1. Solution of Algebric and Transcendental Equations (2)
Newton Raphson Method, Secant method, Solution of system of Nonlinear equation (Newton Raphson Method)

2. Interpolation (5)
Error in polynomial interpolation, finite Difference, Difference of a polynomial, Newton's formulae for interpolation, Bessel's formulae Everett's formula, Relation between Bessel's and Everett's formulae Lagrange's Interpolation formulae

3. Curve fitting, B-splines and Approximation. (3)
Least Squares Curve fitting procedures (Linear, Quadratic & Exponential) B-splines, Approximation of functions.

4. Numerical Differentiation and Integration :- (4)
Numerical Differentiation, Trapezoidal Rule, Simpson 1/3 rule Simpson 3/8 - Rule, volume calculation, Newton-Cotes Integration formulae, General Quadratic formula, Gaussian Integration → Defn + formula

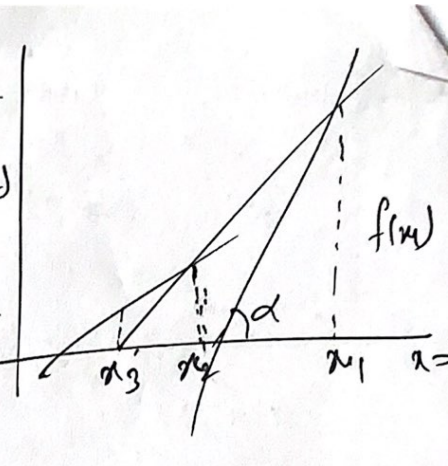
5. Matrices and Linear System of Equation (7)
Solution of Linear System, Direct method (Gauss Jordan) Solution of Linear System - Itterative method (Gauss Seidel) Eigen value problem (Eigen value, Eigen vector.

6. Numerical Solution of Ordinary Differential Equation (6)
Solution of Taylor's Series, Euler's method (Modified method) Predictor - Corrector methods, Simultaneous and Heigher order Equation (4th order Runge Kutta Method) Boundary value problem, ~~BP~~ (Finite Difference method)

7. Numerical Solution of partial Differential Equation (3)
Finite Difference Approximation to derivatives Laplace's Equation, ~~Para~~ Parabolic Equation, Hyperbolic Equation, Itterative method for solution ~~to~~ of Equation.

Newton Raphson Method

Consider a graph $f(x)$. Let us assume x_1 is an approximate root of $f(x)=0$. Draw a tangent at the curve $f(x)$ at $x=x_1$ as shown in figure.
The point of intersection of this tangent line with the x-axis gives the second approximation to the root.



The slope of tangent given by

$$\tan\alpha = \frac{f(x)}{x_1 - x_2} = f'(x_1)$$

where; $f'(x_1)$ is the slope of $f(x)$ at $x=x_1$. Solving for x_2 we obtain;

$$\boxed{x_2 = x_1 - \frac{f(x_1)}{f'(x_1)}}$$

The general form is

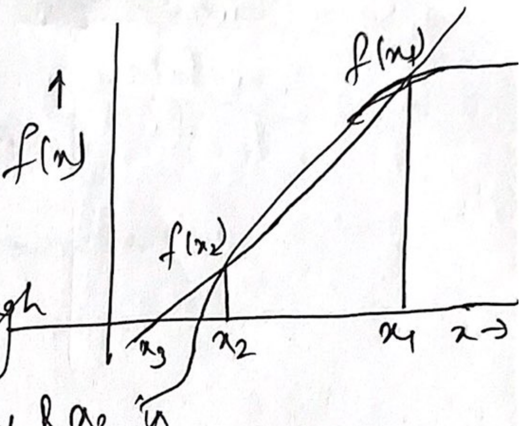
$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

The method of successive approximation is called Newton Raphson Method.

Algorithm

1. Assign an initial value to x say x_0
2. Evaluate $f(x_0)$ & $f'(x_0)$
3. Find the improved estimate to x_0
   $$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$
4. Check for accuracy for the latest estimate
   $$E = \left|\frac{x_1 - x_0}{x_1}\right| \le E$$ stop else continue
5. Replace $x_0 = x_1$ and repeate step (3) & (4)
6. End.

Secant Method



It is like false position & bisection method uses two initial estimates but does not require that they must bracket the root.
The secant method can use the points x_1 & x_2 as starting value although they do not bracket the root.
Slope of secant line passing through x_1 & x_2 is given by

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

$$f(x_1)(x_2 - x_3) = f(x_2)(x_1 - x_3)$$

$$x_3\left[f(x_2) - f(x_1)\right] = f(x_2)\cdot x_1 - f(x_1)\cdot x_2$$

$$x_3 = \frac{f(x_2)\cdot x_1 - f(x_1)\cdot x_2}{f(x_2) - f(x_1)}$$

In general

$$x_{i+1} = \frac{f(x_i)\cdot x_{i-1} - f(x_{i-1})\cdot x_i}{f(x_i) - f(x_{i-1})}$$

Algorithm :-

(1) Decide two initial point x_1 and x_2, accuracy level required, E.
(2) Compute $f_1 = f(x_1)$ and $f_2 = f(x_2)$.
(3) Compute $x_3 = \dfrac{f_2 \cdot x_1 - f_1 x_2}{f_2 - f_1}$
4. Test for accuracy of x_3 if $\left|\dfrac{x_3 - x_2}{x_3}\right| > E$ then
   set $x_1 = x_2$ & $f_1 = f_2$
   set $x_2 = x_3$ & $f_2 = f(x_3)$
   go to step 3.
   otherwise
   set root $= x_3$
   Print result
5. Stop.

Secant formula

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

$$x_3 = x_2 - \frac{f(x_2)(x_2 - x_1)}{f(x_2) - f(x_1)}.$$

Newton Raphson Method (System of non linear Equations).

Recalling the Taylor series of first order form

$$f(x_{i+1}) = f(x_i) + (x_{i+1} - x_i) f'(x) \quad \text{--- (1)}$$

Now; using the above form to calculate x_{i+1} we have

or; $f(x_{i+1}) = f(x_i) + x_{i+1} \cdot f'(x) - x_i f'(x)$

we have; $f(x_{i+1}) = 0$

or; $0 = f(x_i) + x_{i+1} \cdot f'(x) - x_i f'(x)$

or; $x_i \cdot f'(x) - f(x_i) = x_{i+1} \cdot f'(x)$

or; $$x_{i+1} = \frac{x_i f'(x) - f(x_i)}{f'(x)}$$

$$= \frac{x_i f'(x)}{f'(x)} - \frac{f(x_i)}{f'(x_i)}$$

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} \quad \text{--- (2)}$$

For sake of simplicity, let us again consider a two equation non linear system.

$$f(x,y) = 0 \quad , \quad g(x,y) = 0 \quad \text{--- (3)}$$

First order taylor series of these equation can be written as;

$$f(x_{i+1}, y_{i+1}) = f(x_i, y_i) + (x_{i+1} - x_i) \left| \frac{\delta f_i}{\delta x} \right| + (y_{i+1} - y_i) \left| \frac{\delta f_i}{\delta y} \right| \quad \text{- (4)}$$

$$g(x_{i+1}, y_{i+1}) = g(x_i, y_i) + (x_{i+1} - x_i) \left| \frac{\delta g_i}{\delta x} \right| + (y_{i+1} - y_i) \left| \frac{\delta g_i}{\delta y} \right| \quad \text{- (5)}$$

If the root estimates are x_{i+1} and y_{i+1} then

$$f(x_{i+1}, y_{i+1}) = g(x_{i+1}, y_{i+1}) = 0$$

Substituting this in eqⁿ ④ & ⑤ we get the following two linear equation.

$$\Delta x f_1 + \Delta y f_2 + f = 0 \quad \text{— ⑥}$$

$$\Delta x g_1 + \Delta y g_2 + g = 0 \quad \text{— ⑦}$$

where we denote

$$\Delta x = x_{i+1} - x_i \qquad \Delta y = y_{i+1} - y_i$$

$$f_1 = \left|\frac{\delta f_i}{\delta x}\right| \quad f_2 = \left|\frac{\delta f_i}{\delta y}\right| \quad g_1 = \left|\frac{\delta g_i}{\delta x}\right| \quad g_2 = \left|\frac{\delta g_i}{\delta y}\right|$$

$$f = f(x_i, y_i) \quad g = g(x_i, y_i)$$

Solving for x and y we get

$$\left.\begin{aligned} \Delta x &= -\frac{f g_2 - g f_2}{f_1 g_2 - f_2 g_1} = -\frac{D_x}{D} \\ \Delta y &= -\frac{g f_1 - f g_1}{f_1 g_2 - f_2 g_1} = -\frac{D_y}{D} \end{aligned}\right\} \text{— ⑧}$$

where;

$$D = \begin{vmatrix} f_1 & f_2 \\ g_1 & g_2 \end{vmatrix} = f_1 g_2 - f_2 g_1$$

is the jacobian matrix.

we can establish following recurrence relation:

$$\left.\begin{aligned} x_{i+1} &= x_i - \frac{D_x}{D} \\ y_{i+1} &= y_i - \frac{D_y}{D} \end{aligned}\right\} \text{— ⑨}$$

is required formula & these eqⁿ can be used to find root of $f(x,y)$ & $g(x,y)$

Two equation of Newton Raphson Method

1. Define the function f & g
2. Define the jacobian elements f_1, f_2, g_1, g_2
3. Decide starting point x_0 & y_0 and error tolerance E
4. Evaluate f, g, f_1, f_2, g_1, g_2 at (x_0, y_0) and compute D_x, D_y and D

   $x_1 = x_0 - \frac{D_x}{D}$ $\quad y_1 = y_0 - \frac{D_y}{D}$

5. Test for accuracy

   if $|x_1 - x_0| < E$ and

   $|y_1 - y_0| < E$ then solution obtained & goto step 6

   else, set

   $x_0 = x_1$

   $y_0 = y_1$

   and go to step 4
6. write result
7. End.

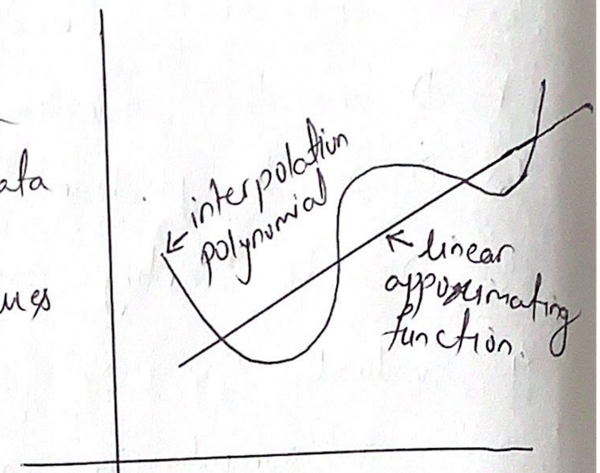
for formula :-

$$\boxed{\begin{aligned} x_1 &= x_0 - \frac{f g_2 - g f_2}{f_1 g_2 - f_2 g_1} \\ y_1 &= y_0 - \frac{g f_1 - f g_1}{f_1 g_2 - f_2 g_1} \end{aligned}}$$

$\longrightarrow$ is the main formula used in computation.

Interpolation :-

The function is constructed such that it passes through all the data points. The method of constructing such function and estimating values at non tabular points is called interpolation.



The most common form of n^{th} order polynomial is :

$$\boxed{p(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n}$$

This form is known as power form

Different methods of interpolation

- Lagrange interpolation
- Newton interpolation
- Bessal's interpolation
- Everett's interpolation

Lagrange interpolation polynomial

We derive a formula for the polynomial of degree n which take specified values at a given set of n+1 points :
~~$p(x_k) = f_k$~~

Let $x_0, x_1, \cdots x_n$ denote n distinct real numbers and let $f_0, f_1, \cdots f_n$ be arbitrary real number. The point (x_0, f_0) $(x_1, f_1) \cdots (x_n, f_n)$ can be imagined to be data values connected by a curve. Any function $p(x)$ satisfying the condition

$$p(x_k) = f_k \quad \text{for } k = 0, 1, \cdots n$$

is called an interpolation polynomial. i.e an interpolation polynomial is a curve that passes through all the data points.

Let us consider a second order polynomial of the form

$$P_2(x) = b_1(x-x_0)(x-x_1) + b_2(x-x_1)(x-x_2) + b_3(x-x_2)(x-x_0) \quad \text{— (1)}$$

if (x_0, f_0) (x_1, f_1) and (x_2, f_2) are the three interpolating points then we have

$$\left.\begin{aligned} P_2(x_0) &= f_0 = b_2(x_0-x_1)(x_0-x_2) \\ P_2(x_1) &= f_1 = b_3(x_1-x_2)(x_1-x_0) \\ P_3(x_2) &= f_2 = b_1(x_2-x_0)(x_2-x_1) \end{aligned}\right\} \text{— (2)}$$

substituting for b_1, b_2 and b_3 in eq^n (1) we get;

$$P_2(x) = f_0 \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} + f_1 \frac{(x-x_2)(x-x_0)}{(x_1-x_2)(x_1-x_0)} +$$

$$f_2 \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} \quad \text{—— (3)}$$

Eq^n (3) can be represented as;

$$P_2(x) = f_0 l_0(x) + f_1 l_1(x) + f_2 l_2(x)$$

$$= \sum_{i=0}^{2} f_i l_i(x)$$

where; $$l_i(x) = \prod_{j=0;\, j\neq i}^{2} \frac{(x-x_j)}{x_i-x_j}$$

In general, for $(n+1)$ point we have n^{th} degree polynomial as

$$\boxed{P_n(x) = \sum_{i=0}^{n} f_i l_i(x)} \quad \text{—— (4)}$$

where;

$$l_i(x) = \prod_{j=0;\, j\neq i}^{n} \frac{x-x_j}{x_i-x_j} \quad \text{—— (5)}$$

Eq^n (4) is the lagrange intepolation polynomial & eq^n (5) is lagrange basis polynomials.

we observe that;

$$l_i(x) = \begin{cases} 1 & \text{for } i=j \\ 0 & \text{for } i\neq j \end{cases}$$

Now; Consider the case $n=1$

$$l_0(x) = \frac{x - x_1}{x_0 - x_1}$$

$$l_1(x) = \frac{x - x_0}{x_1 - x_0}$$

Therefore;
$$P_1(x) = f_0 \frac{(x-x_1)}{(x_0-x_1)} + f_1 \frac{(x-x_0)}{(x_1-x_0)}$$

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

$$\boxed{P_1(x) = f_0 + \frac{f_1-f_0}{x_1-x_0}(x-x_0)}$$

is the linear interpolation formula

Newton Interpolation Polynomial.

- It overcomes the drawback of Lagrange interpolation.
- It improve the accuracy of estimation.

Let us consider newton form of polynomial as:-

$$P_n(x) = a_0 + a_1(x-x_0) + a_2(x-x_0)(x-x_1) + \cdots + a_n(x-x_0)(x-x_1) \cdots (x-x_{n-1}) \quad \text{--- (1)}$$

when the interpolation points $x_0, x_1, x_2, \ldots x_{n-1}$ acts as centres.

To construct interpolation polynomial we determine coefficient $a_0, a_1, \ldots a_n$

Let us assume that $(x_0, f_0), (x_1, f_1), \ldots (x_{n-1}, f_{n-1})$ are the interpolating points. That is

$$P_n(x_k) = f_k \quad k = 0, 1, \ldots n-1$$

Now, at $x = x_0$ we have

$$P_n(x_0) = \boxed{a_0 = f_0} \quad \text{--- (2)}$$

~~Subst~~ Similarly, at $x = x_1$

$$P_n(x_1) = a_0 + a_1(x_1 - x_0) = f_1 \quad \text{--- (3)}$$

Substituting for a_0 from eq^n (2) in eq^n (3) we get;

$$\boxed{a_1 = \frac{f_1 - f_0}{x_1 - x_0}} \quad \text{--- (4)}$$

At $x = x_2$

$$P_n(x_2) = a_0 + a_1(x_2 - x_0) + a_2(x_2 - x_0)(x_2 - x_1) = f_2$$

Substituting for a_0 and a_1 from eq^n (2) & (4) and rearranging the term

$$a_2 = \frac{[(f_2 - f_1)/(x_2 - x_1)] - [(f_1 - f_0)/(x_1 - x_0)]}{x_2 - x_0} \quad \text{--- (5)}$$

Let us define a notation:-

$$f[x_k] = f_k$$

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

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

$$f[x_k, x_{k+1}, \dots x_i, x_{i+1}] = \frac{f[x_{k+1}, \dots x_{i+1}] - f[x_k, \dots x_i]}{x_{i+1} - x_k} \quad (6)$$

These quantities are divide difference.

$$a_0 = f_0 = f[x_0]$$

$$a_1 = \frac{f_1 - f_0}{x_1 - x_0} = f[x_0, x_1]$$

$$a_2 = \frac{\frac{f_2 - f_1}{x_2 - x_1} - \frac{f_1 - f_0}{x_1 - x_0}}{x_2 - x_0}$$

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

$$= f[x_0, x_1, x_2]$$

Thus,

$$a_n = f[x_0, x_1, x_2, \dots x_n] \quad (6)$$

a_1 is first divide difference a_2 is second divide difference & so on.

Substituting for a_i coeff in eqn (1) we get

$$P_n(x) = f[x_0] + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1) + \dots + f[x_0, x_1, x_2, \dots x_n](x - x_0)(x - x_1) \dots (x - x_{n-1})$$

Thus it can be written as,

$$\boxed{P_n(x) = \sum_{i=1}^{n} f[x_0, x_1, \dots x_i] \prod_{j=0}^{i-1} (x - x_j)} \quad (7)$$

Eqⁿ (7) is newton divide difference interpolation polynomial.

(A) For divide difference table

i	x_i	$f[x_i]$	first difference	second difference	third difference	fouth difference
0	x_0	$f[x_0]$	$f[x_0,x_1]$	$f[x_0,x_1,x_2]$		
1	x_1	$f[x_1]$	$f[x_1,x_2]$		$f[x_0,x_1,x_2,x_3]$	$f[x_0,x_1,x_2,x_3,x_4]$
2	x_2	$f[x_2]$	$f[x_2,x_3]$	$f[x_1,x_2,x_3]$	$f[x_1,x_2,x_3,x_4]$	
3	x_3	$f[x_3]$	$f[x_3,x_4]$	$f[x_2,x_3,x_4]$		
4	x_4	$f[x_4]$				

Divide difference table.

shortcut method is in next page

For Gregory-Newton Forward difference formula:

Let us assume that;

$$x = x_0 + kh$$

The first forward difference Δf_i is defined as

$$\Delta f_i = f_{i+1} - f_i$$

The second forward difference is defined as

$$\Delta^2 f_i = \Delta f_{i+1} - \Delta f_i$$

In general;

$$\Delta^j f_i = \Delta^{j-1} f_{i+1} - \Delta^{j-1} f_i \quad \text{——— (1)}$$

We can now express the simple forward difference in term of divided difference: We know that;

$$f[x_0, x_1] = \frac{f(x_1) - f(x_0)}{x_1 - x_0} = \frac{f_1 - f_0}{h}$$

$$\therefore \; f_1 - f_0 = h f[x_0, x_1]$$

Then; $\Delta f_0 = f_1 - f_0 = h f[x_0, x_1]$

Simillarly; $\Delta f_1 = h f[x_1, x_2]$

Now;

$$\begin{aligned}
\Delta^2 f_0 &= \Delta f_1 - \Delta f_0 \\
&= hf[x_1, x_2] - hf[x_0, x_1] \\
&= h\{f[x_1, x_2] - f[x_0, x_1]\} \\
&= h \cdot 2h\, f[x_0, x_1, x_2] \\
&= 2h^2 f[x_0, x_1, x_2]
\end{aligned}$$

In general by induction

$$\Delta^j f_i = \cancel{2} j!\, h^j f[x_i, x_{i+1}, \ldots x_{i+j}]$$

Therefore;

$$f[x_0, x_1, \ldots x_j] = \frac{\Delta^j f_0}{j!\, h^j} \quad \text{—} \quad (2)$$

~~Subst~~ As we have;

$$P_n(x) = \sum_{i=1}^{n} f[x_0, x_1, \ldots x_i] \prod_{k=0}^{i-1} (x - x_k) \quad - \quad (3)$$

Substituting from eq^n (2) in eq^n (3) we get

$$P_n(x) = \sum_{j=0}^{n} \frac{\Delta^j f_0}{j!\, h^j} \prod_{k=0}^{j-1} (x - x_k) \quad \text{—} \quad (4)$$

Let us set $x = x_0 + sh$ and $P_n(s) = P_n(x)$

we know that;

$$x_k = x_0 + kh$$

Thus we have

$$x - x_k = (s-k)h \quad — ⑤$$

Substituting value of $(x - x_k)$ from ⑤ in eq^n ④

$$P_n(s) = \sum_{j=0}^{n} \frac{\Delta^j f_0}{j!\, h^j} \prod_{k=0}^{j-1} (s-k)h$$

$$= \sum_{j=0}^{n} \frac{\Delta^j f_0}{j!\, h^j} \left[s(s-1) \cdots (s-j+1) \right] h^j$$

Thus; $$P_n(s) = \sum_{j=0}^{n} \frac{\Delta^j f_0}{j!} \left[s(s-1) \cdots (s-j+1) \right]$$

$$= \cancel{\sum_{j=0}^{n} \binom{s}{j} \frac{\Delta^j f_0}{j!}} = \sum_{j=0}^{n} \binom{s}{j} \Delta^j f_0$$

$$\therefore \boxed{P_n(s) = \sum_{j=0}^{n} \binom{s}{j} \Delta^j f_0} \quad — ⑥$$

$$\binom{s}{j} = \frac{s(s-1) \cdots (s-j+1)}{j!}$$ is binomial coefficent

Eq^n ④ and ⑥ are Gregary-Newton forward difference formula.

* Newton Forward Interpolation

Let us consider $y = f(x)$ and $y_0, y_1, \dots y_n$ are value corresponding to point $x_0, x_0+h, x_0+2h, \dots x_0+nh$. Suppose we want to find ~~$x = x_0+uh$~~ $f(x) = y$ at point $x = x_0 + uh$

i.e; $u = \frac{x - x_0}{h}$

By the definition of shift operator E we have

$$E^u f(x) = f(x+uh) \quad \text{— (1)}$$

i.e; $E^u f(x_0) = f(x_0+uh)$ [At point $x = x_0$]

or; $f(x_0+uh) = E^u f(x_0)$

or; $f(x) = E^u y_0$ [Here; $x = x_0 + uh$ & $y_0 = f(x_0)$]

$f(x) = (1+\Delta)^u y_0$ [for forward $E = 1+\Delta$]

$$= \left[1 + u\Delta + \frac{u(u-1)}{2!}\Delta^2 + \frac{u(u-1)(u-2)}{3!}\Delta^3 + \dots \right] y_0$$

Binomial expansion of $(1+\Delta)^u$.

$$f(x) = y_0 + u\Delta y_0 + \frac{u(u-1)}{2!}\Delta^2 y_0 + \frac{u(u-1)(u-2)}{3!}\Delta^3 y_0 + \dots \quad \text{— (2)}$$

Here; Eq^n (2) is Newton forward Interpolation formula #

$\boxed{\nabla \text{- nebla}}$ $\boxed{\Delta \text{- delta}}$

* Newton Forward Interpolation

Dhornidhar Tha

Let us consider $y=f(x)$ and $y_0, y_1, \ldots y_n$ are value corresponding to point $x_0, x_0+h, x_0+2h, \ldots x_0+nh$. Suppose we want to find ~~$x=x_0+uh$~~ $f(x)=y$ at point $x=x_0+uh$

$$\text{i.e; } u = \frac{x-x_0}{h}$$

By the definition of shift operator E we have

$$E^u f(x) = f(x+uh) \quad \text{— (1)}$$

$$\text{i.e; } E^u f(x_0) = f(x_0+uh) \quad [\text{At point } x=x_0]$$

$$\text{or; } f(x_0+uh) = E^u f(x_0)$$

$$\text{or; } f(x) = E^u y_0 \quad \left[\text{Here; } x = x_0+uh \ \& \ y_0 = f(x_0)\right]$$

$$f(x) = (1+\Delta)^u y_0 \quad [\text{for forward } E = 1+\Delta]$$

$$= \left[1 + u\Delta + \frac{u(u-1)}{2!}\Delta^2 + \frac{u(u-1)(u-2)}{3!}\Delta^3 + \ldots \right] y_0$$

Binomial expansion of $(1+\Delta)^u$.

$$f(x) = y_0 + u\Delta y_0 + \frac{u(u-1)}{2!}\Delta^2 y_0 + \frac{u(u-1)(u-2)}{3!}\Delta^3 y_0 + \ldots \quad \text{— (2)}$$

Here; Eq^n (2) is Newton forward Interpolation formula #

$\boxed{\nabla \text{- nebla}}$ $\boxed{\Delta \text{- delta}}$

Newton Backward Interpolation

Let us Consider $y=f(x)$ and $y_0, y_1, \ldots y_n$ are value corresponding to $x_n, x_n+h, x_n+2h, \ldots x_n+nh$

Suppose, we want to find $f(x)=y$ at point

$x=x_n+uh$ i.e; $u=\frac{x-x_n}{h}$

By the definition of shift operator E, we know that

$$E^u f(x) = f(x+uh) \quad \text{— (1)}$$

or; $f(x+uh) = E^u f(x)$

or; $f(x_n+uh) = E^u f(x_n)$ [At point $x=x_n$]

or; $f(x) = E^u f(x_n)$

or; $f(x) = (E^{-1})^{-u} f(x_n)$

or; $f(x) = (1-\nabla)^{-u} f(x_n)$ — (2)

Using Binomial expansion of $(1-\nabla)^{-u}$ in eqn (2) we have

$$f(x) = \left(1 + u\nabla + \frac{u(u+1)}{2!}\nabla^2 + \frac{u(u+1)(u+2)}{3!}\nabla^3 + \ldots\ldots\right) f(x_n)$$

$$= \left(1 + u\nabla + \frac{u(u+1)}{2!}\nabla^2 + \frac{u(u+1)(u+2)}{3!}\nabla^3 + \ldots\ldots\right) y_n$$

$$= y_n + u\nabla y_n + \frac{u(u+1)}{2!}\nabla^2 y_n + \frac{u(u+1)(u+2)}{3!}\nabla^3 y_n + \ldots\ldots \quad \text{— (3)}$$

The obtained eqn (3) is newton Backward interpolation formula.

Central difference formula:-

Bessel's Interpolation formula:-

$$
\begin{array}{cccccc}
\vdots & \vdots & & & & \\
x_{-2} & y_{-2} & & & & \\
 & & \Delta y_{-2} & & & \\
 & & & \Delta^2 y_{-2} & & \\
 & & & & \Delta^3 y_{-2} & \\
x_{-1} & y_{-1} & & & & \Delta^4 y_{-2} \\
 & & \Delta y_{-1} & & & \\
 & & & \Delta^2 y_{-1} & & \\
x_0 & y_0 & & & \Delta^3 y_{-1} & \\
 & & \Delta y_0 & & & \Delta^4 y_{-1} \\
 & & & \Delta^2 y_0 & & \\
x_1 & y_1 & & & & \\
 & & \Delta y_1 & & & \\
x_2 & y_2 & & & & \\
\vdots & \vdots & & & &
\end{array}
$$

Here; Bessel's formula can be assumed in the form;

$$y_p = \frac{y_0 + y_1}{2} + B_1 \Delta y_0 + B_2 \frac{\Delta^2 y_{-1} + \Delta^2 y_0}{2} + B_3 \Delta^3 y_{-1} + B_4 \frac{\Delta^4 y_{-2} + \Delta^4 y_{-1}}{2} + \cdots$$

$$= \frac{y_0}{2} + \frac{y_1}{2} + B_1 \Delta y_0 + B_2 \frac{\Delta^2 y_{-1} + \Delta^2 y_0}{2} + B_3 \Delta^3 y_{-1} + B_4 \frac{\Delta^4 y_{-2} + \Delta^4 y_{-1}}{2} + \cdots$$

$$= y_0 + \frac{y_1}{2} - \frac{y_0}{2} + B_1 \Delta y_0 + B_2 \frac{\Delta^2 y_{-1} + \Delta^2 y_0}{2} + B_3 \Delta^3 y_{-1} + B_4 \frac{\Delta^4 y_{-2} + \Delta^4 y_{-1}}{2} + \cdots \quad (1)$$

$$y_p = y_0 + \left(B_1 - \frac{1}{2}\right) \Delta y_0 + B_2 \frac{\Delta^2 y_{-1} + \Delta^2 y_0}{2} + B_3 \Delta^3 y_{-1} + B_4 \frac{\Delta^4 y_{-2} + \Delta^4 y_{-1}}{2} + \cdots$$

Using Gauss forward formula we obtain

$$\left.\begin{aligned} \beta_1 + \tfrac{1}{2} &= p \\ \beta_2 &= \frac{p(p-1)}{2!} \\ \beta_3 &= \frac{p(p-1)(p-\tfrac{1}{2})}{3!} \\ \beta_4 &= \frac{(p+1)p(p-1)(p-2)}{4!} \\ &\vdots \end{aligned}\right\} \quad — (2)$$

Hence Bessel interpolation formula may be cont... as;

$$y_p = y_0 + p\Delta y_0 + \frac{p(p-1)}{2!}\frac{\Delta^2 y_{-1} + \Delta^2 y_0}{2} + \frac{p(p-1)(p-\tfrac{1}{2})}{3!}\Delta^3 y_{-1} + \frac{(p+1)p(p-1)(p-2)}{4!}\frac{\Delta^4 y_{-2} + \Delta^4 y_{-1}}{2} + \cdots \quad — (3)$$

Eq^n (3) is the Bessel's Interpolation formula

Everett's formula

This is an extensively used interpolation formula and uses only even order difference, as shown in following table:

			$\Delta^2 y_{-1}$		$\Delta^4 y_{-2}$		$\Delta^4 y_{-3}$	
x_0	y_0	$-$		$-$		$-$		$\ldots$
x_1	y_1		$\Delta^2 y_0$		$\Delta^4 y_{-1}$		$\Delta^4 y_{-2}$	

Hence the formula can ~~be~~ have the form.

$$y_p = E_0 y_0 + E_2 \Delta^2 y_{-1} + E_4 \Delta^4 y_{-2} + \cdots + F_0 y_1 + F_2 \Delta^2 y_0 + F_4 \Delta^4 y_{-1} + \cdots \quad -(1)$$

The coefficient $E_0, E_2, E_4, F_0, F_2, F_4 \ldots$ can be determined by the same method as in preceding cases as we obtain.

$$\left.\begin{aligned} E_0 &= 1-p = q & F_0 &= p \\ E_2 &= \frac{q(q^2-1^2)}{3!} & F_1 &= \frac{p(p^2-1^2)}{3!} \\ E_4 &= \frac{q(q^2-1^2)(q^2-2^2)}{5!} & F_2 &= \frac{p(p^2-1^2)(p^2-2^2)}{5!} \end{aligned}\right\} -(2)$$

Hence; Everett's formula is given by:

$$\left.\begin{aligned} y_p = {} & q y_0 + \frac{q(q^2-1^2)}{3!}\Delta^2 y_{-1} + \frac{q(q^2-1^2)(q^2-2^2)}{5!}\Delta^4 y_{-2} + \cdots \\ & + p y_1 + \frac{p(p^2-1^2)}{3!}\Delta^2 y_0 + \frac{p(p^2-1^2)(p^2-2^2)}{5!}\Delta^4 y_{-1} + \cdots \end{aligned}\right\} (3)$$

Thus, the eqn (3) is Everett's formula! where
$1-p=q$

<u>Relation between Bessel's & Everett's formula:-</u>

To see we start with Bessel's formula:-

$$y_p = y_0 + p\Delta y_0 + \frac{p(p-1)}{2!}\frac{\Delta^2 y_{-1} - \Delta^2 y_0}{2} + \frac{p(p-1)(p-\frac{1}{2})}{3!}\Delta^3 y_{-1}$$

$$+ \frac{(p+1)p(p-1)(p-2)}{4!}\frac{\Delta^4 y_{-2} + \Delta^4 y_{-1}}{2} + \cdots$$

$$\Rightarrow y_0 + p(y_1 - y_0) + \frac{p(p-1)}{2!}\frac{\Delta^2 y_{-1} - \Delta^2 y_0}{2} + \frac{p(p-1)(p-\frac{1}{2})}{3!}(\Delta^2 y_0 - \Delta^2 y_{-1})$$

$$+ \frac{(p+1)p(p-1)(p-2)}{4!}\frac{\Delta^4 y_{-2} - \Delta^4 y_{-1}}{2}$$

$$\Rightarrow (1-p)y_0 + \left[\frac{p(p-1)}{4} - \frac{p(p-1)(p-\frac{1}{2})}{6}\right]\Delta^2 y_{-1} + \cdots$$

$$+ py_1 + \left[\frac{p(p-1)}{4} + \frac{p(p-1)(p-\frac{1}{2})}{6}\right]\Delta^2 y_0 + \cdots$$

$$\Rightarrow qy_0 + \frac{q(q^2-1^2)}{3!}\Delta^2 y_{-1} + \cdots + py_1 + \frac{p(p^2-1^2)}{3!}\Delta^2 y_0$$

which is everetes formula truncated after second difference

Note :- ~~Newton~~ Bessel's Interpolation formula is similar to Newton's forward and Backward Interpolation formula where the difference between each value of x is same, i.e interval gap of h is fixed.

when interval gap is not same we can use, Newton divide difference, Lagrange interpolation formula.

3 Curve fitting :-

(i) Fitting linear equation

Fitting a straight line is the simplest approach of regression analysis

Let us consider the mathematical equation for straight line

$$y = a + bx = f(x) \quad \text{— (I)}$$

Here; a is intercept
b is slope

Consider a point (x_i, y_i). The vertical distance of this point from the line $f(x) = a + bx$ is the error q_i

$$q_i = y_i - f(x_i)$$
$$= y_i - a - bx_i \quad \text{— (II)}$$

There are various approach for fitting a "best" line through the data. They include

(1) Minimizing the sum of error i.e, minimise

$$\sum q_i = \sum (y_i - a - bx_i) \quad \text{— (III)}$$

(2) Minimise the sum of absolute value of errors

$$\sum |q_i| = \sum |(y_i - a - bx_i)| \quad \text{— (IV)}$$

(3) Minimise the sum of squares of errors.

$$\sum q_i^2 = \sum (y_i - a - bx_i)^2 \quad \text{— (V)}$$

The first two strategy do not yield a unique line for a given set of data. The third strategy overcomes this problem & gurantees a unique line.

The technique of minimizing the sum of square of errors is known as least square regression