ME 704

Computational Methods in Thermal and Fluids Engineering

(Interpolation and Regression)

Kannan Iyer

Kiyer@iitb.ac.in

Department of Mechanical Engineering

Indian Institute of Technology, Bombay

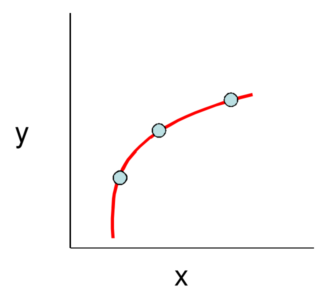
General-I

- Many a time we have values of a function at some discrete intervals of a variable and we need to obtain the value of the function at some value of the variable within the range.
- The question is whether one should interpolate or should one estimate by curve fitting using regression?
- The answer is another question? How accurate is the data?
- If very accurate – interpolate, if not, fit a curve by regression

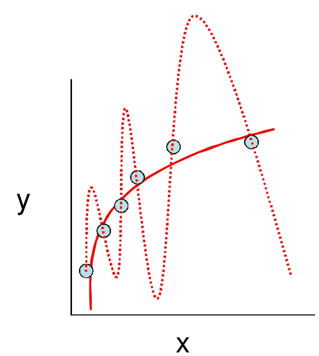
General-II

- What is common between both?
- Both involve functional approximation
- What is the difference?
- In interpolation the curve passes through all points, while in regression, it may not pass through any point!

General-III



Interpolation



Regression

Interpolation

- Interpolation fits a curve passing through the existing data
- The number of degrees of fredom would be equal to the number of points
- The functions chosen are at the discretion of the user
- Polynomials are the most common functions
- N^{th} order polynomial is unique which passes through N+1 points.
- Many concepts exist. Lagrangian interpolation is popular and most easy to code.

Lagrange Interpolation-I

First order Lagrange interpolation

$$f_1(x) = \frac{x - x_2}{x_1 - x_2} f(x_1) + \frac{x - x_1}{x_2 - x_1} f(x_2)$$

Second order Lagrange interpolation

$$f_2(x) = \frac{(x - x_2)(x - x_3)}{(x_1 - x_2)(x_1 - x_3)} f(x_1) + \frac{(x - x_1)(x - x_3)}{(x_2 - x_1)(x_2 - x_3)} f(x_2) + \frac{(x - x_1)(x - x_2)}{(x_3 - x_1)(x_3 - x_2)} f(x_3)$$

Lagrange Interpolation-II

General form

$$f_N(x) = \sum_{i=1}^{N+1} L_i(x) f(x_i)$$

Where, $L_i(x) = \prod_{\substack{j=1 \\ j \neq i}}^{N+1} \frac{x - x_j}{x_i - x_j}$

- Usually it is better to use piecewise lower order polynomials than using a single higher order polynomial

Newton's Forward Difference Polynomial-I

Similarly Newton's forward difference polynomials can be expressed as

- First Order

$$P_1(x) = f(0) + s\Delta f(0)$$

where $s = \frac{x - x_0}{h}$

- Second Order

$$P_2(x) = f(0) + s\Delta f(0) + \frac{s(s-1)}{2!} \Delta^2 f(0)$$

- N^{th} Order

$$P_N(x) = f(x_0) + \sum_{i=1}^{N} (sCi) \Delta^i f(x_0)$$

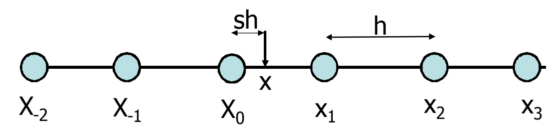
Newton's Forward Difference Polynomial-II

The forward difference operator is illustrated in the following table

x	f	Δf	$\Delta^2 f$	$\Delta^3 f$
x_o	f_0	$(f_1-f_0)= \Delta f_0$	$(f_2-2f_1+f_0)= \Delta^2 f_0$	$(f_3-3f_2+3f_1-f_0)= \Delta^3 f_0$
x_1	f_1	$(f_2-f_1)= \Delta f_1$	$(f_3-2f_2+f_1)= \Delta^2 f_1$	--
x_2	f_2	$(f_3-f_2)= \Delta f_2$	--	--
x_3	f_3	--	--	--

Newton's Forward Difference Polynomial-III

The concept that is employed is to expand the function at a point x around x_0.



$$f(x_0+sh) = f(x_0) + f'(x_0)sh + f''(x_0)\frac{(sh)^2}{2!} + \dots$$

$$+ f^n(x_0)\frac{(sh)^n}{n!} + f^{n+1}(\xi)\frac{(sh)^{n+1}}{n+1!}$$

It should also be clear that a polynomial of a n^{th} order using n+1 points is unique

Newton's Forward Difference Polynomial-IV

- The Newton's forward interpolating Polynomial formula using certain n+1 points is equivalent to the Taylor series formula that satisfies the function at the n+1 points
- To check this out for the first order polynomial is straight forward as

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

$$\Rightarrow f(x) = f(x_0) + \left(\frac{f(x_1) - f(x_0)}{h}\right)sh$$

$$P_1(x) = f_0 + s\Delta f_0$$

Example-I

f=1/x

x	f	Δf	$\Delta^2 f$	$\Delta^3 f$
3.4	0.294118	-0.008404	0.000468	0.000040
3.5	0.285714	-0.007936	0.000428	--
3.6	0.277778	-0.007508	--	--
3.7	0.270270	--	--	--

Find the value of function at 3.44

Example-II

s = (3.44-3.40)/0.1 = 0.4

(1/3.44) = 0.294418	0.294418
+ (0.4) (-0.008404)	0.290756
+ [(0.4)(0.4-1)/2] (0.000468)	0.290700
+ [(0.4)(0.4-1)(0.4-2)/6] (0.000040)	0.290698
Exact Value	0.290697674

Comments

- For a table having four points we could construct up to four terms
- The n+1th term corresponds to term of s^n if expanded in Taylor Series. Further s~1
- Therefore, from Taylor series expansion we can conclude that the leading error term will be $(h^{n+1}/n!)\ (d^{n+1}f/dx^{n+1})$
- Such an approximation is said to be n+1th order approximation
- This implies that if h is reduced by a factor of 2 the error will decrease by a factor 2^{n+1}

Newton's Backward Difference Polynomial-I

Similarly Newton's backward difference polynomials can be expressed as

- First Order

$$P_1(x) = f(0) + s\nabla f(0) \qquad where \quad s = \frac{x - x_0}{h}$$

- Second Order

$$P_2(x) = f(0) + s\nabla f(0) + \frac{(s)(s+1)}{2!}\nabla^2 f(0)$$

- Nth Order

$$P_N(x) = f(x_0) + \sum_{i=0}^{N}((s+i-1)Ci)\nabla^i f(x_0)$$

Newton's Backward Difference Polynomial-II

The backward difference operator is illustrated in the following table

x	f	∇f	$\nabla^2 f$	$\nabla^3 f$
x_o	f_0	--	--	--
x_1	f_1	$(f_1-f_0) = \nabla f_1$	--	--
x_2	f_2	$(f_2-f_1) = \nabla f_2$	$(f_2-2f_1+f_0) = \nabla^2 f_2$	--
x_3	f_3	$(f_3-f_2) = \nabla f_3$	$(f_3-2f_2+f_1) = \nabla^2 f_3$	$(f_3-3f_2+3f_1-f_0) = \nabla^3 f_3$

Example-I

f=1/x

x	f	∇f	$\nabla^2 f$	$\nabla^3 f$
3.2	0.312500			
3.3	0.303030	-0.009470		
3.4	0.294118	-0.008912	0.000558	
3.5	0.285714	-0.008404	0.000508	0.000050

Find the value of function at 3.44

Example-II

s = (3.50-3.44)/0.1 = -0.6

(1/3.44) = 0.285714	0.285714
+ (-0.6) (-0.008404)	0.290756
+ [(-0.6+1)(-0.6)/2] (0.000508)	0.290695
+[(-0.6+2)(-0.6+1)(-0.6)/6] (0.000050)	0.290698
Exact Value	0.290697674

Physical Interpretations