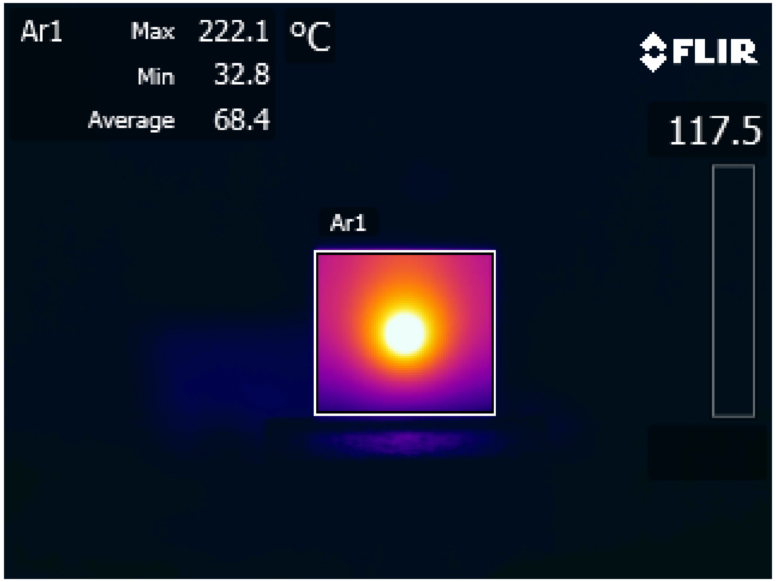
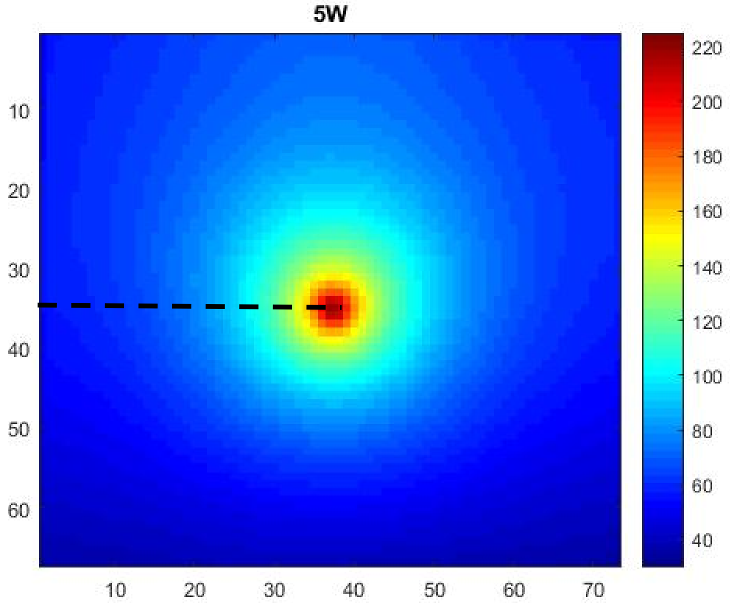
實例：雷射加熱與加工

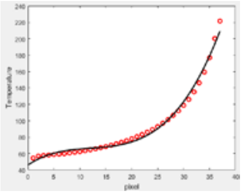
- Laser heating (5W) on an aluminum metal plate
- Obtain the centerline temperature profile using curve fitting
- Using the **2nd** and **6th** order polynomial





實例：雷射加熱與加工

能源與未來生活科技一:熱科學



機械製造

熱處理
金屬/雷射加工

串接實驗與模擬

熱學原理
系統觀念

數值分析

矩陣運算
曲線擬合
內插與外插

實務專題

發現問題
定義需求
產生解決方案

熱力學

熱力循環
相平衡
能量傳遞
熱力學定律

理論分析

問題解析
模擬與分析

工廠實習

雷射加工

大學部專題

實務問題與產品

理論與實務結合

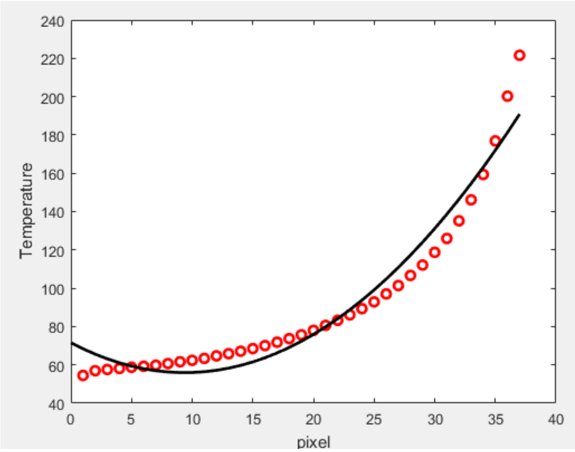
熱傳學

實例：雷射加熱與加工

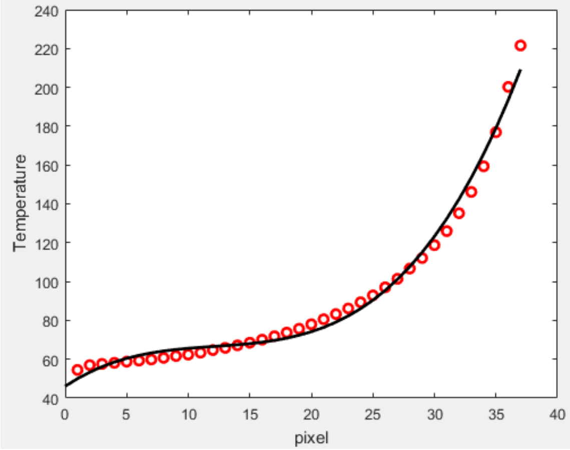
- 該數值分析程式需要能調整多項式次數n (degree)

$$f(x)=a_n x^n + a_{n-1}x^{n-1} + \ldots + a_1 x + a_0$$

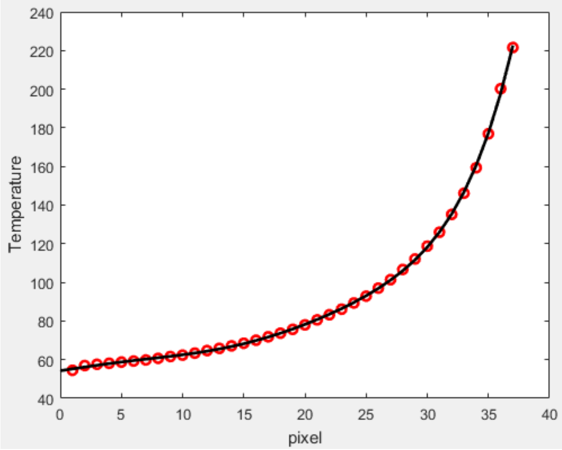
- 流程:
  - 將溫度分佈(矩陣)中的中心線溫度取出，並存成陣列
  - 使用多項式回歸(Polynomial regression)方式取得擬合曲線
  - 將原始數據與擬合曲線畫在同一張圖上



2nd order



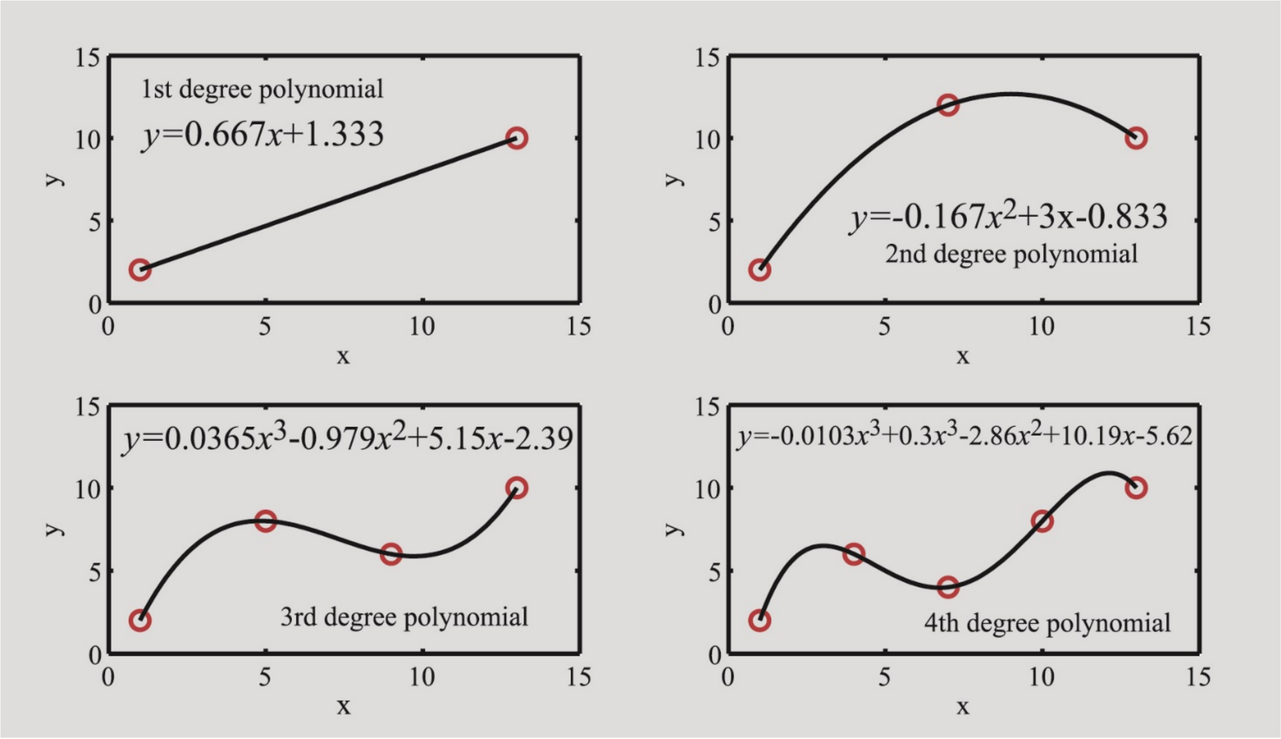
3rd order



6th order

4.5 Interpolation using a Single Polynomial

- Determine a mathematical formula such that (1) it gives the exact value at all the data points (2) an estimate value **between** the points
- For n data points, the polynomial is of order (n-1)
- By solving the system of linear equations, the coefficients can be determined.



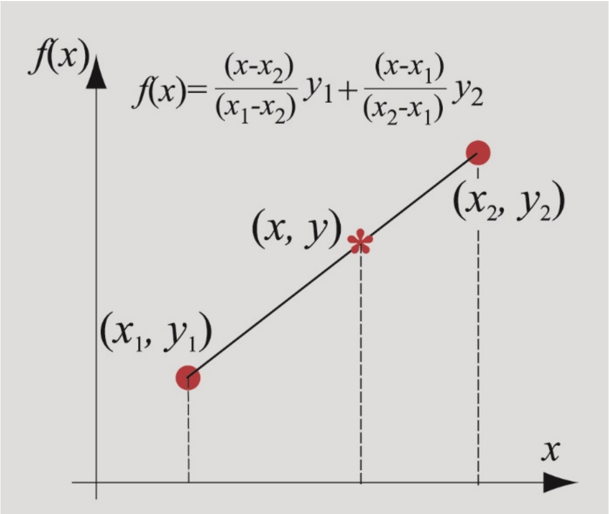
Lagrange Interpolating Polynomials

- 1^{st} order Lagrange polynomial:

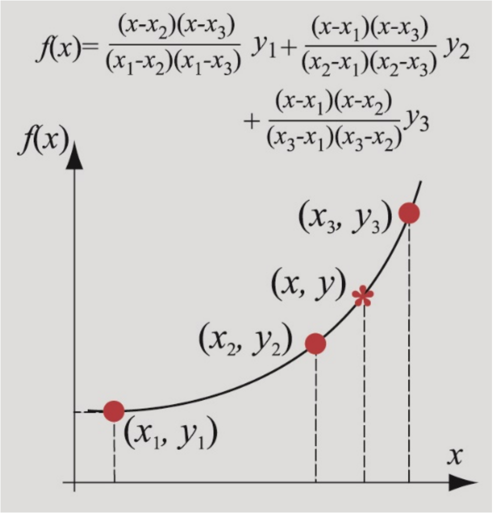
$$f(x) = \frac{(x - x_2)}{(x_1 - x_2)} y_1 + \frac{(x - x_1)}{(x_2 - x_1)} y_2$$

- 2^{nd} order Lagrange polynomial:

$$f(x) = \frac{(x - x_2)(x - x_3)}{(x_1 - x_2)(x_1 - x_3)} y_1 + \frac{(x - x_1)(x - x_3)}{(x_2 - x_1)(x_2 - x_3)} y_2 + \frac{(x - x_1)(x - x_2)}{(x_3 - x_1)(x_3 - x_2)} y_3$$



1st order Lagrange polynomial



2nd order Lagrange polynomial

Lagrange Interpolating Polynomials

- The n[th] order Lagrange polynomial:

$$f(x)=\frac{(x-x_2)(x-x_3)\ldots(x-x_n)}{(x_1-x_2)(x_1-x_3)\ldots(x_1-x_n)}y_1+\frac{(x-x_1)(x-x_3)\ldots(x-x_n)}{(x_2-x_1)(x_2-x_3)\ldots(x_2-x_n)}y_2+\ldots+\frac{(x-x_1)(x-x_2)\ldots(x-x_{n-1})}{(x_n-x_1)(x_n-x_2)\ldots(x_n-x_{n-1})}y_n$$

- The equation can be rewritten as:

$$f(x)=\sum_{i=1}^{n}y_iL_i(x)=\sum_{i=1}^{n}y_i\underbrace{\prod_{\substack{j=1\\j\neq i}}^{n}\frac{(x-x_j)}{(x_i-x_j)}}_{\text{Lagrange function}}$$

- Additional Note:
  - The spacing between the data points does not have to be equal
  - The interpolation polynomial has to be calculated for every x
  - The Lagrange polynomial has to be *calculated again* if additional points are included

Example 4-4: Lagrange Interpolating Polynomial

- For the following five data points:

x	1	2	4	5	7
y	52	5	-5	-40	10

(a) Determine the fourth-order polynomial in Lagrange form

(b) Use the polynomial obtained in (a) to determine the interpolated value for x=3

(*a*) Following the form of Eq. (4.44), the Lagrange polynomial for the five given points is:

$$f(x)=\frac{(x-2)(x-4)(x-5)(x-7)}{(1-2)(1-4)(1-5)(1-7)}52+\frac{(x-1)(x-4)(x-5)(x-7)}{(2-1)(2-4)(2-5)(2-7)}5+\frac{(x-1)(x-2)(x-5)(x-7)}{(4-1)(4-2)(4-5)(4-7)}(-5)+$$
$$\frac{(x-1)(x-2)(x-4)(x-7)}{(5-1)(5-2)(5-4)(5-7)}(-40)+\frac{(x-1)(x-2)(x-4)(x-5)}{(7-1)(7-2)(7-4)(7-5)}10$$

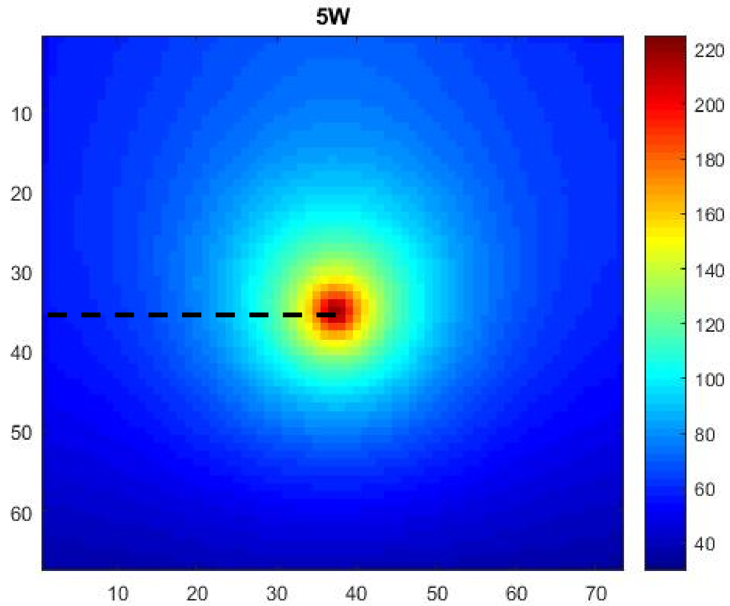
(*b*) The interpolated value for $x=3$ is obtained by substituting the x in the polynomial:

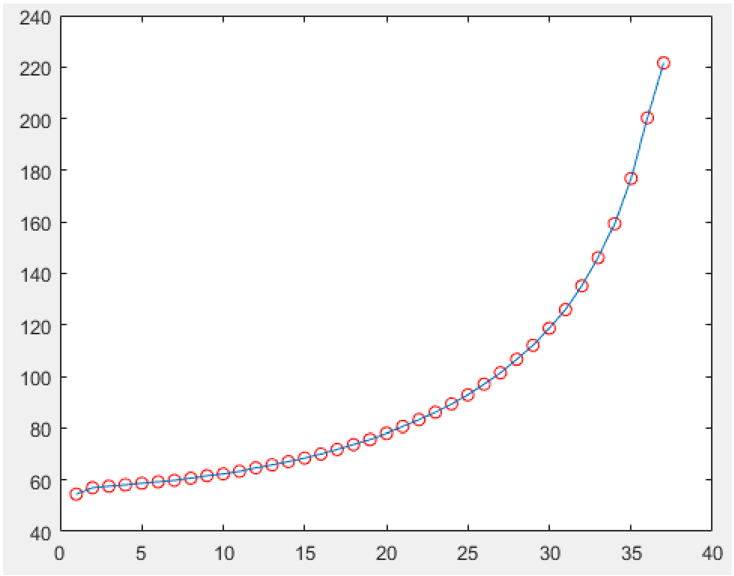
$$f(3)=\frac{(3-2)(3-4)(3-5)(3-7)}{(1-2)(1-4)(1-5)(1-7)}52+\frac{(3-1)(3-4)(3-5)(3-7)}{(2-1)(2-4)(2-5)(2-7)}5+\frac{(3-1)(3-2)(3-5)(3-7)}{(4-1)(4-2)(4-5)(4-7)}(-5)+$$
$$\frac{(3-1)(3-2)(3-4)(3-7)}{(5-1)(5-2)(5-4)(5-7)}(-40)+\frac{(3-1)(3-2)(3-4)(3-5)}{(7-1)(7-2)(7-4)(7-5)}10$$

$$f(3)=-5.778+2.667-4.444+13.333+0.222=6$$

實例：雷射加熱與加工

- 使用Lagrange多項式進行中心線溫度分佈的內差
- 將溫度分佈(矩陣)中的中心線溫度取出，並存成陣列
- 將原始數據與內差曲線畫在同一張圖上
- 數據點之間距並不需要相同





Newton's Interpolating Polynomials

- General form of n-1 order Newton's polynomial:

$$f(x) = a_1 + a_2(x - x_1) + a_3(x - x_1)(x - x_2) + \ldots + a_n(x - x_1)(x - x_2)\ldots(x - x_{n-1})$$

- The coefficients (a_1 through a_n) can be determined
- Newton's polynomials have the features:
  - The data points do not have to be in any descending or ascending order
  - Once the coefficients are determined, they can be used for interpolation at an point between the data points
  - After the coefficients are determined, **additional data points** can be added, and only the **additional coefficients** have to be determined

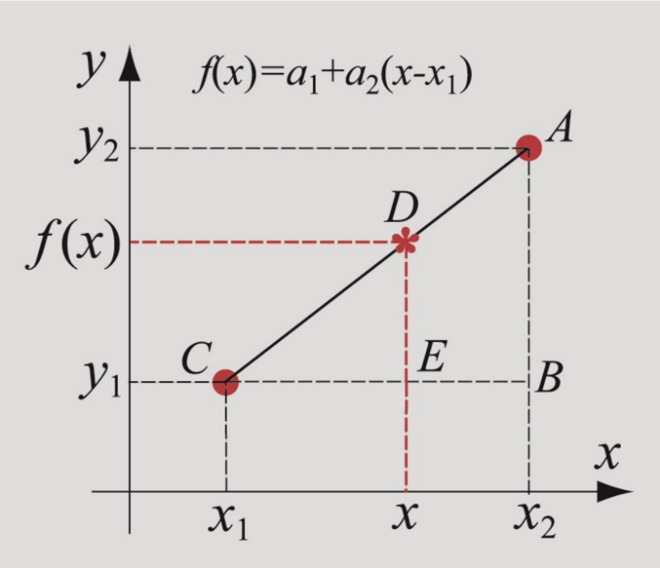
First Order Newton's Polynomial

- For two given points (x_1, y_1) and (x_2, y_2), the first order Newton's polynomial:

$$f(x) = a_1 + a_2(x - x_1)$$

- The coefficients (a_1, a_2) can be solved:

$$a_1 = y_1 \qquad a_2 = \frac{y_2 - y_1}{x_2 - x_1}$$



Second Order Newton's Polynomial

- For three given points (x_1, y_1) (x_2, y_2) and (x_3, y_3), the second order Newton's polynomial:

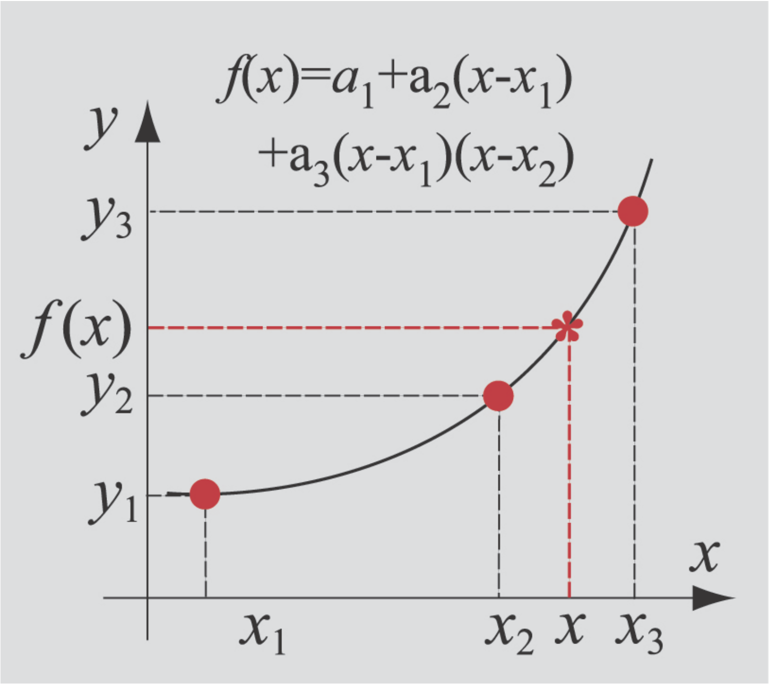
$$f(x) = a_1 + a_2(x - x_1) + a_3(x - x_1)(x - x_2)$$

- The coefficients $(a_1\ a_2\ a_3)$ can be solved:

$$a_1 = y_1$$

$$a_2 = \frac{y_2 - y_1}{x_2 - x_1}$$

$$a_3 = \frac{\dfrac{y_3 - y_2}{x_3 - x_2} - \dfrac{y_2 - y_1}{x_2 - x_1}}{x_3 - x_1}$$



A General Form of Newton’s Polynomial

- The Newton’s polynomial coefficient can be clarified by defining the **divided difference**: $f[x_2, x_1]$

$$f[x_2, x_1] = \frac{y_2 - y_1}{x_2 - x_1} = a_2$$

$$f[x_3, x_2, x_1] = \frac{f[x_3, x_2] - f[x_2, x_1]}{x_3 - x_1} = \frac{\frac{y_3 - y_2}{x_3 - x_2} - \frac{y_2 - y_1}{x_2 - x_1}}{x_3 - x_1} = a_3$$

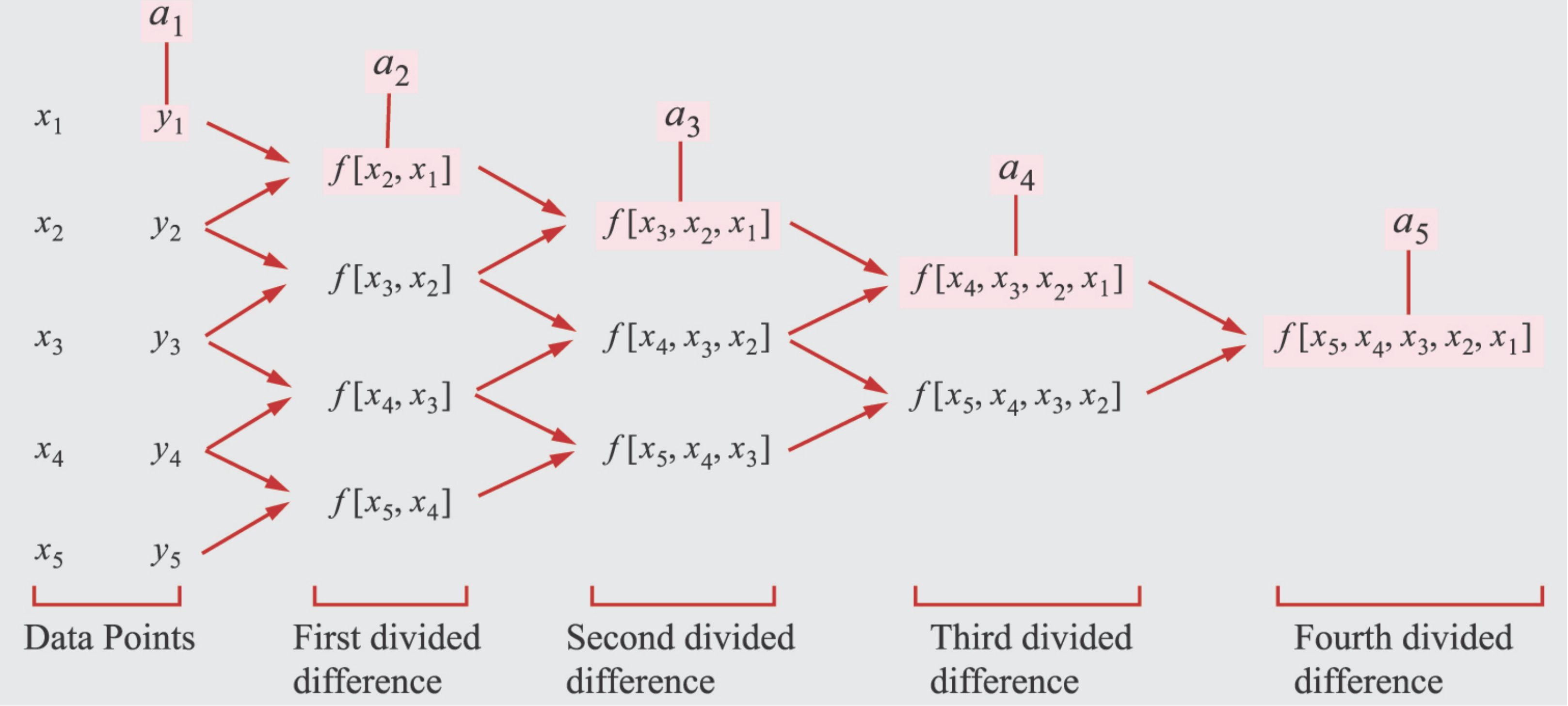
$$f[x_4, x_3, x_2, x_1] = \frac{f[x_4, x_3, x_2] - f[x_3, x_2, x_1]}{x_4 - x_1}$$

$$= \frac{\frac{f[x_4, x_3] - f[x_3, x_2]}{x_4 - x_2} - \frac{f[x_3, x_2] - f[x_2, x_1]}{x_3 - x_1}}{x_4 - x_1} = a_4$$

$$f[x_5, x_4, x_3, x_2, x_1] = \frac{f[x_5, x_4, x_3, x_2] - f[x_4, x_3, x_2, x_1]}{x_5 - x_1} = a_5$$

Divided Difference

- Start by calculating (n-1) first divided differences
- Then calculate (n-2) second divided differences
- Continue for third, fourth… divided differences
- Until one nth divided difference is calculated to give a_n



General Terms for Divided Differences

- In general terms, for n given data pints, the first divided differences between two points (x_i, y_i) and (x_j, y_j) are:

$$f\left[x_j, x_i\right] = \frac{y_j - y_i}{x_j - x_i}$$

- The kth divided difference for second and higher divided differences up to to (n-1) difference is given by:

$$f\left[x_k, x_{k-1}, \ldots, x_2, x_1\right] = \frac{f\left[x_k, x_{k-1}, \ldots, x_3, x_2\right] - f\left[x_{k-1}, x_{k-2}, \ldots, x_2, x_1\right]}{x_k - x_1}$$

- The (n-1) order Newton's polynomial is given by:

$$f(x) = y = \underbrace{y_1}_{a_1} + \underbrace{f[x_2, x_1]}_{a_2}(x - x_1) + \underbrace{f[x_3, x_2, x_1]}_{a_3}(x - x_1)(x - x_2) + \ldots + \underbrace{f[x_n, x_{n-1}, \ldots, x_2, x_1]}_{a_n}(x - x_1)(x - x_2)\ldots(x - x_{n-1})$$

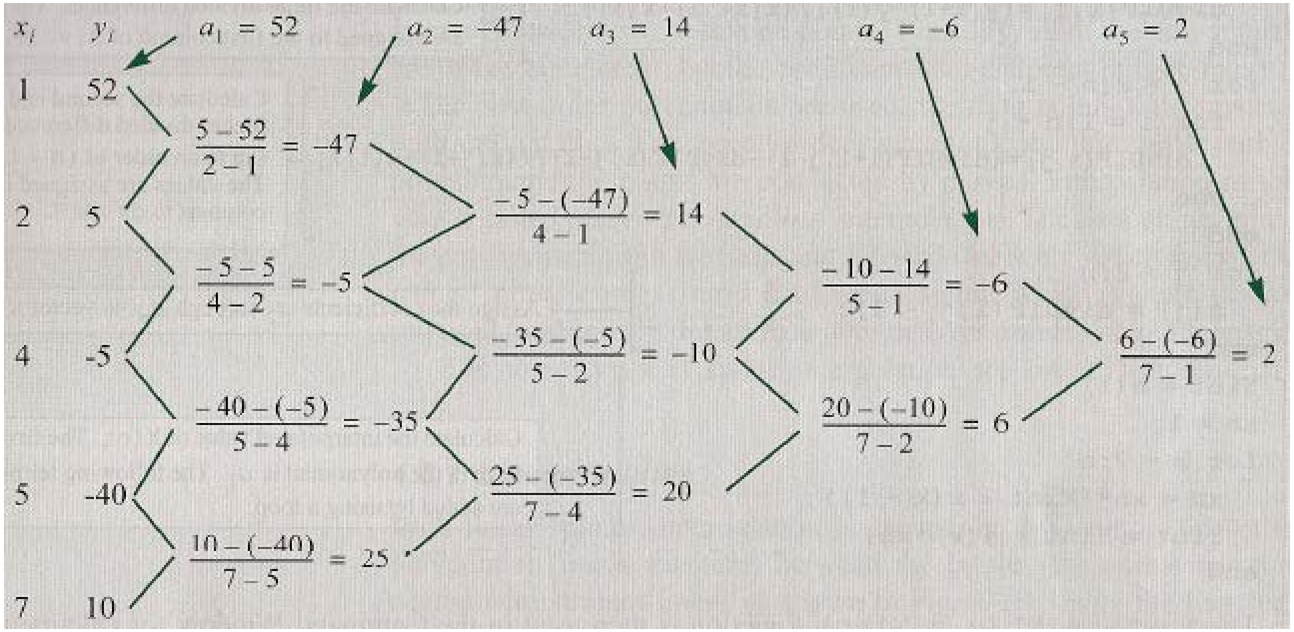
Example 4-5: Newton’s Interpolating Polynomial

- For the set of the following five data points:

x	1	2	4	5	7
y	52	5	-5	-40	10

(a) Determine the fourth-order polynomial in Newton’s form that passes through the points

(b) Use the polynomial obtained in (a) to determine the interpolated value for x=3



4.6 Piecewise (Spline) Interpolation

- Since large error might occur when a high order polynomial is used for interpolation involving a large number of data points
- A better interpolation can be obtained by using **many low-order polynomials** instead of a single high-order polynomial
- Each low-order polynomial is valid in one interval between two or several points
- Typically, all the polynomials are of the same order but with *different coefficient* in each interval
  - When first-order polynomials are used, the data points are connected with straight lines
  - For second order (quadratic) or third order (cubic) polynomials, the points are connected by curves
- Interpolation in this way is called ***piecewise***, or ***splin***e, interpolation
- The data points where the polynomials from two adjacent intervals meet are called **knots**

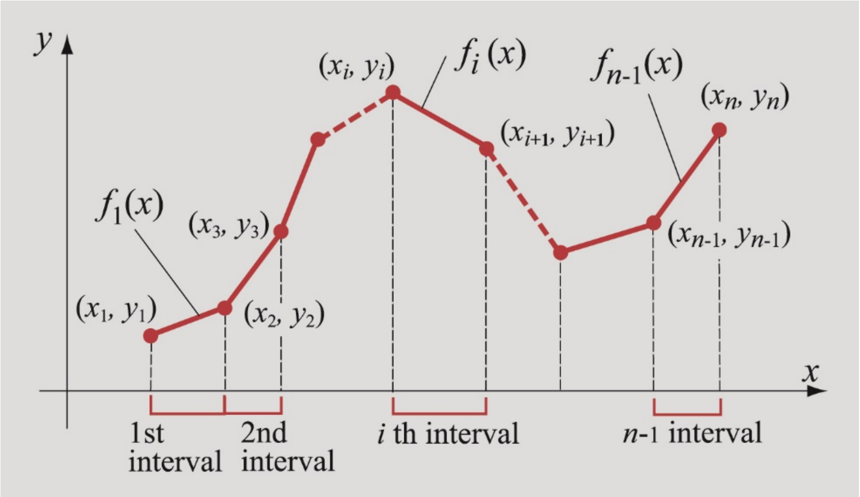
Linear splines

- Using a first order polynomial (linear function) between the points
- Using the Lagrange form, the equation is:

$$f_1(x) = \frac{(x - x_2)}{(x_1 - x_2)} y_1 + \frac{(x - x_1)}{(x_2 - x_1)} y_2$$

- For n given points, there are n-1 intervals. The general form for the straight line connecting (x_i, y_i) and (x_{i+1}, y_{i+1}) :

$$f_i(x) = \frac{(x - x_{i+1})}{(x_i - x_{i+1})} y_i + \frac{(x - x_i)}{(x_{i+1} - x_i)} y_{i+1}$$



Example 4-6: Linear Splines

- The set of the following four data points are:

x	8	11	15	18
y	5	9	10	8

(a) Determine the linear spline that fit the data

(b) Determine the interpolated value for x=12.7

(*a*) There are four points and thus three splines. Using Eq. (4.65) the equations of the splines are:

$$f_1(x) = \frac{(x-x_2)}{(x_1-x_2)}y_1 + \frac{(x-x_1)}{(x_2-x_1)}y_2 = \frac{(x-11)}{(8-11)}5 + \frac{(x-8)}{(11-8)}9 = \frac{5}{-3}(x-11) + \frac{9}{2}(x-8) \quad \text{for} \quad 8 \le x \le 11$$

$$f_2(x) = \frac{(x-x_3)}{(x_2-x_3)}y_2 + \frac{(x-x_2)}{(x_3-x_2)}y_3 = \frac{(x-15)}{(11-15)}9 + \frac{(x-11)}{(15-11)}10 = \frac{9}{-4}(x-15) + \frac{10}{4}(x-11) \quad \text{for} \quad 11 \le x \le 15$$

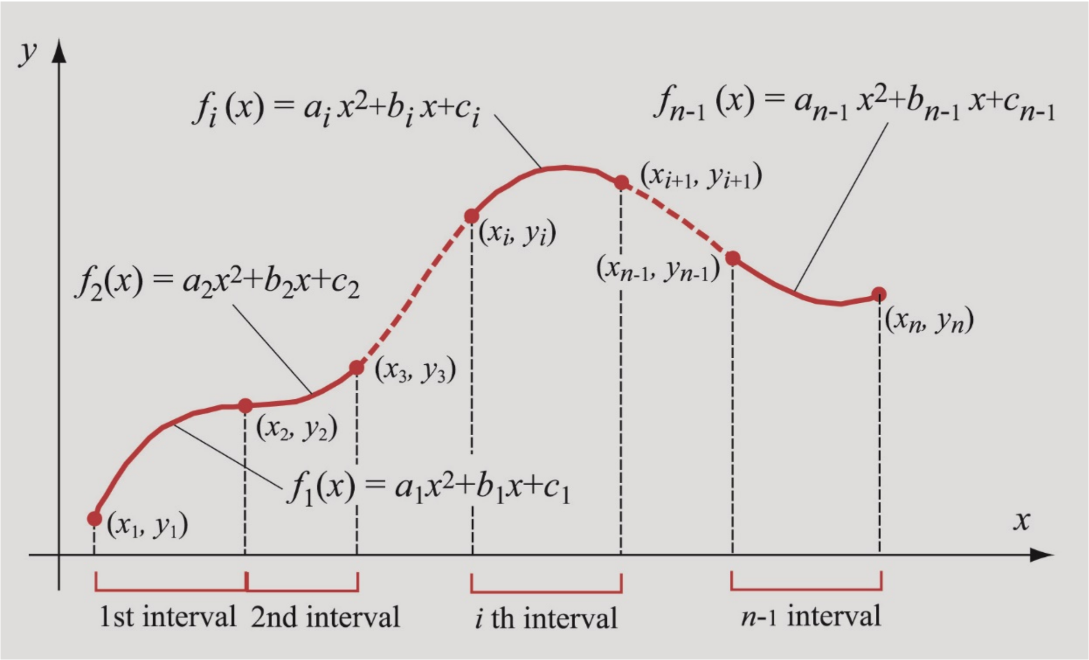
$$f_3(x) = \frac{(x-x_4)}{(x_3-x_4)}y_3 + \frac{(x-x_3)}{(x_4-x_3)}y_4 = \frac{(x-18)}{(15-18)}10 + \frac{(x-15)}{(18-15)}8 = \frac{10}{-3}(x-18) + \frac{8}{3}(x-15) \quad \text{for} \quad 15 \le x \le 18$$

Quadratic Splines

- Second order polynomials are used for interpolation
- For *n* given points, there are *n-1* intervals

$$f_i(x) = a_i x^2 + b_i x + c_i$$

- There are *n-1* equations and each equation has *three* coefficients. Therefore, a total of *3*(n-1)* coefficients have to be determined



Quadratic Splines

- Procedure:

(1) Each polynomial $f_i(x)$ must pass through the end points of the interval (x_i, y_i) and (x_{i+1}, y_{i+1})

2n-2 equations

$$f_i(x_i) = y_i \qquad f_i(x_{i+1}) = y_{i+1}$$

(2) At the interior knots, the slope (first derivative) of the polynomials from the adjacent intervals are equal

n-2 equations

$$f'(x) = \frac{df}{dx} = 2a_i x + b_i \quad \Rightarrow \quad 2a_{i-1}x + b_{i-1} = 2a_i x + b_i$$

(3) Second derivative at either the first point or the last point is zero

1 equation

$$f_1''(x) = 2a_1 = 0$$

Example 4-7: Quadratic Splines

- The set of the following five data points:

x	8	11	15	18	22
y	5	9	10	8	7

(a) Determine the quadratic splines that fit the data
(b) Determine the interpolated value of y for x = 12.7
(c) Make a plot of the data points and the interpolating polynomials

- Procedure 1: eight equations are obtained

$$i = 1 \quad f_1(x) = a_1x_1^2 + b_1x_1 + c_1 = b_1\,8 + c_1 = 5$$
$$f_1(x) = a_1x_2^2 + b_1x_2 + c_1 = b_1\,11 + c_1 = 9$$
$$i = 2 \quad f_2(x) = a_2x_2^2 + b_2x_2 + c_2 = a_2\,11^2 + b_2\,11 + c_2 = 9$$
$$f_2(x) = a_2x_3^2 + b_2x_3 + c_2 = a_2\,15^2 + b_2\,15 + c_2 = 10$$
$$i = 3 \quad f_3(x) = a_3x_3^2 + b_3x_3 + c_3 = a_3\,15^2 + b_3\,15 + c_3 = 10$$
$$f_3(x) = a_3x_4^2 + b_3x_4 + c_3 = a_3\,18^2 + b_3\,18 + c_3 = 8$$
$$i = 4 \quad f_4(x) = a_4x_4^2 + b_4x_4 + c_4 = a_4\,18^2 + b_4\,18 + c_4 = 8$$
$$f_4(x) = a_4x_5^2 + b_4x_5 + c_4 = a_4\,22^2 + b_4\,22 + c_4 = 7$$

Example 4-7: Quadratic Splines

- Procedure 2: three equations are obtained

$$i=2\quad 2a_1x_2+b_1=2a_2x_2+b_2 \longrightarrow b_1=2a_2 11+b_2 \quad \text{or:} \quad b_1-2a_2 11-b_2=0$$

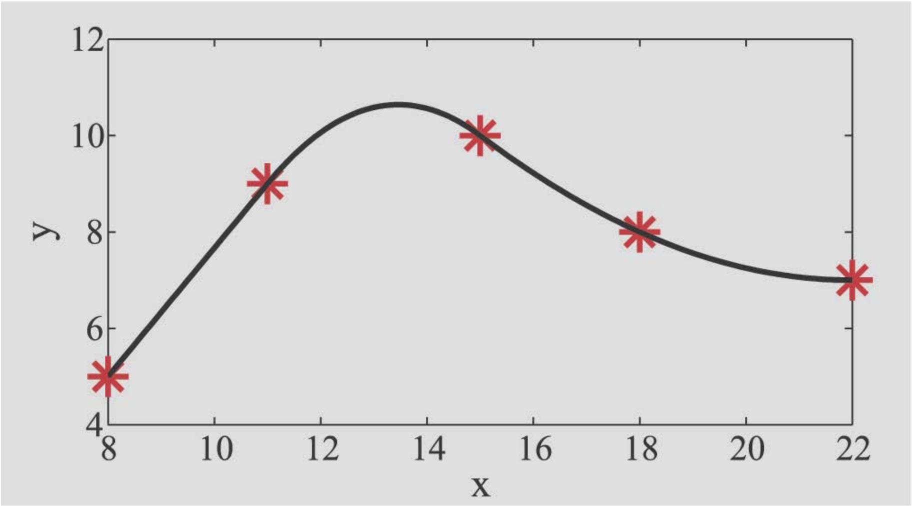
$$i=3\quad 2a_2x_3+b_2=2a_3x_3+b_3 \longrightarrow 2a_2 15+b_2=2a_3 15+b_3 \quad \text{or:} \quad 2a_2 15+b_2-2a_3 15-b_3=0$$

$$i=4\quad 2a_3x_4+b_3=2a_4x_4+b_4 \longrightarrow 2a_3 18+b_3=2a_4 18+b_4 \quad \text{or:} \quad 2a_3 18+b_3-2a_4 18-b_4=0$$

- Procedure 3: set $a_1 = 0$

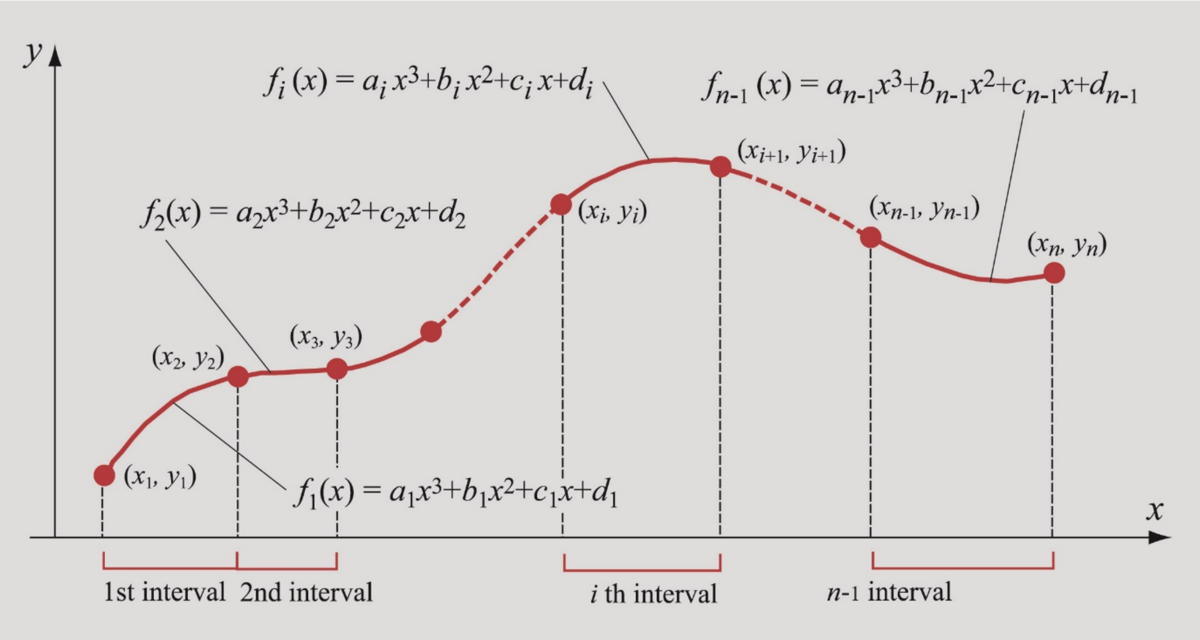
- Solve the 11 linear equations

$$\begin{bmatrix} 8 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 11 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 11^2 & 11 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 15^2 & 15 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 15^2 & 15 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 18^2 & 18 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 18^2 & 18 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 22^2 & 22 & 1 \\ 1 & 0 & -22 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 1 & 0 & -30 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 36 & 1 & 0 & -36 & -1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ c_1 \\ a_2 \\ b_2 \\ c_2 \\ a_3 \\ b_3 \\ c_3 \\ a_4 \\ b_4 \\ c_4 \end{bmatrix} = \begin{bmatrix} 5 \\ 9 \\ 9 \\ 10 \\ 10 \\ 8 \\ 8 \\ 7 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$



Cubic Splines

- Third-order polynomials are used for interpolation
- The determination of all the coefficients may require a large number of calculations
- Two derivations of cubic splines:
  - Standard form of polynomials
  - A variation of the Lagrange form



Cubic Splines: Standard Form Polynomial (natural cubic spline)

- Procedure: $\boxed{f_i(x) = a_i x^3 + b_i x^2 + c_i x + d_i}$

(1) Each polynomial $f_i(x)$ must pass through the end points of the interval (x_i, y_i) and (x_{i+1}, y_{i+1})

$$\boxed{f_i(x_i) = y_i} \qquad \boxed{f_i(x_{i+1}) = y_{i+1}}$$

(2) At the interior knots, the slope (first derivative) of the polynomials from the adjacent intervals are equal

$$f'(x) = \frac{df}{dx} = 3a_i x^2 + 2b_i x + c_i$$

$$3a_{i-1}x_i^{\,2} + 2b_{i-1}x_i + c_{i-1} = 3a_i x_i^{\,2} + 2b_i x_i + c_i$$

(3) Second derivative at the interior knots from adjacent intervals are equal

$$6a_{i-1}x_i + 2b_{i-1} = 6a_i x_i + 2b_i$$

(4) The 2nd derivative is zero at the first and the last point

$$6a_1 x_1 + 2b_1 = 0 \qquad 6a_{n-1}x_n + 2b_{n-1} = 0$$

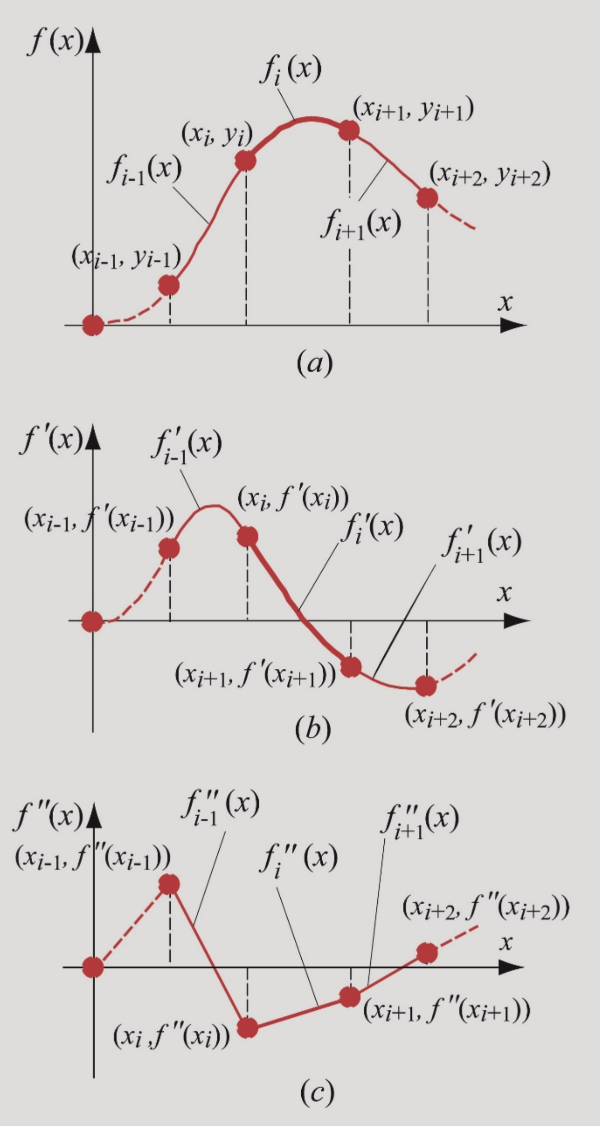
Cubic Splines: Lagrange Form Polynomial

- Lagrange form:

$$f''(x) = \frac{x - x_{i+1}}{x_i - x_{i+1}} f_i''(x_i) + \frac{x - x_i}{x_{i+1} - x_i} f_i''(x_{i+1})$$

- The second derivate is a linear function of x
- The third-order polynomial can be determined by integrating the above equation twice
- The two constants in the integration can be solved by:

$$f_i(x_i) = y_i \qquad f_i(x_{i+1}) = y_{i+1}$$



Cubic Splines: Lagrange Form Polynomial

- The equation of the third order polynomial is:

$$f_i(x) = \frac{f_i''(x_i)}{6(x_{i+1}-x_i)}(x_{i+1}-x)^3 + \frac{f_i''(x_{i+1})}{6(x_{i+1}-x_i)}(x-x_i)^3$$
$$+\left[\frac{y_i}{x_{i+1}-x_i} - \frac{f_i''(x_i)(x_{i+1}-x_i)}{6}\right](x_{i+1}-x)$$
$$+\left[\frac{y_{i+1}}{x_{i+1}-x_i} - \frac{f_i''(x_{i+1})(x_{i+1}-x_i)}{6}\right](x-x_i)$$

- Each interval contains two unknowns
- It can be solved by the continuity of the first derivatives of polynomials from adjacent intervals:

$$f_i'(x_{i+1}) = f_{i+1}'(x_{i+1})$$

- Therefore, the equation becomes:

$$(x_{i+1}-x_i)\,f''(x) + 2(x_{i+2}-x_i)\,f''(x_{i+1}) + (x_{i+2}-x_{i+1})\,f''(x_{i+2})$$
$$= 6\left[\frac{y_{i+2}-y_{i+1}}{x_{i+2}-x_{i+1}} - \frac{y_{i+1}-y_i}{x_{i+1}-x_i}\right]$$

4.7 MATLAB Built-in Functions

- The *polyfit* command: for curve fitting a given set of n points
  - **p = polyfit(x, y, m)**
  - p: vector of coefficients of the polynomial that best fits the data
  - M: degree of the polynomial

- The *interp1* command: one dimensional interpolation at one point
  - **yi = interp1(x, y, xi, ‘method’)**
  - yi: interpolated value
  - Method: method of interpolation (*nearest, linear, spline, pchip*), type as a string (optional)
  - The vector x must be monotonic (ascending/descending order)

4.7 MATLAB Built-in Functions

```
>> x = 0:0.4:6;
>> y = [0 3 4.5 5.8 5.9 5.8 6.2 7.4 9.6 15.6 20.7 26.7 31.1
35.6 39.3 41.5];
>> p = polyfit(x,y,4)
p =
   -0.2644   3.1185 -10.1927  12.8780  -0.2746
```

The polynomial that corresponds to these coefficients is:
$f(x)=(-0.2644)x^4+3.1185x^3-10.1927x^2+12.878x-0.2746$.

```
>> x = [8 11 15 18 22];
>> y = [5 9 10 8 7];
>> xint=8:0.1:22;
>> yint=interp1(x,y,xint,'pchip');
>> plot(x,y,'*',xint,yint)
```

Assign the data points to x and y.

Vector with points for interpolation.

Calculate the interpolated values.

Create a plot with the data points and interpolated values.

4.8 Curve Fitting with a Linear Combination of Nonlinear Functions

- Curve fitting with a linear combination of m nonlinear functions:

$$F(x) = C_1 f_1(x) + C_2 f_2(x) + \ldots + C_m f_m(x) = \sum_{j=1}^{m} C_j f_j(x)$$

- The coefficients (C_1, C_2,…) can be determined by minimizing the total error:

$$E = \sum_{i=1}^{n} \left[y_i - \sum_{j=1}^{m} C_j f_j(x_i) \right]^2 \qquad \frac{\partial E}{\partial C_k} = 0$$

- Solve the equation to get:

$$\boxed{\sum_{i=1}^{n} \sum_{j=1}^{m} C_j f_j(x_i) f_k(x_i) = \sum_{i=1}^{n} y_i f_k(x_i)}$$