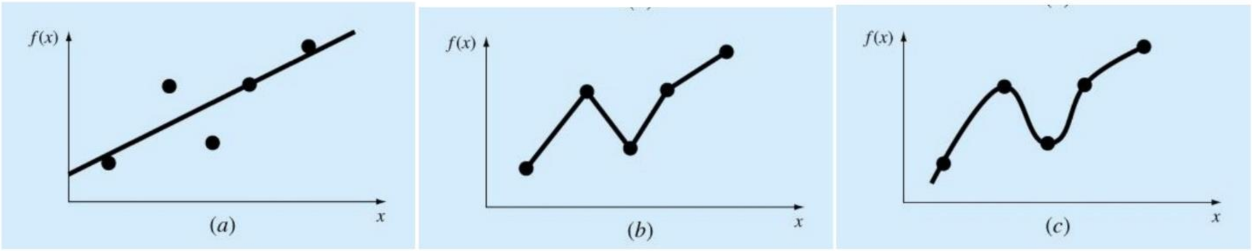
KU KONKUK UNIVERSITY

Numerical Analysis: Curve-Fitting Techniques
Interpolation (보간법)

Lecture Note for Numerical Analysis (9) Interpolation

1. Regression and Interpolation (Curve Fitting)

- Given n data points : $\{(x_0, y_0), (x_1, y_1), (x_2, y_2), (x_3, y_3), \cdots, (x_j, y_j), \cdots, (x_n, y_n)\}$
- Regression: find a curve fitting best to the points $(x_j, y_j), \quad j = 1, 2, \cdots, n$
- Interpolation: find a curve fitting best to and passing the points $(x_j, y_j), \quad j = 1, 2, \cdots, n$



(a) linear regression (b) linear interpolation (c) nonlinear interpolation

2. Basic concept of the polynomial interpolation

- General form

$$f(x;\mathbf{a}) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 \cdots + a_{m-1} x^{m-1} + a_m x^m \quad \mathbf{a} = [a_0, a_1, \cdots, a_{m-1}, a_m]$$

There should be (m+1)-independent points to determine coefficients $\mathbf{a} = [a_0, a_1, \cdots, a_{m-1}, a_m]$ such as

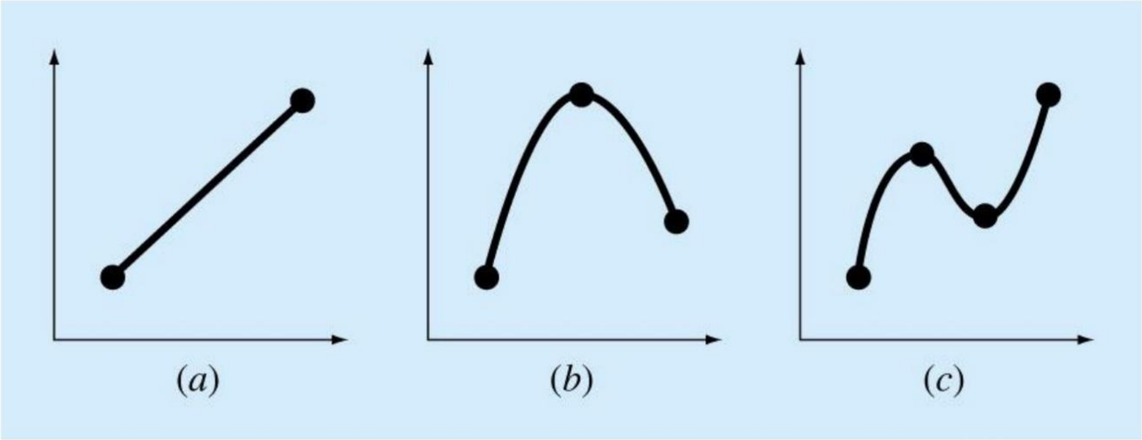
$\{(x_0, y_0), (x_1, y_1), (x_2, y_2), (x_3, y_3), \cdots, (x_j, y_j), \cdots, (x_m, y_m)\}$

1

Curve-Fitting Techniques: Interpolation

$$y_0 = f(x_0, \mathbf{a})$$
$$y_1 = f(x_1, \mathbf{a})$$
$$y_2 = f(x_2, \mathbf{a})$$
$$\vdots$$
$$y_m = f(x_m, \mathbf{a})$$

- Various polynomial interpolations



(a) Linear interpolation(m=1) (b) quadratic interpolation(m=2) (c) cubic interpolation(m=3)

$f(x;\mathbf{a}) = a_0 + a_1 x$ $\quad f(x;\mathbf{a}) = a_0 + a_1 x + a_2 x^2$ $\quad f(x;\mathbf{a}) = a_0 + a_1 x + a_2 x^2 + a_3 x^3$

2

3. Newton's Interpolating Polynomial

(3-1) General form

Data :

$$\{(x_0, y_0), (x_1, y_1), (x_2, y_2), (x_3, y_3), \cdots, (x_j, y_j), \cdots, (x_m, y_m)\} \tag{1}$$

Basic form of the Newton's Interpolating Polynomial

$$\begin{aligned} f(x; \mathbf{a}) &= a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + a_3(x - x_0)(x - x_1)(x - x_2) \\ &\quad + \cdots + a_m(x - x_0)(x - x_1)(x - x_2)\cdots(x - x_{m-2})(x - x_{m-1}) \end{aligned} \tag{2}$$

$$\mathbf{a} = [a_0, a_1, \cdots, a_{m-1}, a_m]$$

(3-2) Computation of the polynomial coefficients of the Newton's Interpolating

With the data $\{(x_0, y_0), (x_1, y_1), (x_2, y_2), (x_3, y_3), \cdots, (x_j, y_j), \cdots, (x_m, y_m)\}$

$$\boxed{\begin{aligned} f(x_0; \mathbf{a}) &= y_0 = a_0 \\ f(x_1; \mathbf{a}) &= y_1 = a_0 + a_1(x_1 - x_0) \\ f(x_2; \mathbf{a}) &= y_2 = a_0 + a_1(x_2 - x_0) + a_2(x_2 - x_0)(x_2 - x_1) \\ f(x_3; \mathbf{a}) &= y_3 = a_0 + a_1(x_3 - x_0) + a_2(x_3 - x_0)(x_3 - x_1) \\ &\quad + a_3(x_3 - x_0)(x_3 - x_1)(x_3 - x_2) \\ &\quad \vdots \\ f(x_m; \mathbf{a}) &= y_m = a_0 + a_1(x_m - x_0) + a_2(x_m - x_0)(x_m - x_1) \\ &\quad + a_3(x_m - x_0)(x_m - x_1)(x_m - x_2) \\ &\quad + \cdots + \\ &\quad a_m(x_m - x_0)(x_m - x_1)(x_m - x_2)\cdots(x_m - x_{m-2})(x - x_{m-1})(x - x_m) \end{aligned}} \tag{3}$$

3

Curve-Fitting Techniques: Interpolation

Therefore, the coefficients can be computed by the following sequential process as

$$\boxed{\begin{aligned}
a_0 &= y_0 \\
a_1 &= \frac{y_1 - a_0}{(x_1 - x_0)} = \frac{y_1 - y_0}{(x_1 - x_0)} \\
a_2 &= \frac{y_2 - a_0 - a_1(x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)} \\
a_3 &= \frac{y_3 - a_0 - a_1(x_3 - x_0) - a_2(x_3 - x_0)(x_3 - x_1)}{(x_3 - x_0)(x_3 - x_1)(x_3 - x_2)} \\
&\vdots \\
a_m &= \frac{y_m - a_0 - a_1(x_m - x_0) - a_2(x_m - x_0)(x_m - x_1) - \cdots}{(x_m - x_0)(x_m - x_1)(x_m - x_2)\cdots(x_m - x_{m-2})(x - x_{m-1})(x - x_m)}
\end{aligned}} \tag{4}$$

Curve-Fitting Techniques: Interpolation

(3-3) Another form of the polynomial coefficients of the Newton's Interpolating: Divided Difference Formula

(a) Linear interpolation

- Given data: $\{(x_0, y_0), (x_1, y_1)\}$
- Interpolation function: $y \approx f(x) = b_0 + b_1 x$
- Constraints: $y_0 = f(x_0), y_1 = f(x_1) \rightarrow f(x) = a_0 + a_1(x - x_0)$
- Calculation of b_0, b_1

$$\begin{aligned} y_0 &= f(x_0) = a_0 \\ y_1 &= f(x_1) = a_0 + a_1(x_1 - x_0) \end{aligned} \rightarrow \boxed{\begin{aligned} a_0 &= f(x_0) \\ a_1 &= \frac{f(x_1) - f(x_0)}{x_1 - x_0} \end{aligned}} \rightarrow \begin{aligned} b_0 &= a_0 - a_1 x_0 \\ b_1 &= a_1 \end{aligned}$$

(b) Quadratic interpolation

- Given data: $\{(x_0, y_0), (x_1, y_1), (x_2, y_2)\}$
- Constraints: $y_0 = f(x_0), y_1 = f(x_1), y_2 = f(x_2)$
- Interpolation function(m=2): $y \approx f(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1)$
- Calculation of a_0, a_1, a_2

$$\begin{aligned} y_0 &= f(x_0) = a_0 \\ y_1 &= f(x_1) = a_0 + a_1(x_1 - x_0) \\ y_2 &\approx f(x_2) = a_0 + a_1(x_2 - x_0) + a_2(x_2 - x_0)(x_2 - x_1) \end{aligned} \rightarrow \boxed{\begin{aligned} a_0 &= f(x_0) \\ a_1 &= \frac{f(x_1) - f(x_0)}{x_1 - x_0} = f[x_1, x_0] \end{aligned}}$$

5

$$a_2 = \frac{f(x_2) - a_0 - a_1(x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)} = \frac{f(x_2) - f(x_0) - \dfrac{f(x_1) - f(x_0)}{x_1 - x_0}(x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)}$$

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

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

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

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

$$= \frac{f[x_2, x_1] - f[x_1, x_0]}{(x_2 - x_0)} \leftarrow f[x_2, x_1] = \frac{\{f(x_2) - f(x_1)\}}{(x_2 - x_1)}, f[x_1, x_0] = \frac{\{f(x_1) - f(x_0)\}}{(x_1 - x_0)}$$

$$\boxed{\begin{aligned} &a_2 = \frac{f[x_2, x_1] - f[x_1, x_0]}{(x_2 - x_0)} = f[x_2, x_1, x_0] \\ &\text{where } f[x_2, x_1] = \frac{\{f(x_2) - f(x_1)\}}{(x_2 - x_1)}, \quad f[x_1, x_0] = \frac{\{f(x_1) - f(x_0)\}}{(x_1 - x_0)} \end{aligned}}$$

6

(c) n^th order Polynomial interpolation: Divided-Difference Interpolation Formula

In general, if we define the following divided difference formula,

0^th order: $\boxed{f[x_i]=f(x_i)}$ 1^st order: $\boxed{f[x_i,x_j]=\dfrac{f[x_i]-f[x_j]}{(x_i-x_j)}}$

2^nd order: $\boxed{f[x_i,x_j,x_k]=\dfrac{f[x_i,x_j]-f[x_j,x_k]}{(x_i-x_k)}}$ ……

n^th order: $\boxed{f[x_n,x_{n-1},x_{n-1},\cdots,x_1,x_0]=\dfrac{f[x_n,x_{n-1},x_{n-2},\cdots,x_1]-f[x_{n-1},x_{n-2},\cdots,x_1,x_0]}{(x_n-x_0)}}$

The n^th order polynomial interpolation function can be defined using the divided difference formula

$$f(x)=a_0+a_1(x-x_0)+a_2(x-x_0)(x-x_1)+\cdots+a_m(x-x_0)(x-x_1)(x-x_2)\cdots(x-x_{m-2})(x-x_{m-1})$$

$$a_0=f[x_0] \qquad a_{m-2}=f[x_{m-2},\cdots x_2,x_1,x_0]$$

$$a_1=f[x_1,x_0] \qquad \cdots \qquad a_{m-1}=f[x_{m-1},x_{m-2},\cdots x_2,x_1,x_0]$$

$$a_3=f[x_2,x_1,x_0] \qquad a_m=f[x_m,x_{m-1},x_{m-2},\cdots x_2,x_1,x_0] \tag{5}$$

where

$$f[x_0]=f(x_0)$$

$$f[x_n,x_{n-1},x_{n-1},\cdots,x_1,x_0]=\frac{f[x_n,x_{n-1},x_{n-2},\cdots,x_1]-f[x_{n-1},x_{n-2},\cdots,x_1,x_0]}{(x_n-x_0)} \quad \text{for } n=1,2,3,4,\cdots$$

7

In a real application, it is more convenient to compute the coefficients in a following sequential process.

$$
\begin{aligned}
&a_0 = f(x_0) \\
&f(x_1) = a_0 + a_1(x_1 - x_0) \rightarrow a_1 = \{f(x_1) - a_0\} / (x_1 - x_0) \\
&f(x_2) = a_0 + a_1(x_2 - x_0) + a_2(x_2 - x_0)(x_2 - x_1) \rightarrow a_2 = \{f(x_1) - a_0 - a_1(x_2 - x_0)\} / \{(x_2 - x_0)(x_2 - x_1)\} \\
&\vdots
\end{aligned}
$$

(3-4) Derivation of Divided Difference Formula using Eq (5)

$$
\begin{aligned}
a_0 &= y_0 = y[x_0] \\
a_1 &= \frac{y_1 - a_0}{(x_1 - x_0)} = \frac{y_1 - y_0}{(x_1 - x_0)} = \frac{y[x_1] - y[x_0]}{(x_1 - x_0)} = y[x_1, x_0] \\
a_2 &= \frac{y_2 - y_0 - a_1(x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)} = \frac{y_2 - y_0 - \dfrac{y_1 - y_0}{(x_1 - x_0)}(x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)} = \frac{\dfrac{y_2 - y_1 + (y_1 - y_0)}{(x_2 - x_1)} - \dfrac{y_1 - y_0}{(x_1 - x_0)(x_2 - x_1)}(x_2 - x_0)}{(x_2 - x_0)} \\
&= \frac{\dfrac{y_2 - y_1 + (y_1 - y_0)}{(x_2 - x_1)} - \dfrac{y_1 - y_0}{(x_1 - x_0)(x_2 - x_1)}(x_2 - x_0)}{(x_2 - x_0)} = \frac{\dfrac{y_2 - y_1}{(x_2 - x_1)} - \dfrac{y_1 - y_0}{(x_1 - x_0)}}{(x_2 - x_0)} = \frac{y[x_2, x_1] - y[x_1, x_0]}{(x_2 - x_0)} \\
&= y[x_2, x_1, x_0]
\end{aligned}
$$

8

Curve-Fitting Techniques: Interpolation

$$a_3=\frac{y_3-y_0-a_1(x_3-x_0)-a_2(x_3-x_0)(x_3-x_1)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)}=\frac{y_3-y_0-y[x_1,x_0](x_3-x_0)-y[x_2,x_1,x_0](x_3-x_0)(x_3-x_1)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)}$$

$$=\frac{(y_3-y_2)+(y_2-y_1)+(y_1-y_0)-y[x_1,x_0](x_3-x_0)-y[x_2,x_1,x_0](x_3-x_0)(x_3-x_1)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)}$$

$$=\frac{\dfrac{y[x_3,x_2]}{(x_3-x_1)}+\dfrac{y[x_2,x_1](x_2-x_1)}{(x_3-x_1)(x_3-x_2)}+\dfrac{y[x_1,x_0](x_1-x_0)}{(x_3-x_1)(x_3-x_2)}-\dfrac{y[x_1,x_0](x_3-x_0)}{(x_3-x_1)(x_3-x_2)}-\dfrac{\big[y[x_2,x_1]-y[x_1,x_0]\big](x_3-x_0)}{(x_3-x_2)(x_2-x_0)}}{(x_3-x_0)}$$

$$=\frac{\dfrac{y[x_3,x_2]}{(x_3-x_1)}-\dfrac{y[x_2,x_1]}{(x_3-x_1)}+\dfrac{y[x_2,x_1](x_3-x_1)}{(x_3-x_1)(x_3-x_2)}+\dfrac{y[x_1,x_0](x_1-x_3)}{(x_3-x_1)(x_3-x_2)}-\dfrac{\big[y[x_2,x_1]-y[x_1,x_0]\big](x_3-x_0)}{(x_3-x_2)(x_2-x_0)}}{(x_3-x_0)}$$

$$=\frac{y[x_3,x_2,x_1]+\dfrac{y[x_2,x_1]}{(x_3-x_2)}\left\{1-\dfrac{(x_3-x_0)}{(x_2-x_0)}\right\}+\dfrac{y[x_1,x_0]}{(x_3-x_2)}\left\{-1+\dfrac{(x_3-x_0)}{(x_2-x_0)}\right\}}{(x_3-x_0)}$$

$$=\frac{y[x_3,x_2,x_1]+\dfrac{y[x_2,x_1]}{(x_3-x_2)}\left\{\dfrac{(x_2-x_3)}{(x_2-x_0)}\right\}+\dfrac{y[x_1,x_0]}{(x_3-x_2)}\left\{\dfrac{(x_3-x_2)}{(x_2-x_0)}\right\}}{(x_3-x_0)}=\frac{y[x_3,x_2,x_1]-\dfrac{y[x_2,x_1]}{(x_2-x_0)}+\dfrac{y[x_1,x_0]}{(x_2-x_0)}}{(x_3-x_0)}$$

$$=\frac{y[x_3,x_2,x_1]-\dfrac{y[x_2,x_1]-y[x_1,x_0]}{(x_2-x_0)}}{(x_3-x_0)}=\frac{y[x_3,x_2,x_1]-y[x_2,x_1,x_0]}{(x_3-x_0)}=y[x_3,x_2,x_1,x_0]$$

$$\vdots$$

4. Lagrange Interpolating Polynomial

○ Exercises

(1) Find a 2nd order polynomial $y = a_0 + a_1 x + a_2 x^2$ satisfying the following condition

(1-1) Passing points given: $(x_0,1),(x_1,0),(x_2,0)$

$$y = a_2(x-x_1)(x-x_2) \leftarrow (x_0,1)$$

Answer→ $$= \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} = \prod_{\substack{k=0 \\ k\neq 0}}^{3} \frac{(x-x_k)}{(x_0-x_k)} = L_0(x)$$

(1-2) Passing points given: $(x_0,0),(x_1,1),(x_2,0)$

$$y = a_2(x-x_0)(x-x_2) \leftarrow (x_1,1)$$

Answer→ $$= \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} = \prod_{\substack{k=0 \\ k\neq 1}}^{3} \frac{(x-x_k)}{(x_1-x_k)} = L_1(x)$$

(1-3) Passing points given: $(x_0,0),(x_1,0),(x_2,1)$

$$y = a_2(x-x_0)(x-x_1) \leftarrow (x_2,1)$$

Answer→ $$= \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} = \prod_{\substack{k=0 \\ k\neq 2}}^{3} \frac{(x-x_k)}{(x_2-x_k)} = L_2(x)$$

10

Curve-Fitting Techniques: Interpolation

(2) Find a 2nd order polynomial $y = a_0 + a_1 x + a_2 x^2$ passing points of $(x_0, y_0), (x_1, y_1), (x_2, y_2)$

Answer→ $y = \sum_{j=1}^{3} y_j L_j(x) \leftarrow L_j(x) = \prod_{\substack{k=0 \\ k \neq j}}^{3} \frac{(x - x_k)}{(x_j - x_k)}$

Proof)

$$
\begin{aligned}
y &= \sum_{j=1}^{3} y_j L_j(x) \\
&= y_0 L_0(x) + y_1 L_1(x) + y_2 L_2(x) \\
&= y_0 \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} + y_1 \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} + y_2 \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)}
\end{aligned}
$$

which passes three given points $(x_0, y_0), (x_1, y_1), (x_2, y_2)$

(3) General Lagrange interpolating function $L_j(x)$ satisfies $L_j(x_k) = \delta_{jk}, \quad j, k = 1, 2, \cdots, n$

$$
L_j(x) = \prod_{\substack{k=0 \\ k \neq j}}^{n} \frac{(x - x_k)}{(x_j - x_k)} = \frac{(x - x_0)(x - x_1)\cdots(x_j - x_{j-1})(x_j - x_{j+1})\cdots(x - x_{n-1})(x - x_n)}{(x_j - x_0)(x_j - x_1)\cdots(x_j - x_{j-1})(x_j - x_{j+1})\cdots(x_j - x_{n-1})(x_j - x_n)}
$$

Where the Dirac delta function δ_{jk} satisfies $\delta_{jk} = \begin{cases} 1 & \text{if } j = k \\ 0 & \text{if } j \neq k \end{cases}$

11

Curve-Fitting Techniques: Interpolation

- Definition of n-th order Lagrange polynomial and its property

$$L_j(x)=\prod_{\substack{k=0\\k\neq j}}^{n}\frac{(x-x_k)}{(x_j-x_k)}=\frac{(x-x_0)(x-x_1)(x-x_2)\cdots(x_j-x_{j-1})(x_j-x_{j+1})\cdots(x-x_{n-1})(x-x_n)}{(x_j-x_0)(x_j-x_1)(x-x_2)\cdots(x_j-x_{j-1})(x_j-x_{j+1})\cdots(x_j-x_{n-1})(x_j-x_n)}$$

For $j=0,1,2,3,\cdots,n$

In case n=3

$$L_0(x)=\frac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)}\rightarrow L_0(x_0)=1,\ L_0(x_1)=L_0(x_2)=L_0(x_3)=0$$

$$L_1(x)=\frac{(x-x_0)(x-x_2)(x-x_3)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)}\rightarrow L_1(x_1)=1,\ L_1(x_0)=L_1(x_2)=L_1(x_3)=0$$

$$L_2(x)=\frac{(x-x_0)(x-x_1)(x-x_3)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)}\rightarrow L_2(x_2)=1,\ L_2(x_0)=L_2(x_1)=L_2(x_3)=0$$

$$L_3(x)=\frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)}\rightarrow L_3(x_3)=1,\ L_2(x_0)=L_3(x_1)=L_3(x_2)=0$$

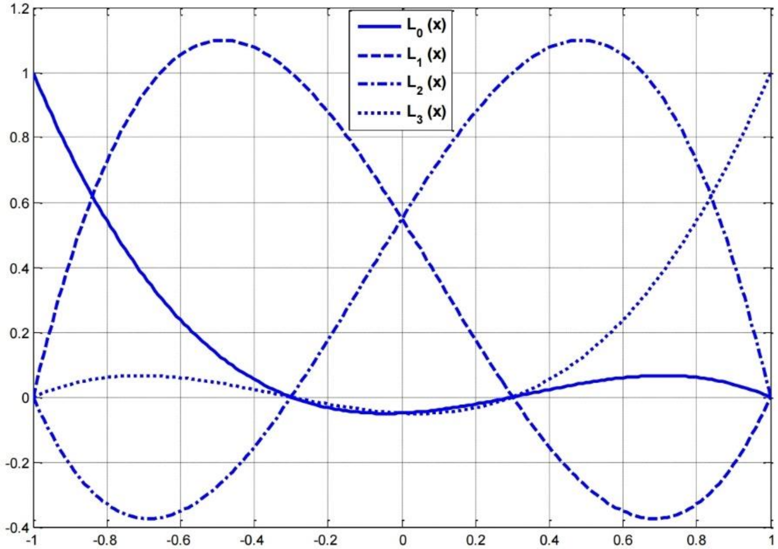
$$\boxed{L_j(x_k)=\delta_{jk},\quad j,k=0,1,2,3,\cdots,n}$$

The Kronecker delta function δ_{jk} is defined as $\delta_{jk}=\begin{cases}1, & \text{if } j=k\\ 0, & \text{if } j\neq k\end{cases}$

$$n=3,\quad x_0=-1.0,\quad x_1=-0.3,\quad x_2=0.3,\quad x_3=1.0$$

12

Curve-Fitting Techniques: Interpolation



- Interpolation for the given data $\{(x_0, f_0), (x_1, f_1), (x_2, f_2), (x_3, f_3), \cdots, (x_j, f_j), \cdots, (x_n, f_n)\}$

$$\boxed{f_n(x) = \sum_{j=0}^{n} L_j(x) f(x_j)}$$

13

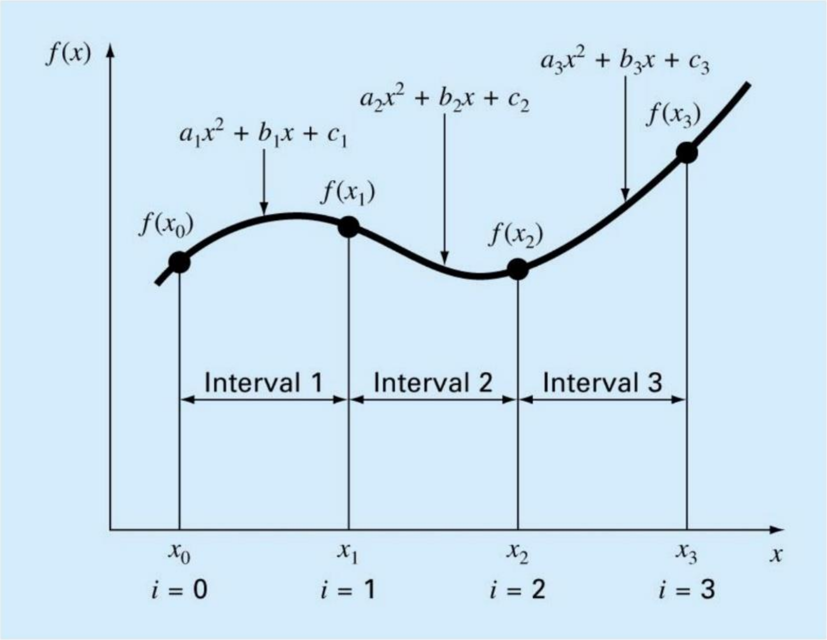
5. Spline Interpolation (Spline means a thin flexible strip to draw smooth curves in drafting).

- Definition of Spline interpolation function: Local polynomial interpolation

$$f(x) \approx f_j(x) = a_j + b_j x + c_j x^2 + d_j x^3 + \cdots\cdots \quad (x_j \le x \le x_{j+1}, \quad j = 0,1,2,\cdots,n)$$

with the given (n+1)-data points $\{(x_0, f(x_0)),(x_1, f(x_1)),(x_2, f(x_2)),(x_3, f(x_3)),\cdots,(x_n, f(x_n))\}$

(Example for quadratic spline (n=2))



14

(4-1) Linear Spline

$$\begin{aligned} f_0(x) &= a_0 + b_0 x, \quad (x_0 \le x < x_1) \\ f_1(x) &= a_1 + b_1 x, \quad (x_1 \le x < x_2) \\ f_2(x) &= a_2 + b_2 x, \quad (x_2 \le x < x_3) \\ &\vdots \\ f_{n-1}(x) &= a_{n-1} + b_{n-1} x, \quad (x_{n-1} \le x \le x_n) \end{aligned}$$

- Unkown 2n (n for a_j , n for b_j) with (n+1) data point.
  → (n-1) additional relations are required: continuity conditions at each point $x_j \ (j = 1,2,\cdots,n-1)$

$$\begin{aligned} a_0 + b_0 x_0 &= f(x_0) \\ a_0 + b_0 x_1 &= f(x_1) \\ a_1 + b_1 x_1 &= f(x_1) \\ a_1 + b_1 x_2 &= f(x_2) \\ &\vdots \\ a_{n-2} + b_{n-2} x_{n-1} &= f(x_{n-1}) \\ a_{n-1} + b_{n-1} x_{n-1} &= f(x_{n-1}) \\ a_{n-1} + b_{n-1} x_n &= f(x_n) \end{aligned}$$

$$\rightarrow \begin{pmatrix} 1 & x_0 & & & & & \\ 1 & x_1 & & & & & \\ & & 1 & x_1 & & & \\ & & 1 & x_2 & & & \\ & & & & \ddots & & \\ & & & & & 1 & x_{n-1} \\ & & & & & 1 & x_n \end{pmatrix} \begin{pmatrix} a_0 \\ b_0 \\ a_1 \\ b_1 \\ \vdots \\ a_{n-1} \\ b_{n-1} \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f(x_1) \\ f(x_1) \\ f(x_2) \\ \vdots \\ f(x_{n-1}) \\ f(x_n) \end{pmatrix}$$

- The results become

$$b_j = \frac{f(x_{j+1}) - f(x_j)}{x_{j+1} - x_j}$$

$$a_j = f(x_j) - b_j x_j$$

15

(4-2) Quadratic Spline

$$
\begin{aligned}
f_0(x) &= a_0 + b_0 x + c_0 x^2, \quad (x_0 \le x < x_1) \\
f_1(x) &= a_1 + b_1 x + c_1 x^2, \quad (x_1 \le x < x_2) \\
f_2(x) &= a_2 + b_2 x + c_2 x^2, \quad (x_2 \le x < x_3) \\
f_3(x) &= a_3 + b_3 x + c_3 x^2, \quad (x_3 \le x < x_4) \\
f_4(x) &= a_4 + b_4 x + c_4 x^2, \quad (x_4 \le x < x_5) \\
&\vdots \\
f_{n-1}(x) &= a_{n-1} + b_{n-1} x + c_{n-1} x^2, \quad (x_{n-1} \le x \le x_n)
\end{aligned}
$$

→ Unknown 3n (n for a_j , n for b_j , n for c_j)

(a) (n+1) given data points.

$$
\begin{aligned}
a_0 + b_0 x_0 + c_0 x_0{}^2 &= f(x_0) \\
a_1 + b_1 x_1 + c_1 x_1{}^2 &= f(x_1) \\
&\vdots \\
a_{n-1} + b_{n-1} x_n + c_{n-1} x_n{}^2 &= f(x_n)
\end{aligned}
$$

(b) (n-1)-continuity conditions at each point

$x_j \ (j = 1, 2, \cdots, n-1)$

$$
\begin{aligned}
a_0 + b_0 x_1 + c_0 x_1{}^2 &= f(x_1) \\
a_1 + b_1 x_2 + c_1 x_2{}^2 &= f(x_2) \\
&\vdots \\
a_{n-2} + b_{n-2} x_{n-1} + c_{n-2} x_{n-1}{}^2 &= f(x_{n-1})
\end{aligned}
$$

16

(c) (n-1)-continuity conditions for 1st derivatives at each point $x_j \ (j = 1, 2, \cdots, n-1)$

$$\begin{array}{c} f_0'(x_1) = f_1'(x_1) \\ f_1'(x_2) = f_2'(x_2) \\ f_2'(x_3) = f_3'(x_3) \\ \vdots \\ f_{n-2}'(x_{n-1}) = f_{n-1}'(x_{n-1}) \end{array} \rightarrow \begin{array}{c} b_0 + 2c_0 x_1 = b_1 + 2c_1 x_1 \\ b_1 + 2c_1 x_2 = b_2 + 2c_2 x_2 \\ b_2 + 2c_2 x_3 = b_3 + 2c_3 x_3 \\ \vdots \\ b_{n-2} + 2c_{n-2} x_{n-1} = b_{n-1} + 2c_{n-1} x_{n-1} \end{array}$$

(d) 1-smooth condition at $x_0 : f_0''(x_0) = 0 \rightarrow c_0 = 0$

17

Curve-Fitting Techniques: Interpolation

Resultant system of equations for the quadratic spline curve

$$
\begin{aligned}
a_0 + b_0 x_0 &= f(x_0) \\
a_0 + b_0 x_1 &= f(x_1) \\
b_0 - b_1 - 2c_1 x_1 &= 0 \\
a_1 + b_1 x_1 + c_1 x_1^{\ 2} &= f(x_1) \\
a_1 + b_1 x_2 + c_1 x_2^{\ 2} &= f(x_2) \\
b_1 + 2c_1 x_2 - b_2 - 2c_2 x_2 &= 0 \\
&\vdots \\
a_{n-2} + b_{n-2} x_{n-1} + c_{n-2} x_{n-1}^{\ 2} &= f(x_{n-1}) \\
b_{n-2} + 2c_{n-2} x_{n-1} - b_{n-1} - 2c_{n-1} x_{n-1} &= 0 \\
a_{n-1} + b_{n-1} x_{n-1} + c_{n-1} x_{n-1}^{\ 2} &= f(x_{n-1}) \\
a_{n-1} + b_{n-1} x_n + c_{n-1} x_n^{\ 2} &= f(x_n)
\end{aligned}
$$

$$
\begin{pmatrix}
1 & x_0 & & & & & & & & & & & & \\
1 & x_1 & & & & & & & & & & & & \\
0 & 1 & 0 & -1 & -2x_1 & & & & & & & & & \\
 & & 1 & x_1 & x_1^{\ 2} & & & & & & & & & \\
 & & 1 & x_2 & x_2^{\ 2} & & & & & & & & & \\
 & & & & & \ddots & & & & & & & & \\
 & & & & & & 0 & -1 & -2x_{n-2} & & & & & \\
 & & & & & & 1 & x_{n-2} & x_{n-2}^{\ 2} & & & & & \\
 & & & & & & 1 & x_{n-1} & x_{n-1}^{\ 2} & & & & & \\
 & & & & & & & 1 & 2x_{n-1} & 0 & -1 & -2x_{n-1} & & \\
 & & & & & & & & & 1 & x_{n-1} & x_{n-1}^{\ 2} & & \\
 & & & & & & & & & 1 & x_n & x_n^{\ 2} & &
\end{pmatrix}
\begin{pmatrix} a_0 \\ b_0 \\ a_1 \\ b_1 \\ c_1 \\ \\ a_{n-2} \\ b_{n-2} \\ c_{n-2} \\ a_{n-1} \\ b_{n-1} \\ c_{n-1} \end{pmatrix}
=
\begin{pmatrix} f(x_0) \\ f(x_1) \\ 0 \\ f(x_1) \\ f(x_2) \\ \\ 0 \\ f(x_{n-2}) \\ f(x_{n-1}) \\ 0 \\ f(x_{n-1}) \\ f(x_n) \end{pmatrix}
$$

18

(4-3) Cubic Spline

$$
\begin{aligned}
f_0(x) &= a_0 + b_0 x + c_0 x^2 + d_0 x^3, \quad (x_0 \le x < x_1) \\
f_1(x) &= a_1 + b_1 x + c_1 x^2 + d_1 x^3, \quad (x_1 \le x < x_2) \\
f_2(x) &= a_2 + b_2 x + c_2 x^2 + d_2 x^3, \quad (x_2 \le x < x_3) \\
&\vdots \\
f_{n-1}(x) &= a_{n-1} + b_{n-1} x + c_{n-1} x^2 + d_{n-1} x^3, \quad (x_{n-1} \le x \le x_n)
\end{aligned}
$$

→ Unknown 4n (n for a_j , n for b_j , n for c_j, n for d_j): 4n conditions are required

(a) (n+1) given data points.

$$
\begin{aligned}
a_0 + b_0 x_0 + c_0 x_0^{\ 2} + d_0 x_0^{\ 3} &= f(x_0) \\
a_1 + b_1 x_1 + c_1 x_1^{\ 2} + d_1 x_1^{\ 3} &= f(x_1) \\
&\vdots \\
a_{n-1} + b_{n-1} x_n + c_{n-1} x_n^{\ 2} + d_{n-1} x_n^{\ 3} &= f(x_n)
\end{aligned}
$$

(b) (n-1)-continuity conditions at each point $x_j \ (j = 1, 2, \cdots, n-1)$

$$
\begin{aligned}
a_0 + b_0 x_1 + c_0 x_1^{\ 2} + d_0 x_1^{\ 3} &= f(x_1) \\
a_1 + b_1 x_2 + c_1 x_2^{\ 2} + d_1 x_2^{\ 3} &= f(x_2) \\
&\vdots \\
a_{n-2} + b_{n-2} x_{n-1} + c_{n-2} x_{n-1}^{\ 2} + d_{n-2} x_{n-1}^{\ 3} &= f(x_{n-1})
\end{aligned}
$$

19

Curve-Fitting Techniques: Interpolation

(c) (n-1)-continuity conditions for 1st derivatives at each point $x_j \ (j=1,2,\cdots,n-1)$

$$\boxed{\begin{gathered} f_0'(x_1)=f_1'(x_1) \\ f_1'(x_2)=f_2'(x_2) \\ \vdots \\ f_{n-2}'(x_{n-1})=f_{n-1}'(x_{n-1}) \end{gathered}} \rightarrow \boxed{\begin{gathered} b_0+2c_0x_1+3d_0x_1^{\ 2}=b_1+2c_1x_1+3d_1x_1^{\ 2} \\ b_1+2c_1x_2+3d_1x_2^{\ 2}=b_2+2c_2x_2+3d_2x_2^{\ 2} \\ \vdots \\ b_{n-2}+2c_{n-2}x_{n-1}+3d_{n-2}x_{n-1}^{\ 2}=b_{n-1}+2c_{n-1}x_{n-1}+3d_{n-1}x_{n-1}^{\ 2} \end{gathered}}$$

(d) (n-1)-continuity conditions for 2nd derivatives at each point $x_j \ (j=1,2,\cdots,n-1)$

$$\boxed{\begin{gathered} f_0''(x_1)=f_1''(x_1) \\ f_1''(x_2)=f_2''(x_2) \\ \vdots \\ f_{n-2}''(x_{n-1})=f_{n-1}''(x_{n-1}) \end{gathered}} \rightarrow \boxed{\begin{gathered} 2c_0+6d_0x_1=2c_1+6d_1x_1 \\ 2c_1+6d_1x_2=2c_2+6d_2x_2 \\ \vdots \\ 2c_{n-2}+6d_{n-2}x_{n-1}=2c_{n-1}+6d_{n-1}x_{n-1} \end{gathered}}$$

(e) 1-smooth condition at $x_0 : f_0'''(x_0)=0 \rightarrow d_0=0$

(f) 1-smooth condition at $x_n : f_{n-1}'''(x_n)=0 \rightarrow d_{n-1}=0$

20

End of Lecture