. \tag{37}$$

Spline Interpolation - Alternative III

By requiring that adjacent $S_i(x)$ have common values at the interior points, we find that

$$b_0 = \frac{a_1 - a_0}{h_0} - c_0 h_0 \tag{38}$$

$$b_1 = \frac{a_2 - a_1}{h_1} - c_1 h_1 \tag{39}$$

$$b_2 = \frac{a_3 - a_2}{h_2} - c_2 h_2 \ , \tag{40}$$

where $h_i = x_{i+1} - x_i$.
By requiring that the derivatives of the adjacent $S_i(x)$ have common values at the interior points, we obtain

$$b_0 + 2c_0 h_0 = b_1 \tag{41}$$

Spline Interpolation - Alternative IV

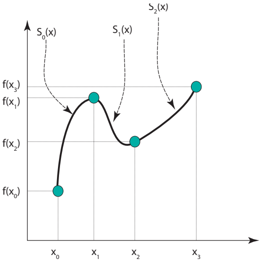
$$b_1 + 2c_1h_1 = b_2 \tag{42}$$

Using Eqs. (38) to (42) together with the assumption that $c_0 = 0$, we get c_i's by solving the following matrix equation

$$\begin{bmatrix} 1 & 0 & 0 \\ h_0 & h_1 & 0 \\ 0 & h_1 & h_2 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{a_2}{h_1} - a_1\left(\frac{1}{h_0} + \frac{1}{h_1}\right) + \frac{a_0}{h_0} \\ \frac{a_3}{h_2} - a_2\left(\frac{1}{h_1} + \frac{1}{h_2}\right) + \frac{a_1}{h_1} \end{bmatrix} \tag{43}$$

Thus, with c_0, c_1 and c_2 known (and $a_i = f(x_i)$), can use (38)-(40) to solve for b_0, b_1 and b_2.

Cubic Spline Interpolation I



Spline consists of a cubic polynomials:

- $S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3$
- $S(x) = S_i(x)$ for $x_i < x < x_{i+1}$.
- $S(x)$, it first and second derivatives $S'(x)$, $S''(x)$ are all continuous at $x = x_i$.
- For $n + 1$ data points, there are n number of defined $S_i(x)$.

Cubic Spline Interpolation II

Find a_i, b_i, c_i and d_i for $i = 0...n-1$ ($4n$ unknowns)

1. $S(x)$ have the values of f at $x = x_i$, hence,

$$S_i(x_i) = a_i = f(x_i) \quad i = 0, ..., n-1 \tag{44}$$

Equation (44) will give n conditions. In addition we know function value at the end of the last segment which gives another condition and we write $a_n = f(x_n)$. This notation will be handy later even though we do not have a spline S_n.

2. The function values at adjacent polynomials must be equal at the interior nodes. Defining $h_i = x_{i+1} - x_i$:

$$\begin{aligned} S_i(x_{i+1}) &= S_{i+1}(x_{i+1}) \\ a_i + b_i h_i + c_i h_i^2 + d_i h_i^3 &= a_{i+1} \end{aligned} \tag{45}$$

for $i = 0...n-2$. This gives us another $n-1$ conditions.

Cubic Spline Interpolation III

3. The derivative of $S_i(x)$ shall be continuous at interior nodes

$$S_i^{'}(x_{i+1}) = S_{i+1}^{'}(x_{i+1}) \tag{46}$$

which leads to

$$b_i + 2c_ih_i + 3d_ih_i^2 = b_{i+1} \tag{47}$$

for $i = 0 ... n-2$. Note that for most cubic splines, b_n does not exist. Equation (47) will give us another $n-1$ equations.

4. The second derivative of $S_i(x)$ needs to be continuous at interior nodes.

$$S_i^{''}(x_{i+1}) = S_{i+1}^{''}(x_{i+1}) \tag{48}$$

which gives

$$c_i + 3d_ih_i = c_{i+1} \tag{49}$$

where $i = 0 ... n-2$. This will give us another $n-1$ equations.

Cubic Spline Interpolation IV

The above conditions provide $4n-2$ equations. we still need two additional equations to solve the $4n$ number of unknowns.

- **clamped spline**: if more information about the function we are trying to approximate (e.g. its derivative at the end points) is known, it is used to get two equations
- **natural spline**: in the absence of any information, it is common to set the second derivative of the cubic spline to zero at the two end points. Thus:

$$c_0 = 0 \tag{50}$$

$$c_n = 0 \tag{51}$$

Cubic Spline Interpolation V

We are now ready to solve a_i, b_i, c_i and d_i. Re-arranging (49):

$$\boxed{d_i = \frac{c_{i+1} - c_i}{3h_i}} \tag{52}$$

Put Eq. (52) into Eq. (47) to get

$$b_{i+1} = b_i + (c_i + c_{i+1})\, h_i \tag{53}$$

Put Eq. (52) into Eq. (45) to get

$$\boxed{b_i = \frac{1}{h_i}(a_{i+1} - a_i) - \frac{h_i}{3}(2c_i + c_{i+1})} \tag{54}$$

Cubic Spline Interpolation VI

Substituting Eq. (54) into Eq. (53) and rearranging leads to the following relationship between a_i's and c_i's

$$
\begin{aligned}
h_{j-1}c_{j-1} + 2\left(h_j + h_{j-1}\right)c_j + h_j c_{j+1} = \\
\frac{3}{h_j}\left(a_{j+1} - a_j\right) + \frac{3}{h_{j-1}}\left(a_{j-1} - a_j\right)
\end{aligned} \tag{55}
$$

Equation (55) is a tridiagonal system and can be solved if we assume that $c_0 = c_n = 0$. The linear system of equation equation that we need to solve looks like the following

$$
[A]\{X\} = \{C\} \tag{56}
$$

where

Cubic Spline Interpolation VII

$$
\underbrace{\begin{bmatrix}
1 & 0 & 0 & 0 & \cdots & 0 \\
h_0 & 2\left(h_0+h_1\right) & h_1 & 0 & \cdots & 0 \\
0 & h_1 & 2\left(h_1+h_2\right) & h_2 & \ddots & 0 \\
0 & 0 & \ddots & \ddots & \ddots & \vdots \\
0 & 0 & \cdots & h_{n-2} & 2\left(h_{n-2}+h_{n-1}\right) & h_{n-1} \\
0 & 0 & 0 & \cdots & 0 & 1
\end{bmatrix}}_{A}
\underbrace{\begin{Bmatrix} c_0 \\ c_1 \\ \vdots \\ \vdots \\ \vdots \\ c_n \end{Bmatrix}}_{X}
$$

$$
= \underbrace{\begin{Bmatrix}
0 \\
\frac{3}{h_1}\left(a_2-a_1\right)+\frac{3}{h_0}\left(a_0-a_1\right) \\
\frac{3}{h_2}\left(a_3-a_2\right)+\frac{3}{h_1}\left(a_1-a_2\right) \\
\vdots \\
\frac{3}{h_{n-1}}\left(a_n-a_{n-1}\right)+\frac{3}{h_{n-2}}\left(a_{n-2}-a_{n-1}\right) \\
0
\end{Bmatrix}}_{C}
$$

Cubic Spline Interpolation VIII

In summary, to construct cubic splines, do the following

1. Make use of Eq. (44) and set $a_i = f(x_i)$.
2. Solve Eq. (55) to obtain all other values of c_i's. Note that by solving Eq. (55) we are assuming that $c_0 = 0$ and $c_n = 0$. This is equivalent to saying that the second derivative at x_0 and x_n are zero !
3. Use Eq. (54) to obtain the b_i's.
4. Use Eq. (52) to obtain the d_i's.

Subsequently, we can evaluate the function values at the any point x from the formula $S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3$. But we have to identify first which set of coefficients we need to use. That is, we have to find the interval $[x_i, x_{i+1}]$ that encompasses point x.

Next week

Differentiation/Integration

POSTERA CRESCAM LAUDE

THE UNIVERSITY OF

MELBOURNE