Periodic Orthogonal Basis Functions

$$\varphi_k(x) = \cos kx\,; \qquad 0 \le x \le 2\pi \quad \text{or} \quad -\pi \le x \le \pi$$

$$\langle \varphi_j, \varphi_k \rangle = \int_{-\pi}^{\pi} \cos jx \cos kx \, dx = \begin{cases} 0 & \text{if} \quad j \ne k \\ \pi & \text{if} \quad j = k \ne 0 \\ 2\pi & \text{if} \quad j = k = 0 \end{cases}$$

$$\psi_k(x) = \sin kx\,; \qquad 0 \le x \le 2\pi \quad \text{or} \quad -\pi \le x \le \pi$$

$$\langle \psi_j, \psi_k \rangle = \int_{-\pi}^{\pi} \sin jx \sin kx \, dx = \begin{cases} 0 & \text{if} \quad j \ne k \\ \pi & \text{if} \quad j = k \end{cases}$$

Wave Number: $k = 0, 1, 2, \cdots \infty$

Least Square Approximation of Periodic Functions

Consider a periodic function $f(x)$ of period 2π

We shall find the *least square approximation* of this function $f^*(x)$ using orthogonal basis functions as follows:

$$f^*(x) = \sum_{k=0}^{\infty} a_k \varphi_k(x) + \sum_{k=0}^{\infty} b_k \psi_k(x)$$

where, $\varphi_k(x) = \cos kx$ and $\psi_k(x) = \sin kx$

We have: $\langle \varphi_j, \varphi_k \rangle = \begin{cases} 0 & \text{if} \quad j \neq k \\ \pi & \text{if} \quad j = k \neq 0 \\ 2\pi & \text{if} \quad j = k = 0 \end{cases}$ and $\langle \psi_j, \psi_k \rangle = \begin{cases} 0 & \text{if} \quad j \neq k \\ \pi & \text{if} \quad j = k \end{cases}$

$$\langle \varphi_j, \psi_k \rangle = 0 \quad \forall\, j, k$$

Least Square Approximation of Periodic Functions

If $f^*(x)$ is the *least square approximation* of $f(x)$, normal equations must be satisfied:

$$\left\langle (f - f^*), \varphi_j \right\rangle = \left\langle (f - f^*), \psi_j \right\rangle = 0 \quad \forall\, j$$

$$\langle f, \varphi_j \rangle = \sum_{k=0}^{\infty} a_k \langle \varphi_j, \varphi_k \rangle + \sum_{k=0}^{\infty} b_k \langle \varphi_j, \psi_k \rangle \quad \Longrightarrow \quad a_k = \frac{\langle f, \varphi_k \rangle}{\langle \varphi_k, \varphi_k \rangle}$$

$$\langle f, \psi_j \rangle = \sum_{k=0}^{\infty} a_k \langle \psi_j, \varphi_k \rangle + \sum_{k=0}^{\infty} b_k \langle \psi_j, \psi_k \rangle \quad \Longrightarrow \quad b_k = \frac{\langle f, \psi_k \rangle}{\langle \psi_k, \psi_k \rangle}$$

Least Square Approximation of Periodic Functions

$$a_k = \frac{\langle f, \varphi_k \rangle}{\langle \varphi_k, \varphi_k \rangle} = \frac{1}{\langle \varphi_k, \varphi_k \rangle} \int_{-\pi}^{\pi} f(x)\varphi_k(x)\, dx = \frac{1}{\langle \varphi_k, \varphi_k \rangle} \int_{-\pi}^{\pi} f(x) \cos kx\, dx$$

$$a_0 = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x)\, dx; \qquad a_k = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos kx\, dx; \quad k = 1,2,\cdots\infty$$

$$b_k = \frac{\langle f, \psi_k \rangle}{\langle \psi_k, \psi_k \rangle} = \frac{1}{\langle \psi_k, \psi_k \rangle} \int_{-\pi}^{\pi} f(x)\psi_k(x)\, dx = \frac{1}{\langle \psi_k, \psi_k \rangle} \int_{-\pi}^{\pi} f(x) \sin kx\, dx$$

$$b_0 = 0; \qquad b_k = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin kx\, dx; \quad k = 1,2,\cdots\infty$$

Least Square Approximation of Periodic Functions

$$f^*(x) = \sum_{k=0}^{\infty} a_k \varphi_k(x) + \sum_{k=0}^{\infty} b_k \psi_k(x)$$

where, $\varphi_k(x) = \cos kx$ and $\psi_k(x) = \sin kx$

$$f^*(x) = \sum_{k=0}^{\infty} a_k \cos kx + \sum_{k=0}^{\infty} b_k \sin kx$$

$$= a_0 + \sum_{k=1}^{\infty} a_k \cos kx + \sum_{k=1}^{\infty} b_k \sin kx$$

where,

$$a_0 = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x)\, dx; \quad a_k = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos kx\, dx; \quad b_k = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin kx\, dx$$

Fourier Series is the Least Square Approximation of a periodic function!

Least Square Approximation of Periodic Functions

Consider a periodic function $f(x)$ of period 2π

We shall find the *least square approximation* of this function $f^*(x)$ using orthogonal basis functions as follows:

$$f^*(x) = \sum_{k=-\infty}^{\infty} c_k \phi_k(x)\,; \qquad \phi_k(x) = e^{ikx}$$

We have: $\langle \phi_j, \phi_k \rangle = \begin{cases} 0 & \text{if} \quad j \neq k \\ \pi & \text{if} \quad j = k \end{cases}$

Least Square Approximation of Periodic Functions

If $f^*(x)$ is the *least square approximation* of $f(x)$, normal equations must be satisfied:

$$\langle (f - f^*), \phi_j \rangle = 0 \quad \forall\, j$$

$$\langle f, \phi_j \rangle = \sum_{k=-\infty}^{\infty} c_k \langle \phi_j, \phi_k \rangle \quad \Longrightarrow \quad c_k = \frac{\langle f, \phi_k \rangle}{\langle \phi_k, \phi_k \rangle}$$

$$f^*(x) = \sum_{k=-\infty}^{\infty} c_k e^{ikx}$$

$$c_k = \frac{\langle f, \phi_k \rangle}{\langle \phi_k, \phi_k \rangle} = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x)\, e^{-ikx} dx$$

Least Square Approximation of Periodic Functions

$$f^*(x) = \sum_{k=-\infty}^{\infty} c_k e^{ikx} = c_0 + \sum_{k=-\infty}^{-1} c_k e^{ikx} + \sum_{k=1}^{\infty} c_k e^{ikx}$$

$$= c_0 + \sum_{k=1}^{\infty} c_{-k} e^{-ikx} + \sum_{k=1}^{\infty} c_k e^{ikx}$$

$$= c_0 + \sum_{k=1}^{\infty} (c_k + c_{-k}) \cos kx + \sum_{k=1}^{\infty} i(c_k - c_{-k}) \sin kx$$

$$= a_0 + \sum_{k=1}^{\infty} a_k \cos kx + \sum_{k=1}^{\infty} b_k \sin kx$$

where, $a_0 = c_0; a_k = (c_k + c_{-k}); b_k = i(c_k - c_{-k})$

Least Square Approximation of Periodic Functions

$$c_k = \frac{1}{2\pi}\int_{-\pi}^{\pi} f(x)\, e^{-ikx} dx$$

$$a_0 = c_0 = \frac{1}{2\pi}\int_{-\pi}^{\pi} f(x)\, dx$$

$$a_k = (c_k + c_{-k}) = \frac{1}{2\pi}\int_{-\pi}^{\pi} f(x)\left(e^{-ikx} + e^{ikx}\right)dx = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\cos kx\, dx$$

$$b_k = i(c_k - c_{-k}) = \frac{1}{2\pi}\int_{-\pi}^{\pi} f(x)\,(i)\left(e^{-ikx} - e^{ikx}\right)dx = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\sin kx\, dx$$

Both are Fourier Series, one is the exponential form and the other is the sine-cosine form!

The Fourier Series

✓For a periodic function $f(x)$ with a period of 2π, *i.e.*, $(-\pi, \pi)$ or $(0, 2\pi)$

$$f(x) = \frac{a_0}{2} + \sum_{k=1}^{\infty} a_k \cos kx + \sum_{k=1}^{\infty} b_k \sin kx$$

$$a_0 = \frac{1}{\pi}\int_{-\pi}^{\pi} f(t)\,dt; \quad a_k = \frac{1}{\pi}\int_{-\pi}^{\pi} f(t)\cos kt\,dt\,; \quad b_k = \frac{1}{\pi}\int_{-\pi}^{\pi} f(t)\sin kt\,dt$$

✓Alternatively:

$$f(x) = \sum_{k=-\infty}^{\infty} c_k e^{ikx} \qquad c_k = \frac{1}{2\pi}\int_{-\pi}^{\pi} f(x)\,e^{-ikx}dx$$

$$c_k = \frac{1}{2}(a_k - ib_k); \qquad c_{-k} = \frac{1}{2}(a_k + ib_k)$$

The Fourier Series

For a periodic function of any period $2L$ *i.e.*, $(-L, L)$ or $(0, 2L)$, simply scale the variable x. That is, replace x by $\pi x/L$

$$f(x) = \frac{a_0}{2} + \sum_{k=1}^{\infty} a_k \cos\frac{k\pi x}{L} + \sum_{k=1}^{\infty} b_k \sin\frac{k\pi x}{L}$$

$$a_k = \frac{1}{L}\int_{-L}^{L} f(t)\cos\frac{k\pi t}{L}\,dt\,; \quad b_k = \frac{1}{L}\int_{-L}^{L} f(t)\sin\frac{k\pi t}{L}\,dt$$

$$f(x) = \sum_{k=-\infty}^{\infty} c_k e^{i\frac{k\pi x}{L}}\;; \qquad c_k = \frac{1}{2L}\int_{-L}^{L} f(x)\, e^{-i\frac{k\pi x}{L}}\,dx$$

The Discrete and Finite Fourier Series

$$f(x) = \sum_{k=-\infty}^{\infty} c_k e^{ikx} \qquad\qquad c_k = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x)\, e^{-ikx} dx$$

Finite (only finite frequencies or wave numbers):

$$f(x) = \sum_{k=a}^{b} c_k e^{ikx}$$

Discrete points in *x*:

$$x_m = \frac{2\pi m}{M+1}; \qquad c_k = \frac{\langle f, \phi_k \rangle}{\langle \phi_k, \phi_k \rangle} = \frac{1}{M+1} \sum_{m=0}^{M} f(x_m) e^{-ikx_m}$$

The Fourier Series

- ✓ Conditions for convergence:
  - ➢ $f(x)$ is continuous or piecewise continuous in $(-\pi, \pi)$ or $(-L, L)$
  - ➢ **finite number** of **finite discontinuities**, i.e., finite number of maxima, minima
  - ➢ $f(x)$ is periodic on the entire x-axis with period 2π or $2L$

- ✓ Sufficient Condition (Dirichlet Condition for Fourier Series):

$$\|f(x)\|^2 = \int_{-\pi \text{ or } -L}^{\pi \text{ or } L} |f(x)|^2 w(x) dx < \infty$$

- ✓ What if the function is not periodic?

From the exponential form of the Fourier Series!

✓ Exponential form:

$$f(x) = \sum_{k=-\infty}^{\infty} c_k e^{i\frac{k\pi}{L}x} \ ; \ \ c_k = \frac{1}{2L}\int_{-L}^{L} f(t)e^{-i\frac{k\pi}{L}t}dt$$

✓ Now, let $L \to \infty$. So, the finite interval [-L, L] transforms to infinite interval (-∞, ∞)

✓ Define: $\frac{\pi}{L} = \Delta\omega$ and $\frac{k\pi}{L} = \omega$ as $L \to \infty$

$$f(x) = \frac{1}{2\pi}\sum_{k=-\infty}^{\infty} \frac{\pi}{L}\int_{-L}^{L} f(t)e^{-i\frac{k\pi}{L}t}dt\, e^{i\frac{k\pi}{L}x} = \frac{1}{2\pi}\int_{-\infty}^{\infty} d\omega \int_{-\infty}^{\infty} f(t)e^{-i\omega t}dt\, e^{i\omega x}$$

$$f(x) = \frac{1}{2\pi}\int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} f(t)e^{-i\omega t}dt\right]e^{i\omega x}d\omega = \frac{1}{2\pi}\int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} f(t)e^{i\omega t}dt\right]e^{-i\omega x}d\omega$$

The Fourier Transform Pairs

$$f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} d\omega \int_{-\infty}^{\infty} f(t) e^{i\omega(t-x)} dt$$

$$f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} f(t) e^{i\omega t} dt \right] e^{-i\omega x} d\omega$$

- ✓ Define:

$$F(i\omega) = \int_{-\infty}^{\infty} f(t) e^{i\omega t} dt \qquad f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega)\, e^{-i\omega x} d\omega$$

- ✓ $F(i\omega)$ is the *Fourier transform* and the 2nd integral is the *inverse transform*.
- ✓ The following are equivalent:
  - ➢ $F(i\omega) = \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt$ and $f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(i\omega)\, e^{i\omega x} d\omega$
  - ➢ $F(i\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t) e^{i\omega t} dt$ and $f(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} F(i\omega)\, e^{-i\omega x} d\omega$
  - ➢ $F(i\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt$ and $f(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} F(i\omega)\, e^{i\omega x} d\omega$

Discrete Fourier Transform (DFT)

$$F(i\omega) = \int_{-\infty}^{\infty} f(t)e^{-i\omega t}dt \qquad f(x) = \frac{1}{2\pi}\int_{-\infty}^{\infty} F(i\omega)\, e^{i\omega x}d\omega$$

- ✓ DFT is equivalent to the continuous Fourier Transform for signals known at *discrete* equi-spaced instances or locations.
- ✓ Continuous signal $f(t)$. If we have samples $f(t_j)$ at intervals of Δt, $t_j = j\Delta t$, $j = 0, 1, 2, N - 1$
- ✓ Continuous Fourier Transform can be approximated as:

$$F(i\omega) = \int_{0}^{(N-1)\Delta t} f(t)e^{-i\omega t}dt = \sum_{j=0}^{N-1} f(j\Delta t)e^{-i\omega j\Delta t}$$

- ✓ In principle, this can be evaluated for any ω, but with only N data points to start with, only N final inputs will be significant.
- ✓ Continuous Fourier Transform: can be evaluated over a finite interval (fundamental period T) rather than $(-\infty, \infty)$ if the wave form was periodic
- ✓ DFT Assumption: data is periodic, $f(N\Delta t)$ to $f((2N-1)\Delta t)$ is same as $f(0)$ to $f((N-1)\Delta t)$

Discrete Fourier Transform (DFT)

$$F(i\omega) = \int_0^{(N-1)\Delta t} f(t)e^{-i\omega t}dt = \sum_{j=0}^{N-1} f(j\Delta t)e^{-i\omega j\Delta t}$$

- ✓ DFT Assumption: data is periodic, $f(N\Delta t)$ to $f((2N\text{-}1)\Delta t)$ is same as $f(0)$ to $f((N\text{-}1)\Delta t)$
- ✓ So, we evaluate the DFT equation for the fundamental frequency (e.g., 1 cycle/sequence or $(1/N\Delta t)$ Hz or $(2\pi/N\Delta t)$ rad/sec) and its harmonics:

$$\omega = 0, \frac{2\pi}{N\Delta t}, \frac{2\pi}{N\Delta t} \times 2, \frac{2\pi}{N\Delta t} \times 3 \cdots, \frac{2\pi}{N\Delta t} \times k \cdots, \frac{2\pi}{N\Delta t} \times (N-1)$$

- ✓ Therefore, the DFT is:

$$F(k) = \sum_{j=0}^{N-1} f(j\Delta t)e^{-i\omega j\Delta t} = \sum_{j=0}^{N-1} f(j\Delta t)e^{-i\frac{2\pi k}{N\Delta t}j\Delta t} = \sum_{j=0}^{N-1} f(j\Delta t)e^{-i\frac{2\pi k}{N}j}$$

$$k = 0, 1, 2, \cdots N-1$$

Discrete Fourier Transform (DFT)

$$F(k)=\sum_{j=0}^{N-1} f(j\Delta t)e^{-i\frac{2\pi}{N}kj}\,; \qquad k=0,1,2,\cdots N-1$$

$$\begin{bmatrix} F(0)\\ F(1)\\ F(2)\\ \vdots\\ F(k)\\ \vdots\\ F(N-1)\end{bmatrix}=\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 & \cdots & 1\\ 1 & A & A^2 & \cdots & A^j & \cdots & A^{N-1}\\ 1 & A^2 & A^4 & \cdots & A^{2j} & \cdots & A^{2(N-1)}\\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots\\ 1 & A^k & A^{2k} & \cdots & A^{kj} & \cdots & A^{k(N-1)}\\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots\\ 1 & A^{(N-1)} & A^{2(N-1)} & \cdots & A^{j(N-1)} & \cdots & A^{(N-1)(N-1)}\end{bmatrix}\begin{bmatrix} f(0)\\ f(\Delta t)\\ f(2\Delta t)\\ \vdots\\ f(j\Delta t)\\ \vdots\\ f\big((N-1)\Delta t\big)\end{bmatrix}$$

Where,

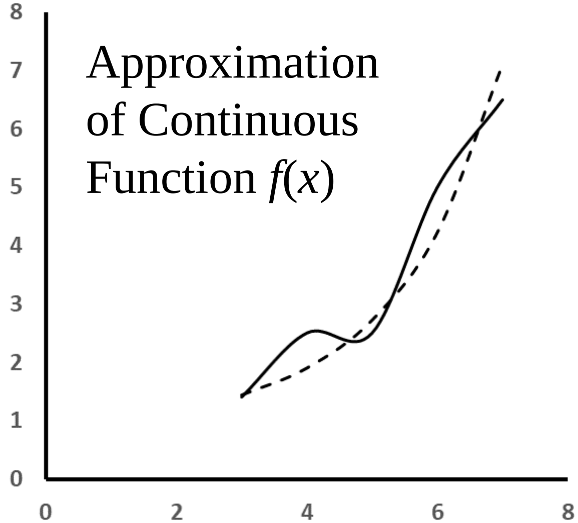
$$A=e^{-i\frac{2\pi}{N}}$$

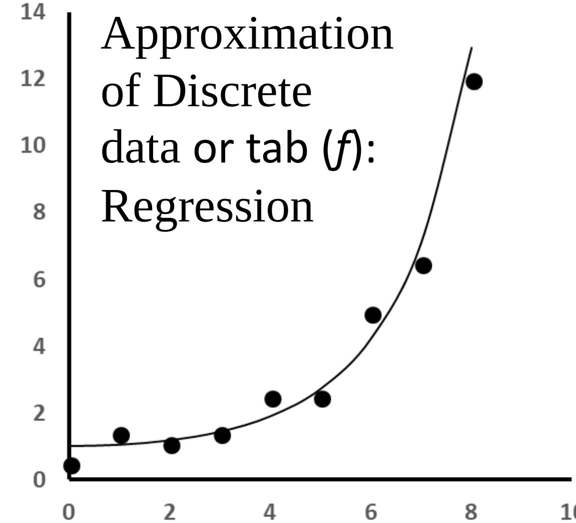
ESO 208A: Computational Methods in Engineering

Interpolation

Saumyen Guha

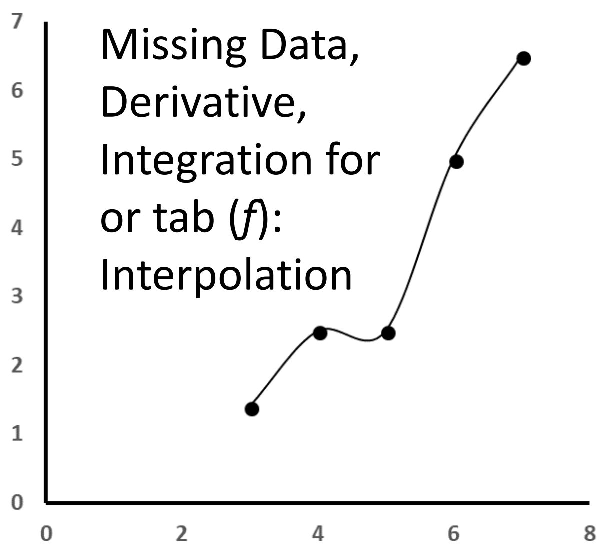
Department of Civil Engineering
IIT Kanpur

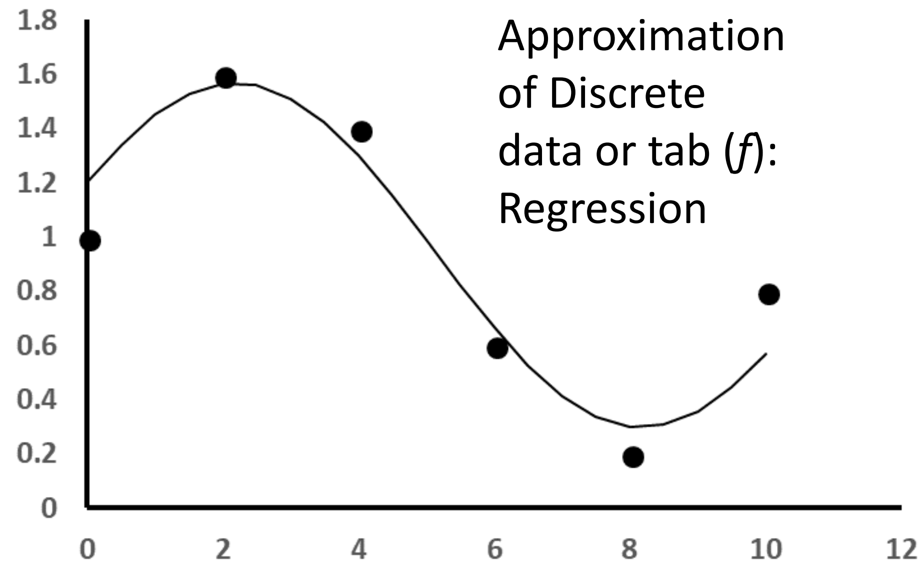




Complicated Analytical Function, Analog Signal from a measuring device

Discrete measurements of continuous experiments or phenomena





Discrete Data

- ✓ (n + 1) observations or data pairs [(x_0, y_0), (x_1, y_1), (x_2, y_2) … (x_n, y_n)]
- ✓ (m + 1) basis functions: $\{\varphi_0, \varphi_1, \varphi_2, \cdots \varphi_m\}$
- ✓ Approximating polynomial: $p(x) = \sum_{j=0}^{m} c_j \varphi_j(x)$

$$c_0\varphi_0(x_0) + c_1\varphi_1(x_0) + c_2\varphi_2(x_0) + \cdots + c_m\varphi_m(x_0) = y_0$$
$$c_0\varphi_0(x_1) + c_1\varphi_1(x_1) + c_2\varphi_2(x_1) + \cdots + c_m\varphi_m(x_1) = y_1$$

::

$$c_0\varphi_0(x_n) + c_1\varphi_1(x_n) + c_2\varphi_2(x_n) + \cdots + c_m\varphi_m(x_n) = y_n$$

- ✓ n equations, m unknowns:
  - ✓ **m < n: over-determined system, least square regression**
  - ✓ **m = n: unique solution, interpolation**
  - ✓ m > n: under-determined system

Interpolation Polynomials

- ✓ Newton’s Divided Difference
- ✓ Lagrange Polynomials
- ✓ Gram’s polynomials (introduced earlier)
- ✓ Spline Interpolation: piecewise continuous, smoothing

Newton's Divided Difference

Triangular set of basis polynomials

$$
\begin{aligned}
\varphi_0(x) &= 1 \\
\varphi_1(x) &= (x - x_0) \\
\varphi_2(x) &= (x - x_0)(x - x_1) \\
\varphi_3(x) &= (x - x_0)(x - x_1)(x - x_2) \\
&\vdots \\
\varphi_i(x) &= (x - x_0)(x - x_1)\cdots(x - x_{i-1}) \\
&\vdots \\
\varphi_n(x) &= (x - x_0)(x - x_1) \quad \cdots \quad \cdots \quad (x - x_{n-1})
\end{aligned}
$$

Newton's Divided Difference

Consider a net of points $\{x_0, x_1, x_2, \cdots x_n\}$ and the corresponding function values as $\{f_0, f_1, f_2, \cdots f_n\}$

Newton's polynomial is:

$$p(x) = c_0 + c_1(x - x_0) + c_2(x - x_0)(x - x_1)$$
$$+c_3(x - x_0)(x - x_1)(x - x_2) \cdots \quad \cdots$$
$$+c_n(x - x_0)(x - x_1) \cdots (x - x_{n-1})$$

True function:

$$f(x) = p(x) + \frac{f^{n+1}(\xi)}{(n+1)!}(x - x_0)(x - x_1) \cdots (x - x_n)$$

for some $\xi \in \text{int}(x, x_0, x_1 \ \cdots x_n)$

Newton's Divided Difference

Newton's polynomial with the remainder term:

$$f(x) = c_0 + c_1(x - x_0) + c_2(x - x_0)(x - x_1)$$
$$+c_3(x - x_0)(x - x_1)(x - x_2)\cdots \quad \cdots$$
$$+c_n(x - x_0)(x - x_1)\cdots(x - x_{n-1})$$
$$+a(x)(x - x_0)(x - x_1)\cdots(x - x_n)$$

$$f(x_0) = f_0 = c_0$$

$$f[x_0, x] = \frac{f(x) - f(x_0)}{x - x_0} = c_1 + c_2(x - x_1)$$
$$+ c_3(x - x_1)(x - x_2)\cdots \quad \cdots$$
$$+ c_n(x - x_1)\cdots(x - x_{n-1})$$
$$+ a(x)(x - x_1)\cdots(x - x_n)$$

$$f[x_0, x_1] = \frac{f_1 - f_0}{x_1 - x_0} = c_1$$

Newton’s Divided Difference

$$f[x_0, x] = \frac{f(x) - f(x_0)}{x - x_0} = c_1 + c_2(x - x_1)$$
$$+ c_3(x - x_1)(x - x_2) \cdots \quad \cdots$$
$$+ c_n(x - x_1) \cdots (x - x_{n-1})$$
$$+ a(x)(x - x_1) \cdots (x - x_n)$$

$$f[x_0, x_1, x] = \frac{f[x_0, x] - f[x_0, x_1]}{x - x_1} = c_2 + c_3(x - x_2) + \cdots$$
$$+ c_n(x - x_2) \cdots (x - x_{n-1}) + a(x)(x - x_2) \cdots (x - x_n)$$

$$f[x_0, x_1, x_2] = \frac{f[x_0, x_2] - f[x_0, x_1]}{x_2 - x_1} = c_2$$

Newton's Divided Difference

$$f[x_0, x_1, x] = \frac{f[x_0, x] - f[x_0, x_1]}{x - x_1} = c_2 + c_3(x - x_2) + \cdots$$

$$+c_n(x - x_2)\cdots(x - x_{n-1}) + a(x)(x - x_2)\cdots(x - x_n)$$

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

$$= c_3 + c_4(x - x_3) + \cdots$$

$$+ c_n(x - x_3)\cdots(x - x_{n-1}) + a(x)(x - x_3)\cdots(x - x_n)$$

$$f[x_0, x_1, x_2, x_3] = \frac{f[x_0, x_1, x_3] - f[x_0, x_1, x_2]}{(x_3 - x_2)} = c_3$$

Newton's Divided Difference

$$f[x_0, x_1, \ldots x_{k-1}, x] = \frac{f[x_0,x_1,\ldots x_{k-2},x] - f[x_0,x_1,\ldots x_{k-2},x_{k-1}]}{(x - x_{k-1})} = c_k + c_{k+1}(x - x_k) + \cdots + c_n(x - x_k)\cdots(x - x_{n-1}) + a(x)(x - x_k)\cdots(x - x_n)$$

$$f[x_0, x_1, x_2, \cdots x_k] = \frac{f[x_0, x_1, \ldots x_{k-2}, x_k] - f[x_0, x_1, \ldots x_{k-2}, x_{k-1}]}{(x_k - x_{k-1})} = c_k$$

$$f[x_0, x_1, x_2, \cdots x_n] = c_n; \qquad f[x_0, x_1, x_2, \cdots x_n, x] = a(x)$$

Newton's polynomial without the remainder term:

$$p(x) = f_0 + \sum_{j=1}^{n} f\left[x_0, x_1, \cdots x_j\right](x - x_0)(x - x_1)\cdots\left(x - x_{j-1}\right)$$

Properties of Divided Differences

2nd Divided Difference: $f[x_k, x_{k-1}, x_{k-2}] = \frac{f[x_k, x_{k-1}] - f[x_{k-1}, x_{k-2}]}{x_k - x_{k-2}}$

$$a = \frac{f[x_k, x_{k-1}] - f[x_k, x_{k-2}]}{x_{k-1} - x_{k-2}}$$

$$= \frac{\frac{f_k - f_{k-1}}{x_k - x_{k-1}} - \frac{f_k - f_{k-2}}{x_k - x_{k-2}}}{x_{k-1} - x_{k-2}}$$

$$+x_{k-1}f_{k-1} - x_{k-1}f_{k-1}$$

$$= \frac{\cancel{x_k f_k} - x_k f_{k-1} - x_{k-2} f_k + x_{k-2} f_{k-1} - \cancel{x_k f_k} + x_k f_{k-2} + x_{k-1} f_k - x_{k-1} f_{k-2}}{(x_k - x_{k-1})(x_{k-1} - x_{k-2})(x_k - x_{k-2})}$$

$$= \frac{(x_{k-1} - x_{k-2}) f_k - (x_{k-1} - x_{k-2}) f_{k-1} - (x_k - x_{k-1}) f_{k-1} + (x_k - x_{k-1}) f_{k-2}}{(x_k - x_{k-1})(x_{k-1} - x_{k-2})(x_k - x_{k-2})}$$

$$= \frac{\frac{f_k - f_{k-1}}{x_k - x_{k-1}} - \frac{f_{k-1} - f_{k-2}}{x_{k-1} - x_{k-2}}}{x_k - x_{k-2}} = \frac{f[x_k, x_{k-1}] - f[x_{k-1}, x_{k-2}]}{x_k - x_{k-2}} = f[x_k, x_{k-1}, x_{k-2}]$$

Note: Properties of Divided Differences

✓ 1st Divided Difference:

$$f[x_k, x_{k-1}] = \frac{f_k - f_{k-1}}{x_k - x_{k-1}} = \frac{f_{k-1} - f_k}{x_{k-1} - x_k} = f[x_{k-1}, x_k]$$

✓ 2nd Divided Difference:

$$f[x_k, x_{k-1}, x_{k-2}] = \frac{f[x_k, x_{k-1}] - f[x_{k-1}, x_{k-2}]}{x_k - x_{k-2}} = \frac{f[x_k, x_{k-1}] - f[x_k, x_{k-2}]}{x_{k-1} - x_{k-2}} = \frac{f[x_k, x_{k-2}] - f[x_{k-1}, x_{k-2}]}{x_k - x_{k-1}}$$

$$f[x_k, x_{k-1}, x_{k-2}] = f[x_{k-1}, x_k, x_{k-2}] = f[x_{k-2}, x_{k-1}, x_k] = f[x_k, x_{k-2}, x_{k-1}]$$

We shall use these properties for the *Theory of Approximation*!

Newton's Divided Difference

Recursion Formula for Divided Difference:

(Recall: discussion during Muller's method)

The order of the points within the divided difference is immaterial. To see it generally, consider this:

$$f[x_0, x_1, \ldots x_k, x] = \frac{f[x_0, x_1, \ldots x_{k-1}, x] - f[x_0, x_1, \ldots x_{k-1}, x_k]}{(x - x_k)}$$

For generalization: replace the index zero with $(i + 1)$ evaluate the divided difference at $x = x_i$

$$f[x_i, x_{i+1}, \ldots x_k] = \frac{f[x_i, x_{i+1}, \ldots x_{k-1}] - f[x_{i+1}, \ldots x_{k-1}, x_k]}{(x_i - x_k)}$$

Newton's Divided Difference

$$f[x_i, x_{i+1}, \dots x_k] = \frac{f[x_i, x_{i+1}, \dots x_{k-1}] - f[x_{i+1}, \dots x_{k-1}, x_k]}{(x_i - x_k)}$$

$$= \frac{f[x_{i+1}, \dots x_{k-1}, x_k] - f[x_i, x_{i+1}, \dots x_{k-1}]}{(x_k - x_i)}$$

or

$$f[x_k, x_{k-1}, \dots x_i] = \frac{f[x_{k-1}, \dots x_{i+1}, x_i] - f[x_k, x_{k-1}, \dots x_{i+1}]}{(x_i - x_k)}$$

$$\boxed{= \frac{f[x_k, x_{k-1}, \dots x_{i+1}] - f[x_{k-1}, \dots x_{i+1}, x_i]}{(x_k - x_i)}}$$

Newton's Divided Difference

Examples:

$$f[x_k, x_{k-1}, \dots x_i] = \frac{f[x_k, x_{k-1}, \dots x_{i+1}] - f[x_{k-1}, \dots x_{i+1}, x_i]}{(x_k - x_i)}$$

$$k = 1, i = 0\colon f[x_1, x_0] = \frac{f(x_1) - f(x_0)}{(x_1 - x_0)}$$

$$k = 2, i = 1\colon f[x_2, x_1] = \frac{f(x_2) - f(x_1)}{(x_2 - x_1)}$$

$$k = 2, i = 0\colon f[x_2, x_1, x_0] = \frac{f[x_2, x_1] - f[x_1, x_0]}{(x_2 - x_0)}$$

$$k = 3, i = 1\colon f[x_3, x_2, x_1] = \frac{f[x_3, x_2] - f[x_2, x_1]}{(x_3 - x_1)}$$

$$k = 3, i = 0\colon f[x_3, x_2, x_1, x_0] = \frac{f[x_3, x_2, x_1] - f[x_2, x_1, x_0]}{(x_3 - x_0)}$$

Newton’s Divided Difference:Example

$$
\begin{tikzcd}
x_0\,,\, f(x_0) \arrow[r] & f[x_1, x_0] \arrow[r] & f[x_2, x_1, x_0] \arrow[r] & f[x_3, x_2, x_1, x_0] \\
x_1\,,\, f(x_1) \arrow[r] \arrow[ur] & f[x_2, x_1] \arrow[r] \arrow[ur] & f[x_3, x_2, x_1] \arrow[ur] & \\
x_2\,,\, f(x_2) \arrow[r] \arrow[ur] & f[x_3, x_2] \arrow[ur] & & \\
x_3\,,\, f(x_3) \arrow[ur] & & &
\end{tikzcd}
$$

$p(x) = f(x_0) + f[x_1, x_0](x - x_0) + f[x_2, x_1, x_0](x - x_0)(x - x_1) +$
$f[x_3, x_2, x_1, x_0](x - x_0)(x - x_1)(x - x_2)$

Newton’s Divided Difference:Example

1 , -48	2	9	2
2 , -46	29	17	
4, 12	80		
5 , 92			

$$p(x) = -48 + 2(x-1) + 9(x-1)(x-2) + 2(x-1)(x-2)(x-4)$$
$$= 2x^3 - 5x^2 + 3x - 48$$

Newton's Divided Difference:Error Estimate

Recall the Newton's polynomial with the remainder:

$$f(x) = p(x) + \frac{f^{n+1}(\xi)}{(n+1)!}(x - x_0)(x - x_1)\cdots(x - x_n)$$

$$\text{for some } \xi \in \text{int}(x, x_0, x_1 \ \cdots x_n)$$

$$f(x) = p(x) + a(x)(x - x_0)(x - x_1)\cdots(x - x_n)$$

We derived:

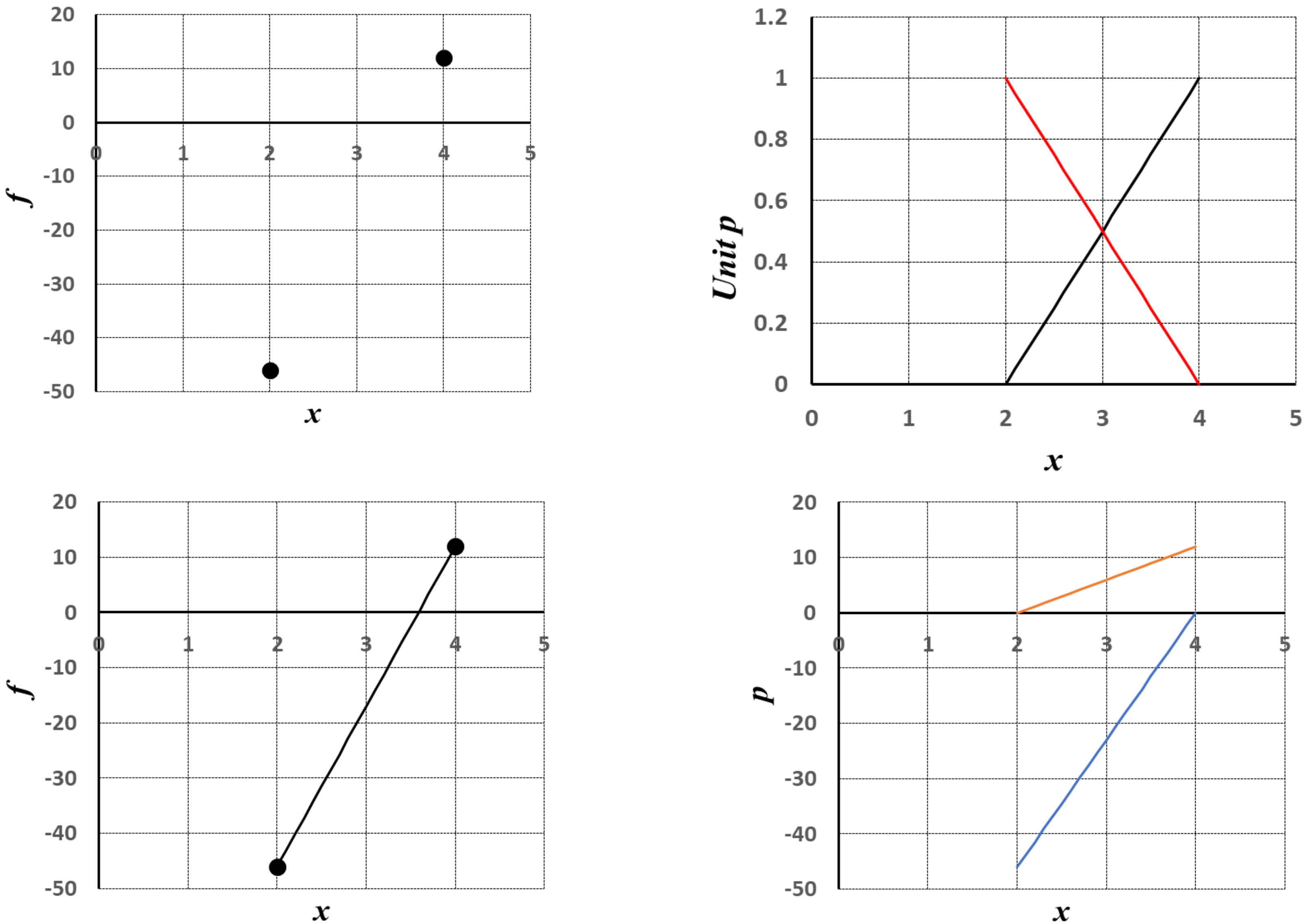
$$f[x_0, x_1, x_2, \cdots x_n, x] = a(x)$$

If an extra-data $\{x_{n+1}, f(x_{n+1})\}$ is available, it is possible to make an approximate estimate of the error as:

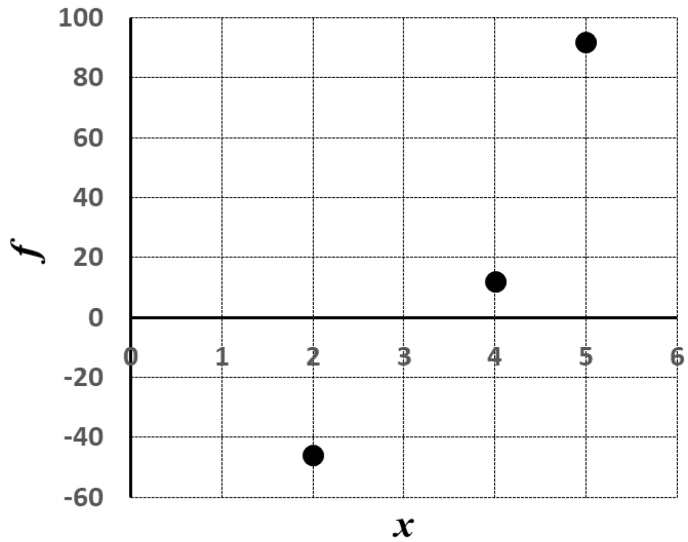
$f[x_0, x_1, x_2, \cdots x_n, x_{n+1}] = a(x_{n+1}) \approx a(x)$ and the error (E) as:

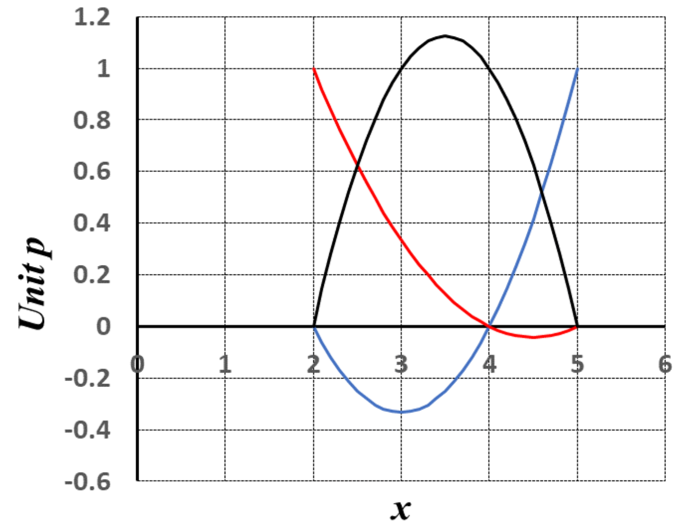
$$E \approx f[x_0, x_1, x_2, \cdots x_n, x_{n+1}](x - x_0)(x - x_1)\cdots(x - x_n)$$

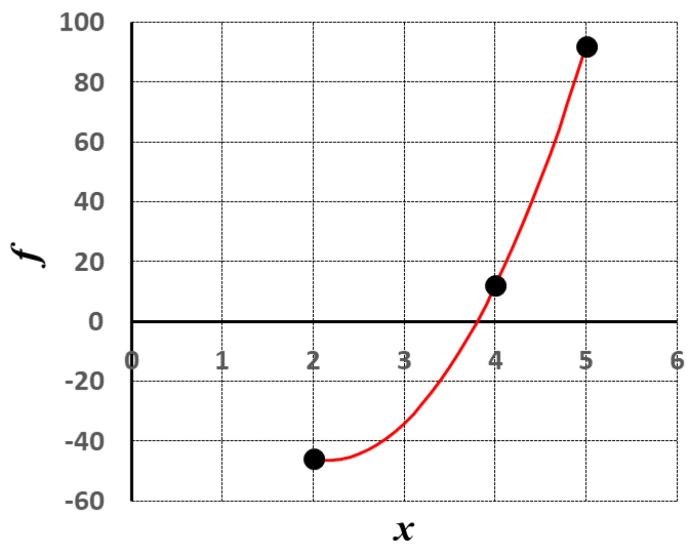
Lagrange Polynomials: Linear

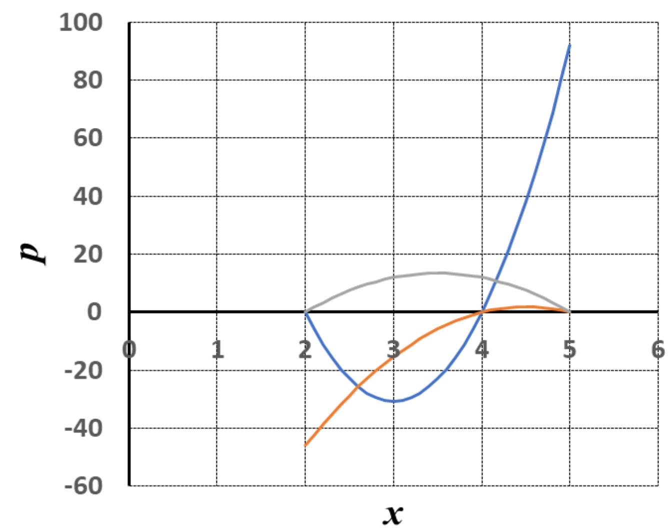


Lagrange Polynomials: Quadratic









Lagrange Polynomials

Unit linear polynomials for two nodes: $\{x_0, x_1\}$

$$\delta_0(x) = \frac{x - x_1}{x_0 - x_1}; \quad \delta_1(x) = \frac{x - x_0}{x_1 - x_0}$$

Unit quadratic for three nodes: $\{x_0, x_1, x_2\}$

$$\delta_0(x) = \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)}; \quad \delta_1(x) = \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)}$$

$$\delta_2(x) = \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)}$$

Polynomials of order *n* for the mesh of nodes $\{x_0, x_1, x_2, \cdots x_n\}$

$$\delta_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^{n} \frac{x - x_j}{x_i - x_j} \qquad \delta_i(x_j) = \begin{cases} 0 & for \quad j \neq i \\ 1 & for \quad j = i \end{cases}$$

Lagrange Polynomials

Polynomials to be fitted to a mesh of nodes $\{x_0, x_1, x_2, \cdots x_n\}$ and the corresponding function values as $\{f_0, f_1, f_2, \cdots f_n\}$

Lagrange polynomial is:

$$p(x) = \sum_{i=0}^{n} f_i \delta_i(x) \qquad \delta_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^{n} \frac{x - x_j}{x_i - x_j}$$

Lagrange Polynomial: Example

Write the cubic polynomial using Lagrange polynomials that passes through the following four points: (1 , -48), (2 , -46), (4, 12), (5 , 92)?

$$p(x) = (-48)\frac{(x-2)(x-4)(x-5)}{(1-2)(1-4)(1-5)} + (-46)\frac{(x-1)(x-4)(x-5)}{(2-1)(2-4)(2-5)} + 12\frac{(x-1)(x-2)(x-5)}{(4-1)(4-2)(4-5)} + 92\frac{(x-1)(x-2)(x-4)}{(5-1)(5-2)(5-4)}$$

$$= 4(x-2)(x-4)(x-5) - \frac{23}{3}(x-1)(x-4)(x-5) - 2(x-1)(x-2)(x-5) + \frac{23}{3}(x-1)(x-2)(x-4) = 2x^3 - 5x^2 + 3x - 48$$

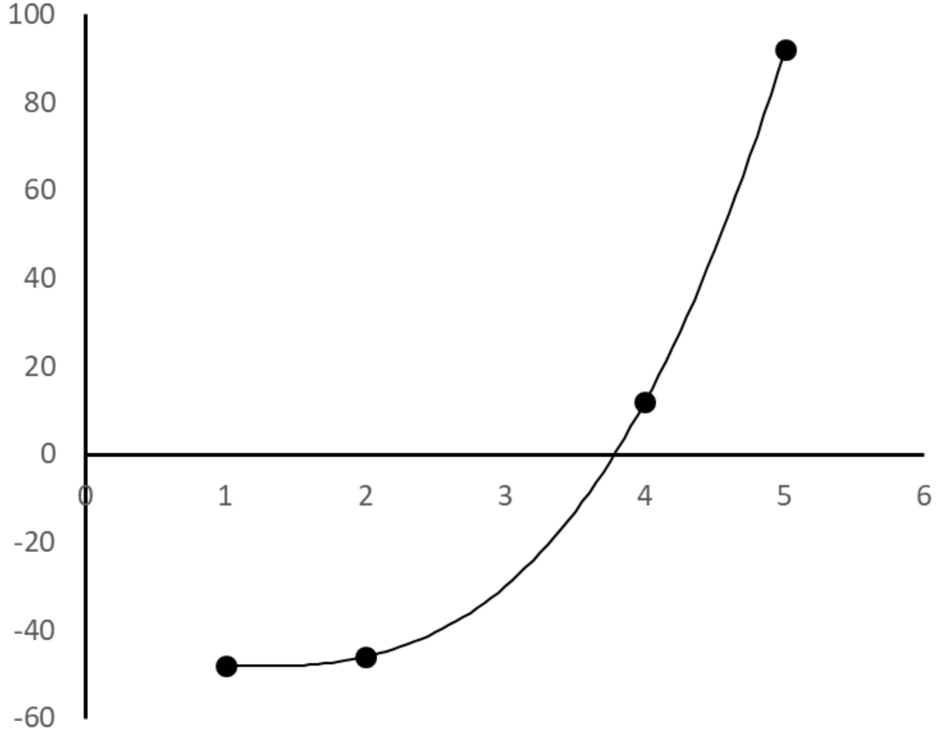
Recall Newton's polynomial through the same set of points:

$$p(x) = -48 + 2(x-1) + 9(x-1)(x-2) + 2(x-1)(x-2)(x-4)$$
$$= 2x^3 - 5x^2 + 3x - 48$$

Example Fit

Fit an interpolation polynomial through the following four points: (1 , -48), (2 , -46), (4, 12), (5 , 92)

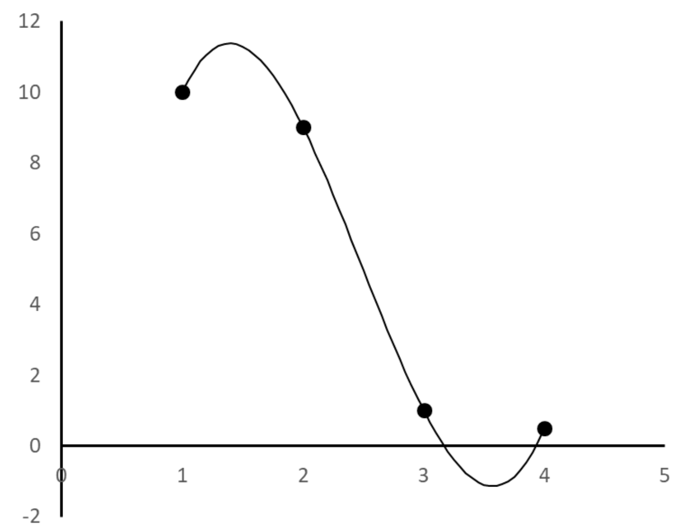
$p(x) = 2x^3 - 5x^2 + 3x - 48$



Interpolation: Example2

Fit an interpolation polynomial through the following four points: (1 , 10), (2 , 9), (3, 1), (4 , 0.5)

1 , 10	-1	-3.5	2.416667
2 , 9	-8	3.75	
3, 1	-0.5		
4 , 0.5			



$$p(x) = 10 - (x-1) - 3.5(x-1)(x-2) + 2.41667(x-1)(x-2)(x-3)$$

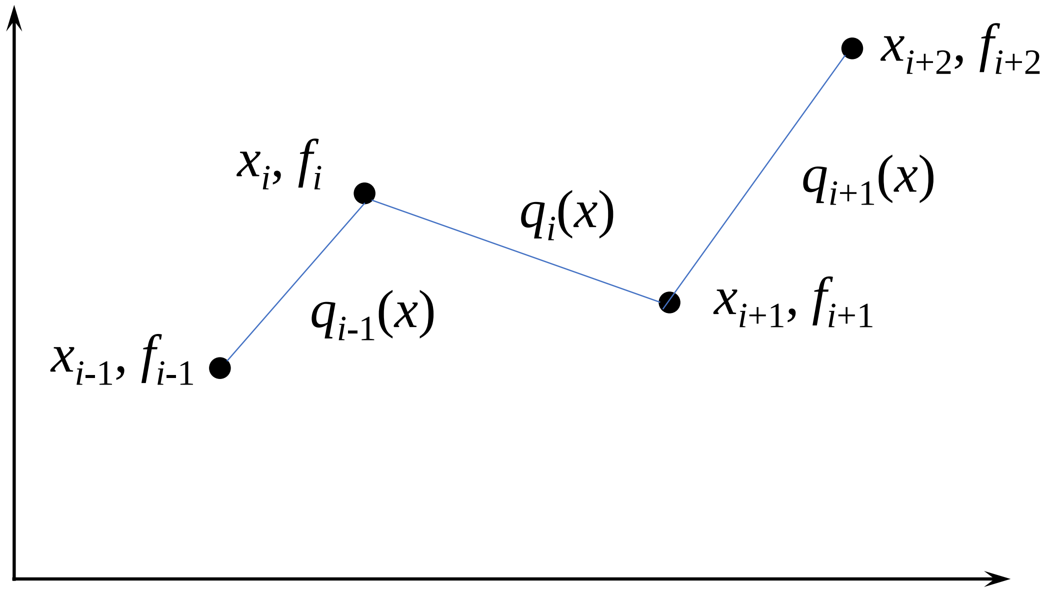
Single Interpolation Polynomial

- ✓ If a single polynomial is fitted through a given set of points, the polynomial is always the same, irrespective of the basis used.
- ✓ Computation involved in obtaining the polynomial may vary depending on the basis.
- ✓ Since the polynomials are same, the error in approximation may be obtained using the remainder term of the Newton's polynomial for all cases.
- ✓ So far, we have fitted one polynomial through all the n points.
- ✓ For a large number of points, this may lead to oscillations between the points, especially at the end intervals for the equally spaced data.
- ✓ Next we fit *piecewise continuous polynomials* or *splines* to the data!

Spline Interpolation

- ✓ Given: $(n + 1)$ observations or data pairs $[(x_0, f_0), (x_1, f_1), (x_2, f_2) \ldots (x_n, f_n)]$
- ✓ This gives a mesh of *nodes* $\{x_0, x_1, x_2, \cdots x_n\}$ on the independent variable and the corresponding function values as $\{f_0, f_1, f_2, \cdots f_n\}$
- ✓ Goal: fit an independent polynomial in each interval (between two points) with certain continuity requirements at the nodes.
  - ✓ *Linear spline*: continuity in function values, C^0 continuity
  - ✓ *Quadratic spline*: continuity in function values and 1st derivatives, C^1 continuity
  - ✓ *Cubic spline:* continuity in function values, 1st and 2nd derivatives, C^2 continuity
- ✓ Denote for node i or at x_i: functional value f_i, first derivative u_i, second derivative v_i

Spline Interpolation: Linear



$q_i(x_i) = f_i$

$q_{i-1}(x_i) = q_i(x_i)$

- ✓ A straight line in each interval: $(n+1)$ points, n straight lines, $2n$ unknowns
- ✓ Available conditions: $(n+1)$ function values, $(n-1)$ function continuity conditions
- ✓ Straight lines can be uniquely estimated!

Spline Interpolation: Linear

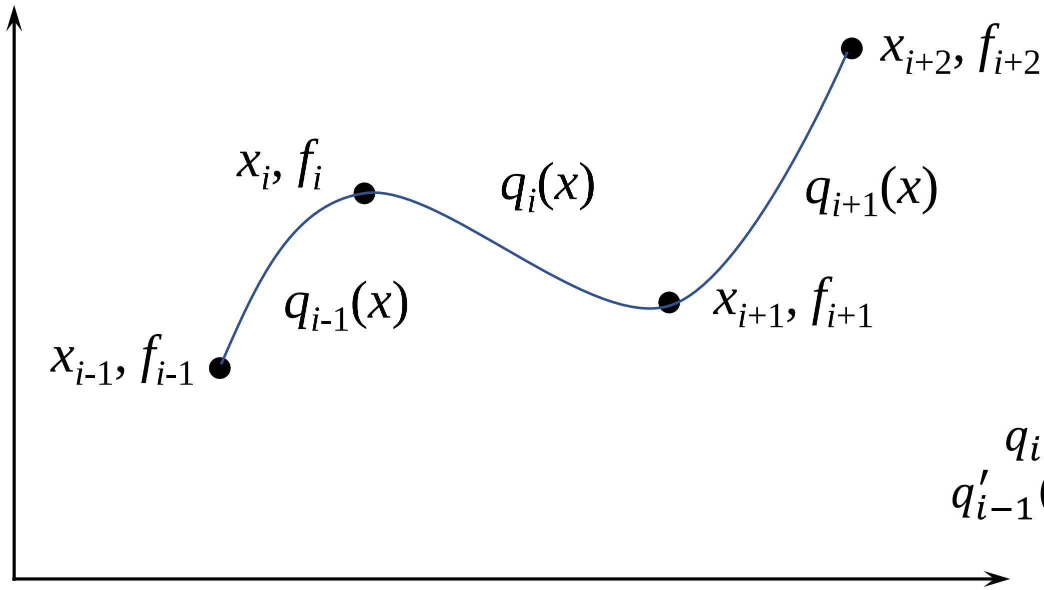
- ✓ Let us denote the Linear Spline in the interval $\{x_i, x_{i+1}\}$ as $q_i(x)$
- ✓ We know: $q_i(x_i) = f(x_i) = f_i$ and $q_i(x_{i+1}) = f(x_{i+1}) = f_{i+1}$
- ✓ Therefore, the Linear Spline in the interval $\{x_i, x_{i+1}\}$:

$$q_i(x) = f_i \frac{x-x_{i+1}}{x_i-x_{i+1}} + f_{i+1} \frac{x-x_i}{x_{i+1}-x_i}$$, the Lagrange polynomial

- ✓ Let us denote the length of the i^{th} interval $\{x_i, x_{i+1}\}$ as $h_i = x_{i+1} - x_i$
- ✓ The linear spline at the i^{th} interval is then given by:

$$q_i(x) = \frac{f_{i+1}}{h_i}(x - x_i) - \frac{f_i}{h_i}(x - x_{i+1}) = f[x_{i+1}, x_i](x - x_i) + f_i$$

Spline Interpolation: Quadratic



$$q_i(x_i) = f_i$$
$$q_{i-1}(x_i) = q_i(x_i)$$
$$q'_{i-1}(x_i) = u_i = q'_i(x_i)$$

- ✓ A quadratic polynomial in each interval: (n+1) points, n quadratic polynomials, 3n unknowns
- ✓ Available conditions: (n + 1) function values, (n - 1) function continuity and (n - 1) 1st derivative continuity conditions, total 3n-1 conditions.
- ✓ **1 free condition to be chosen by the user!**

Spline Interpolation: Quadratic

- ✓ Quadratic Spline in the interval $\{x_i, x_{i+1}\}$: $q_i(x)$
- ✓ $q_i'(x)$ is a set of linear splines.
- ✓ Let us denote the derivative (u) of the function at the i^{th} node as

$$f'(x_i) = u(x_i) = u_i$$

- ✓ Therefore: $q_i'(x_i) = u(x_i) = u_i$ and $q_i'(x_{i+1}) = u(x_{i+1}) = u_{i+1}$
- ✓ We may write:

$$q_i'(x) = \frac{u_{i+1}}{h_i}(x - x_i) - \frac{u_i}{h_i}(x - x_{i+1})$$

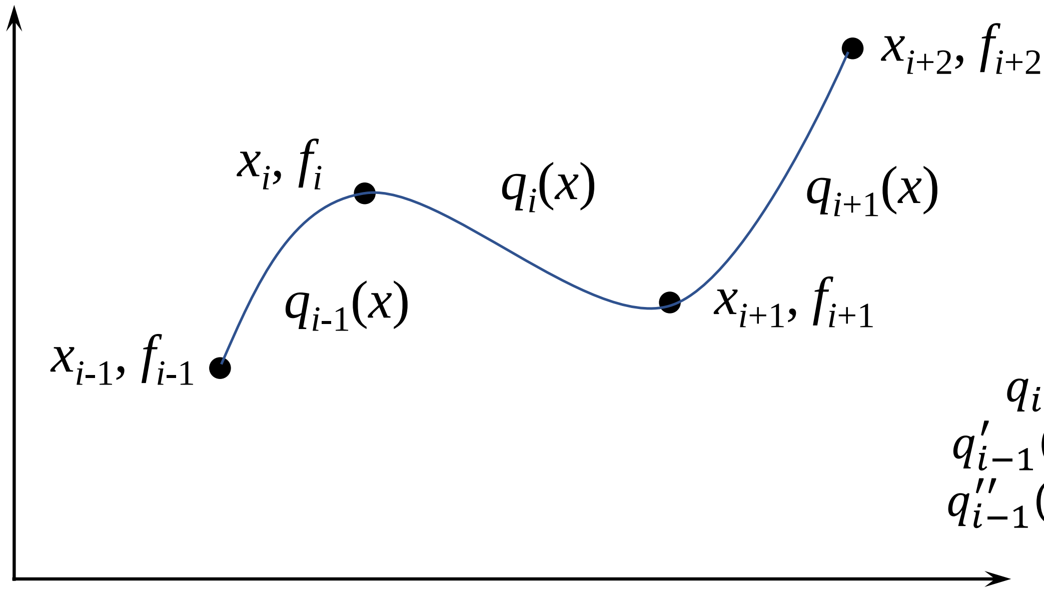
$$q_i(x) = \frac{u_{i+1}}{2h_i}(x - x_i)^2 - \frac{u_i}{2h_i}(x - x_{i+1})^2 + c$$

$$q_i(x_i) = f_i \qquad \Longrightarrow \qquad c = f_i + \frac{u_i h_i}{2}$$

$$q_i(x_{i+1}) = f_{i+1} \qquad \Longrightarrow \qquad u_{i+1} = 2f[x_{i+1}, x_i] - u_i$$

- ✓ Recall one free condition, either u_0 or u_n is specified!

Spline Interpolation: Cubic



$$q_i(x_i) = f_i$$
$$q_{i-1}(x_i) = q_i(x_i)$$
$$q'_{i-1}(x_i) = u_i = q'_i(x_i)$$
$$q''_{i-1}(x_i) = v_i = q''_i(x_i)$$

- ✓ A cubic polynomial in each interval: (n+1) points, n cubic polynomials, 4n unknowns
- ✓ Available conditions: (n + 1) function values, (n - 1) function continuity, (n - 1) 1st derivative continuity conditions and (n - 1) 2nd derivative continuity conditions, total 4n - 2 conditions.
- ✓ **2 free conditions to be chosen by the user!**

Spline Interpolation: Cubic

- ✓ Cubic Spline in the interval $\{x_i, x_{i+1}\}$: $q_i(x)$
- ✓ $q_i''(x)$ is a set of linear splines.
- ✓ Let us denote the 2nd derivative (v) of the function at the ith node as

$$f''(x_i) = v(x_i) = v_i$$

- ✓ Therefore: $q_i''(x_i) = v(x_i) = v_i$ and $q_i''(x_{i+1}) = v(x_{i+1}) = v_{i+1}$
- ✓ We may write:

$$q_i''(x) = \frac{v_{i+1}}{h_i}(x - x_i) - \frac{v_i}{h_i}(x - x_{i+1})$$

$$q_i(x) = \frac{v_{i+1}}{6h_i}(x - x_i)^3 - \frac{v_i}{6h_i}(x - x_{i+1})^3 + c_1 x + c_2$$

$$q_i(x_i) = f_i = \frac{v_i h_i^2}{6} + c_1 x_i + c_2$$

$$q_i(x_{i+1}) = f_{i+1} = \frac{v_{i+1} h_i^2}{6} + c_1 x_{i+1} + c_2$$

Spline Interpolation: Cubic

$$q_i(x) = \frac{v_{i+1}}{6h_i}(x - x_i)^3 - \frac{v_i}{6h_i}(x - x_{i+1})^3 + c_1 x + c_2$$

$$q_i(x_i) = f_i = \frac{v_i h_i^2}{6} + c_1 x_i + c_2; \quad q_i(x_{i+1}) = f_{i+1} = \frac{v_{i+1} h_i^2}{6} + c_1 x_{i+1} + c_2$$

$$c_1 = f[x_{i+1}, x_i] - \frac{h_i}{6}(v_{i+1} - v_i) = \frac{f_{i+1}}{h_i} - \frac{f_i}{h_i} + \frac{h_i}{6} v_i - \frac{h_i}{6} v_{i+1}$$

$$c_2 = -\frac{f_{i+1}}{h_i} x_i + \frac{f_i}{h_i} x_{i+1} - \frac{h_i}{6} v_i x_{i+1} + \frac{h_i}{6} v_{i+1} x_i$$

$$\begin{aligned} &q_i(x) \\ &= \frac{v_{i+1}}{6}\left[\frac{(x - x_i)^3}{h_i} - h_i(x - x_i)\right] + \frac{v_i}{6}\left[-\frac{(x - x_{i+1})^3}{h_i} + h_i(x - x_{i+1})\right] \\ &+ \frac{f_{i+1}}{h_i}(x - x_i) - \frac{f_i}{h_i}(x - x_{i+1}) \end{aligned}$$

We need to estimate (n + 1) unknown v_i. We have (n – 1) conditions from the continuity of the first derivative.

Spline Interpolation: Cubic

$$q_i'(x_i) = q_{i-1}'(x_i); \qquad i = 1, 2, 3, \cdots n-1$$

$$q_i(x) = \frac{v_{i+1}}{6}\left[\frac{(x-x_i)^3}{h_i} - h_i(x-x_i)\right] + \frac{v_i}{6}\left[-\frac{(x-x_{i+1})^3}{h_i} + h_i(x-x_{i+1})\right] + \frac{f_{i+1}}{h_i}(x-x_i) - \frac{f_i}{h_i}(x-x_{i+1})$$

$$q_i'(x) = \frac{v_{i+1}}{6}\left[\frac{3(x-x_i)^2}{h_i} - h_i\right] + \frac{v_i}{6}\left[-\frac{3(x-x_{i+1})^2}{h_i} + h_i\right] + f[x_{i+1}, x_i]$$

$$q_{i-1}'(x) = \frac{v_i}{6}\left[\frac{3(x-x_{i-1})^2}{h_{i-1}} - h_{i-1}\right] + \frac{v_{i-1}}{6}\left[-\frac{3(x-x_i)^2}{h_{i-1}} + h_{i-1}\right] + f[x_i, x_{i-1}]$$

$$-\frac{h_i}{6}v_{i+1} - \frac{h_i}{3}v_i + f[x_{i+1}, x_i] = \frac{h_{i-1}}{3}v_i + \frac{h_{i-1}}{6}v_{i-1} + f[x_i, x_{i-1}]$$

$$-\frac{h_{i-1}}{6}v_{i-1} - \left(\frac{h_i + h_{i-1}}{3}\right)v_i - \frac{h_i}{6}v_{i+1} = f[x_i, x_{i-1}] - f[x_{i+1}, x_i]$$

$$\frac{h_{i-1}}{6}v_{i-1} + \left(\frac{h_i + h_{i-1}}{3}\right)v_i + \frac{h_i}{6}v_{i+1} = f[x_{i+1}, x_i] - f[x_i, x_{i-1}]$$

Spline Interpolation: Cubic

$$h_{i-1}v_{i-1} + 2(h_i + h_{i-1})v_i + h_i v_{i+1} = 6f[x_{i+1}, x_i] - 6f[x_i, x_{i-1}]$$

- ✓ $i = 1, 2, 3, \cdots n-1$. So, (n – 1) equations, (n + 1) unknowns, two conditions have to be provided by the users. They decide the type of cubic splines
- ✓ *Natural Spline:*

$$v_0 = v_n = 0$$

- ✓ *Parabolic Runout:*

$$v_0 = v_1 \qquad \text{and} \qquad v_{n-1} = v_n$$

- ✓ *Not-a-knot:*

$$q_0(x) = q_1(x) \qquad \Longrightarrow \qquad \frac{v_1 - v_0}{h_0} = \frac{v_2 - v_1}{h_1}$$

$$q_{n-2}(x) = q_{n-1}(x) \qquad \Longrightarrow \qquad \frac{v_{n-1} - v_{n-2}}{h_{n-2}} = \frac{v_n - v_{n-1}}{h_{n-1}}$$

- ✓ *Periodic:*

$$v_0 = v_{n-1} \qquad \text{and} \qquad v_1 = v_n$$

Spline Interpolation: Cubic

$$h_{i-1}v_{i-1} + 2(h_i + h_{i-1})v_i + h_i v_{i+1} = 6f[x_{i+1}, x_i] - 6f[x_i, x_{i-1}]$$

- ✓ $i = 1, 2, 3, \cdots n-1.$ So, (n – 1) equations, (n + 1) unknowns, two conditions have to be provided by the users. They decide the type of cubic splines

- ✓ *Clamped Spline:*

$$u_0 = q_0'(x_0) = \alpha \qquad \text{and} \qquad u_n = q_{n-1}'(x_n) = \beta$$

$$q_0'(x) = \frac{v_1}{6}\left[\frac{3(x-x_0)^2}{h_0} - h_0\right] + \frac{v_0}{6}\left[-\frac{3(x-x_1)^2}{h_0} + h_0\right] + f[x_1, x_0]$$

$$q_0'(x_0) = -\frac{v_1 h_0}{6} - \frac{v_0 h_0}{3} + f[x_1, x_0] = \alpha \quad \Longrightarrow \quad 2v_0 + v_1 = \frac{6}{h_0}\left(f[x_1, x_0] - \alpha\right)$$

$$q_{n-1}'(x) = \frac{v_n}{6}\left[\frac{3(x-x_{n-1})^2}{h_{n-1}} - h_{n-1}\right] + \frac{v_{n-1}}{6}\left[-\frac{3(x-x_n)^2}{h_{n-1}} + h_{n-1}\right] + f[x_n, x_{n-1}]$$

$$q_{n-1}'(x_n) = \frac{v_n h_{n-1}}{3} + \frac{v_{n-1} h_{n-1}}{6} + f[x_n, x_{n-1}] = \beta$$

$$\Longrightarrow \quad 2v_n + v_{n-1} = \frac{6}{h_{n-1}}\left(\beta - f[x_n, x_{n-1}]\right)$$