Interpolations

Ghulam Nabi Dar
Department of Physics KU Srinagar

August 29, 2020

Outline

1. Interpolations
   - How to do Interpolations: Motivations
   - **Methods of finding a function (Polynomial)**
   - **Direct Method**
   - Interpolation Techniques: Langrange Interpolation
     - Langrane: Quadratic Interpolation
     - Interpolation:
   - Assignment
   - Issues with Langrange interpolation formula
   - Forward Difference Table
   - Backward Difference Table
2. Newtons forward interpolation formula (NFIF)
3. Newtons Backward interpolation formula

Interpolations

Interpolations

- Interpolation is a technique to fit the data set. However there is difference between curve fitting and interpolation.

Interpolations

- Interpolation is a technique to fit the data set. However there is difference between curve fitting and interpolation.
- It is a study of algorithms using numerical approximations for mathematical analysis.

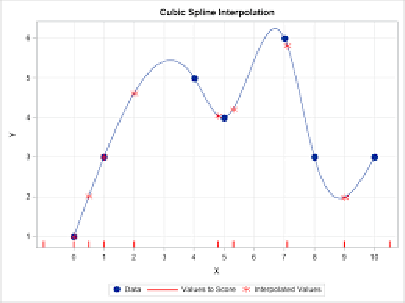
Interpolations

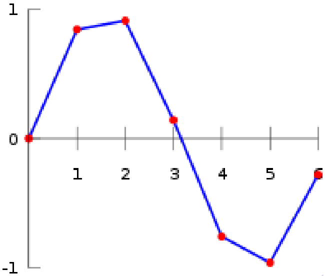
- Interpolation is a technique to fit the data set. However there is difference between curve fitting and interpolation.
- It is a study of algorithms using numerical approximations for mathematical analysis.
- In this course we will see number of methods for fitting a given data.

Interpolations

- Interpolation is a technique to fit the data set. However there is difference between curve fitting and interpolation.
- It is a study of algorithms using numerical approximations for mathematical analysis.
- In this course we will see number of methods for fitting a given data.
- Each method involves some amount of error and looking or exploring for improvement gives birth to a new set of techniques or methods.

Interpolation





cubic interpolation

How to do interpolation: Motivations

How to do interpolation: Motivations

- To find a *polynomial* $f(x)$ or $p_n(x)$ for a given data set (x_i, y_i) in such a way that at given values of x

$$f(x_i) = y_i = P_n(x)$$

How to do interpolation: Motivations

- To find a *polynomial* $f(x)$ or $p_n(x)$ for a given data set (x_i, y_i) in such a way that at given values of x

$$f(x_i) = y_i = P_n(x)$$

- $i = 0, 1, 2, \cdots$ ....n

How to do interpolation: Motivations

- To find a *polynomial* $f(x)$ or $p_n(x)$ for a given data set (x_i, y_i) in such a way that at given values of x

$$f(x_i) = y_i = P_n(x)$$

- $i = 0, 1, 2, \cdots$ ....n
- For example

How to do interpolation: Motivations

- To find a *polynomial* $f(x)$ or $p_n(x)$ for a given data set (x_i, y_i) in such a way that at given values of x

$$f(x_i) = y_i = P_n(x)$$

- $i = 0, 1, 2, \cdots$ ....n
- For example
- $$f(x_0) = y_0, f(x_1) = y_1, f(x_2) = y_2, \cdots f(x_n) = y_n$$

Missing term estimation using interpolation

Missing term estimation using interpolation

- This polynomial is used to find the missing terms in the data set or predict the expected values of y_i for x_i

Missing term estimation using interpolation

- This polynomial is used to find the missing terms in the data set or predict the expected values of y_i for x_i
- If the missing term lies in between the given data set say x_0 to x_n

Missing term estimation using interpolation

- This polynomial is used to find the missing terms in the data set or predict the expected values of y_i for x_i
- If the missing term lies in between the given data set say x_0 to x_n The process is known as interpolation.

Missing term estimation using interpolation

- This polynomial is used to find the missing terms in the data set or predict the expected values of y_i for x_i
- If the missing term lies in between the given data set say x_0 to x_n The process is known as interpolation.
- If the missing term lies outside the given data set, the process is known as Extrapolation

Missing term estimation using interpolation

- This polynomial is used to find the missing terms in the data set or predict the expected values of y_i for x_i
- If the missing term lies in between the given data set say x_0 to x_n The process is known as interpolation.
- If the missing term lies outside the given data set, the process is known as Extrapolation
- The process involves to find the coefficients of fitted polynomial using given data set.

Missing term estimation : Interpolation

Missing term estimation : Interpolation

- As interpolation is the techniques of connecting points accordingly we have

Missing term estimation : Interpolation

- As interpolation is the techniques of connecting points accordingly we have
- linear fit (using two data point together for for fitting)

Missing term estimation : Interpolation

- As interpolation is the techniques of connecting points accordingly we have
- linear fit (using two data point together for for fitting)
- parabolic fit (using three data point)

Missing term estimation : Interpolation

- As interpolation is the techniques of connecting points accordingly we have
- linear fit (using two data point together for for fitting)
- parabolic fit (using three data point)
- cubic (using four data point)

Missing term estimation : Interpolation

- As interpolation is the techniques of connecting points accordingly we have
- linear fit (using two data point together for for fitting)
- parabolic fit (using three data point)
- cubic (using four data point)
- quad (using five data point)

Missing term estimation : Interpolation

- As interpolation is the techniques of connecting points accordingly we have
- linear fit (using two data point together for for fitting)
- parabolic fit (using three data point)
- cubic (using four data point)
- quad (using five data point)
- and n^{th} degree polynomial using $n + 1$ data points

Methods for finding the polynomial- Direct Method-1

Methods for finding the polynomial- Direct Method-1

- For a given data set of $(n+1)$ points $x_i, y_i, i = 0, 1, 2, \cdots n$

Methods for finding the polynomial- Direct Method-1

- For a given data set of $(n+1)$ points $x_i, y_i, i = 0, 1, 2, \cdots n$
- Let

$$P_n(x) = f(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4 + .. + a_n x^n$$

is the required polynomial setisfying the given data such that

Methods for finding the polynomial- Direct Method-1

- For a given data set of $(n+1)$ points $x_i, y_i, i = 0, 1, 2, \cdots n$
- Let

$$P_n(x) = f(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4 + .. + a_n x^n$$

is the required polynomial setisfying the given data such that

- $P_n(x_i) = y_i$ for each i , we can write for example

Finding Polynomial...(contd...)

Finding Polynomial...(contd...)

- $$P_n(x_0) = f(x_0) = a_0 + a_1x_0 + a_2x_0^2 + a_3x_0^3 + .. + a_nx_0^n$$

Finding Polynomial...(contd...)

- $$P_n(x_0) = f(x_0) = a_0 + a_1x_0 + a_2x_0^2 + a_3x_0^3 + .. + a_nx_0^n$$
- $$P_n(x_1) = f(x_1) = a_0 + a_1x_1 + a_2x_1^2 + a_3x_1^3 + .. + a_nx_1^n$$

Finding Polynomial...(contd...)

- $$P_n(x_0) = f(x_0) = a_0 + a_1 x_0 + a_2 x_0^2 + a_3 x_0^3 + .. + a_n x_0^n$$
- $$P_n(x_1) = f(x_1) = a_0 + a_1 x_1 + a_2 x_1^2 + a_3 x_1^3 + .. + a_n x_1^n$$
- $$P_n(x_2) = f(x_2) = a_0 + a_1 x_2 + a_2 x_2^2 + a_3 x_2^3 + .. + a_n x_2^n$$

. . . .

Finding Polynomial...(contd...)

- $$P_n(x_0) = f(x_0) = a_0 + a_1 x_0 + a_2 x_0^2 + a_3 x_0^3 + .. + a_n x_0^n$$
- $$P_n(x_1) = f(x_1) = a_0 + a_1 x_1 + a_2 x_1^2 + a_3 x_1^3 + .. + a_n x_1^n$$
- $$P_n(x_2) = f(x_2) = a_0 + a_1 x_2 + a_2 x_2^2 + a_3 x_2^3 + .. + a_n x_2^n$$

. . . .

- $$P_n(x_n) = f(x_n) = a_0 + a_1 x_n + a_2 x_n^2 + a_3 x_n^3 + .. + a_n x_n^n$$

Finding Polynomial...(contd...)

Finding Polynomial...(contd...)

- this can be solved for the coefficients $a_0, a_1, a_2, \cdots a_n$ and the polynomial is obtained.

Finding Polynomial...(contd...)

- this can be solved for the coefficients $a_0, a_1, a_2, \cdots a_n$ and the polynomial is obtained.
- We can write above set of equations in a matrix form and write the value of given parameters and solve for coefficients

Finding Polynomial...(contd...)

- this can be solved for the coefficients $a_0, a_1, a_2, \cdots a_n$ and the polynomial is obtained.
- We can write above set of equations in a matrix form and write the value of given parameters and solve for coefficients
- $$\begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ \vdots \\ f_n \end{bmatrix} = \begin{bmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ 1 & x_2 & x_2^2 & \cdots & x_2^n \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{bmatrix} \times \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}$$

Example

Example

Example

- Home Work

Example

- Home Work
- Let x_i and y_i is given as $x_i = 1, 2, 3$ and $y_i = 1, 4, 9$. Fit a polynomial through direct method. Write your expression and find $p_n(x)$ at $x = 4$

Interpolation Techniques: Langrange Interpolation

Interpolation Techniques: Langrange Interpolation

- Aim is to determine the polynomial for a data set which is not equally spaced.

Interpolation Techniques: Langrange Interpolation

- Aim is to determine the polynomial for a data set which is not equally spaced.

- Let x_i amd f_i , where $i = 0, 1, 2, 3, \cdots n$ is a given data set with arbitary spaced.

Interpolation Techniques: Langrange Interpolation

- Aim is to determine the polynomial for a data set which is not equally spaced.

- Let x_i amd f_i , where $i = 0, 1, 2, 3, \cdots n$ is a given data set with arbitary spaced.

- *Langrange* idea is to multiply each f_j by a *polynomial* that is just 1 at x_j and just 0 elsewhere other than x_j .

Interpolation Techniques: Langrange Interpolation

- Aim is to determine the polynomial for a data set which is not equally spaced.
- Let x_i amd f_i , where $i = 0, 1, 2, 3, \cdots n$ is a given data set with arbitary spaced.
- *Langrange* idea is to multiply each f_j by a *polynomial* that is just 1 at x_j and just 0 elsewhere other than x_j .
- Then take sum of these $n + 1$ *polynomials* to get the *unique* interpolation formula of degree n or else. Let us work it out in steps.

Langrange Linear Interpolation

Langrange Linear Interpolation

- Consider two point interpolation through (x_0, y_o) and (x_1, y_1).

Langrange Linear Interpolation

- Consider two point interpolation through (x_0, y_o) and (x_1, y_1).
- Then the *Linear Langrange interpolation* is

Langrange Linear Interpolation

- Consider two point interpolation through (x_0, y_o) and (x_1, y_1).
- Then the *Linear Langrange interpolation* is
- $$P_1 = L_0(x)y_0 + L_1(x)y_1$$

Langrange Linear Interpolation

- Consider two point interpolation through (x_0, y_o) and (x_1, y_1).
- Then the *Linear Langrange interpolation* is
- $$P_1 = L_0(x)y_0 + L_1(x)y_1$$
- where $L_0(x)$ and $L_1(x)$ is a polynoial given as

Langrange Linear Interpolation

- Consider two point interpolation through (x_0, y_o) and (x_1, y_1).
- Then the *Linear Langrange interpolation* is
- $$P_1 = L_0(x)y_0 + L_1(x)y_1$$
- where $L_0(x)$ and $L_1(x)$ is a polynoial given as
- $L_0(x) = 1$ at $x = x_0$ and $\mathrm{L}_0(x) = 0$ eles where and

Langrange Linear Interpolation

- Consider two point interpolation through (x_0, y_o) and (x_1, y_1).
- Then the *Linear Langrange interpolation* is
- $$P_1 = L_0(x)y_0 + L_1(x)y_1$$
- where $L_0(x)$ and $L_1(x)$ is a polynoial given as
- $L_0(x) = 1$ at $x = x_0$ and $\mathrm{L}_0(x) = 0$ eles where and
- $L_1(x) = 1$ *at* x= x_1 and $\mathrm{L}_1(x) = 0$ eles where

Langrange Linear Interpolation

- Consider two point interpolation through (x_0, y_o) and (x_1, y_1).
- Then the *Linear Langrange interpolation* is
- $$P_1 = L_0(x)y_0 + L_1(x)y_1$$
- where $L_0(x)$ and $L_1(x)$ is a polynoial given as
- $L_0(x) = 1$ at $x = x_0$ and $\mathrm{L}_0(x) = 0$ eles where and
- $L_1(x) = 1$ *at* $\mathrm{x}= \mathrm{x}_1$ and $\mathrm{L}_1(x) = 0$ eles where
- the expressions for polynomials directed by the rule are

Langrange Linear Interpolation

Langrange Linear Interpolation

- $$L_0(x) = \frac{x - x_1}{x_0 - x_1}$$

Langrange Linear Interpolation

- $$L_0(x) = \frac{x - x_1}{x_0 - x_1}$$

- $$L_1(x) = \frac{x - x_0}{x_1 - x_0}$$

Langrange Linear Interpolation

- $$L_0(x) = \frac{x - x_1}{x_0 - x_1}$$

- $$L_1(x) = \frac{x - x_0}{x_1 - x_0}$$

- with this our solution or Linear interpolation formula is

Langrange Linear Interpolation

- $$L_0(x) = \frac{x - x_1}{x_0 - x_1}$$
- $$L_1(x) = \frac{x - x_0}{x_1 - x_0}$$
- with this our solution or Linear interpolation formula is
- $$P_1(x) = (\frac{x - x_0}{x_0 - x_1})y_0 + (\frac{x - x_0}{x_1 - x_0})y_1$$

Assignment1:

Assignment1:

Assignment1:

- Replace x_0 by x_1 and x_1 by x_2

Assignment1:

- Replace x_0 by x_1 and x_1 by x_2
- Show above equation $P_1(x)$ is a straight line given as (in terms of x and y)

Assignment1:

- Replace x_0 by x_1 and x_1 by x_2
- Show above equation $P_1(x)$ is a straight line given as (in terms of x and y)
- $$y - y_1 = \frac{y_2 - y_1}{x_2 - x_1}(x - x_1)$$

Assignment1:

- Replace x_0 by x_1 and x_1 by x_2
- Show above equation $P_1(x)$ is a straight line given as (in terms of x and y)
- $$y - y_1 = \frac{y_2 - y_1}{x_2 - x_1}(x - x_1)$$
- Which is like
$$y = mx + c$$

Langrange interpolation: Quadratic Equation

Langrange interpolation: Quadratic Equation

- Second degree interpolation takes care of three points instead of two as in linear interpolation.

Langrange interpolation: Quadratic Equation

- Second degree interpolation takes care of three points instead of two as in linear interpolation.

- for example consider the data string as $(x_0, y_0), (x_1, y_1), (x_2, y_2)$

Langrange interpolation: Quadratic Equation

- Second degree interpolation takes care of three points instead of two as in linear interpolation.

- for example consider the data string as $(x_0, y_0), (x_1, y_1), (x_2, y_2)$

  Then $P_2(x)$ is obtained using Langrange Interpolation rule as

Langrange interpolation: Quadratic Equation

- Second degree interpolation takes care of three points instead of two as in linear interpolation.

- for example consider the data string as $(x_0, y_0), (x_1, y_1), (x_2, y_2)$

  Then $P_2(x)$ is obtained using Langrange Interpolation rule as

- $$P_2(x) = L_0(x)y_0 + L_1(x)y_1 + L_2(x)y_2$$

Langrange interpolation: Quadratic Equation

- Second degree interpolation takes care of three points instead of two as in linear interpolation.

- for example consider the data string as $(x_0, y_0), (x_1, y_1), (x_2, y_2)$

  Then $P_2(x)$ is obtained using Langrange Interpolation rule as

- $$P_2(x) = L_0(x)y_0 + L_1(x)y_1 + L_2(x)y_2$$

- such that each $L_0(x), L_1(x)$ and $\mathrm{L}_2(x)$ is a polynoial given respectively as

Langrange: Quad Equation

Langrange: Quad Equation

- $L_0(x) = 1$ or o at $x = x_0$ and elesewhere x_0

Langrange: Quad Equation

- $L_0(x) = 1$ or o at $x = x_0$ and elesewhere x_0
- $L_1(x) = 1$ or o at $x = x_1$ and elesewhere x_1

Langrange: Quad Equation

- $L_0(x) = 1$ or o at $x = x_0$ and elesewhere x_0
- $L_1(x) = 1$ or o at $x = x_1$ and elesewhere x_1
- $L_2(x) = 1$ or o at $x = x_2$ and elesewhere x_2

Langrange: Quad Equation

- $L_0(x) = 1$ or o at $x = x_0$ and elesewhere x_0
- $L_1(x) = 1$ or o at $x = x_1$ and elesewhere x_1
- $L_2(x) = 1$ or o at $x = x_2$ and elesewhere x_2
- The only choice to have the above is as

$$L_0(x) = \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} = \frac{l_0(x)}{l_0(x_0)}$$

Langrange: Quad Equation

Langrange: Quad Equation

- $$L_1(x) = \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} = \frac{l_1(x)}{l_0(x_1)}$$

Langrange: Quad Equation

- $$L_1(x) = \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} = \frac{l_1(x)}{l_0(x_1)}$$

- $$L_2(x) = \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)} = \frac{l_2(x)}{l_0(x_2)}$$

Langrange: Quad Equ contd...

Langrange: Quad Equ contd...

- In general

$$L_k(x_j) = \frac{(x_j - x_l)(x_j - x_m)}{(x_j - x_l)(x_j - x_m)}$$

Langrange: Quad Equ contd...

- In general

$$L_k(x_j) = \frac{(x_j - x_l)(x_j - x_m)}{(x_j - x_l)(x_j - x_m)}$$

- Note that $L_k(x_j) = 0$ if $k \neq j$ otherwise 1

Langrange: Cubic Interpolation

Langrange: Cubic Interpolation

- Similar to above interpolation here we take 04 points together and develop a polynoial of degree 3 as per *Langrange rule* as

Langrange: Cubic Interpolation

- Similar to above interpolation here we take 04 points together and develop a polynoial of degree 3 as per *Langrange rule* as
-
$$P_3(x) = L_0(x)y_0 + L_1(x)y_1 + L_2(x)y_2 + L_3(x)y_3$$

  with the parameter defined as

Langrange: Cubic Interpolation

- Similar to above interpolation here we take 04 points together and develop a polynoial of degree 3 as per *Langrange rule* as
- $$P_3(x) = L_0(x)y_0 + L_1(x)y_1 + L_2(x)y_2 + L_3(x)y_3$$

  with the parameter defined as
- $$L_0(x) = \frac{(x - x_1)(x - x_2)(x - x_3)}{(x_0 - x_1)(x_0 - x_2)(x_0 - x_3)} = \frac{l_0(x)}{l_0(x_0)}$$

Langrange: Cubic Interpolation

- Similar to above interpolation here we take 04 points together and develop a polynoial of degree 3 as per *Langrange rule* as
- $$P_3(x) = L_0(x)y_0 + L_1(x)y_1 + L_2(x)y_2 + L_3(x)y_3$$

  with the parameter defined as
- $$L_0(x) = \frac{(x - x_1)(x - x_2)(x - x_3)}{(x_0 - x_1)(x_0 - x_2)(x_0 - x_3)} = \frac{l_0(x)}{l_0(x_0)}$$
- $$L_1(x) = \frac{(x - x_0)(x - x_2)(x - x_3)}{(x_1 - x_0)(x_1 - x_2)(x_1 - x_3)} = \frac{l_1(x)}{l_1(x_1)}$$

Cubic. Interpolation

Cubic. Interpolation

- $$L_2(x) = \frac{(x - x_0)(x - x_1)(x - x_3)}{(x_2 - x_0)(x_2 - x_1)(x_2 - x_3)} = \frac{l_2(x)}{l_2(x_2)}$$

Cubic. Interpolation

- $$L_2(x) = \frac{(x - x_0)(x - x_1)(x - x_3)}{(x_2 - x_0)(x_2 - x_1)(x_2 - x_3)} = \frac{l_2(x)}{l_2(x_2)}$$

- $$L_3(x) = \frac{(x - x_0)(x - x_1)(x - x_2)}{(x_3 - x_0)(x_3 - x_1)(x_3 - x_2)} = \frac{l_3(x)}{l_3(x_3)}$$

Quad. Interpolation

Quad. Interpolation

- In short we can write

$$P_3(x) = \sum_{k=0}^{3} L_k(x) y_k = L_0(x)y_0 + L_1(x)y_1 + L_2(x)y_2 + L_3(x)y_3$$

Quad. Interpolation

- In short we can write

$$P_3(x) = \sum_{k=0}^{3} L_k(x) y_k = L_0(x)y_0 + L_1(x)y_1 + L_2(x)y_2 + L_3(x)y_3$$

- The general n^{th} degree polynoial is given as

Quad. Interpolation

- In short we can write

$$P_3(x) = \sum_{k=0}^{3} L_k(x)y_k = L_0(x)y_0 + L_1(x)y_1 + L_2(x)y_2 + L_3(x)y_3$$

- The general n^{th} degree polynoial is given as
-

$$P_n(x) = \sum_{k=0}^{n} L_k(x)y_k = L_0(x)y_0 + L_1(x)y_1 + L_2(x)y_2 + \cdots + L_n(x)y_n$$

n order. Interpolation

- $$P_n(x) = \sum_{k=0}^{n} \frac{l_k(x)}{l_k(x_k)} y_k$$

n order. Interpolation

- $$P_n(x) = \sum_{k=0}^{n} \frac{l_k(x)}{l_k(x_k)} y_k$$
- where $L_k(x) = \frac{l_k(x)}{l_k(x_k)} y_k$. Such that $L_k(x_k) = 1$ and 0 at other nodes.

n order. Interpolation

- $$P_n(x) = \sum_{k=0}^{n} \frac{l_k(x)}{l_k(x_k)} y_k$$
- where $L_k(x) = \frac{l_k(x)}{l_k(x_k)} y_k$. Such that $L_k(x_k) = 1$ and 0 at other nodes.
- Note down for polynoial of degree n

n order. Interpolation

- $$P_n(x) = \sum_{k=0}^{n} \frac{l_k(x)}{l_k(x_k)} y_k$$
- where $L_k(x) = \frac{l_k(x)}{l_k(x_k)} y_k$. Such that $L_k(x_k) = 1$ and 0 at other nodes.
- Note down for polynoial of degree n
- $$l_0(x) = (x - x_1)(x - x_2)(x - x_3) \cdots (x - x_n)$$

n order. Interpolation

- $$P_n(x) = \sum_{k=0}^{n} \frac{l_k(x)}{l_k(x_k)} y_k$$
- where $L_k(x) = \frac{l_k(x)}{l_k(x_k)} y_k$. Such that $L_k(x_k) = 1$ and 0 at other nodes.
- Note down for polynoial of degree n
- $$l_0(x) = (x - x_1)(x - x_2)(x - x_3) \cdots (x - x_n)$$
- $$l_1(x) = (x - x_0)(x - x_2)(x - x_3) \cdots (x - x_n)$$

n order. Interpolation

- $$P_n(x) = \sum_{k=0}^{n} \frac{l_k(x)}{l_k(x_k)} y_k$$
- where $L_k(x) = \frac{l_k(x)}{l_k(x_k)} y_k$. Such that $L_k(x_k) = 1$ and 0 at other nodes.
- Note down for polynoial of degree n
- $$l_0(x) = (x - x_1)(x - x_2)(x - x_3) \cdots (x - x_n)$$
- $$l_1(x) = (x - x_0)(x - x_2)(x - x_3) \cdots (x - x_n)$$
- $$l_2(x) = (x - x_0)(x - x_1)(x - x_3) \cdots (x - x_n)$$

n^{th}. *Interpolation*

n^{th}.*Interpolation*

- $$l_3(x) = (x - x_0)(x - x_1)(x - x_2)\cdots(x - x_n)$$

nth. *Interpolation*

- $$l_3(x) = (x - x_0)(x - x_1)(x - x_2)\cdots(x - x_n)$$

- $$l_n(x) = (x - x_0)(x - x_1)(x - x_2)\cdots(x - x_{n-1})$$

n^{th}. *Interpolation*

- $$l_3(x) = (x - x_0)(x - x_1)(x - x_2)\cdots(x - x_n)$$

- $$l_n(x) = (x - x_0)(x - x_1)(x - x_2)\cdots(x - x_{n-1})$$

- Also note that or check

$$P_n(x_0) = y_0$$

$$P_n(x_1) = y_1$$

$$P_n(x_n) = y_n$$

Assignment

Assignment

- For the given table find the Langrange interpolation formula

$x_k \rightarrow$	1	0	2	3
$y_k \rightarrow$	-1	2	10	35

Assignment

- For the given table find the Langrange interpolation formula

$x_k \rightarrow$	1	0	2	3
$y_k \rightarrow$	-1	2	10	35

- Show that

$$P_3(x) = \frac{5}{3}x^3 - \frac{4}{3}x^2 + 2$$

Verify $P_3(x_0) = y_0 = -1$

Issues with Langrange Interpolation formula

Issues with Langrange Interpolation formula

- Amount of computations required is large. It involves $2(n+1)$ multiple / divisional and $(2n+1)$ additional operations

Issues with Langrange Interpolation formula

- Amount of computations required is large. It involves $2(n+1)$ multiple / divisional and $(2n+1)$ additional operations
- Where as power polynoial $a_0 + a_1 x + \cdots + a_n x^n$ involves only n multiple and n additional operations

Issues with Langrange Interpolation formula

- Amount of computations required is large. It involves $2(n+1)$ multiple / divisional and $(2n+1)$ additional operations
- Where as power polynoial $a_0 + a_1 x + \cdots + a_n x^n$ involves only n multiple and n additional operations
- Interpolation for additional value of x requires same amount of effort as first value. It does not take it from previous calculations and starts from begning.

Issues with Langrange Interpolation formula

- Amount of computations required is large. It involves $2(n+1)$ multiple / divisional and $(2n+1)$ additional operations
- Where as power polynoial $a_0 + a_1x + \cdots + a_nx^n$ involves only n multiple and n additional operations
- Interpolation for additional value of x requires same amount of effort as first value. It does not take it from previous calculations and starts from begning.
- If in the middle of calculations, number of interpolation points are changed . Previous results are not used.

Issues with Langrange Interpolation formula

- Amount of computations required is large. It involves $2(n+1)$ multiple / divisional and $(2n+1)$ additional operations
- Where as power polynoial $a_0 + a_1 x + \cdots + a_n x^n$ involves only n multiple and n additional operations
- Interpolation for additional value of x requires same amount of effort as first value. It does not take it from previous calculations and starts from begning.
- If in the middle of calculations, number of interpolation points are changed . Previous results are not used.
- And error is large

Newtons Method of Interpolation Technique

Newtons Method of Interpolation Technique

Newtons Method of Interpolation Technique

- Here in this method we are able to increase order of polynoial

Newtons Method of Interpolation Technique

- Here in this method we are able to increase order of polynoial
- whereas in Langrange entirely a new operation has to start up from the begning.

Newtons Method of Interpolation Technique

- Here in this method we are able to increase order of polynoial
- whereas in Langrange entirely a new operation has to start up from the begning.
- This method is used for equally spaced arguments $(x_{i+1} - x_i = h)$ say is always uniform.

Newtons Method of Interpolation Technique

- Here in this method we are able to increase order of polynoial
- whereas in Langrange entirely a new operation has to start up from the begning.
- This method is used for equally spaced arguments $(x_{i+1} - x_i = h)$ say is always uniform.
- Since Interpolation is needed to estimate the missing terms.

Newtons Method of Interpolation Technique

- Here in this method we are able to increase order of polynoial
- whereas in Langrange entirely a new operation has to start up from the begning.
- This method is used for equally spaced arguments $(x_{i+1} - x_i = h)$ say is always uniform.
- Since Interpolation is needed to estimate the missing terms.
- Missing terms can be in the begning , end or in the middle of given data.

Newtons Method of Interpolation Technique

- Here in this method we are able to increase order of polynoial
- whereas in Langrange entirely a new operation has to start up from the begning.
- This method is used for equally spaced arguments $(x_{i+1} - x_i = h)$ say is always uniform.
- Since Interpolation is needed to estimate the missing terms.
- Missing terms can be in the begning , end or in the middle of given data.
- for example through census i know the population from 1981 till 2011. If I am interested to know before 1981 or in the middle say 2004 or at the end around say 2014

Newtons Method of Interpolation Technique

- Here in this method we are able to increase order of polynoial
- whereas in Langrange entirely a new operation has to start up from the begning.
- This method is used for equally spaced arguments $(x_{i+1} - x_i = h)$ say is always uniform.
- Since Interpolation is needed to estimate the missing terms.
- Missing terms can be in the begning , end or in the middle of given data.
- for example through census i know the population from 1981 till 2011. If I am interested to know before 1981 or in the middle say 2004 or at the end around say 2014
- Accordingly we have three types of Newtons interpolation methods defined through difference tables, taken as

Difference Tables

Difference Tables

- Forward differences

Difference Tables

- Forward differences
- Backward differences

Difference Tables

- Forward differences
- Backward differences
- Central differences

Difference Tables

- Forward differences
- Backward differences
- Central differences
- Interpolations defined through differences depend on difference tables

Difference Tables

- Forward differences
- Backward differences
- Central differences
- Interpolations defined through differences depend on difference tables
- Before going to Newtons Methods we will learn first of difference tables

Difference Tables

- Forward differences
- Backward differences
- Central differences
- Interpolations defined through differences depend on difference tables
- Before going to Newtons Methods we will learn first of difference tables
- difference of dependent variable is involved say $y_i - y_{i+1}$ as per the tables .

Forward Difference Table

Forward Difference Table

-

x	y	Δ (first difference)	Δ^2 (second difference)	Δ^3(third difference)
x_0	y_0			
x_1	y_1	$\Delta y_0 = y_1 - y_0$		
x_2	y_2	$\Delta y_1 = y_2 - y_1$	$\Delta^2 y_0 = \Delta y_1 - \Delta y_0$	
x_3	y_3	$\Delta y_2 = y_3 - y_2$	$\Delta^2 y_1 = \Delta y_2 - \Delta y_1$	$\Delta^3 y_0 = \Delta^2 y_1 - \Delta^2 y_0$

Backward Difference Table

Backward Difference Table

- **Note: Symbolic representation y_0 which is subtracted.**

x	y	∇ (first difference)	∇^2 (second difference)	∇^3(third difference)
$\mathbf{x}_0$	$\mathbf{y}_0$			
$\mathbf{x}_1$	$\mathbf{y}_1$	$\nabla y_1 = y_1 - y_0$		
$\mathbf{x}_2$	$\mathbf{y}_2$	$\nabla y_2 = y_2 - y_1$	$\nabla^2 y_2 = \nabla y_2 - \nabla y_1$	
$\mathbf{x}_3$	$\mathbf{y}_3$	$\nabla y_3 = y_3 - y_2$	$\nabla^2 y_3 = \nabla y_3 - \nabla y_2$	$\nabla^3 y_3 = \nabla^2 y_3 - \nabla^2 y_2$

Assignment

Assignment

- **Note: Symbolic representation** y_1, y_2, y_3 **from which subtracted**
  Assignment: choose your data from x_0, y_0 **to** x_6, y_6**. Derive the result for**
  $\Delta y_0, \Delta y_4, \Delta^2 y_0, \Delta y_4, \Delta y_5, \Delta^4 y_0,$

And for

$\nabla y_1, \nabla y_4, \nabla^2 y_2, \nabla^3 y_3$

Note:

Assignment

- **Note: Symbolic representation y_1, y_2, y_3 from which subtracted**
  Assignment: choose your data from x_0, y_0 to x_6, y_6. Derive the result for
  $\Delta y_0, \Delta y_4, \Delta^2 y_0, \Delta y_4, \Delta y_5, \Delta^4 y_0,$

And for

$\nabla y_1, \nabla y_4, \nabla^2 y_2, \nabla^3 y_3$

Note:

- **See forward difference table end at y_0 used for forward difference interpolation formula.**

Assignment

- **Note: Symbolic representation y_1, y_2, y_3 from which subtracted**
  Assignment: choose your data from x_0, y_0 to x_6, y_6. Derive the result for
  $\Delta y_0, \Delta y_4, \Delta^2 y_0, \Delta y_4, \Delta y_5, \Delta^4 y_0,$

And for
$\nabla y_1, \nabla y_4, \nabla^2 y_2, \nabla^3 y_3$
Note:

- **See forward difference table end at y_0 used for forward difference interpolation formula.**
- **Backward difference table ends at y_n , the last data point used for Backward difference interpolation formula.**

Newtons forward interpolation formula (NFIF)

Newtons forward interpolation formula (NFIF)

- If $x_0, x_1, x_2, ..., x_k and$ $y_0, y_1, y_2, ...y_k$ are give set of observations with arguments equally spaced.

Newtons forward interpolation formula (NFIF)

- If $x_0, x_1, x_2, ..., x_k and$ $y_0, y_1, y_2, ...y_k$ are give set of observations with arguments equally spaced.
- The NFIF is given as

$$P_n(x) = y_0 + p\Delta y_0 + \frac{p(p-1)}{2!}\Delta^2 y_0 + \frac{p(p-1)(p-2)}{3!}\Delta^3 y_0 + \cdots \frac{p(p-1)(p-2)....(p-(n-1))}{n!}\Delta^n y_0$$

Newtons forward interpolation formula (NFIF)

- If $x_0, x_1, x_2, ..., x_k and$ $y_0, y_1, y_2, ...y_k$ are give set of observations with arguments equally spaced.
- The NFIF is given as

$$P_n(x) = y_0 + p\Delta y_0 + \frac{p(p-1)}{2!}\Delta^2 y_0 + \frac{p(p-1)(p-2)}{3!}\Delta^3 y_0 + \cdots \frac{p(p-1)(p-2)....(p-(n-1))}{n!}\Delta^n y_0$$

- where $p = \frac{x-x_0}{h}$, h a uniform spacing between arguments.

Newtons forward interpolation formula (NFIF)

Newtons forward interpolation formula (NFIF)

- Proof: Assume n^{th} degree polynoial for a given data as

Newtons forward interpolation formula (NFIF)

- Proof: Assume n^{th} degree polynoial for a given data as
- $$P_n(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \cdots + a_n(x - x_0)(x - x_1)(x - x_2)\cdots(x - x_{n-1})$$

Newtons forward interpolation formula (NFIF)

- Proof: Assume n^{th} degree polynoial for a given data as
-

$$P_n(x) = a_0+a_1(x-x_0)+a_2(x-x_0)(x-x_1)+\cdots+a_n(x-x_0)(x-x_1)(x-x_2)\cdots(x-x_{n-1})$$

- As per the requirement of passing through given points we have

x=x_0, $P_n(x_0) = a_0 = y_0$

Newtons forward interpolation formula (NFIF)

- Proof: Assume n^{th} degree polynoial for a given data as
-

$$P_n(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \cdots + a_n(x - x_0)(x - x_1)(x - x_2)\cdots(x - x_{n-1})$$

- As per the requirement of passing through given points we have

$$\text{x=x}_0,\ P_n(x_0) = a_0 = y_0$$

-

$$\text{x=x}_1,\ P_n(x_1) = y_1 = a_0 + a_1(x_1 - x_0) = y_0 + a_1 * h$$

Newtons forward interpolation formula (NFIF)

Newtons forward interpolation formula (NFIF)

- from this we get

$$a_1 = \frac{(y_1 - y_0)}{h} = \frac{\Delta y_0}{h}$$

Newtons forward interpolation formula (NFIF)

- from this we get

$$a_1 = \frac{(y_1 - y_0)}{h} = \frac{\Delta y_0}{h}$$

- Similarly at

$$x = x_2$$

$$P_n(x_2) = y_2 = a_0 + a_1(x_2 - x)) + a_2(x_2 - x_0)(x_2 - x_1) = y_0 + 2\Delta y_0 + 2h^2 a_2$$

Newtons forward interpolation formula (NFIF)

- from this we get

$a_1 = \frac{(y_1 - y_0)}{h} = \frac{\Delta y_0}{h}$

- Similarly at

$x = x_2$

$$P_n(x_2) = y_2 = a_0 + a_1(x_2 - x)) + a_2(x_2 - x_0)(x_2 - x_1) = y_0 + 2\Delta y_0 + 2h^2 a_2$$

-

$a_2 = \frac{\Delta^2 y_0}{2h^2}$

Newtons forward interpolation formula (NFIF)

Newtons forward interpolation formula (NFIF)

- And at

$$x = x_3$$

$$P_n(x_3) = y_3$$

gives expression for a_3 as

Newtons forward interpolation formula (NFIF)

- And at
$$x = x_3$$
$$P_n(x_3) = y_3$$
gives expression for a_3 as
- $$a_3 = \frac{\Delta^3 y_0}{3! h^3}$$
and soon

Newtons forward interpolation formula (NFIF)

- And at

$$x = x_3$$

$$P_n(x_3) = y_3$$

gives expression for a_3 as

- $$a_3 = \frac{\Delta^3 y_0}{3! h^3}$$

and soon

- therefore with all these coefficients so evaluated we end up with an expression involving various difference operators and y_0 called as Newtons forward interpolation polynomial as

Newtons forward interpolation formula (NFIF)

Newtons forward interpolation formula (NFIF)

- $$P_n(x) = y_0 + p\Delta y_0 + \frac{p(p-1)}{2!}\Delta^2 y_0 + \frac{p(p-1)(p-2)}{3!}\Delta^3 y_0 + \cdots + \frac{p(p-1)\cdots(p-(n-1))}{n!}\Delta^n y_0$$

Newtons forward interpolation formula (NFIF)

- $$P_n(x) = y_0 + p\Delta y_0 + \frac{p(p-1)}{2!}\Delta^2 y_0 + \frac{p(p-1)(p-2)}{3!}\Delta^3 y_0 + \cdots + \frac{p(p-1)\cdots(p-(n-1))}{n!}\Delta^n y_0$$

- where $p = \frac{x-x_0}{h}$
  The above formula is also known as Newton -Gregory forward polynomial. This is like a Binomial.

Newtons forward interpolation formula (NFIF)

Newtons forward interpolation formula (NFIF)

- Example : Home work : Develop forward difference table and hence a NFIF for

Newtons forward interpolation formula (NFIF)

- Example : Home work : Develop forward difference table and hence a NFIF for
-

x	y	Δ	Δ^2	Δ^3
2	-7			
4	-3			
6	6			

Newtons forward interpolation formula (NFIF)

- Example : Home work : Develop forward difference table and hence a NFIF for

x	y	Δ	Δ^2	Δ^3
2	-7			
4	-3			
6	6			

- Write NFIF and substitute the values from table and find the polynomial.

Newtons Backward interpolation formula (NBIF)

Newtons Backward interpolation formula (NBIF)

- $$P_n(x) = y_n + (x - x_n)\frac{\nabla y_n}{h} + \frac{1}{2!}(x - x_n)(x - x_{n-1})\frac{\nabla^2 y_n}{h^2} + \frac{1}{n!}(x - x_n)(x - x_{n-1})...(x - x_1)\frac{\nabla^n y_n}{h^n}$$

Newtons Backward interpolation formula (NBIF)

- $$P_n(x) = y_n + (x - x_n)\frac{\nabla y_n}{h} + \frac{1}{2!}(x - x_n)(x - x_{n-1})\frac{\nabla^2 y_n}{h^2} + \frac{1}{n!}(x - x_n)(x - x_{n-1})...(x - x_1)\frac{\nabla^n y_n}{h^n}$$
- Derive the above formula using the assumption

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

Newtons Backward interpolation formula (NBIF)

- $$P_n(x) = y_n + (x - x_n)\frac{\nabla y_n}{h} + \frac{1}{2!}(x - x_n)(x - x_{n-1})\frac{\nabla^2 y_n}{h^2} + \frac{1}{n!}(x - x_n)(x - x_{n-1})...(x - x_1)\frac{\nabla^n y_n}{h^n}$$
- Derive the above formula using the assumption

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

- At given values , polynomial has to pass through given points a required condition for interpolation.

Newtons Backward interpolation formula (NBIF)

- $$P_n(x) = y_n + (x - x_n)\frac{\nabla y_n}{h} + \frac{1}{2!}(x - x_n)(x - x_{n-1})\frac{\nabla^2 y_n}{h^2} + \frac{1}{n!}(x - x_n)(x - x_{n-1})...(x - x_1)\frac{\nabla^n y_n}{h^n}$$
- Derive the above formula using the assumption

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

- At given values , polynomial has to pass through given points a required condition for interpolation.
- Derive NBIF after determine all the coefficients (Home Work).

Newtons Backward interpolation formula (NBIF)

- $$P_n(x) = y_n + (x - x_n)\frac{\nabla y_n}{h} + \frac{1}{2!}(x - x_n)(x - x_{n-1})\frac{\nabla^2 y_n}{h^2} + \frac{1}{n!}(x - x_n)(x - x_{n-1})...(x - x_1)\frac{\nabla^n y_n}{h^n}$$
- Derive the above formula using the assumption

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

- At given values , polynomial has to pass through given points a required condition for interpolation.
- Derive NBIF after determine all the coefficients (Home Work).
- On putting $(x - x_n) = p * h$ and determining all the terms like $(x - x_n), (x - x_{n-1} = (p + 2) * h$ and $(x - x_{n-2}) = (p + 3) * h$ and soon . We can write the NBIF as

Newtons Backward interpolation formula (NBIF)

- $$P_n(x) = y_n + (x - x_n)\frac{\nabla y_n}{h} + \frac{1}{2!}(x - x_n)(x - x_{n-1})\frac{\nabla^2 y_n}{h^2} + \frac{1}{n!}(x - x_n)(x - x_{n-1})...(x - x_1)\frac{\nabla^n y_n}{h^n}$$
- Derive the above formula using the assumption

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

- At given values , polynomial has to pass through given points a required condition for interpolation.
- Derive NBIF after determine all the coefficients (Home Work).
- On putting $(x - x_n) = p * h$ and determining all the terms like $(x - x_n), (x - x_{n-1} = (p + 2) * h$ and $(x - x_{n-2}) = (p + 3) * h$ and soon . We can write the NBIF as
- $$P_n(x) = y_n + p\nabla y_n + \frac{p(p+1)}{2!}\nabla^2 y_n + \cdots + \frac{p(p+1)(p+2).....(p+n-1)}{n!}\nabla^n y_n$$

Newtons Backward interpolation formula (NBIF)

Newtons Backward interpolation formula (NBIF)

- Assignment: Derive above polynomial. And for the given table, estimate $P_n(7.5)$

Newtons Backward interpolation formula (NBIF)

- Assignment: Derive above polynomial. And for the given table, estimate $P_n(7.5)$

-

x	1	2	3	4	5	6	7	8
y	1	8	27	64	125	216	343	512

Newtons Backward interpolation formula (NBIF)

- Assignment: Derive above polynomial. And for the given table, estimate $P_n(7.5)$
x	1	2	3	4	5	6	7	8
y	1	8	27	64	125	216	343	512
- Note: As the point of estimate lies near end so NBIF will be the most suitable. Please evaluate the result by using NFIF also. Answer: $P_n(7.5) = 421.875$, Since from data it is clear that the table is showing cubic behavior which cab be checked also by calculating $(7.5)^3$

Home Work on interpolations

Home Work on interpolations

- Population of town is given below. Estimate the population in 1955 and 1985

year	1951	1961	1971	1981	1991
population	46	66	81	91	101

Home Work on interpolations

- Population of town is given below. Estimate the population in 1955 and 1985

year	1951	1961	1971	1981	1991
population	46	66	81	91	101

- Choose your interpolation as per the question i,e where will you use forward and where will you use Backward interpolation formula.

Home Work on interpolations

- Population of town is given below. Estimate the population in 1955 and 1985

year	1951	1961	1971	1981	1991
population	46	66	81	91	101

- Choose your interpolation as per the question i,e where will you use forward and where will you use Backward interpolation formula.
    - Answer: for 1955 = 54.85 And for 1985 = 96.84