CE 601: Numerical Methods

Lecture 17

(09-Sept-2014)

Cubic Splines (Contd..)
&
Approximate Polynomial Fits

Course Coordinator:
Dr. Suresh A. Kartha,
Associate Professor,
Department of Civil Engineering,
IIT Guwahati.

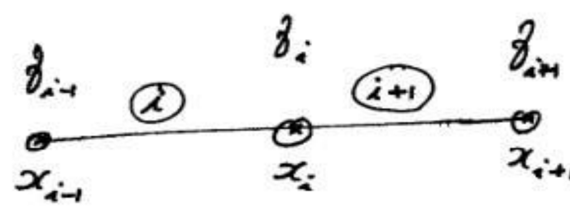
Cubic Splines

- Yesterday we were discussing on Cubic Splines.
- Splines are lower degree polynomials that connect sub-sets of data points.
- While incorporating splines, we have to see that the function is consistent as per actual function (i.e. it should be differentiable, the slopes and curvatures are continuous).

To develop cubic spline

- The cubic polynomial used as approximating function for the ith interval in the (n+1) data points $(x_0,f_0),(x_1,f_1), \ldots,(x_n,f_n)$ is:

$$f_i(x) = a_i + b_i x + c_i x^2 + d_i x^3$$



In actual, we have only the values

$f_0, f_1, f_2, \ldots, f_n$

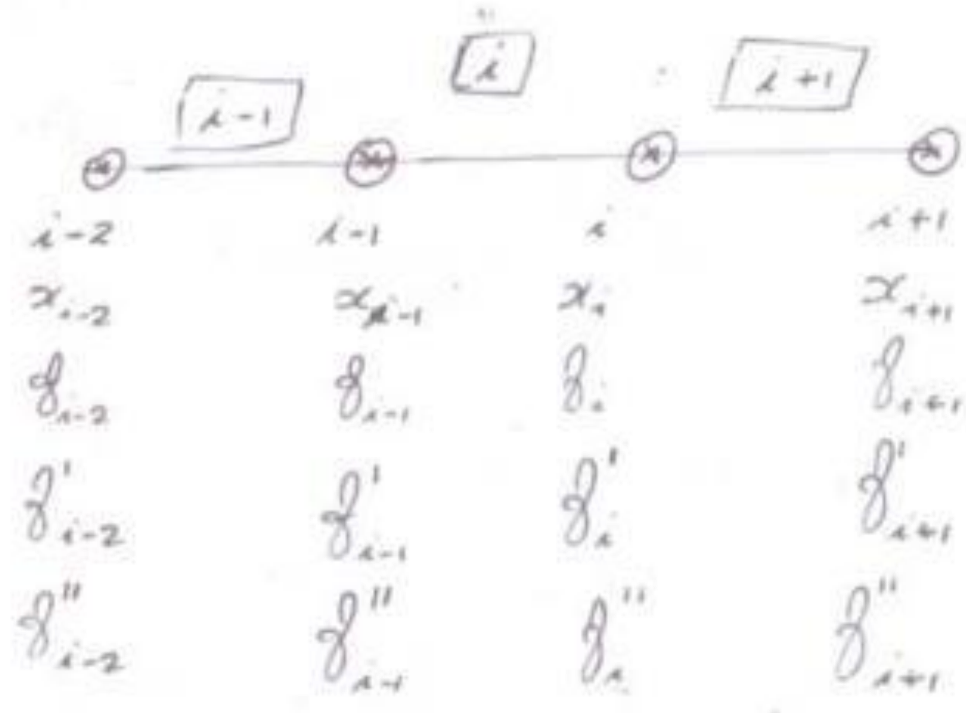
-> *(n+1)* values

f_0' and f_n'

-> 2 values if provided

f_0'' and f_n''

-> 2 values



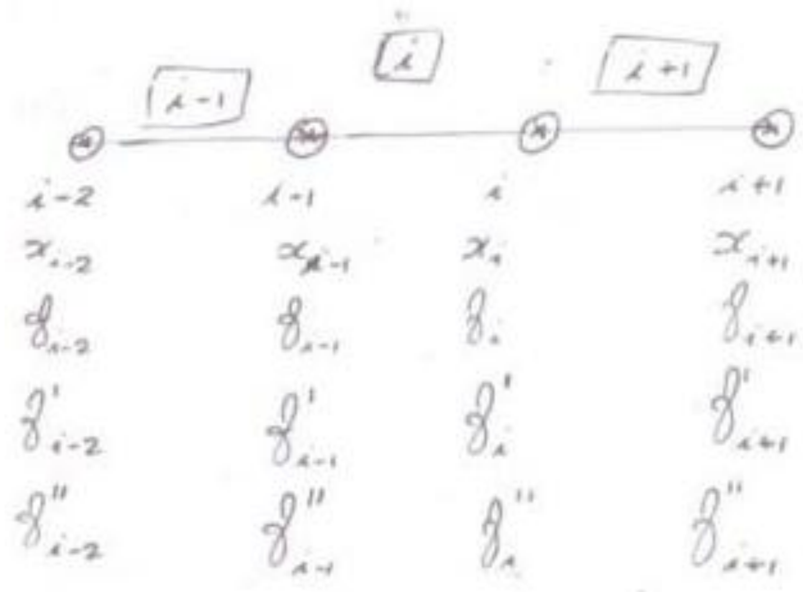
- All other f_i' and f_i'' should be inferred from these values.

So for any i-th interval,

$$f_i(x) = \frac{(x_i - x)^3}{6(x_i - x_{i-1})} f''_{i-1} + \frac{(x - x_{i-1})^3}{6(x_i - x_{i-1})} f''_i$$
$$+ (x_i - x)\left\{\frac{1}{(x_i - x_{i-1})} f_{i-1} - \frac{(x_i - x_{i-1})}{6} f''_{i-1}\right\}$$
$$+ (x - x_{i-1})\left\{\frac{f_i}{(x_i - x_{i-1})} - \frac{(x_i - x_{i-1})}{6} f''_i\right\}$$

- To find expressions for f''_i at interval nodes $i = 1, 2, 3, \ldots, n-1$ obtained by differentiating above equation once to get $f_i'(x)$ and again for $f_i''(x)$.

- The differentiated eq. $f_i'(x)$ is used for two data points f'_{i-1} and f_i'.
- Also differentiated eq. $f_{i+1}'(x)$ is used for f_i'.
- Similarly, $f_{i-1}'(x)$ is used for f_{i-1}'.
- Equating these equations, we get unknown in terms of f_{i-1}'', f_i'' and f_{i+1}''.



$$\rightarrow (x_i - x_{i-1}) f_{i-1}'' + 2(x_{i+1} - x_{i-1}) f_i'' + (x_{i+1} - x_i) f_{i+1}''$$

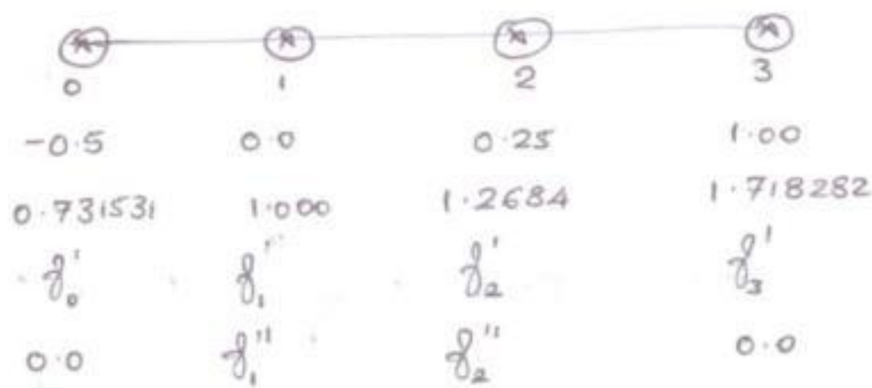
$$= 6\frac{f_{i+1} - f_i}{x_{i+1} - x_i} - 6\frac{f_i - f_{i-1}}{x_i - x_{i-1}} \qquad \rightarrow (2)$$

Eq. (2) is substituted in (n-1) interval nodes

to get systems of equations for unknowns $f_i''\ (i = 1, 2, 3, \ldots, n-1)$.

- <u>Example</u> (As adopted from the course text book Hoffman's Numerical Methods)
- For the given data set

i	*x*	*f(x)*	*f"(x)*
0	***-0.500***	***0.731531***	***0.0***
1	***0.000***	***1.00000***	
2	***0.250***	***1.26840***	
3	***1.000***	***1.718282***	***0.0***



- Soln. There are 4 data points. Three intervals are there *i = 1,2,3*

$$f_i(x) = a_i + b_i x + c_i x^2 + d_i x^3$$

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

$$+ (x - x_{i-1})\left\{\frac{f_i}{(x_i - x_{i-1})} - \frac{(x_i - x_{i-1})}{6} f_i''\right\}$$

We have $f_0'' = f_3'' = 0.0$

$\therefore$ For $i = 1$ interval, we have -0.5 $\leq x \leq 0$

i.e. $(x_1 - x_0) f_0'' + 2(x_2 - x_0) f_1'' + (x_2 - x_1) f_2''$

$$= 6\frac{f_2 - f_1}{x_2 - x_1} - 6\frac{f_1 - f_0}{x_1 - x_0}$$

i.e. $0.5 \times 0.0 + 2 \times 0.95 \times f_1'' + 0.25 \times f_2''$

$$= 6 \times \frac{1.2684 - 1}{0.25} - 6 \times \frac{1 - 0.731531}{0.5}$$

i.e. $1.50 f_1'' + 0.25 f_2'' = 3.219972$

Similarly, for interval $i = 2$, $0 \leq x \leq 0.25$

We have,

$(x_2 - x_1) f_1'' + 2(x_3 - x_1) f_2'' + (x_3 - x_2) f_3''$

$$= 6\frac{f_3 - f_2}{x_3 - x_2} - 6\frac{f_2 - f_1}{x_2 - x_1}$$

i.e. $0.25 \times f_1'' + 2 \times 1 \times f_2'' + 0.75 \times 0.0 = -2.842544.$

Solving for f_1'' and f_2'', we get:

$f_1'' = 2.434240$ and $f_2'' = -1.725552$

$$\therefore\ f_1(x) = \frac{(x_1 - x)^3}{6(x_1 - x_0)} f_0'' + \frac{(x - x_0)^3}{6(x_1 - x_0)} f_1'' + \frac{(x_1 - x)}{(x_1 - x_0)} f_0$$

$$- \frac{(x_1 - x)(x_1 - x_0)}{6} f_0'' + \frac{(x - x_0)}{(x_1 - x_0)} f_1 - \frac{(x - x_0)(x_1 - x_0)}{6} f_1''$$

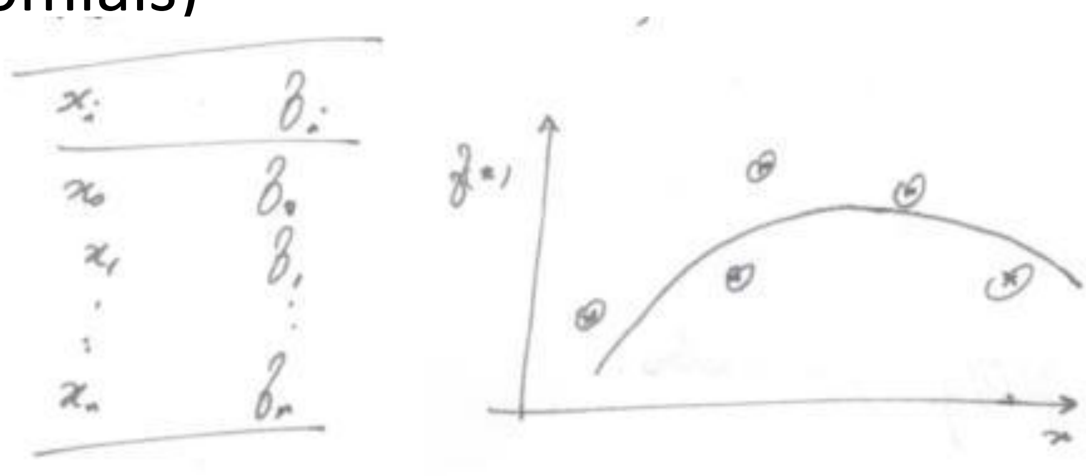
$$\text{i.e. } f_1(x) = \frac{(x + 0.5)^3}{6 \times 0.5} f_1'' + \frac{(0 - x)}{0.5} f_0 + \frac{(x + 0.5)}{0.5} f_1$$

$$- \frac{0.5 \times (x + 0.5)}{6} f_1''$$

Similarly obtain cubic polynomial for interval $i = 2$ and 3.

Approximate Polynomial Fits

- Earlier we mentioned that for a given set of data, we can have polynomials that can approximate a function:
  - exactly passes through all the data points (exactly fit polynomial)
  - may not pass through all data points (approximately fit polynomials)



- The most popular method to develop approximately fit polynomials are Least Squares Method.

Least Squares Approximation

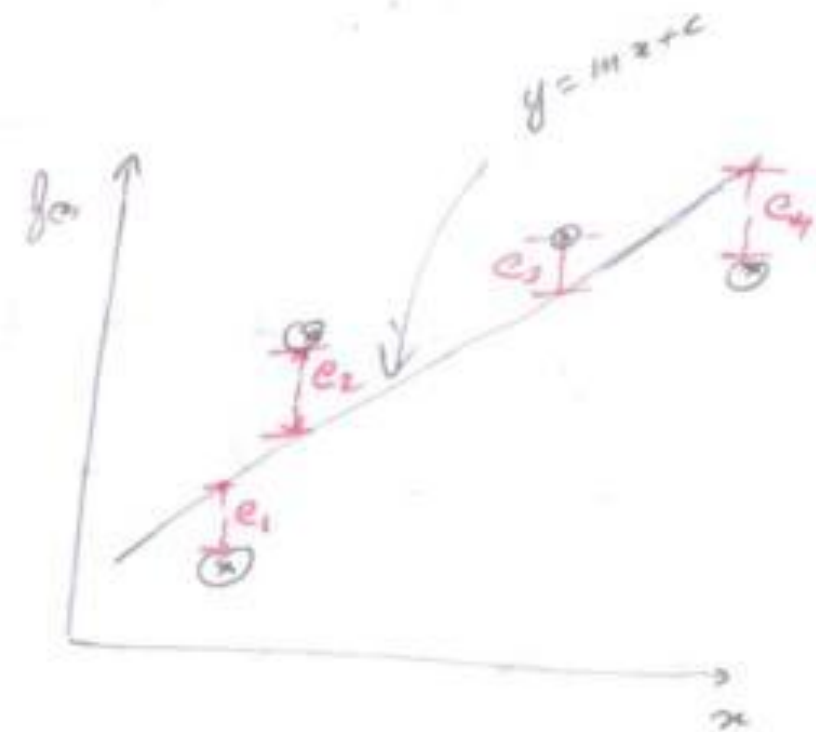
The objective is to

- Minimize the sum of the squares of the deviations.

- From the figure, you can see that a straight line is approximated to the given data set.
- These are errors $e_1, e_2, \ldots$ for each data set w.r.t. the predicted function value from straight line

$$e_i = f_i - y_i$$

- We can also use higher degree polynomials as well in method of least squares.



Straight Line Approximation

i	x	f	y	e	e^2
0	x_0	f_0	y_0	e_0	$(f_0 - y_0)^2$
1	x_1	f_1	y_1	e_1	$(f_1 - y_1)^2$
2	x_2	f_2	y_2	e_2	$(f_2 - y_2)^2$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
n	x_n	f_n	y_n	e_n	$(f_n - y_n)^2$

We are fitting a straight line for the given (x_i, f_i) data.

$y = A + Bx$, A and B are now parameters.

$$y_i = A + Bx_i$$

$$e_i = f_i - y_i$$

The error $e_i \to e_i(A, B)$

We have to minimize the sum of squares of errors.

$$\text{Sum } S = \sum_{i=0}^{n} \left(f_i - y_i \right)^2 = \sum_{i=0}^{n} e_i^2$$

i.e. $S \to S(A, B)$

To have the value of S as minimum for this case.

$$\frac{\partial S}{\partial A} = 0 \text{ and } \frac{\partial S}{\partial B} = 0$$

$$S = \sum_{i=0}^{n} \left(f_i - y_i \right)^2 = \sum_{i=0}^{n} \left(f_i - A - B \times x_i \right)^2$$

$$\therefore \frac{\partial S}{\partial A} = 0 = \sum_{i=0}^{n} 2 \left(f_i - A - B \times x_i \right)(-1)$$

$$\text{i.e. } \sum_{i=0}^{n} f_i - A \sum_{i=0}^{n} (1) - B \sum_{i=0}^{n} x_i = 0$$

Similarly,

$$\frac{\partial S}{\partial B} = 0 = \sum_{i=0}^{n} 2\left(f_i - A - B\times x_i\right)(-x_i)$$

i.e. $$\sum_{i=0}^{n}\left(f_i x_i\right) - A\sum_{i=0}^{n} x_i - B\sum_{i=0}^{n} x_i^2 = 0$$

As we have told (n+1) data points, we get

$$A\times(n+1) + B\times\sum_{i=0}^{n} x_i = \sum_{i=0}^{n} f_i$$

$$A\times\sum_{i=0}^{n} x_i + B\times\sum_{i=0}^{n} x_i^2 = \sum_{i=0}^{n} f_i x_i$$

These are the normal equations that need to be solved to get A and B, the parameters for straight line fit.

Example

x	f
0	10
5	17
10	25
15	31

- To fit a straight line curve for the given data.

i	*x*	*f*	x^2	*xf*
0	*0*	*10*	*0*	*0*
1	*5*	*17*	*25*	*85*
2	*10*	*25*	*100*	*250*
3	*15*	*31*	*225*	*465*
	$\sum x = 30$	$\sum f = 83$	$\sum x^2 = 350$	$\sum xf = 800$

$$\therefore 83 - A\times 4 - B\times 30 = 0$$

$$800 - A\times 30 - B\times 350 = 0$$

$$\text{i.e.} \begin{bmatrix} 4 & 30 \\ 30 & 350 \end{bmatrix} \begin{Bmatrix} A \\ B \end{Bmatrix} = \begin{Bmatrix} 83 \\ 800 \end{Bmatrix}$$

Solve for A and B.

Higher Order Fit

- In a similar form one can also go for higher order approximately fitting polynomials for given data set.

e.g. $f(x) = y = A + Bx + Cx^2$

or, $f(x) \approx y = A + Bx + Cx^2 + Dx^3 + \cdots$

- For straight line fit there are two normal equations. For second-degree polynomial you require three normal equations i.e. The number of normal equations increases on increase of polynomial degree

$$y = a_0 + a_1 x + a_2 x^2 + \cdots a_m x^m$$

- If for a given $(n+1)$ data points, if we would like to fit m^{th} degree polynomial, then you require (m+1) normal equations.
- Then the systems to solve for coefficients of m^{th} degree polynomial will look like:

$$\begin{bmatrix} (n+1) & \sum x_i & \sum x_i^2 & \cdots & \sum x_i^m \\ \sum x_i & \sum x_i^2 & \sum x_i^3 & \cdots & \sum x_i^{m+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \sum x_i^m & \sum x_i^{m+1} & \sum x_i^{m+2} & \cdots & \sum x_i^{2m} \end{bmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ \vdots \\ a_m \end{Bmatrix} = \begin{Bmatrix} \sum f_i \\ \sum x_i f_i \\ \vdots \\ \sum x_i^m f_i \end{Bmatrix}$$

Coefficient of Determination or Correlation Coefficient

x_i	f_i
x_0	f_0
x_1	f_1
x_2	f_2
$\vdots$	$\vdots$
x_n	f_n

You can identify the mean of function values,

$$\bar{f} = \frac{\sum_{i=0}^{n} f_i}{(n+1)}$$

Spread of data about its mean, we can represent through sum of squares of the deviations of function value from its mean,

$$\text{Let } S_t = \sum_{i=0}^{n} (f_i - \bar{f})^2$$

Sum of squares of the deviation of actual function values of f_i from the best fit polynomial y_i

$$\text{Let } S_r = \sum_{i=0}^{n} (f_i - y_i)^2$$

when $y_i = a_0 + a_1 x + \cdots + a_m x^m$

As the polynomial is to be best-fit polynomial, we can see that $S_r \le S_t$

We can define now, $R^2 = \dfrac{S_t - S_r}{S_t}$

where $R^2 \rightarrow$ correlation coefficient or coefficient of determination.

Usually if $R^2 \ge 0.80$, we can accept the fit.
If $R^2 \le 0.30$, we need to totally discard the fit.