Stirling and Bessel Interpolation

Nguyen Viet Anh 20200039

January 10, 2022

CONTENTS

GENERAL KNOWLEDGE

1. Difference

- Given a function $f(x)$ with n known data points with constant x-interval ($x_k - x_{k-1} = h$):

$$f(x_i) = y_i \quad \forall i \in [1, n]$$

- Difference at y_k (Δy_k) (difference of order 1):

$$\Delta y_k = y_{k+1} - y_k$$

- Difference of order p:

$$\Delta^p y_k = \Delta^{p-1} y_{k+1} - \Delta^{p-1} y_k$$

GENERAL KNOWLEDGE

2. Difference table

- A difference table represents the differences of all orders generated from the known set of data points, such as following:

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$	$\Delta^4 y$	$\dots$
$\vdots$	$\vdots$					
x_{-2}	y_{-2}					
		Δy_{-2}				
x_{-1}	y_{-1}		$\Delta^2 y_{-2}$			
		Δy_{-1}		$\Delta^3 y_{-2}$		
x_0	y_0		$\Delta^2 y_{-1}$		$\Delta^4 y_{-2}$	$\dots$
		Δy_0		$\Delta^3 y_{-1}$		
x_1	y_1		$\Delta^2 y_0$			
		Δy_1				
x_2	y_2					
$\vdots$	$\vdots$					

GENERAL KNOWLEDGE

3. Polynomial interpolation

- **Definition:** Interpolation which approximate a function $f(x)$ with a polynomial $P(x)$ (known as interpolating polynomial) based on a set of known data points $\{(x_i, y_i)\}_{i=1}^{n}$.
- $P(x_i) = f(x_i) \quad , i = \overline{1, n}$
- $\deg P = n - 1$ (n: number of data points)
- The deduction of polynomial $P(x)$ from a fixed set of data points $\{(x_i, y_i)\}_{i=1}^{n}$ is unique.

GENERAL KNOWLEDGE

4. Gauss Forward and Backward Interpolation

Suppose a set of $2n+1$ points is given: $\{(x_i, y_i)\}_{i=-n}^{n}$

- Gauss Forward Interpolation:

$$P(x_0+ht) = y_0 + t\Delta y_0 + \frac{t(t-1)}{2!}\Delta^2 y_{-1} + \frac{(t+1)t(t-1)}{3!}\Delta^3 y_{-1} + \frac{(t+1)t(t-1)(t-2)}{4!}\Delta^4 y_{-2} + \cdots + \frac{(t+n-1)(t+n-1)\ldots(t-n)}{(2n)!}\Delta^{2n} y_{-n}$$

$$\textbf{Error} = \left| \frac{\Delta^{2n} y_{-n}}{(2n)!}.t\prod_{i=1}^{n}(t^2 - i^2) \right|$$

- Gauss Backward Interpolation

$$P(x_0+th) = y_0 + t\Delta y_{-1} + \frac{t(t+1)}{2!}\Delta^2 y_{-1} + \frac{(t+1)t(t-1)}{3!}\Delta^3 y_{-2} + \frac{(t+2)(t+1)t(t-1)}{4!}\Delta^4 y_{-2} + \cdots + \frac{(t+n)(t+n-1)\ldots(t-n+1)}{(2n)!}\Delta^{2n} y_{-n}$$

$$\textbf{Error} = \left| \frac{\Delta^{2n} y_{-n}}{(2n)!}.t\prod_{i=1}^{n}(t^2 - i^2) \right|$$

GENERAL KNOWLEDGE

5. Difference table for Gauss Forward Interpolation

- Assume that a set of 5 points $\{(x_i, y_i)\}_{i=-2}^{2}$ is given. Hence the difference table of Gauss Forward Interpolation:

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$	$\Delta^4 y$	$\dots$
$\vdots$	$\vdots$					
x_{-2}	y_{-2}					
		Δy_{-2}				
x_{-1}	y_{-1}		$\Delta^2 y_{-2}$			
		Δy_{-1}		$\Delta^3 y_{-2}$		
x_0	y_0		$\Delta^2 y_{-1}$		$\Delta^4 y_{-2}$	$\dots$
		Δy_0		$\Delta^3 y_{-1}$		
x_1	y_1		$\Delta^2 y_0$			
		Δy_1				
x_2	y_2					
$\vdots$	$\vdots$					

GENERAL KNOWLEDGE

5. Difference table for Gauss Backward Interpolation

- Consider the same set above for Gauss Backward Interpolation:

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$	$\Delta^4 y$	$\dots$
$\vdots$	$\vdots$					
x_{-2}	y_{-2}					
		Δy_{-2}				
x_{-1}	y_{-1}		$\Delta^2 y_{-2}$			
		Δy_{-1}		$\Delta^3 y_{-2}$		
x_0	y_0		$\Delta^2 y_{-1}$		$\Delta^4 y_{-2}$	$\dots$
		Δy_0		$\Delta^3 y_{-1}$		
x_1	y_1		$\Delta^2 y_0$			
		Δy_1				
x_2	y_2					
$\vdots$	$\vdots$					

STIRLING INTERPOLATION (THEORY)

1. Idea

- Stirling interpolation imputes an approach for a set of odd number of data points through which the computational expense and storage, comparing to Gauss Forward and Backward Interpolation, is heavily reduced.
- **Idea overview:** Transform non-odd and non-even polynomials to odd or even polynomials, therefore using x^2 as a variable instead of x

STIRLING INTERPOLATION (THEORY)

2. Formulas

- Stirling interpolation is the arithmetic mean of Gauss Forward and Backward Interpolation

$$\text{Stirling} = \frac{\text{Gauss Forward} + \text{Gauss Backward}}{2}$$

$$\begin{cases} x_0 : \text{central data point} \\ h : \text{difference between 2 consecutive data points} \end{cases}$$

Let: $t = \dfrac{x - x_0}{h} \implies x = x_0 + ht$

STIRLING INTERPOLATION (THEORY)

2. Formulas

- Stirling interpolation is the arithmetic mean of Gauss Forward and Backward Interpolation

$$\text{Stirling} = \frac{\text{Gauss Forward} + \text{Gauss Backward}}{2}$$

$$\begin{cases} x_0 : \text{central data point} \\ h : \text{difference between 2 consecutive data points} \end{cases}$$

Let: $t = \dfrac{x - x_0}{h} \Longrightarrow x = x_0 + ht$

- Gauss Forward Interpolation:

$$P(x_0 + ht) = y_0 + t\Delta y_0 + \tfrac{t(t-1)}{2!}\Delta^2 y_{-1} + \tfrac{(t+1)t(t-1)}{3!}\Delta^3 y_{-1} + \tfrac{(t+1)t(t-1)(t-2)}{4!}\Delta^4 y_{-2} + \cdots + \tfrac{(t+n-1)(t+n-1)\ldots(t-n)}{(2n)!}\Delta^{2n} y_{-n}$$

STIRLING INTERPOLATION (THEORY)

2. Formulas

- Stirling interpolation is the arithmetic mean of Gauss Forward and Backward Interpolation

$$\text{Stirling} = \frac{\text{Gauss Forward} + \text{Gauss Backward}}{2}$$

$$\begin{cases} x_0 : \text{central data point} \\ h : \text{difference between 2 consecutive data points} \end{cases}$$

Let: $t = \dfrac{x - x_0}{h} \implies x = x_0 + ht$

- Gauss Forward Interpolation:

$$P(x_0 + ht) = y_0 + t\Delta y_0 + \frac{t(t-1)}{2!}\Delta^2 y_{-1} + \frac{(t+1)t(t-1)}{3!}\Delta^3 y_{-1} + \frac{(t+1)t(t-1)(t-2)}{4!}\Delta^4 y_{-2} + \cdots + \frac{(t+n-1)(t+n-1)\ldots(t-n)}{(2n)!}\Delta^{2n} y_{-n}$$

- Gauss Backward Interpolation:

$$P(x_0 + ht) = y_0 + t\Delta y_{-1} + \frac{t(t+1)}{2!}\Delta^2 y_{-1} + \frac{(t+1)t(t-1)}{3!}\Delta^3 y_{-2} + \frac{(t+2)(t+1)t(t-1)}{4!}\Delta^4 y_{-2} + \cdots + \frac{(t+n)(t+n-1)\ldots(t-n+1)}{(2n)!}\Delta^{2n} y_{-n}$$

STIRLING INTERPOLATION (THEORY)

2. Formulas

- Consider the differences of odd order $2k+1$:

$$\begin{cases} \textbf{Gauss Forward: } \dfrac{\prod\limits_{i=-k}^{k}(t-i)}{(2k+1)!}\Delta^{2k+1}y_{-k} \\ \textbf{Gauss Backward: } \dfrac{\prod\limits_{i=-k}^{k}(t-i)}{(2k+1)!}\Delta^{2k+1}y_{-k-1} \end{cases}$$

STIRLING INTERPOLATION (THEORY)

2. Formulas

- Consider the differences of odd order $2k+1$:

$$\begin{cases} \textbf{Gauss Forward: } \dfrac{\prod\limits_{i=-k}^{k}(t-i)}{(2k+1)!}\Delta^{2k+1}y_{-k} \\ \textbf{Gauss Backward: } \dfrac{\prod\limits_{i=-k}^{k}(t-i)}{(2k+1)!}\Delta^{2k+1}y_{-k-1} \end{cases}$$

$\Longrightarrow$ The difference of odd order $2k+1$ for Stirling Interpolation:

$$\frac{\prod\limits_{i=-k}^{k}(t-i)}{(2k+1)!}\frac{\Delta^{2k+1}y_{-k}+\Delta^{2k+1}y_{-k-1}}{2}$$

$$=\frac{t\prod\limits_{i=1}^{k}(t^2-i^2)}{(2k+1)!}\frac{\Delta^{2k+1}y_{-k}+\Delta^{2k+1}y_{-k-1}}{2}$$

STIRLING INTERPOLATION (THEORY)

2. Formulas

- Consider the differences of even order $2k$:

$$\begin{cases} \textbf{Gauss Forward: } \dfrac{\prod\limits_{i=-k}^{k-1}(t-i)}{(2k)!}\Delta^{2k}y_{-k} \\ \textbf{Gauss Backward: } \dfrac{\prod\limits_{i=-k+1}^{k}(t-i)}{(2k)!}\Delta^{2k}y_{-k} \end{cases}$$

STIRLING INTERPOLATION (THEORY)

2. Formulas

- Consider the differences of even order $2k$:

$$\begin{cases} \textbf{Gauss Forward: } \dfrac{\prod\limits_{i=-k}^{k-1}(t-i)}{(2k)!}\Delta^{2k}y_{-k} \\ \textbf{Gauss Backward: } \dfrac{\prod\limits_{i=-k+1}^{k}(t-i)}{(2k)!}\Delta^{2k}y_{-k} \end{cases}$$

$\Longrightarrow$ The difference of even order $2k$ for Stirling Interpolation:

$$\frac{1}{2}(t+k+t-k)\frac{\prod\limits_{i=-k+1}^{k-1}(t-i)}{(2k)!}\Delta^{2k}y_{-k}$$

$$=\frac{\prod\limits_{i=0}^{k-1}(t^2-i^2)}{(2k)!}\Delta^{2k}y_{-k}$$

STIRLING INTERPOLATION (THEORY)

2. Formulas

- The above deduces Stirling Interpolation:

$$P(x_0+ht) = y_0 + t\frac{\Delta y_0+\Delta y_{-1}}{2} + \frac{t^2}{2!}\Delta^2 y_{-1} + \frac{(t+1)t(t-1)}{3!}\frac{\Delta^3 y_{-1}+\Delta^3 y_{-2}}{2} + \dots + \frac{t(t^2-1^2)(t^2-2^2)\dots(t^2-(n-1)^2)}{(2n-1)!}\frac{\Delta^{2n-1}y_{-n}+\Delta^{2n-1}y_{-n+1}}{2} + \frac{t^2(t^2-1^2)(t^2-2^2)\dots(t^2-(n-1)^2)}{(2n)!}\Delta^{2n}y_{-n}$$

$$\implies P(x_0+ht) = y_0 + \sum_{k=0}^{n-1}\left(\frac{t\prod_{i=1}^{k}(t^2-i^2)}{(2k+1)!}\frac{\Delta^{2k+1}y_{-k}+\Delta^{2k+1}y_{-k-1}}{2}\right) + \sum_{k=1}^{n}\left(\frac{\prod_{i=0}^{k-1}(t^2-i^2)}{(2k)!}\Delta^{2k}y_{-k}\right)$$

STIRLING INTERPOLATION (THEORY)

2. Formulas

- The difference table for Stirling Interpolation

  Example: Given a 5-point set $\{(x_i, y_i)\}_{i=-2}^{2}$

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$	$\Delta^4 y$	$\dots$
$\vdots$	$\vdots$					
x_{-2}	y_{-2}					
		Δy_{-2}				
x_{-1}	y_{-1}		$\Delta^2 y_{-2}$			
		Δy_{-1}		$\Delta^3 y_{-2}$		
x_0	y_0		$\Delta^2 y_{-1}$		$\Delta^4 y_{-2}$	$\dots$
		Δy_0		$\Delta^3 y_{-1}$		
x_1	y_1		$\Delta^2 y_0$			
		Δy_1				
x_2	y_2					
$\vdots$	$\vdots$					

$\dashrightarrow$: **Gauss Backward**

$\longrightarrow$: **Gauss Forward**

STIRLING INTERPOLATION (THEORY)

3. Error

Error of Stirling interpolation equals the mean of errors of Gauss Forward and Gauss Backward Interpolation:

$$\Longrightarrow |R(x)| \leq \left| \frac{\Delta^{2n} y_{-n}}{(2n)!} . t \prod_{i=1}^{n} (t^2 - i^2) \right|$$

STIRLING INTERPOLATION (THEORY)

4. Conclusion

- Stirling interpolation formula:

$$P(x_0 + ht) = y_0 + t\frac{\Delta y_0 + \Delta y_{-1}}{2} + \frac{t^2}{2!}\Delta^2 y_{-1} + \frac{t(t^2-1)}{3!}\frac{\Delta^3 y_{-1} + \Delta^3 y_{-2}}{2} + \cdots + \frac{t(t^2-1^2)(t^2-2^2)...(t^2-(n-1)^2)}{(2n-1)!}\frac{\Delta^{2n-1}y_{-n} + \Delta^{2n-1}y_{-n+1}}{2} + \frac{t^2(t^2-1^2)(t^2-2^2)...(t^2-(n-1)^2)}{(2n)!}\Delta^{2n}y_{-n}$$

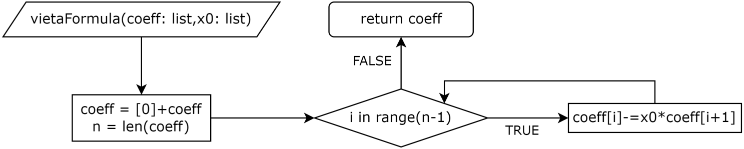
- Error of Stirling interpolation:

$$|R(x)| \leq \left|\frac{\Delta^{2n}y_{-n}}{(2n)!}.t\prod_{i=1}^{n}(t^2 - i^2)\right|$$

STIRLING INTERPOLATION (ALGORITHM)

1. Packages

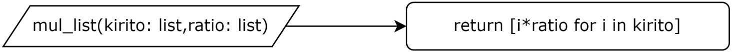
- vietaFormula(coeff,x0):
  Input: a list 'roots' and a real number x0
  Output: list with coefficients of the product polynomial of 'coeff' and (x-x0)



STIRLING INTERPOLATION (ALGORITHM)

1. Packages

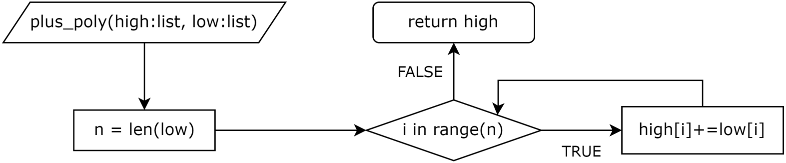
- mul_list(kirito, ratio):
  Input: List 'kirito' and real number 'ratio'
  Output: List 'kirito' with every element is multiplied by 'ratio'



STIRLING INTERPOLATION (ALGORITHM)

1. Packages

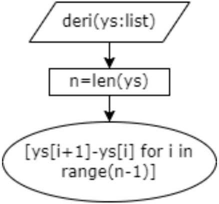
- plus_poly(high,low):
  Input: List 'high' and list 'low'
  Output: Coefficients of sum polynomial of 'high' and 'low'



STIRLING INTERPOLATION (ALGORITHM)

1. Packages

- deri(ys)
  Input: a list 'ys'
  Output: the next order of difference



STIRLING INTERPOLATION (ALGORITHM)

1. Packages

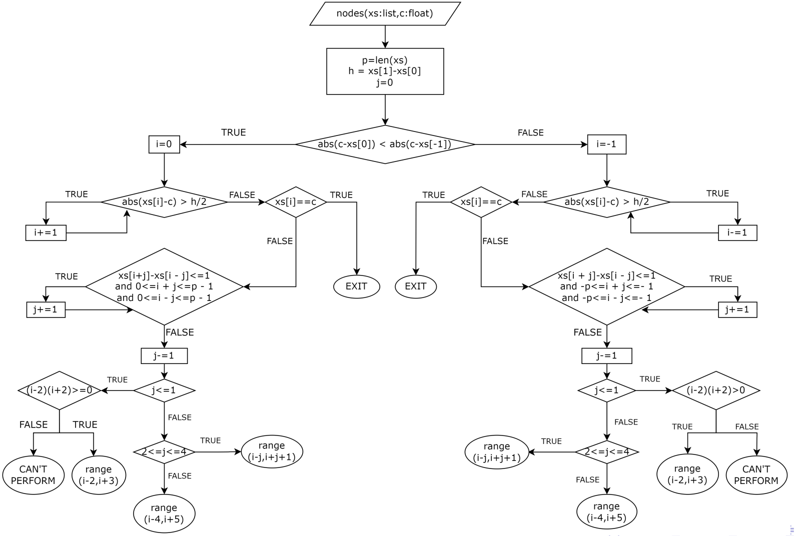
- nodes(xs,c)

  Input: a list xs and a real number c

  Output: Interpolation nodes indices used for Stirling interpolation

STIRLING INTERPOLATION (ALGORITHM)

1. Packages



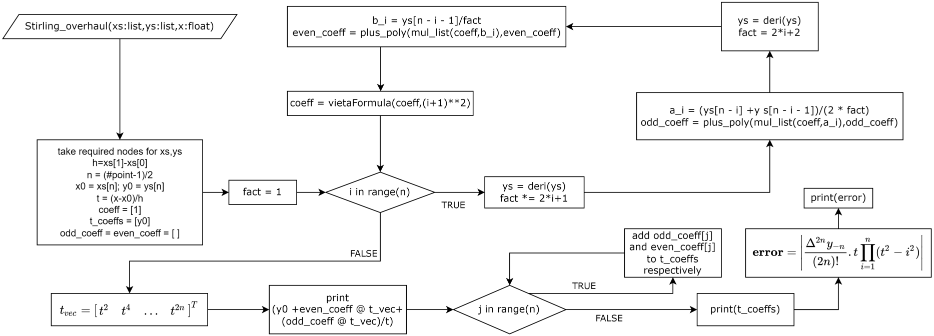
STIRLING INTERPOLATION (ALGORITHM)

1.Packages

- stirling(xs,ys,x)
  Input: 2 lists xs,ys and a real number x
  Output:
  - Approximated value for $f(x)$ using Stirling Interpolation
  - Coefficients for $P(x_0 + ht)$ with respect to t
  - Error range

STIRLING INTERPOLATION (ALGORITHM)

1. Packages



STIRLING INTERPOLATION (ALGORITHM)

2. Program

- **Input:**
  1. List xs: contains all x values
  2. List ys: contains all corresponding y values
  3. Real number start: to be approximated
- **Output:**

  If start is not between x_{min} and x_{max} $\longrightarrow$ EXIT PROGRAM

  Else: stirling(xs,ys,start)

BESSEL INTERPOLATION (THEORY)

1. Idea

- Similar to Stirling, except Bessel Interpolation interpolates sets with even number of data points

BESSEL INTERPOLATION (THEORY)

2. Formulas

- Bessel Interpolation is also equivalent to the arithmetic mean of Gauss Forward and Backward Interpolation, except the central data point of Gauss Backward is shifted to the following position

$$\text{Bessel} = \frac{\text{Gauss Forward} + \text{Gauss Backward(shifted)}}{2}$$

Given a set of $2n+2$ data points $\{(x_i, y_i)\}_{i=-n}^{n+1}$

- Gauss Forward Interpolation (centered at x_0)

$$P(x_0+ht) = y_0 + t\Delta y_0 + \frac{t(t-1)}{2!}\Delta^2 y_{-1} + \frac{(t+1)t(t-1)}{3!}\Delta^3 y_{-1} + \frac{(t+1)t(t-1)(t-2)}{4!}\Delta^4 y_{-2} + \cdots + \frac{(t+n)(t+n-1)\ldots(t-n)}{(2n+1)!}\Delta^{2n+1} y_{-n}$$

- Gauss Backward Interpolation (centered at $x_1 = x_0 + h$)

$$P(x_1+ht) = y_1 + t\Delta y_0 + \frac{t(t+1)}{2!}\Delta^2 y_0 + \frac{(t+1)t(t-1)}{3!}\Delta^3 y_{-1} + \frac{(t+2)(t+1)t(t-1)}{4!}\Delta^4 y_{-1} + \cdots + \frac{(t+n)(t+n-1)\ldots(t-n)}{(2n+1)!}\Delta^{2n+1} y_{-n}$$

BESSEL INTERPOLATION (THEORY)

2. Formulas

- Gauss Forward Interpolation (centered at x_0)

$$P(x_0 + ht) = y_0 + t\Delta y_0 + \frac{t(t-1)}{2!}\Delta^2 y_{-1} + \frac{(t+1)t(t-1)}{3!}\Delta^3 y_{-1} + \frac{(t+1)t(t-1)(t-2)}{4!}\Delta^4 y_{-2} + \cdots + \frac{(t+n)(t+n-1)...(t-n)}{(2n+1)!}\Delta^{2n+1} y_{-n}$$

BESSEL INTERPOLATION (THEORY)

2. Formulas

- Gauss Forward Interpolation (centered at x_0)

$$P(x_0 + ht) = y_0 + t\Delta y_0 + \frac{t(t-1)}{2!}\Delta^2 y_{-1} + \frac{(t+1)t(t-1)}{3!}\Delta^3 y_{-1} + \frac{(t+1)t(t-1)(t-2)}{4!}\Delta^4 y_{-2} + \cdots + \frac{(t+n)(t+n-1)\ldots(t-n)}{(2n+1)!}\Delta^{2n+1} y_{-n}$$

Substitute t with $t + \dfrac{1}{2}$:

BESSEL INTERPOLATION (THEORY)

2. Formulas

- Gauss Forward Interpolation (centered at x_0)

$$P(x_0 + ht) = y_0 + t\Delta y_0 + \frac{t(t-1)}{2!}\Delta^2 y_{-1} + \frac{(t+1)t(t-1)}{3!}\Delta^3 y_{-1} + \frac{(t+1)t(t-1)(t-2)}{4!}\Delta^4 y_{-2} + \cdots + \frac{(t+n)(t+n-1)\ldots(t-n)}{(2n+1)!}\Delta^{2n+1} y_{-n}$$

Substitute t with $t + \dfrac{1}{2}$:

$$P\left(x_0 + \frac{h}{2} + ht\right) = y_0 + \left(t + \frac{1}{2}\right)\Delta y_0 + \frac{\left(t+\frac{1}{2}\right)\left(t-\frac{1}{2}\right)}{2!}\Delta^2 y_{-1} + \frac{\left(t+\frac{3}{2}\right)\left(t+\frac{1}{2}\right)\left(t-\frac{1}{2}\right)}{3!}\Delta^3 y_{-1} + \frac{\left(t+\frac{3}{2}\right)\left(t+\frac{1}{2}\right)\left(t-\frac{1}{2}\right)\left(t-\frac{3}{2}\right)}{4!}\Delta^4 y_{-2} + \cdots + \frac{\left(t+\frac{2n+1}{2}\right)\left(t+\frac{2n-1}{2}\right)\ldots\left(t-\frac{2n-1}{2}\right)}{(2n+1)!}\Delta^{2n+1} y_{-n}$$

BESSEL INTERPOLATION (THEORY)

2. Formulas

- Gauss Backward Interpolation (centered at $x_1 = x_0 + h$)

$$P(x_1 + ht) = y_1 + t\Delta y_0 + \frac{t(t+1)}{2!}\Delta^2 y_0 + \frac{(t+1)t(t-1)}{3!}\Delta^3 y_{-1} + \frac{(t+2)(t+1)t(t-1)}{4!}\Delta^4 y_{-1} + \cdots + \frac{(t+n)(t+n-1)...(t-n)}{(2n+1)!}\Delta^{2n+1} y_{-n}$$

BESSEL INTERPOLATION (THEORY)

2. Formulas

- Gauss Backward Interpolation (centered at $x_1 = x_0 + h$)

$$P(x_1 + ht) = y_1 + t\Delta y_0 + \frac{t(t+1)}{2!}\Delta^2 y_0 + \frac{(t+1)t(t-1)}{3!}\Delta^3 y_{-1} + \frac{(t+2)(t+1)t(t-1)}{4!}\Delta^4 y_{-1} + \cdots + \frac{(t+n)(t+n-1)\ldots(t-n)}{(2n+1)!}\Delta^{2n+1} y_{-n}$$

Substituting t with $t - \dfrac{1}{2}$

BESSEL INTERPOLATION (THEORY)

2. Formulas

- Gauss Backward Interpolation (centered at $x_1 = x_0 + h$)

$$P(x_1 + ht) = y_1 + t\Delta y_0 + \frac{t(t+1)}{2!}\Delta^2 y_0 + \frac{(t+1)t(t-1)}{3!}\Delta^3 y_{-1} + \frac{(t+2)(t+1)t(t-1)}{4!}\Delta^4 y_{-1} + \cdots + \frac{(t+n)(t+n-1)\ldots(t-n)}{(2n+1)!}\Delta^{2n+1} y_{-n}$$

Substituting t with $t - \dfrac{1}{2}$

$$P\left(x_1 - \frac{h}{2} + ht\right) = y_1 + \left(t - \frac{1}{2}\right)\Delta y_0 + \frac{\left(t-\frac{1}{2}\right)\left(t+\frac{1}{2}\right)}{2!}\Delta^2 y_0 + \frac{\left(t+\frac{1}{2}\right)\left(t-\frac{1}{2}\right)\left(t-\frac{3}{2}\right)}{3!}\Delta^3 y_{-1} + \frac{\left(t+\frac{3}{2}\right)\left(t+\frac{1}{2}\right)\left(t-\frac{1}{2}\right)\left(t-\frac{3}{2}\right)}{4!}\Delta^4 y_{-1} + \cdots + \frac{\left(t+\frac{2n-1}{2}\right)\left(t+\frac{2n-3}{2}\right)\ldots\left(t-\frac{2n+1}{2}\right)}{(2n+1)!}\Delta^{2n+1} y_{-n}$$

BESSEL INTERPOLATION (THEORY)

2. Formulas

- Consider the difference of odd order $2k+1$

$$\begin{cases} \textbf{Gauss Forward: } \dfrac{\prod\limits_{i=-k-1}^{k-1}\left(t-\frac{1}{2}-i\right)}{(2k+1)!}\Delta^{2k+1}y_{-k} \\ \textbf{Gauss Backward:} \dfrac{\prod\limits_{i=-k}^{k}\left(t-\frac{1}{2}-i\right)}{(2k+1)!}\Delta^{2k+1}y_{-k} \end{cases}$$

BESSEL INTERPOLATION (THEORY)

2. Formulas

- Consider the difference of odd order $2k+1$

$$\begin{cases} \textbf{Gauss Forward: } \dfrac{\prod\limits_{i=-k-1}^{k-1}\left(t-\frac{1}{2}-i\right)}{(2k+1)!}\Delta^{2k+1}y_{-k} \\ \textbf{Gauss Backward:} \dfrac{\prod\limits_{i=-k}^{k}\left(t-\frac{1}{2}-i\right)}{(2k+1)!}\Delta^{2k+1}y_{-k} \end{cases}$$

Hence the difference of odd order $2k+1$ for Bessel Interpolation

$$\frac{t\prod\limits_{i=0}^{k-1}\left(t^2-\left(i+\frac{1}{2}\right)^2\right)}{(2k+1)!}\Delta^{2k+1}y_{-k}$$

BESSEL INTERPOLATION (THEORY)

2. Formulas

- Consider the difference of even order 2k

$$\begin{cases} \textbf{Gauss Forward: } \dfrac{\prod_{i=-k}^{k-1}\left(t-\frac{1}{2}-i\right)}{(2k)!}\Delta^{2k}y_{-k} \\ \textbf{Gauss Backward: } \dfrac{\prod_{i=-k}^{k-1}\left(t-\frac{1}{2}-i\right)}{(2k)!}\Delta^{2k}y_{-(k-1)} \end{cases}$$

BESSEL INTERPOLATION (THEORY)

2. Formulas

- Consider the difference of even order 2k

$$\begin{cases} \textbf{Gauss Forward: } \dfrac{\prod_{i=-k}^{k-1}\left(t-\frac{1}{2}-i\right)}{(2k)!}\Delta^{2k}y_{-k} \\ \textbf{Gauss Backward: } \dfrac{\prod_{i=-k}^{k-1}\left(t-\frac{1}{2}-i\right)}{(2k)!}\Delta^{2k}y_{-(k-1)} \end{cases}$$

Hence the difference of even order $2k$ for Bessel Interpolation

$$\frac{\prod_{i=0}^{k-1}\left(t^2-\left(i+\frac{1}{2}\right)^2\right)}{(2k)!}\,\frac{\Delta^{2k}y_{-k}+\Delta^{2k}y_{-(k-1)}}{2}$$

BESSEL INTERPOLATION (THEORY)

2. Formulas

$\Longrightarrow$ Bessel Interpolation formula:

$$P\left(\frac{x_0+x_1}{2}+ht\right)=\frac{y_0+y_1}{2}+t\Delta y_0+\frac{t^2-\frac{1}{4}}{2!}\frac{\Delta^2 y_{-1}+\Delta^2 y_0}{2}+\frac{t\left(t^2-\frac{1}{4}\right)}{3!}\Delta^3 y_{-1}+$$
$$\frac{\left(t^2-\frac{1}{4}\right)\left(t^2-\frac{9}{4}\right)}{4!}\frac{\Delta^4 y_{-2}+\Delta^4 y_{-1}}{2}+\cdots+$$
$$\frac{\left(t^2-\frac{1}{4}\right)\left(t^2-\frac{9}{4}\right)\ldots\left(t^2-\left(n-\frac{1}{2}\right)^2\right)}{(2n)!}\frac{\Delta^{2n} y_{-n}+\Delta^{2n} y_{-(n-1)}}{2}+$$
$$\frac{t\left(t^2-\frac{1}{4}\right)\left(t^2-\frac{9}{4}\right)\ldots\left(t^2-\left(n-\frac{1}{2}\right)^2\right)}{(2n+1)!}\Delta^{2n+1} y_{-n}$$

BESSEL INTERPOLATION (THEORY)

2. Formulas

- Difference table for Bessel Interpolation

  Example: Given a 6-point set $\{(x_i, y_i)\}_{i=-2}^{3}$

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$	$\Delta^4 y$	$\Delta^5 y$	$\dots$
$\vdots$	$\vdots$						
x_{-2}	y_{-2}						
		Δy_{-2}					
x_{-1}	y_{-1}		$\Delta^2 y_{-2}$				
		Δy_{-1}		$\Delta^3 y_{-2}$			
x_0	y_0		$\Delta^2 y_{-1}$		$\Delta^4 y_{-2}$		
		Δy_0		$\Delta^3 y_{-1}$		$\Delta^5 y_{-2}$	$\dots$
x_1	y_1		$\Delta^2 y_0$		$\Delta^4 y_{-1}$		
		Δy_1		$\Delta^3 y_0$			
x_2	y_2		$\Delta^2 y_1$				
		Δy_2					
x_3	y_3						
$\vdots$	$\vdots$						

$\dashrightarrow$: **Gauss Backward**

$\longrightarrow$: **Gauss Forward**

BESSEL INTERPOLATION (THEORY)

3. Error

- Similar to Stirling, Bessel Interpolation error is as follows:

$$|R(x)| \leq \left| \frac{\Delta^{2n+1} y_{-n}}{(2n+1)!} \prod_{i=0}^{n} \left(t^2 - \left(i + \frac{1}{2} \right)^2 \right) \right|$$

BESSEL INTERPOLATION (THEORY)

4. Conclusion

- Bessel Interpolation formula:

$$P\left(\frac{x_0+x_1}{2}+ht\right)=\frac{y_0+y_1}{2}+t\Delta y_0+\frac{t^2-\frac{1}{4}}{2!}\frac{\Delta^2 y_{-1}+\Delta^2 y_0}{2}+\frac{t\left(t^2-\frac{1}{4}\right)}{3!}\Delta^3 y_{-1}+$$

$$\frac{\left(t^2-\frac{1}{4}\right)\left(t^2-\frac{9}{4}\right)}{4!}\frac{\Delta^4 y_{-2}+\Delta^4 y_{-1}}{2}+\cdots+$$

$$\frac{\left(t^2-\frac{1}{4}\right)\left(t^2-\frac{9}{4}\right)\ldots\left(t^2-\left(n-\frac{1}{2}\right)^2\right)}{(2n)!}\frac{\Delta^{2n} y_{-n}+\Delta^{2n} y_{-(n-1)}}{2}+$$

$$\frac{t\left(t^2-\frac{1}{4}\right)\left(t^2-\frac{9}{4}\right)\ldots\left(t^2-\left(n-\frac{1}{2}\right)^2\right)}{(2n+1)!}\Delta^{2n+1} y_{-n}$$

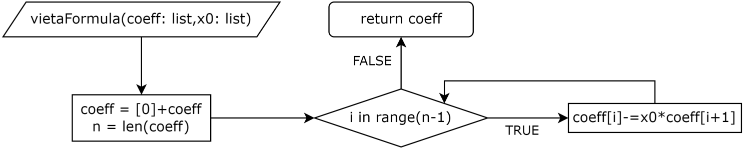
- Error range for Bessel Interpolation:

$$|R(x)|\leq\left|\frac{\Delta^{2n+1}y_{-n}}{(2n+1)!}\prod_{i=0}^{n}\left(t^2-\left(i+\frac{1}{2}\right)^2\right)\right|$$

BESSEL INTERPOLATION (ALGORITHM)

1. Packages

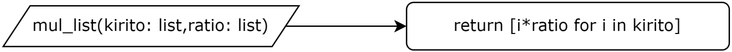
- vietaFormula(coeff,x0):
  Input: a list 'roots' and a real number x0
  Output: list with coefficients of the product polynomial of 'coeff' and (x-x0)



BESSEL INTERPOLATION (ALGORITHM)

1. Packages

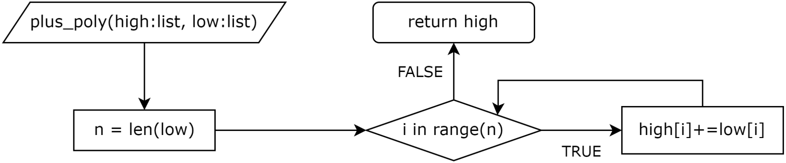
- mul_list(kirito, ratio):
  Input: List 'kirito' and real number 'ratio'
  Output: List 'kirito' with every element is multiplied by 'ratio'



BESSEL INTERPOLATION (ALGORITHM)

1. Packages

- plus_poly(high,low):
  Input: List 'high' and list 'low'
  Output: Coefficients of sum polynomial of 'high' and 'low'



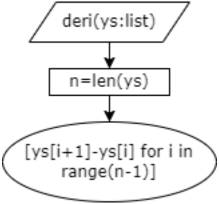
BESSEL INTERPOLATION (ALGORITHM)

1. Packages

- deri(ys)

  Input: list ys

  Output: the next order of difference



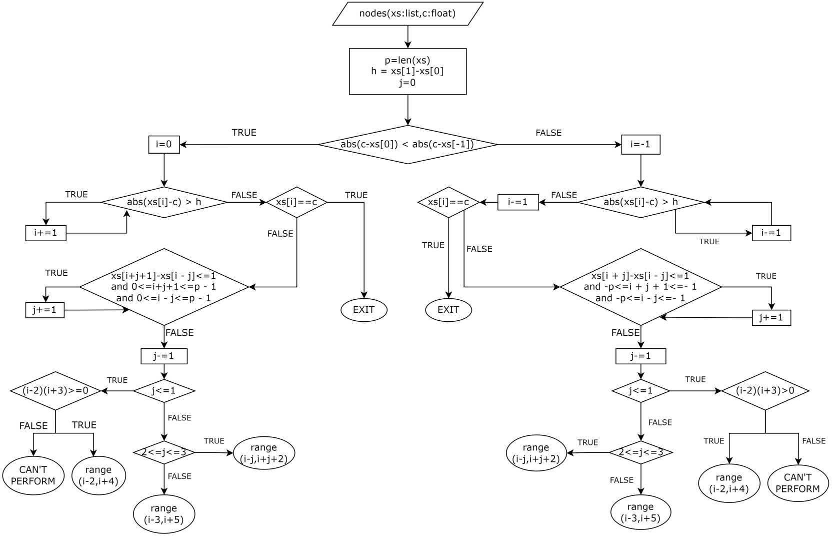
BESSEL INTERPOLATION (ALGORITHM)

1. Packages

- nodes(xs,c)
  Input: a list xs and a real number c
  Output: Interpolation nodes indices used for Bessel Interpolation

BESSEL INTERPOLATION (ALGORITHM)

1.Packages



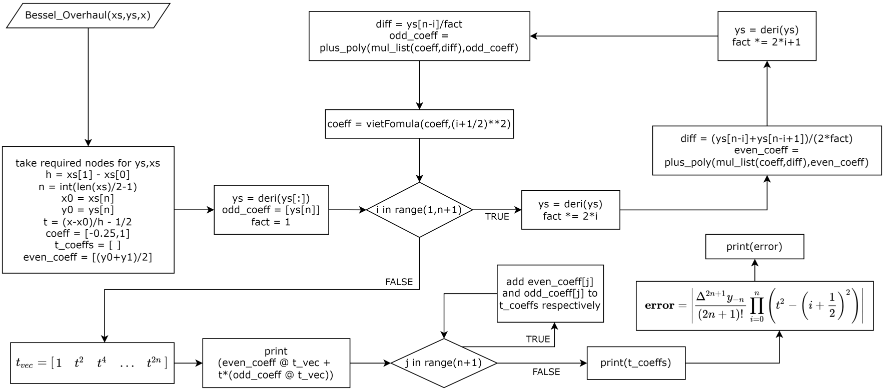
BESSEL INTERPOLATION (ALGORITHM)

1. Packages

- bessel(xs,ys,x)
  Input: 2 lists xs,ys and a real number x
  Output:
  - Approximated value for $f(x)$ using Bessel Interpolation
  - Coefficients for $P(\frac{x_0+x_1}{2} + ht)$ with respect to t
  - Error range

BESSEL INTERPOLATION (ALGORITHM)

1. Packages



BESSEL INTERPOLATION (ALGORITHM)

2. Program

- **Input:**
  1. List xs: contains all x values
  2. List ys: contains all corresponding y values
  3. Real number start: to be approximated
- **Output:**

  If start is not between x_{min} and x_{max} $\longrightarrow$ EXIT PROGRAM

  Else: bessel(xs,ys,start)

STIRLING, BESSEL vs GAUSS FORWARD, BACKWARD

- Stirling Interpolation (given $2n + 1$ data points). Thus the computational complexity of Stirling Interpolation (similarly with Bessel) is $o\left(\frac{13n^2}{2}\right)$
- Gauss Forward and Backward Interpolation possess the computational complexity of $o(12n^2)$

STIRLING, BESSEL vs GAUSS FORWARD, BACKWARD

- Stirling Interpolation (given $2n+1$ data points). Thus the computational complexity of Stirling Interpolation (similarly with Bessel) is $o\left(\frac{13n^2}{2}\right)$
- Gauss Forward and Backward Interpolation possess the computational complexity of $o(12n^2)$

ASSESSMENT: Stirling and Bessel interpolation heavily reduce number of calculations, thus reduce computational cost

ADVANTAGES AND DISADVANTAGES

- **Advantages:**
  1. Heavily reduces number of calculations and storage
  2. Limitation on nodes (8 for Bessel and 9 for Stirling) ensures reasonable computations yet promises stable accuracy
  3. Considering function $f(x_0 + ht)$ significantly reduces computational complexity still, due to the linear variation between t and x, functions including integration and derivative can easily be expressed through t

ADVANTAGES AND DISADVANTAGES

- **Advantages:**
  1. Heavily reduces number of calculations and storage
  2. Limitation on nodes (8 for Bessel and 9 for Stirling) ensures reasonable computations yet promises stable accuracy
  3. Considering function $f(x_0 + ht)$ significantly reduces computational complexity still, due to the linear variation between t and x, functions including integration and derivative can easily be expressed through t
- **Disadvantages:**
  1. Only handle constant $x-$interval between data points
  2. Inappropriate for approximating points close to bounds
  3. Unstable error with considerably non-monotonic inputs