Spline Interpolation:

Drawing a cubic curve through the A_i, A_{i+1}, and another curve A_{i+1}, A_{i+2}, such that the slope and curvatures of the 2 curves matches at A_{i+1}. Such a curve is called a spline.

A spline fits a set of n^{th} degree polynomial, $g_i(x)$ between each pair of points, from x_i to x_{i+1}. The point at which splines join are called knots.

If the polynomial are all of degree 1, we have a linear spline and the curve would appear as below, but the slopes are discontinuous where the segments joins.

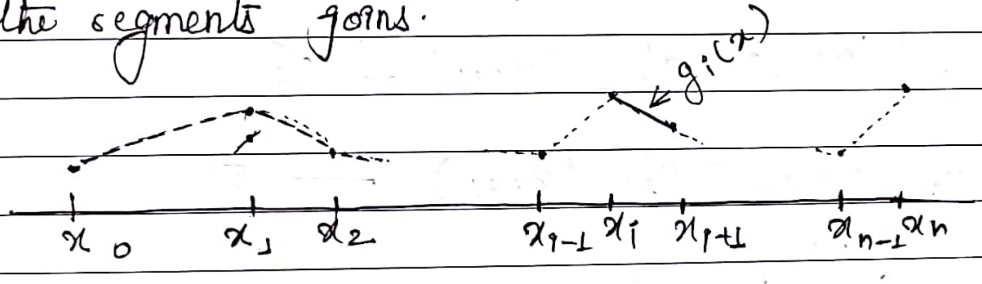


fig: linear spline

Spline of degree greater than one do not have this problem. Most often cubic splines are used.

We can write the eqn for a cubic polynomial, $g_i(x)$, in the i^{th} interval between points (x_i, y_i) and (x_{i+1}, y_{i+1}). It look like solid curve shown below. The dashed curves are other cubic spline polynomials.

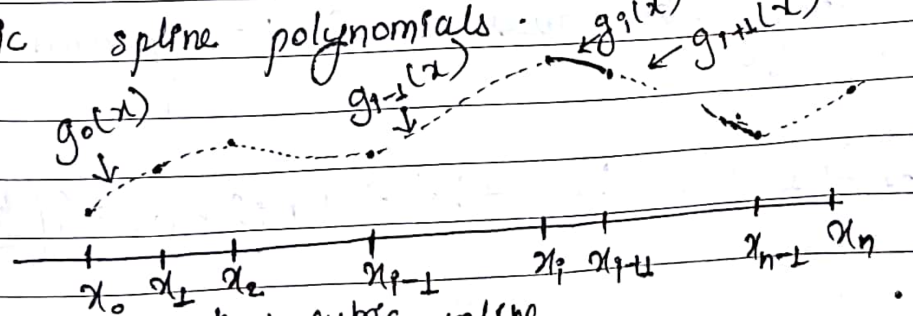


fig: cubic spline

Let, the cubic spline $g_i(x)$ has the eq^n,

$$g_i(x) = a_i(x-x_i)^3 + b_i(x-x_i)^2 + c_i(x-x_i) + d_i \quad \text{--- (1)}$$

This cubic ~~slip~~ spline function is of the form,

$$g(x) = g_i(x) \quad \text{--- (2)} \text{ on interval } [x_i, x_{i+1}]$$

for $i = 0, 1, 2 \dots n$ and meets the condition:

$\therefore g_i(x_i) = y_i$; $i = 0, 1, 2, \dots n-1$ and $g_{n-1}(x_n) = y_n$

$$g_i(x_{i+1}) = g_{i+1}(x_{i+1}) \quad \text{--- (3)}$$

$$g_i'(x_{i+1}) = g_{i+1}'(x_{i+1}) \quad \text{--- (4)}$$

$$g_i''(x_{i+1}) = g_{i+1}''(x_{i+1}) \quad \text{--- (5)}$$

where,

$$a_i = \frac{S_{i+1} - S_i}{6h_i}$$

$$b_i = S_i/2$$

$$c_i = \frac{y_{i+1} - y_i}{h_i} - \frac{2h_i S_i + h_i S_{i+1}}{6}$$

$$d_i = y_i.$$

substituting the values of a_i, b_i, c_i and d_i in eq^n (1) we get

$$h_{i-1} S_{i-1} + 2(h_{i-1} + h_i) S_i + h_i S_{i+1} = 6\left(\frac{y_{i+1} - y_i}{h_i} - \right.$$

$$\frac{y_i - y_{i-1}}{h_{i-1}} \Big)$$

$$= 6\left(f[x_i, x_{i+1}] - f[x_{i-1}, x_i]\right) \quad \text{---} \quad (6)$$

This gives $n-1$ eq^n relating $n+1$ values of S_i.
So, taking $S_0 = 0$ and $S_n = 0$. This makes the end cubic approaches to linearities at their ~~external~~ extermities.
This condition is called natural spl
If we write the eq^n in matrix form, we get

$$\begin{bmatrix} h_0 & 2(h_0+h_1) & h_1 & & \\ & h_1 & 2(h_1+h_2) & h_3 & \\ & & \cdots & & \\ & & & h_{n-2} & 2(h_{n-2}+h_{n-1}) & h_{n-1} \end{bmatrix}$$

$$\begin{bmatrix} S_0 \\ S_1 \\ S_2 \\ \vdots \\ S_n \end{bmatrix} = 6 \begin{bmatrix} f[x_1, x_2] - f[x_0, x_1] \\ f[x_2, x_3] - f[x_1, x_2] \\ \vdots \\ f[x_{n-1}, x_n] - f[x_{n-2}, x_{n-1}] \end{bmatrix}$$

taking $S_0 = S_n = 0$ then matrix becomes,

$$\begin{bmatrix} 2(h_0+h_1) & h_1 & & \\ h_1 & 2(h_1+h_2) & h_2 & \\ & & \cdots & \\ & & h_{n-2} & 2(h_{n-2}+h_{n-1}) \end{bmatrix}$$

$$\begin{bmatrix} S_1 \\ S_2 \\ \vdots \\ S_{n-1} \end{bmatrix} = 6 \begin{bmatrix} f[x_1, x_2] - f[x_0, x_1] \\ f[x_2, x_3] - f[x_1, x_2] \\ \vdots \\ f[x_{n-1}, x_n] - f[x_{n-2}, x_{n-1}] \end{bmatrix}$$

On solving which we get the value of S_i and finally a_i, b_i, c_i and d_i is computed after finding a_i, b_i, c_i, d_i we put this value in eq^n (1) to get required cubic spline.

Q. Fit the data with a natural spline.

x	0	1	1.5	2.25
$f(x)$	2	4.4366	6.7134	13.9130

Sol^n,

Let the cubic spline to be fitted will be of the form,

$$g_i(x) = a_i(x-x_i)^3 + b_i(x-x_i)^2 + c_i(x-x_i) + d_i \quad \text{—(i)}$$

where,

$$a_i = \frac{S_{i+1} - S_i}{6 \cdot h_i} ; \quad b_i = \frac{S_i}{2} ; \quad c_i = \frac{y_{i+1} - y_i}{h_i} - \frac{2h_i S_i + h_i S_{i+1}}{6}$$

$d_i = y_i$

Given,

divided diff.

x_i	h_i	y_i	$[x_i, x_{i+1}]$
0		2	
	1 (h_0)		$2.4366/1 = 2.4366 \, (x_0, x_1)$
1		4.4366	
	0.5 (h_1)		$2.2768/0.5 = 4.5536 \, (x_1, x_2)$
1.5		6.7134	
	0.75 (h_2)		$7.1996/0.75 = 9.5994 \, (x_2, x_3)$
2.25		13.9130	

4.

4 spline coefficient starts from S_0. ~~star~~

for natural ~~c~~ cubic spline,

$s_0 = s_3 = 0$

Then, finding values s_i using matrix, we get,

$$\begin{bmatrix} 2(h_0+h_1) & h_1 \\ h_1 & 2(h_1+h_2) \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} = 6 \begin{bmatrix} f[x_1,x_2]-f[x_0,x_1] \\ f[x_2,x_3]-f[x_1,x_2] \end{bmatrix}$$

or,
$$\begin{bmatrix} 3 & 0.5 \\ 0.5 & 2.5 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} = 6 \begin{bmatrix} 4.5536 - 2.4366 \\ 9.5995 - 4.5536 \end{bmatrix}$$

or,
$$\begin{bmatrix} 3 & 0.5 \\ 0.5 & 2.5 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} = \begin{bmatrix} 12.702 \\ 30.2754 \end{bmatrix}$$

or,
$$\begin{bmatrix} 3s_1 + 0.5s_2 \\ 0.5s_1 + 2.5s_2 \end{bmatrix} = \begin{bmatrix} 12.702 \\ 30.2754 \end{bmatrix}$$

On solving, we get,

$s_1 = 2.292$, $s_2 = 11.6518$, $s_0 = 0$, $s_3 = 0$

Using the values of s_i we compute a_i, b_i, c_i & d_i and substituting these values to eq^n (i), we get,

$i = 0$ (Ivary)

$$\begin{cases} a_0 = \dfrac{s_1 - s_0}{6 * h_0} & c_0 = \dfrac{y_1 - y_0}{h_0} \\ b_0 = \dfrac{s_0}{2} & d_0 = y_0 \end{cases}$$

$g_0^{(3)}$ $\circ$

i	interval	$g_i(x) = a_i(x-x_i)^3 + b_i(x-x_i)^2 + c_i(x-x_i) + d_i$
0	$[0, 1]$	$g_0(x) = 0.382x^3 + 2.0546x + 2$
1	$[1, 1.5]$	$g_1(x) = 3.1206(x-1)^3 + 1.146(x-1)^2 + 3.2(x-1) + 4.4366$
2	$[1.5, 2.25]$	$g_2(x) = -2.589(x-1.5)^3 + 5.8259(x-1.5)^2 + 6.6865($

Q. Find the value of $f(0.7)$ and $f(2)$

for,

$x = 0.7$

lies in interval $[0, 1]$

$g(0.7) = 3.5692$ //

for,

$x = 2$ lies in interval $[1.5, 2.25]$

$g_2(2) = 11.1896$ //

Q. Find the cubic spline for following data

x	1	2	3	4
y	-6	-1	16	25

and evaluate $y(1.5)$, $y'(2)$.

Convergence of N-R Method:

Let us consider x_n be the n^{th} approximated root which deviates from exact root α by small quantity ε_n. then,

$$x_n = \alpha + \varepsilon_n$$

and

$$x_{n+1} = \alpha + \varepsilon_{n+1}$$

Using N-R method, we have,

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

$$\text{or, } \cancel{\alpha} + \varepsilon_{n+1} = \cancel{\alpha} + \varepsilon_n - \frac{f(\alpha+\varepsilon_n)}{f'(\alpha+\varepsilon_n)}$$

$$\text{or, } \varepsilon_{n+1} = \varepsilon_n - \frac{f(\alpha+\varepsilon_n)}{f'(\alpha+\varepsilon_n)}$$

Expanding by taylor's series, we get, (since: α = root)

$$\text{or, } \varepsilon_{n+1} = \varepsilon_n - \frac{\cancelto{0}{f(\alpha)} + \varepsilon_n f'(\alpha) + \frac{\varepsilon_n^2}{2!} f''(\alpha) + \frac{\varepsilon_n^3}{3!} f'''(\alpha) + \dots}{f'(\alpha) + \varepsilon_n f''(\alpha) + \frac{\varepsilon_n^2}{2!} f'''(\alpha) + \frac{\varepsilon_n^3}{3!} f^{iv}(\alpha) + \dots}$$

$$\text{or, } \varepsilon_{n+1} = \varepsilon_n - \frac{\varepsilon_n f'(\alpha) + \frac{\varepsilon_n^2}{2!} f''(\alpha) + \frac{\varepsilon_n^3}{3!} f'''(\alpha) + \dots}{f'(\alpha) + \varepsilon_n f''(\alpha) + \frac{\varepsilon_n^2}{2!} f'''(\alpha) + \dots}$$

or, $$\varepsilon_{n+1} = \frac{\varepsilon_n f'(\alpha) + \varepsilon_n^2 f''(\alpha) + \frac{\varepsilon_n^3}{2!} f'''(\alpha) + \frac{\varepsilon_n^4}{3!} f^{iv}(\alpha) + \cdots - \left(\varepsilon_n f'(\alpha) + \frac{\varepsilon_n^2}{2!} f''(\alpha) + \frac{\varepsilon_n^3}{3!} f'''(\alpha) + \cdots\right)}{f'(\alpha) + \varepsilon_n f''(\alpha) + \frac{\varepsilon_n^2}{2!} f'''(\alpha) + \cdots}$$

or, $$\varepsilon_{n+1} = \frac{\frac{\varepsilon_n^2}{2} f''(\alpha) + \frac{\varepsilon_n^3}{3} f'''(\alpha) + \frac{\varepsilon_n^4}{4} f^{iv}(\alpha) + \cdots}{f'(\alpha) + \varepsilon_n f''(\alpha) + \frac{\varepsilon_n^2}{2!} f'''(\alpha) + \cdots}$$

$$\left[\because \varepsilon_n^2 f''(\alpha) - \frac{\varepsilon_n^2}{2!} f''(\alpha) = \frac{\varepsilon_n^2}{2} f''(\alpha)\right.$$

and so on,]

~~or,~~ neglecting higher ~~order~~ derivatives, we get,

constant

or, $$\varepsilon_{n+1} = \frac{\varepsilon_n^2}{2} \frac{f''(\alpha)}{f'(\alpha)} \quad \text{--- (2)}$$

or, $$\varepsilon_{n+1} = K \varepsilon_n^2$$

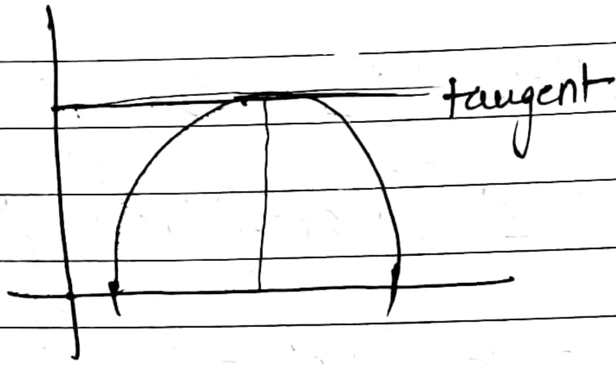
$$\boxed{\text{or, } \varepsilon_{n+1} \propto \varepsilon_n^2} \quad \text{--- (3)}$$

$\therefore$ ~~ε_n~~ Eq^n (3) shows that N-R method has quadratic convergence or 2nd degree of convergence.

Limitations : * computational complexity.

* If error is greater than 1 then it error diverges ~~expo~~ quadratically.

Date ____
Page ____



(never touches the x-axis so no root)

Limitations of N-R method:

1. Initial guess x_0 should be choosen in such a way that it lies near to the exact root otherwise it should diverge.
2. Initial root should be choosen in such a way that tangent on the point should not be parallel to x-axis otherwise root can not be determined.
3. initial root should be choosen in such a way that tangent on the point should not be perpendicular to x-axis otherwise root may deviate from the exact point.

* Fixed Point Iteration Method:

Necessary and sufficient condition:

Theorem: if

(i) $f(x)=0$ be any function which is equivalent to $x=\phi(x)$.

(ii) I be any interval within which $f(x)$ is continuous.

(iii) $|\phi'(x)|<1$ fall all value of x in interval I.

then any sequence of ...