

## ① Round-off error

① Have a well-conditioned matrix, but the solution procedure is bad.

→ We can suggest improvement in algorithm.

② Have an ill-conditioned matrix which is close to being singular.  
→ no hope if it is due to round-off error.

\* Computers store floating point numbers as product of a fractional part  $f$  with  $d$  digits and exponential part  $10^s$ .  $\downarrow 0.1 \leq |f| < 1$

ex) 87.648  $\rightarrow$   $\underbrace{0.87648}_{f} \times 10^{\underline{2}} \rightarrow s \quad d=5$

100  $\rightarrow$   $\underbrace{0.1}_{f} \times 10^3$

rounding is performed on  $f$   
→ round-off error.

Ex)  $\begin{cases} 0.01x_1 + x_2 = 1 \\ 1x_1 - x_2 = 0 \end{cases} \rightarrow \text{exact sol. } x_1 = x_2 = \frac{1}{1.01} = 0.990099\ldots$

Do this example problem w/ a computer that carries 2 significant figures,  $\rightarrow x_1 = x_2 = 0.99$   
best sol.

GE :  $\begin{cases} 0.01x_1 + x_2 = 1 \\ 1x_1 - x_2 = 0 \end{cases}$

$$\begin{array}{rcl} & & \xrightarrow{\quad} \\ \begin{array}{rcl} 0 & - 101x_2 & = -100 \\ \hline & -0.101x_1 & \xrightarrow{3} -0.10x_1 \end{array} & \xrightarrow{\quad} x_2 = 1 \\ & & \xrightarrow{\quad} x_1 = 0 \end{array}$$

wrong!

But do the same prob.

$$\begin{array}{l} \begin{array}{l} \text{pivot} \\ \xrightarrow{1} \begin{cases} 1x_1 - x_2 = 0 \\ 0.01x_1 + x_2 = 1 \end{cases} \end{array} \xrightarrow{\text{GE}} \begin{cases} x_1 - x_2 = 0 \\ \underline{1.01x_2 = 1} \end{cases} \\ \xrightarrow{0.101x_1 \Rightarrow 0.10x_1} \begin{array}{l} x_1 = 1 \\ x_2 = 1 \end{array} \end{array}$$

choice of pivot is very important.  
accurate!

By row changes, make the pivot the largest element in the column.

ex)  $\begin{cases} |x_1| + 100x_2 = 100 \\ |x_1| - x_2 = 0 \end{cases}$   $\xrightarrow{\text{GE}}$   $\xrightarrow{\text{BS}}$   $\begin{cases} x_1 = 0 \\ x_2 = 1 \end{cases}$  ) wrong.

$\Rightarrow$  we need scaling.

Normalize each eq. in advance s.t. the largest element in each row is 1.

Scaling  $\Rightarrow \begin{cases} 0.01x_1 + x_2 = 1 \\ 1x_1 - x_2 = 0 \end{cases} \xrightarrow{\text{pivot}} \begin{cases} x_1 - x_2 = 0 \\ 0.01x_1 + x_2 = 1 \end{cases}$

$\rightarrow \text{GE} \rightarrow \text{BS} \rightarrow x_1 = x_2 = 1$

② example of ill-conditioned matrix

$$\begin{cases} x_1 + x_2 = 2 \\ x_1 + 1.0001x_2 = 2 \end{cases} \xrightarrow{\text{exact sol!}} \begin{cases} x_1 = 2 \\ x_2 = 0 \end{cases}$$

$$A\vec{x} = b$$

↑  
↑  
measurement

perturb the RHS

$$\begin{cases} x_1 + x_2 = 2 \\ x_1 + 1.0001x_2 = 2.0001 \end{cases} \xrightarrow{\text{exact sol.}} \begin{cases} x_1 = 1 \\ x_2 = 1 \end{cases}$$

slight change in b  $\rightarrow$  large change in x.

$\rightarrow$  No numerical method can avoid this sensitivity to small perturbations.

Q : Given  $Ax=b$ , what is the change in x w.r.t. a change in the parameter of the system b?

$Ax=b$   
error in b,  $\delta b \rightarrow x + \delta x$ . find  $\delta x$ !

$$A(x + \delta x) = b + \delta b \quad \& \quad Ax = b$$

$$\rightarrow A\delta x = \delta b \rightarrow \delta x = A^{-1}\delta b$$

$$\|\delta x\| = \|A^{-1}\delta b\| \leq \|A^{-1}\| \|\delta b\| \quad (\text{Schwartz inequality})$$

$$Ax = b \rightarrow \|b\| = \|Ax\| \leq \|A\| \|x\| \quad \|A\| : \text{norm}$$

$$\|A\|^2 = \lambda_{\max}(A^T A)$$

$$\Rightarrow \frac{\|\delta x\|}{\|A\| \|x\|} \leq \frac{\|A^{-1}\| \|\delta b\|}{\|b\|}$$

$$\rightarrow \boxed{\frac{\|\delta x\|}{\|x\|} \leq \underbrace{\|A\| \|A^{-1}\|}_{\text{amplification factor}} \frac{\|\delta b\|}{\|b\|}}$$

amplification factor

condition number of  $A = \frac{\lambda_{\max}}{\lambda_{\min}}$

$$\begin{cases} x_1 + x_2 = 2 \\ x_1 + 1.0001x_2 = 2 \end{cases} \rightarrow A = \begin{pmatrix} 1 & 1 \\ 1 & 1.0001 \end{pmatrix} \rightarrow \|A\| = 2.00005 \approx 2$$

$$A^{-1} = \begin{pmatrix} 1000 & -10000 \\ -10000 & 10000 \end{pmatrix} \rightarrow \|A^{-1}\| = 20,000$$

$$\Rightarrow \frac{\|\delta x\|}{\|x\|} \leq 2 \times 20,000 \frac{\|\delta b\|}{\|b\|} = \underbrace{40,000}_{\gg 1} \frac{\|\delta b\|}{\|b\|}$$

matrix  $A$  is stiff or ill-conditioned.

If the condition number times the order of round-off accuracy of machine is order of 1,  
 $\Rightarrow$  concerned.

ex) condition number =  $10^X$

order of round-off error accuracy =  $10^{-Y}$

## ⑥ Cayley - Hamilton theorem

Every matrix satisfies its own characteristic eq.

$$A_{n \times n} : P(\lambda) = \det(A - \lambda I) = 0$$

$$\rightarrow \lambda^n + c_1 \lambda^{n-1} + c_2 \lambda^{n-2} + \dots + c_{n-1} \lambda + c_n = 0$$

$$\Rightarrow A^n + c_1 A^{n-1} + c_2 A^{n-2} + \dots + c_{n-1} A + c_n I = 0 \quad \text{ $n$ : integer}$$

$$P(A) = 0$$

$$T = f(D) = d_0 I + d_1 D + d_2 D^2 + d_3 D^3 + d_4 D^4 + \dots$$

- Any power of an  $n \times n$  matrix can be represented as a polynomial of degree up to  $n-1$ .

Ex)  $A = \begin{pmatrix} 1 & 2 \\ 0 & 8 \end{pmatrix}$ ,  $\sqrt{A} = ?$      $\lambda_1 = 1$ ,  $\lambda_2 = 8$

$$\sqrt{A} = d_0 I + d_1 A \rightarrow \sqrt{\lambda_1} = d_0 + d_1 \lambda_1 \rightarrow d_0 = \frac{2}{3}$$

$$\sqrt{\lambda_2} = d_0 + d_1 \lambda_2 \quad d_1 = \frac{1}{3}$$

$$\Rightarrow \sqrt{A} = \frac{2}{3} I + \frac{1}{3} A = \begin{pmatrix} 1 & \frac{2}{3} \\ 0 & 2 \end{pmatrix}$$

- Any function of a matrix can be represented as a polynomial of degree up to  $n-1$ .

Ex)  $A = \begin{pmatrix} \pi & 3\pi \\ 2\pi & 2\pi \end{pmatrix}$ ,  $\cos A = ?$      $\lambda_1 = 4\pi$ ,  $\lambda_2 = -\pi$

$$\cos A = d_0 I + d_1 A \rightarrow \cos \lambda_1 = d_0 + d_1 \lambda_1 \rightarrow d_0 = -\frac{3}{5}$$

$$\cos \lambda_2 = d_0 + d_1 \lambda_2 \quad d_1 = \frac{2}{5\pi}$$

$$\cos A = -\frac{3}{5}I + \frac{2}{5}\pi A = \begin{pmatrix} -\frac{1}{5} & \frac{6}{5} \\ \frac{6}{5} & \frac{1}{5} \end{pmatrix}$$

ex)  $A = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$ ,  $\sqrt{A} = ?$      $\lambda_1 = \lambda_2 = 2$  double root

$$\sqrt{A} = \alpha_0 I + \alpha_1 A \rightarrow \sqrt{\lambda}_1 = \alpha_0 + \alpha_1 \lambda_1$$

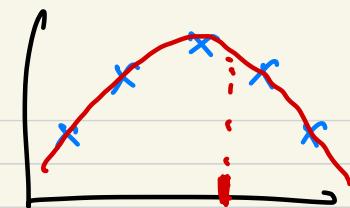
$$P(\lambda) = (\lambda - \lambda_1)^2 (\lambda - \lambda_2) \dots (\lambda - \lambda_{n-1})$$

$$P' \Big|_{\lambda=\lambda_1} = 2(\lambda - \lambda_1) \dots + \dots + \dots = 0$$

$$\left. \begin{array}{l} \sqrt{\lambda}_1 = \alpha_0 + \alpha_1 \lambda_1 \\ \rightarrow \frac{1}{2} \lambda_1^{-\frac{1}{2}} = \alpha_1 \end{array} \right\} \rightarrow \frac{\alpha_0}{\alpha_1} \rightarrow \sqrt{A} = \begin{pmatrix} \sqrt{2} & 0 \\ 0 & \sqrt{2} \end{pmatrix}$$

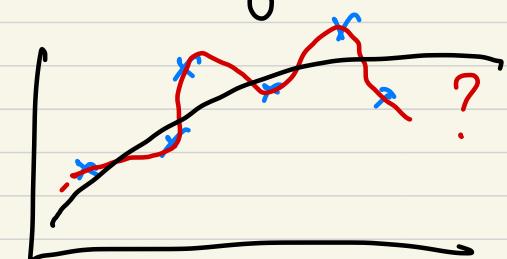
# Ch 1. Interpolation

Discrete data  $(x_i, y_i), i = 0, 1, 2, \dots, n$



- Find value of  $y$  at a point between two data pts.
- Fix a smooth curve through the data.

If a data is from a crude experiment with some uncertainty, it is best to use the method of least square errors.



- Polynomial interpolation  $(x_i, y_i), i = 0, 1, 2, \dots, n$

$n+1$  pts,

Fix a polynomial of degree  $n$  through  $n+1$  pts,

$$P = a_0 + a_1 x + a_2 x^2 + \dots + a_n x^n$$

Find the coeffs.  $a_0, a_1, \dots, a_n$  by passing the polynomial going through the data pts.

$$y_i = \hat{a}_0 + \hat{a}_1 x_i + \hat{a}_2 x_i^2 + \dots + \hat{a}_n x_i^n, \quad i=0, 1, 2, \dots, n$$

$n+1$  eqs

$n+1$  unknowns

$$\begin{pmatrix} 1 & x_0 & x_0^2 & \dots & x_0^n \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^n \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix} \rightarrow O(n^3)$$

expensive !

A full matrix  
 $(n+1) \times (n+1)$

For large  $n$ , the resulting system of eqs.  
becomes ill-conditioned in general.

In practice, we define the polynomial in an explicit way as opposed to solving a matrix system.

- Lagrange polynomial

Define a polynomial of degree  $n$  associated with each pt.  $x_j$ .

$$\begin{aligned} L_j(x) &= \alpha_j (x - x_0)(x - x_1) \dots (x - x_{j-1})(x - x_{j+1}) \dots (x - x_n) \\ &= \alpha_j \prod_{\substack{i=0 \\ i \neq j}}^n (x - x_i) \end{aligned}$$

$$L_j(x_k) = 0 \text{ if } k \neq j \quad (k=0, 1, 2, \dots)$$

$$L_j(x_j) = \alpha_j \prod_{\substack{i=0 \\ i \neq j}}^n (x_j - x_i) \Rightarrow \alpha_j \equiv 1 / \prod_{\substack{i=0 \\ i \neq j}}^n (x_j - x_i)$$

$$\Rightarrow L_j(x_k) = \begin{cases} 0 & k \neq j \\ 1 & k=j \end{cases}$$

The desired polynomial

$$P(x) = \sum_{j=0}^n y_j L_j(x)$$

Lagrange polynomial

$$P(x) = y_0 L_0(x) + y_1 L_1(x) + y_2 L_2(x) + \dots + y_n L_n(x)$$

Ex)

$\bar{x}$	0	1	2	3
$x_i$	1	2	4	8
$y_i$	1	3	7	11

$$g(x=7) = ?$$

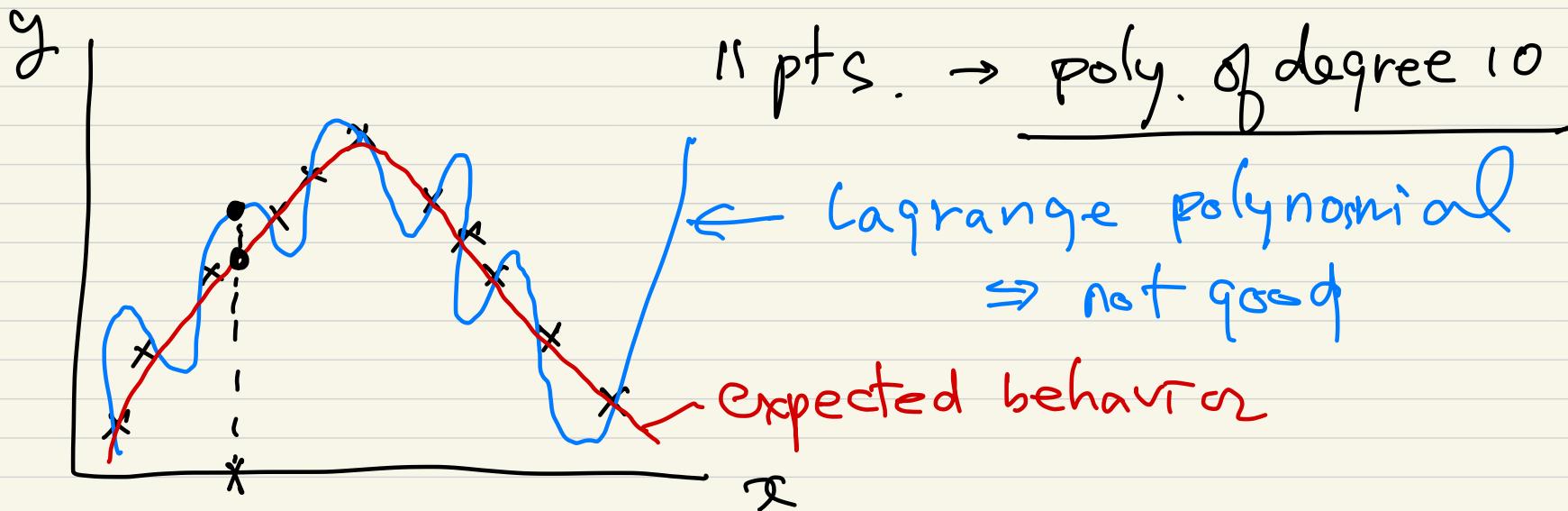
$$P(x) = 1 \cdot L_0(x) + 3 \cdot L_1(x) + 7 \cdot L_2(x) + 11 \cdot L_3(x)$$

$$L_0(x) = \frac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)} \rightarrow L_0(7)$$

"

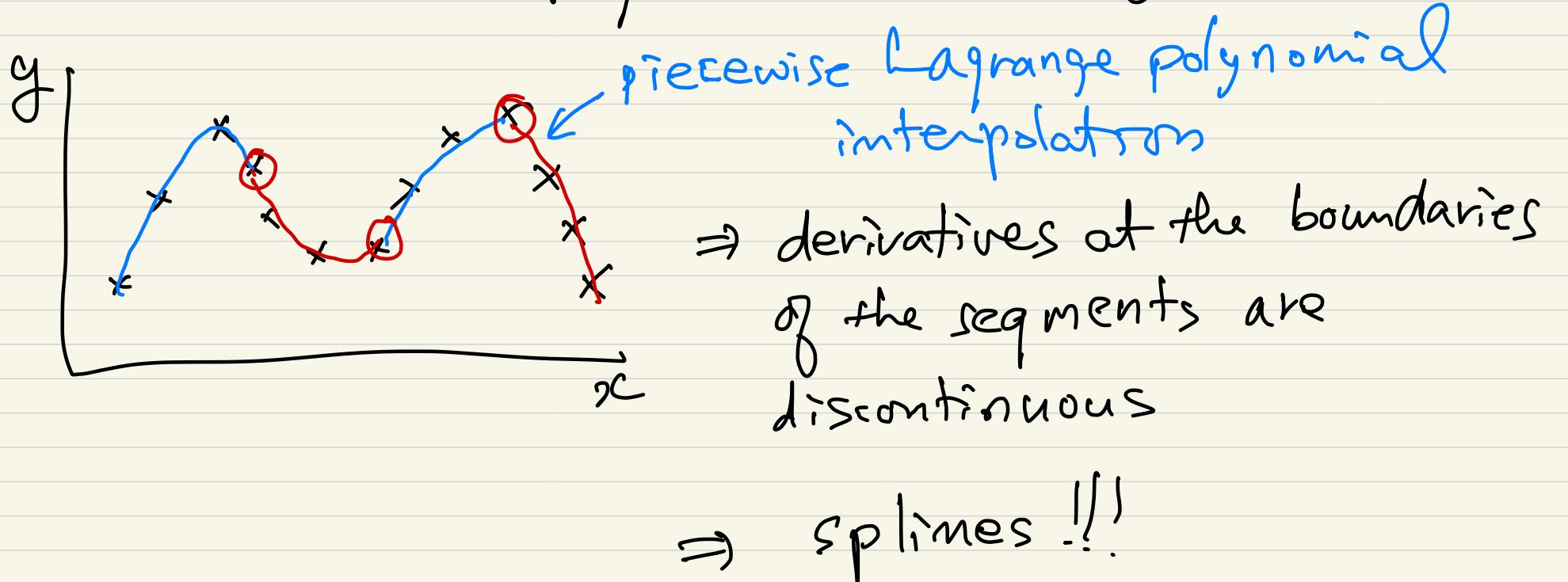
0.714

$$g(7) = P(7) = \underline{L_0(7)} + 3 \underline{L_1(7)} + 7 \underline{L_2(7)} + 11 \underline{L_3(7)} = 10.857$$

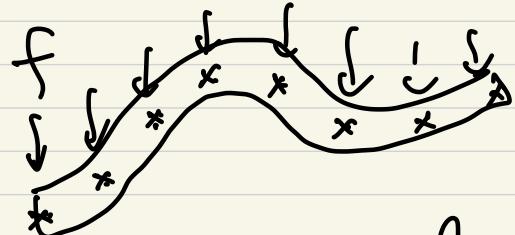


Although the polynomial is tied down at the data pts., it can wander in between.  $\rightarrow$  inaccurate

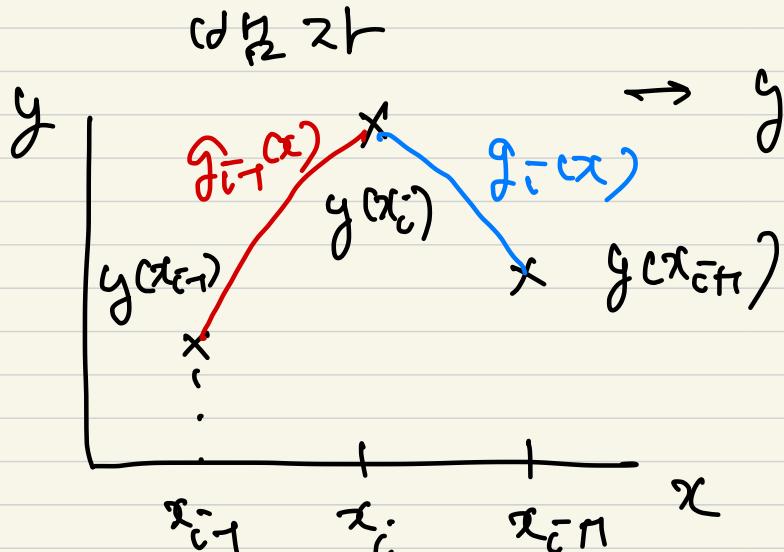
- Piecewise Lagrange polynomial interpolation  
instead of fitting only one poly. to all the data.
- lower-order polys. to sections of data.



## Spline interpolation



elastic scale



$$EI y^{(iv)} = f$$

between the data pts.,  $f=0$

$$EI y^{(iv)} = 0$$

$\rightarrow g$  is cubic in between  $x_i < x < x_{i+1}$ ,

For  $i$ -th interval  $(x_i \leq x \leq x_{i+1})$

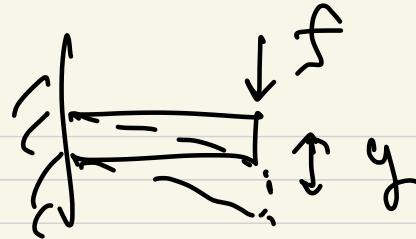
$$g_i(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3$$

$$g_i(x_i) = y(x_i), \quad g_i(x_{i+1}) = y(x_{i+1})$$

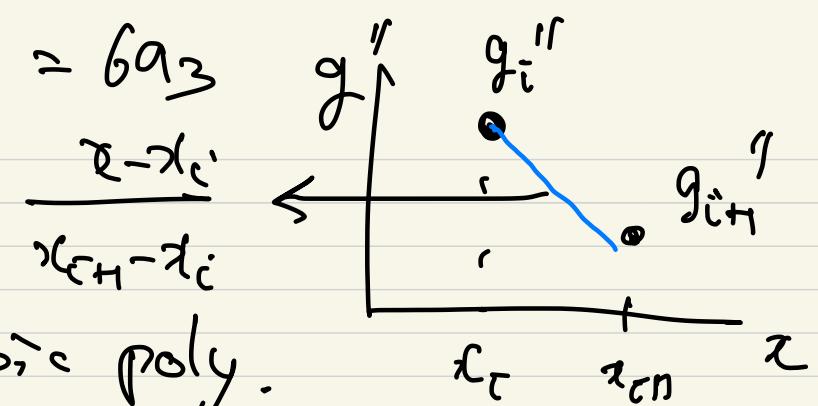
Match the 1st & 2nd derivatives from the adjacent intervals.

$$\begin{aligned} g'_i(x_i) &= g'_{i-1}(x_i) \\ g''_i(x_i) &= g''_{i-1}(x_i) \end{aligned}$$

$$\begin{aligned} g'_i(x) &= a_1 + 2a_2 x + 3a_3 x^2 \\ g''_i(x) &= 2a_2 + 6a_3 x \leftarrow \end{aligned}$$



$$\Rightarrow g_i''(x) = g''(x_c) \frac{x - x_{cH}}{x_c - x_{cH}} + g''(x_{cH}) \frac{x - x_c}{x_{cH} - x_c}$$



Integrate twice to get the cubic poly.

$$g_i(x) = g''(x_c) \frac{(x - x_{cH})^2}{2(x_c - x_{cH})} + g''(x_{cH}) \frac{(x - x_c)^2}{2(x_{cH} - x_c)} + c_1$$

$$g_i(x) = \boxed{g'(x_c)} \frac{(x - x_{cH})^3}{6(x_c - x_{cH})} + \boxed{g''(x_{cH})} \frac{(x - x_c)^3}{6(x_{cH} - x_c)} + \boxed{c_1}x + \boxed{c_2}$$

obtain  $c_1$  and  $c_2$  by  $g_i(x_c) = y(x_c)$  and  $g_i(x_{cH}) = y(x_{cH})$

$$\begin{aligned} \rightarrow g_i(x) &= \frac{1}{6} \boxed{g'(x_c)} \left[ \frac{(x_{cH} - x)^3}{(x_{cH} - x_c)} - (x_{cH} - x_c)(g'_{cH} - x) \right] \\ &\quad + \frac{1}{6} \boxed{g''(x_{cH})} \left[ \frac{(x - x_c)^3}{(x_{cH} - x_c)} - (x_{cH} - x_c)(x - x_c) \right] \\ &\quad + y(x_c) \frac{x_{cH} - x}{x_{cH} - x_c} + y(x_{cH}) \frac{x - x_c}{x_{cH} - x_c} \end{aligned}$$

Apply  $g_i'(x_c) = g_{i-1}'(x_c)$

$$\rightarrow \frac{\Delta_{i-1}}{6} g''(x_{c_1}) + \frac{\Delta_{i-1} + \Delta_i}{3} g''(x_c) + \frac{\Delta_i}{6} g''(x_{c_n}) = \frac{y(x_{c_n}) - y(x_c)}{\Delta_i} + \frac{y(x_{c_{i-1}}) - y(x_c)}{\Delta_{i-1}}$$

for  $i=1, 2, \dots, n-1$

$$g''(x_0), g''(x_1), \dots, g''(x_n)$$

→  $n+1$  unknowns

$$\Delta_i = x_{c_n} - x_i$$

$$\Delta_{i-1} = x_i - x_{c_{i-1}}$$

$n-1$  eqs.

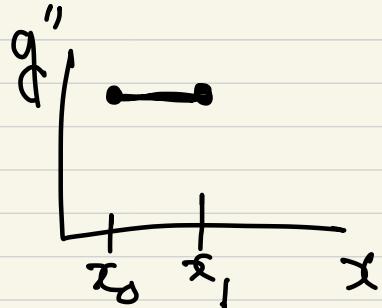
$$\begin{pmatrix} & \ddots & & & 0 \\ & \ddots & \ddots & & \\ & & \ddots & & \\ & & & \frac{\Delta_{i-1}}{6} & \frac{\Delta_{i-1} + \Delta_i}{3} & \frac{\Delta_i}{6} \\ 0 & & & & & \end{pmatrix} \begin{pmatrix} g_1'' \\ g_2'' \\ \vdots \\ g_{n-1}'' \\ g_n'' \end{pmatrix} = \begin{pmatrix} \vdots & + \cdot g_0' \\ \vdots \\ \vdots \\ \vdots \\ \vdots & + \cdot g_n' \end{pmatrix}$$

tri-diagonal matrix

## End conditions

$$\textcircled{1} \quad \begin{cases} g_0'' = g_1'' \\ g_n'' = g_{n-1}'' \end{cases}$$

$g_0''$  and  $g_n''$



$\Rightarrow$  parabolic runout

$$\begin{pmatrix} 1 & -1 & 0 & \dots & 0 \end{pmatrix} \begin{pmatrix} g_0'' \\ g_1'' \\ \vdots \\ g_{n-1}'' \\ g_n'' \end{pmatrix} = \begin{pmatrix} \cdot \\ \cdot \\ \vdots \\ \cdot \\ \cdot \end{pmatrix}$$

$$\textcircled{2} \quad \begin{cases} g_0'' = 0 \\ g_n'' = 0 \end{cases}$$



$\Rightarrow$  natural spline

$$\textcircled{3} \quad \begin{cases} g_0'' = \lambda g_1'' \\ g_n'' = \lambda g_{n-1}'' \end{cases} \quad 0 \leq \lambda \leq 1$$

④  $\begin{cases} g_0'' = g_{n-1}'' \\ g_n'' = g_1'' \end{cases}$  periodic b.c.

$$\begin{cases} g_0'' = g_n'' \\ g_n'' = g_0'' \end{cases}$$

$$\begin{pmatrix} 1 & 0 & 0 & 0 & \sigma & 0 & -1 \\ -1 & 0 & \dots & 0 & 1 \end{pmatrix} \begin{pmatrix} g_0'' \\ g_1'' \\ \vdots \\ g_{n-1}'' \\ g_n'' \end{pmatrix}$$

For equally spaced pts.,  $\sigma_i = \Delta\bar{x}_i = \Delta$

$$\frac{1}{6} [g''(x_{i-1}) + 4g''(x_i) + g''(x_{i+1})] = \frac{1}{\Delta^2} (y_{i-1} - 2y_i + y_{i+1})$$

In some cases, you may get wiggles.

$\rightarrow$  tension splines  $\rightarrow$  pull on each ends ( $g^{(ir)} - \sigma^2 y^{(ir)} = 0$ )

$$\rightarrow g'' - \sigma^2 g = (g_i^{(ir)} - \sigma^2 y_i) \frac{x - x_{i+1}}{x_i - x_{i+1}} + (g_{i+1}^{(ir)} - \sigma^2 y_{i+1}) \frac{x - x_i}{x_{i+1} - x_i}$$

$\sigma \rightarrow \infty$  : straight line interpolation

$\sigma = 0$  : usual spline

wiggles in sol.  $\rightarrow$  increase  $f$ .

- \* sp line has difficulties w/ large slopes

