TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN TOÁN ỨNG DỤNG VÀ TIN HỌC

Báo cáo Giải tích số

Phương pháp nội suy trung tâm Gauss I và Gauss II

Sinh viên	:	Nguyễn Đức Anh
MSSV	:	20172946
Lớp	:	CTTN Toán Tin K62
Giảng viên	:	TS. Hà Thị Ngọc Yến

Hà Nội - 1/2022

Mục lục

Danh sách hình vẽ

Chương 1

Nội dung lý thuyết

1.1 Nội suy trung tâm

Trong tính toán và khoa học, chúng ta thường có một số điểm dữ liệu đã biết giá trị thông qua đo đạc thực nghiệm. Bộ dữ liệu này có thể là đại diện cho một hàm số nào đó phụ thuộc vào một biến số độc lập. Nhu cầu đặt ra là liệu có thể xác định quy luật biến đổi của hàm số đó hoặc dự đoán giá trị các thời điểm chưa đo đạc không có trong bộ dữ liệu. Hoặc trường hợp khác là khi hàm số sinh ra bộ dữ liệu quá phức tạp, ta cần thay thế hàm số đó bằng một hàm trung gian đơn giản hơn để dễ dàng trong việc tính toán và ước lượng. Để giải quyết vấn đề đó, chúng ta sẽ dùng các phương pháp nội suy (ước lượng) và xấp xỉ hàm. Dạng hàm trung gian được chọn thường là đa thức và gọi là đa thức nội suy.

Bài toán nội suy đa thức tổng quát được phát biểu như sau: Cho hàm số $y = f(x)$ và bảng số thu được từ hàm số đó $y_i = f(x_i)$ với $i = \overline{0,n}$, từ bảng số đó hãy xây dựng đa thức nội suy $P_n(x)$ bậc nhỏ hơn hoặc bằng n sao cho $P_n(x_i) = y_i$ với $i = \overline{0,n}$. Trong trường hợp các mốc nội suy x_i có khoảng cách đều nhau thì được gọi là nội suy với mốc cách đều. Phương pháp tiêu biểu để giải bài toán trên có thể kể đến là phương pháp nội suy Newton tiến và lùi . Tuy nhiên, vấn đề của phương pháp Newton là các công thức mang đặc trưng một phía (xem [2]), khi cần tính giá trị gần đúng tại x gần

x_0, ta dùng công thức tiến còn x gần x_n sẽ dùng công thức lùi. Khi đó, trong trường hợp cần tính x tại giữa bảng thì các công thức trên bị hạn chế vì nhiều mốc nội suy không được dùng. Hơn nữa, phép nội suy có sai số nhỏ nhất khi x nằm ở khoảng trung tâm đoạn nội suy. Từ đó, công thức nội suy trung tâm được ra đời.

Dựa trên ý tưởng từ mốc xuất phát nằm ở trung tâm, ta sẽ lần lượt hoặc đồng thời kết nạp các mốc từ hai phía, tùy thuộc vào cách đi mà ta chia ra thành các phương pháp tiêu biểu như Gauss I, Gauss II, Stirlin hoặc Bessel. Các phương pháp nội suy trung tâm sẽ làm việc với bộ số có mốc cách đều và sử dụng chung một bảng sai phân trung tâm như bảng 1.1. Ta có thể phát biểu lại bài toán mà các phương pháp nội suy trung tâm như Gauss I và II sẽ xử lý như sau: Giả sử có $2n+1$ mốc nội suy cách đều và được xếp thứ tự $x_{-n},\ldots,x_{-1},x_0,x_1,\ldots,x_n$ và các giá trị tương ứng y_i cho trong bảng số:

$$y_i = f(x_i), \;\; i = 0, \pm 1, \ldots, \pm n. \tag{1.1}$$

Ta cần xây dựng đa thức nội suy $P(x)$ bậc $\leq 2n$ sao cho:

$$y_i = P(x_i), \;\; i = 0, \pm 1, \ldots, \pm n. \tag{1.2}$$

Có thể thấy số lượng mốc yêu cầu là lẻ với $2n+1$ mốc, điều này cơ bản là giúp cho các phương pháp sẽ kết nạp mốc đều về cả hai phía. Mặc dù vậy, ta vẫn có thể giải quyết với các bài toán có số mốc chẵn. Từ đó, có thể đưa ra điều kiện cho các thuật toán nội suy trung tâm là làm việc với các bộ dữ liệu có mốc cách đều, đã được sắp thứ tự và số mốc nội suy thường là lẻ.

Trong báo cáo này, em xin được phép trình bày về hai phương pháp cơ bản của nội suy trung tâm là Gauss I và Gauss II. Hai phương pháp này được phát triển từ phương pháp nội suy Newton với hai quy luật nạp mốc khác nhau. Đây là cũng hai phương pháp nền tảng cho các phương pháp nội suy trung tâm khác như Stirlin hay Bessel.

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$	$\Delta^4 y$	$\dots$
$\dots$	$\dots$					
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					
$\dots$	$\dots$					

Bảng 1.1: Bảng sai phân trung tâm

1.2 Một số kiến thức chung

Định nghĩa 1.1. *Cho hàm số $f(x)$ xác định trên [a,b], với số gia của dối số là $\Delta x = h$ ($h > 0$ cố định) thì sai phân cấp một của hàm số tại x là:*

$$\Delta f(x) = f(x+h) - f(x)$$

Sai phân của sai phân cấp $m-1$ là sai phân cấp m, ký hiệu là:

$$\Delta^m f(x) = \Delta(\Delta^{m-1} f(x))$$

Định nghĩa 1.2. *Giả sử hàm số $y = f(x)$ được cho trong một bảng số $y_i = f(x_i)$ và các mốc nội suy x_i cách đều, khi đó sai phân các cấp được xác định và có thể được trình bày như trong bảng 1.1 và được gọi là bảng sai phân trung tâm.*

1.3 Phương pháp Gauss I

Phương pháp nội suy trung tâm Gauss I được phát triển từ phương pháp nội suy Newton với đặc trưng là các mốc được nạp theo quy luật từ mốc nội suy trung tâm sẽ đi đến một mốc bên phải rồi đến một mốc bên trái $x_0 \to x_1 \to x_{-1} \to x_2 \to x_{-2} \dots$ Theo đó, đa thức nội suy theo cách nạp mốc của Gauss I có dạng cơ bản như sau:

$$\begin{aligned} P(x) = a_0 + a_1(x-x_0) + a_2(x-x_0)(x-x_1) + a_3(x-x_{-1})(x-x_0)(x-x_1) + \dots \\ + a_{2n-1}(x-x_{-(n-1)}) \dots (x-x_0) \dots (x-x_{n-1}) \\ + a_{2n}(x-x_{-(n-1)}) \dots (x-x_0) \dots (x-x_n). \end{aligned}$$

Do đầu vào là các mốc cách đều, đặt $h = x_i - x_{i-1}$ với $i = 1, \dots, 2n$. Để tìm các tham số a_i, ta sẽ thay lần lượt các x_i trong bảng nội suy vào đa thức.

- Với $x = x_0$ thì $a_0 = y_0$.
- Với $x = x_1$ thì:

  $$y_1 = P(x_1) = y_0 + a_1(x_1 - x_0)$$

  Suy ra

  $$a_1 = \frac{y_1 - y_0}{x_1 - x_0} = \frac{\Delta y_0}{h}.$$

- Với $x = x_{-1}$ thì:

  $$\begin{aligned} y_{-1} = P(x_{-1}) &= y_0 + \frac{\Delta y_0}{h}(x_{-1} - x_0) + a_2(x_{-1} - x_0)(x_{-1} - x_1) \\ &= y_0 - \Delta y_0 + a_2 \times h \times 2h \end{aligned}$$

  Hay

  $$\Delta y_0 - \Delta y_{-1} = a_2 \times 2h^2$$

Suy ra:

$$a_2 = \frac{\Delta y_0 - \Delta y_{-1}}{2h^2} = \frac{\Delta^2 y_{-1}}{2h^2}.$$

Lần lượt như vậy, ta sẽ tìm được công thức của các hệ số a_i theo quy luật như sau:

$$a_{2i-1} = \frac{\Delta^{2i-1} y_{-(i-1)}}{(2i-1)!h^{2i-1}}, \quad a_{2i} = \frac{\Delta^{2i} y_{-i}}{(2i)!h^{2i}}, \quad (i = \overline{1,n})$$

Có thể thấy các hệ số a_i sẽ đi theo từng cặp, tuy nhiên theo công thức trên sẽ nảy sinh vấn đề khi giá trị các mốc nội suy là gần nhau tức h nhỏ, khi đó quá trình chia cho h^{2i-1} và h^{2i} sẽ có thể xảy ra lỗi khi thực hiện cho cho số gần 0. Nên ta cần phương án loại bỏ h trong phép chia của công thức. Sử dụng điều kiện mốc cách đều, ta có thể đặt:

$$t = \frac{x - x_0}{h}$$

Khi đó, các công thức của a_{2i-1} và $(x - x_{-(i-1)}) \dots (x - x_0) \dots (x - x_{i-1})$ là phần tích đi kèm trở thành:

$$\frac{\Delta^{2i-1} y_{-(i-1)}}{(2i-1)!} \times (t + i - 1) \dots (t+1)t(t-1) \dots (t - i + 1).$$

Còn a_{2i} và $(x - x_{-(i-1)}) \dots (x - x_0) \dots (x - x_i)$ trở thành:

$$\frac{\Delta^{2i} y_{-i}}{(2i)!} \times (t + i - 1) \dots (t+1)t(t-1) \dots (t - i + 1)(t - i).$$

Tổng quát lại, công thức theo phương pháp Gauss I theo biến t sẽ là:

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

trong đó:

$$a_{2i-1} = \frac{\Delta^{2i-1} y_{-(i-1)}}{(2i-1)!}, \; a_{2i} = \frac{\Delta^{2i} y_{-i}}{(2i)!}.$$

Nhận xét 1.1. *Với công thức Gauss I, sử dụng bảng sai phân, ta có thể thấy các giá* *trị sai phân được chọn sẽ tạo thành một đường gấp khúc nằm dưới đường "central line" như hình 1.1, trong đó, đường "central line" là đường dóng ngang từ mốc xuất phát x_0*

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

Hình 1.1: Bảng sai phân với Gauss I

Nhận xét 1.2. *Vấn đề sẽ nảy sinh khi xử lý với các bài toán có số mốc là chẵn, khi đó mốc trung tâm không còn là duy nhất có thể dẫn đến không sử dụng hết mốc như hình 1.2 (mốc x_{-3} không được sử dụng). Để tránh hiện tượng này thì với bài toán có $2n$ mốc x_i có chỉ số là $i = 0, 1, 2, \ldots, 2n-1$, Gauss I sẽ ưu tiên chọn mốc xuất phát thứ $i = \left[\dfrac{2n-1}{2}\right]$.*

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

Hình 1.2: Bảng sai phân với Gauss I và số mốc chẵn

Nhận xét 1.3. *Do phương pháp Gauss I dựa trên ý tưởng từ phương pháp Newton* *nên ta có thể biến đổi công thức Newton để nhận được phương pháp Gauss I như sau (xem thêm [1]):*

- *Xuất phát từ công thức Newton tiến:*

$$P(t) = y_0 + t\Delta y_0 + \frac{t(t-1)}{2!}\Delta^2 y_0 + \frac{t(t-1)(t-2)}{3!}\Delta^3 y_0 + \frac{t(t-1)(t-2)(t-3)}{4!}\Delta^4 y_0 + \dots$$

- *Ta lại có:*

$$\begin{aligned}\Delta^2 y_0 &= \Delta^2 y_{-1} + \Delta^3 y_{-1}\\ \Delta^3 y_0 &= \Delta^3 y_{-1} + \Delta^4 y_{-1}\\ \Delta^4 y_0 &= \Delta^4 y_{-1} + \Delta^5 y_{-1}\end{aligned}$$

  Suy ra:

$$\begin{aligned}P(t) = y_0 + t\Delta y_0 &+ \frac{t(t-1)}{2!}(\Delta^2 y_{-1} + \Delta^3 y_{-1}) + \frac{t(t-1)(t-2)}{3!}(\Delta^3 y_{-1} + \Delta^4 y_{-1})\\ &+ \frac{t(t-1)(t-2)(t-3)}{4!}(\Delta^4 y_{-1} + \Delta^5 y_{-1}) + \dots\end{aligned}$$

  Khi đó $P(t)$ trở thành:

$$\begin{aligned}P(t) = y_0 + t\Delta y_0 &+ \frac{t(t-1)}{2!}\Delta^2 y_{-1} + \frac{(t+1)t(t-1)}{3!}\Delta^3 y_{-1}\\ &+ \frac{(t+1)t(t-1)(t-2)}{4!}\Delta^4 y_{-1} + \frac{(t+1)t(t-1)(t-2)(t-3)}{5!}\Delta^5 y_{-1} \dots\end{aligned}$$

- *Tiếp tục áp dụng:*

$$\begin{aligned}\Delta^4 y_{-1} &= \Delta^4 y_{-2} + \Delta^5 y_{-2}\\ \Delta^5 y_{-1} &= \Delta^5 y_{-2} + \Delta^6 y_{-2}\\ &\dots\end{aligned}$$

Cứ như thế, ta sẽ được công thức Gauss I.

1.4 Phương pháp Gauss II

Phương pháp nội suy trung tâm Gauss II cũng tương tự như phương pháp nội suy Gauss I, sự khác biệt nằm ở quy luật kết nạp mốc khi ta sẽ xuất phát từ mốc trung tâm và nạp các mốc theo thứ tự một mốc bên trái rồi đến một mốc bên phải $x_0 \to x_{-1} \to x_1 \to x_{-2} \to x_2 \dots$ Do đó, công thức của đa thức nội suy theo phương pháp Gauss II sẽ có dạng như sau:

$$\begin{aligned} P(x) = a_0 + a_1(x-x_0) + a_2(x-x_0)(x-x_{-1}) + a_3(x-x_{-1})(x-x_0)(x-x_1) + \\ a_4(x-x_{-2})(x-x_{-1})(x-x_0)(x-x_1) + \\ a_5(x-x_{-2})(x-x_{-1})(x-x_0)(x-x_1)(x-x_2) + \dots \end{aligned}$$

Để tìm các hệ số a_i, ta sẽ thế lần lượt các x_i trong bảng đa thức tương tự như phương pháp Gauss I. Với phép đặt $t = \dfrac{x-x_0}{h}$, ta sẽ có công thức Gauss 2 như sau:

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

trong đó:

$$a_{2i-1} = \frac{\Delta^{2i-1} y_{-i}}{(2i-1)!},\ a_{2i} = \frac{\Delta^{2i} y_{-i}}{(2i)!}.$$

Nhận xét 1.4. *Với công thức Gauss II, sử dụng bảng sai phân, ta có thể thấy các giá trị sai phân được chọn sẽ tạo thành một đường gấp khúc nằm trên đường "central line" như hình 1.3.*

Nhận xét 1.5. *Vấn đề bài toán với số mốc chẵn cũng xảy ra với phương pháp nội suy Gauss II, ví dụ như trong hình 1.4 (mốc x_3 không được sử dụng). Để tránh hiện tượng này, với bài toán có* $2n$ *mốc* x_i *với chỉ số là* $i = 0, 1, 2, \dots, 2n-1$*, Gauss II sẽ ưu tiên*

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$	$\Delta^4 y$	$\dots$
$\dots$	$\dots$					
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					
$\dots$	$\dots$					

Hình 1.3: Bảng sai phân với Gauss II

chọn mốc xuất phát thứ $i = n$.

Nhận xét 1.6. *Tương tự như Gauss I, ta cũng có thể biến đổi công thức Newton để nhận được công thức của phương pháp Gauss II (xem thêm [1]).*

Nhận xét 1.7. *Vì cả hai phương pháp Gauss I và Gauss II có bản chất là phương pháp Newton nên sai số của hai công thức Gauss có thể được đánh giá qua sai số của công thức Newton và là:*

$$R_n(x) \approx \frac{\Delta^{n+1} y_0}{(n+1)!} \prod_{k=0}^{n} (t-k)$$

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$	$\Delta^4 y$	$\Delta^5 y$
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		$\Delta^3 y_{-1}$		$\Delta^5 y_{-2}$
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					

Hình 1.4: Bảng sai phân với Gauss II và số mốc chẵn

Chương 2

Thuật toán

2.1 Một số hàm phụ trợ

Trước khi đi vào phần thuật toán chính của hai phương pháp, ta cần xây dựng một số hàm phụ trợ được sử dụng trong thuật toán chính.

- Hàm **saiphan**(a) nhằm mục đích tìm sai phân với đầu vào là một vector, từ đó ta sẽ tính được các cột sai phân các cấp là thành phần của bảng sai phân.
  Ví dụ: Input: $a = [1, 2, 4, 1]$ thì output: $[1, 2, -3]$
  Hàm **saiphan**(a: vector):

  $ans \leftarrow$ empty list
  $n \leftarrow length(a)$
  for $i = 1 \rightarrow n-1$ **do**
      Append $(a[i+1] - a[i])$ to list ans
  end for
  return ans

- Hàm **bangsaiphan**(x, y) để xây dựng bảng sai phân từ dữ liệu đầu, sử dụng hàm **saiphan** để tính liên tiếp các cấp sai phân đến cấp cao nhất. Mỗi thành phần $table[i]$ ứng với một cột sai phân cấp i trong bảng sai phân.

Hàm **bangsaiphan**(x, y: vector):

$n \leftarrow length(x)$
$table \leftarrow list[y]$
for $i = 1 \rightarrow n-1$ **do**
    Append **saiphan**($table[i]$) to $table$
end for
return table

- Hàm **horner**(*poly*, *x*) để tính giá trị đa thức có các hệ số là vector *poly* tại giá trị *x*.

  Ví dụ: Với $poly = [1, 2, 3]$ ứng đa thức $1 + 2x + 3x^2$ có giá trị tại $x = 2$ là 17.

  Hàm **horner**(*poly*: vector, *x*: float):

  $n \leftarrow len(poly)$
  $ans \leftarrow poly[n-1]$
  for $i = n-2 \rightarrow 0$ **do**
      $ans \leftarrow ans * x + poly[i]$
  end for

- Hàm **mulConst**(*a*, *c*) để nhân đa thức có các hệ số là vector *a* với hằng số *c*.

  Ví dụ: Với $a = [1, 2, 3]$ và $c = 2$ thì output là $[2, 4, 6]$.

  Hàm **mulConst**(*a*: vector, *c*: float):

  $ans \leftarrow$ empty list
  for $i = 0 \rightarrow len(a)$ **do**
      Append $c * a[i]$ to ans
  end for
  return ans

- Hàm **sumHorner**(*a*: vector, *b*: vector) là hàm cộng hai đa thức với bộ hệ số là *a* và *b*.

  Ví dụ: Với $a = [1, 2, 3]$ và $b = [2, 3, 4]$ thì output là $[3, 5, 7]$.

Hàm **sumHorner**(a: vector, b: vector):

if $len(a) > len(b)$ **then**
    for $i = 0 \rightarrow len(b)$ **do**
        $a[i] \leftarrow a[i] + b[i]$
    end for
    return a
else
    for $i = 0 \rightarrow len(a)$ **do**
        $b[i] \leftarrow a[i] + b[i]$
    end for
    return b
end if
return ans

- Hàm **mulHorner**(a, m, $case$) dùng để nhân đa thức a với đa thức bậc một $(t + int(m/2))$ nhằm hỗ trợ tính phần tích $t(t-1)(t+1)\dots$ trong công thức Gauss, $case \in \{1, 2\}$ tương ứng với Gauss I hoặc Gauss II.

  Hàm **mulHorner**(a: vector, m: float, $case$: int):

$c \leftarrow int(m/2)$
$ans \leftarrow$ empty list
if ($m\%2 == 1$ and $case == 1$) or ($m\%2 == 0$ and $case == 2$) **then**:
    for $i = 0 \rightarrow len(a)$ **do**
        if $j == 0$ **then**
            Append $a[j] * c$ to ans
        else
            Append $a[j-1] + a[j] * c$ to ans
        end if
    end for
else

for $i = 0 \to len(a)$ **do**
    if $j == 0$ **then**
        Append $a[j] * (-c)$ to *ans*
    else
        Append $a[j-1] + a[j] * (-c)$ to *ans*
    end if
end for
end if
Append 1 to *ans*
return *ans*

2.2 Thuật toán chính

2.2.1 Phương pháp Gauss I

Hàm chính **Gauss1Horner**(x, y, $value$) là hàm tìm đa thức nội suy theo phương pháp Gauss I với bảng dữ liệu là x,y và đưa ra giá trị hàm tại $value$. Kết quả trả ra là hệ số của đa thức nội suy và giá trị đa thức tại $value$. Hàm sẽ chia ra hai trường hợp với số mốc chẵn và số mốc lẻ. Vòng lặp **for** tại dòng số 7 và 16 là quá trình cộng dần đa thức khi kết nạp thêm một mốc theo công thức lý thuyết. Để giảm khối lượng tính toán, thành phần giai thừa $fact$ dưới mẫu của các a_i trong công thức lý thuyết được tích hợp trong vòng **for** này. Thành phần $poly$ chính là thành phần tích $t(t-1)(t+1)\ldots$ trong công thức. Đầu ra bao gồm ans là một vector các hệ số theo cấp tăng dần của đa thức nội suy và $result$ là giá trị đa thức tại $value$.

Hàm chính **Gauss1Horner**(x, y: vector, $value$: float):

1: $n \leftarrow len(x)$, $h \leftarrow x[1] - x[0]$
2: $table \leftarrow$ **bangsaiphan**(x, y)
3: $fact \leftarrow 1$
4: **if** n%2==1 **then**

$poly \leftarrow [0]$
$ans \leftarrow [table[0][int(n/2)]]$
for $i = 1 \rightarrow n$ **do**
$fact \leftarrow fact * i$
$poly \leftarrow$ **mulHorner**($poly$, i, 1)
$ans \leftarrow$ **sumHorner**(ans, **mulConst**($poly$, $table[i][int((n-i)/2)]/fact$))
end for
$result \leftarrow$ **horner**(ans, $(value - x[int(n/2)])/h$)
else
$poly \leftarrow [0]$
$ans \leftarrow [table[0][int((n-1)/2)]]$
for $i = 1 \rightarrow n$ **do**
$fact \leftarrow fact * i$
$poly \leftarrow$ **mulHorner**($poly$, i, 1)
$ans \leftarrow$**sumHorner**(ans, **mulConst**($poly$, $table[i][int((n-i-1)/2)]/fact$))
end for
$result \leftarrow$ **horner**(ans, $(value - x[int((n-1)/2)])/h$)
end if
return ans, $result$

2.2.2 Phương pháp Gauss II

Hàm chính **Gauss2Horner**(x, y, $value$) là hàm tìm đa thức nội suy theo phương pháp Gauss II với bảng dữ liệu là x,y và đưa ra giá trị đa thức tại $value$. Kết quả trả ra là vector hệ số của đa thức nội suy và giá trị đa thức tại $value$. Hàm sẽ chia ra hai trường hợp với số mốc chẵn và số mốc lẻ. Việc xây dựng hàm này tương tự hàm Gauss I, điểm khác biết ở việc chọn mốc xuất phát, thành phần tích $t(t-1)(t+1)\dots$ và giá trị sai phân sẽ chọn trên từng cột cấp sai phân của bảng sai phân.

Hàm chính **Gauss2Horner**(x, y: vector, $value$: float):

```
n ← len(x), h ← x[1] − x[0]
table ← bangsaiphan(x, y)
fact ← 1
if n%2==1 then
    poly ← [0]
    ans ← [table[0][int(n/2)]]
    for i = 1 → n do
        fact ← fact ∗ i
        poly ← mulHorner(poly, i, 2)
        ans ← sumHorner(ans, mulConst(poly, table[i][int((n−i−1)/2)]/fact))
    end for
    result ← horner(ans, (value − x[int(n/2)])/h)
else
    poly ← [0]
    ans ← [table[0][int(n/2)]]
    for i = 1 → n do
        fact ← fact ∗ i
        poly ← mulHorner(poly, i, 2)
        ans ← sumHorner(ans, mulConst(poly, table[i][int((n − i)/2)]/fact))
    end for
    result ← horner(ans, (value − x[int(n/2)])/h)
end if
return ans, result
```

Nhận xét 2.1. *Cả thuật toán chính Gauss I và Gauss II đã xây dựng ở trên đều có độ phức tạp cỡ* $O\left(\frac{5n^2}{2}\right)$. *Nếu chỉ dừng lại ở yêu cầu là tính giá trị đa thức tại một điểm nào đó mà không cần đưa ra đa thức thì thuật toán có thể sửa đổi để đạt độ phức tạp cỡ* $O\left(\frac{3n^2}{2}\right)$.

2.3 Thuật toán với bộ dữ liệu lớn

Một vấn đề đặt ra với các phương pháp nội suy trung tâm chính là việc trích xuất dữ liệu, tức là việc chọn số lượng mốc nội suy đủ dùng khi bảng dữ liệu đã cho có kích thước lớn. Ví dụ như ta có bảng dữ liệu giá nhà trung bình trên một mét vuông từ năm 1800 đến năm 2020 với bước thời gian là 5 năm tức là có 45 điểm dữ liệu. Tuy nhiên, ta cần ước lượng giá nhà tại thời điểm năm 1943 không có trong bảng dữ liệu thì việc sử dụng cả 45 mốc là không cần thiết, do đó việc lựa chọn mốc nội suy trung tâm và lựa chọn số mốc muốn sử dụng là rất quan trọng. Thuật toán sau sẽ chọn mốc nội suy xuất phát và cho phép chúng ta chọn số mốc cần sử dụng tùy ý.

Thuật toán sẽ tính đa thức nội suy và giá trị tại điểm $value$ từ đầu vào x, y với số mốc sử dụng là num. Mốc nội suy xuất phát sẽ được chọn là một trong hai mốc trong bảng dữ liệu mà gần giá trị $value$ nhất. Chỉ số $index$ (chỉ số $i \in \{0, \dots, n-1\}$ của x_i với bộ dữ liệu có n mốc) của mốc nội suy xuất phát có thể dễ dàng tính được nhờ đặc điểm của mốc cách đều theo công thức trong dòng số 5 của thuật toán. Lệnh **If** từ dòng số 8 đến 13 là quá trình chọn mốc với số lượng num đã cho sao cho mốc xuất phát x_{index} là mốc trung tâm. Ta có thể tùy chỉnh phương pháp sử dụng tại dòng số 15, ở đây thuật toán được xây dựng cho phương pháp Gauss I với hàm **GaussIHorner** đã trình bày ở phần trước.

Hàm **GaussBig**(x: vector, y: vector, $value$: float, num: int)

1: $n \leftarrow len(x)$
2: **if** $num \geq n$ **then**:
3:     **return** "No enough nodes"
4: **end if**
5: $index \leftarrow int((value - x[0])/(x[1] - x[0]))$
6: Kiểm tra có đủ mốc với hai mốc trung tâm $x[index]$ hoặc $x[index+1]$.
7: Chọn mốc $x[index]$ hoặc $x[index+1]$ làm mốc trung tâm x_{index} và lưu $index$ phù hợp.

if $num\%2 == 1$ **then**

  $x_{new} = x[index - int(num/2) : index + int(num/2) + 1]$

  $y_{new} = y[index - int(num/2) : index + int(num/2) + 1]$

else

  $x_{new} = x[index - int(num/2) + 1 : index + int(num/2) + 1]$

  $y_{new} = y[index - int(num/2) + 1 : index + int(num/2) + 1]$

end if

$ans =$ **Gauss1Horner**$(x_{new}, y_{new}, value)$

return ans

Nhận xét 2.2. *Bên cạnh các thuật toán trên đã sử dụng nền tảng là phương pháp Horner để tính đa thức, trong chương trình cụ thể, em có sử dụng thêm sự hỗ trợ của thư viện tính toán hình thức symbolic Sympy để đưa ra công thức của đa thức nội suy theo x từ đa thức nội suy theo t đã được tính nhờ Horner. Điều này giúp ta thấy được sự duy nhất của đa thức nội suy theo x trong các trường hợp hai phương pháp nội suy Gauss I và II cho ra hai đa thức theo t khác nhau.*

```python

# dùng sybolic để đưa ra đa thức theo x từ t
def display_poly_gauss_1(poly_t, x):
    # đa thức theo t
    t = sym.Symbol('t')
    ans = 0
    for i in range(len(poly_t)):
        ans = ans + np.round(poly_t[i],4)*t**i

    ans1 = sym.expand(ans)

    # đa thức theo x
    n = len(x)
    h = x[1] - x[0]
    if(n%2==1):
        x0 = x[int(n/2)]
    else:
        x0 = x[int((n-1)/2)]
    x = sym.Symbol('x')
    ans2 = sym.expand(ans.subs(t, (x-x0)/h))
    return ans1, ans2
```

Hình 2.1: Hàm hiển thị đa thức theo t và x sử dụng symbolic của Gauss I.

Chương 3

Chương trình, ví dụ

Chương trình được xây dựng bằng ngôn ngữ Python dựa trên các thuật toán đã trình bày ở chương 2 và được sử dụng để chạy các ví dụ trong chương này (source chương trình có thể được xem tại các file đi kèm cùng báo cáo này). Sau đây, ta sẽ xem xét một các ví dụ để hiểu rõ hơn về hai phương pháp Gauss I và Gauss II.

Ví dụ 3.1. *Tính giá trị tại $x = 345$ với bộ dữ liệu:*

x	310	320	330	340	350	360	370
y	2.4914	2.5052	2.5185	2.5315	2.5441	2.5563	2.5687

Đây là trường hợp bộ dữ liệu với số mốc lẻ, ta có bảng sai phân tương ứng. Kết quả chạy với phương pháp Gauss 1 và Gauss 2 lần lượt cho trong hình 3.1 và 3.2. Do đây là trường hợp số mốc lẻ nên có thể thấy cả hai phương pháp đều chọn mốc xuất phát là $x = 340$ và có hệ số của đa thức theo t cũng như giá trị tại 345 là giống nhau. Phép kiểm tra đa thức bằng cách tích lại giá trị của đa thức tại các mốc cho thấy đa thức nhận được hoàn toàn chính xác.

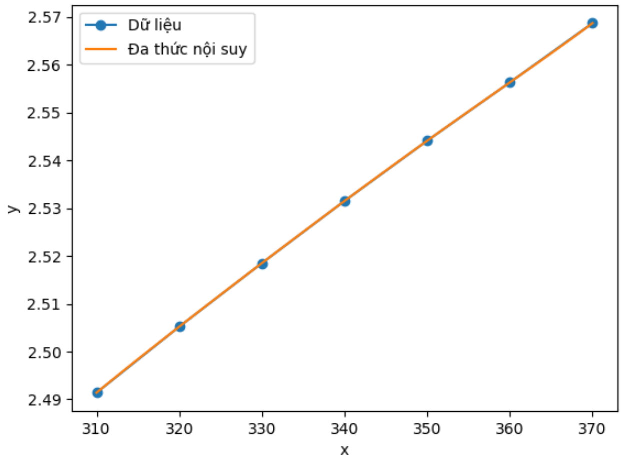
Ngoài ra, thông qua biểu diễn dữ liệu có thể thấy phân bố của các điểm dữ liệu gần như nằm trên một đường cong khá thẳng, do đó dù đa thức nội suy có bậc 6 nhưng các hệ số tại bậc ≥ 4 gần hư bằng 0. Nếu làm tròn và rút gọn, ta có thể nhận lại một đa thức bậc 1. Hình 3.3 là biểu diễn dáng điệu của đa thức nội suy theo x của cả hai phương pháp.

```
Phương pháp Gauss 1
Bảng sai phân:
310.0   2.4914  0.0138  -0.0005 0.0002  -0.0003 0.0004  0.0001
320.0   2.5052  0.0133  -0.0003 -0.0001 0.0001  0.0005
330.0   2.5185  0.013   -0.0004 0.0 0.0006
340.0   2.5315  0.0126  -0.0004 0.0006
350.0   2.5441  0.0122  0.0002
360.0   2.5563  0.0124
370.0   2.5687
-----------------------------------------
Các sai phân đã chọn: [2.5315, 0.0126, -0.0004, 0.0, 0.0001, 0.0005, 0.0001]
Hệ số đa thức theo t: [ 2.5315e+00  1.2800e-02 -2.0000e-04 -0.0000e+00  0.0000e+00  0.0000e+00
  0.0000e+00]
Giá trị tại 345 là: 2.53785771484375
-----------------------------------------
Kiểm tra đa thức:
Predict: [2.4914, 2.5052, 2.5185, 2.5315, 2.5441, 2.5563, 2.5686999999999998]
True: [2.4914, 2.5052, 2.5185, 2.5315, 2.5441, 2.5563, 2.5687]
[Finished in 3.3s]
```

Hình 3.1: Kết quả chạy ví dụ 3.1 với Gauss 1.

```
Phương pháp Gauss 2
Bảng sai phân:
310.0   2.4914  0.0138  -0.0005 0.0002  -0.0003 0.0004  0.0001
320.0   2.5052  0.0133  -0.0003 -0.0001 0.0001  0.0005
330.0   2.5185  0.013   -0.0004 0.0 0.0006
340.0   2.5315  0.0126  -0.0004 0.0006
350.0   2.5441  0.0122  0.0002
360.0   2.5563  0.0124
370.0   2.5687
-----------------------------------------
Các sai phân đã chọn: [2.5315, 0.013, -0.0004, -0.0001, 0.0001, 0.0004, 0.0001]
Hệ số đa thức theo t: [ 2.5315e+00  1.2800e-02 -2.0000e-04 -0.0000e+00  0.0000e+00  0.0000e+00
  0.0000e+00]
Giá trị tại 345 là: 2.53785771484375
-----------------------------------------
Kiểm tra đa thức:
Predict: [2.4914, 2.5052, 2.5185, 2.5315, 2.5441, 2.5563, 2.5686999999999998]
True: [2.4914, 2.5052, 2.5185, 2.5315, 2.5441, 2.5563, 2.5687]
[Finished in 7.0s]
```

Hình 3.2: Kết quả chạy ví dụ 3.1 với Gauss 2.



Hình 3.3: Dáng điệu của đa thức nội suy trong đoạn nội suy trong ví dụ 1.

Ví dụ 3.2.

Tính giá trị tại $x = 34$ *với bộ dữ liệu:*

x	20	25	30	35	40	45
y	354	332	291	260	231	204

Ví dụ 3.2 là trường hợp số mốc chẵn, kết quả chạy với phương pháp Gauss 1 và Gauss 2 lần lượt cho trong hình 3.4 và 3.5. Khi này vì việc chọn mốc nội suy xuất phát trung tâm để dùng hết mốc của hai phương pháp Gauss 1 và Gauss 2 là khác nhau nên đa thức theo t của hai phương pháp sẽ khác nhau, tuy nhiên do đa thức nội suy theo x là duy nhất nên có thể thấy giá trị tại 34 của cả hai vẫn giống nhau. Hình 3.6 là biểu diễn dáng điệu của đa thức nội suy theo x của cả hai phương pháp.

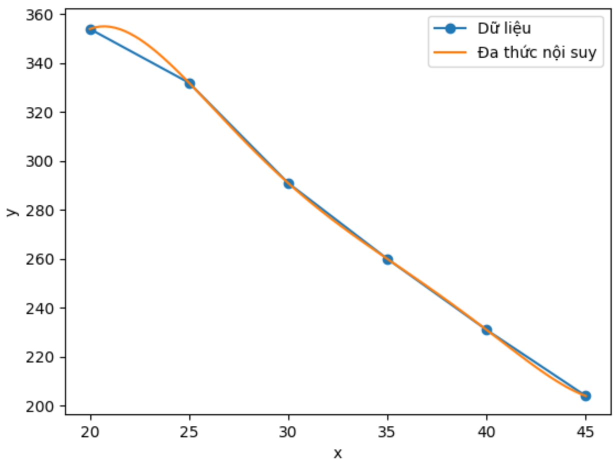
Đặc biệt với bài toán có số mốc chẵn có thể dẫn đến trường hợp không sử dụng hết mốc. Với ví dụ này, theo phương pháp Gauss 2 nếu chọn mốc xuất phát là $x = 30$ như trong hình 3.7, ta sẽ kết nạp đến mốc $x = 45$ từ đó đa thức nội suy chỉ còn bậc 4. Và theo phép kiểm tra đa thức trong hình thì tính giá trị đa thức tại $x = 45$ sẽ trở thành ngoại suy với sai số lớn khi chỉ đạt ≈ 159 so với giá trị chính xác 204. Do đó việc lựa chọn mốc xuất phát để sử dụng hết các mốc cần thiết là rất quan trọng.

```
Phương pháp Gauss 1
Bảng sai phân:
20.0    354.0   -22.0   -19.0   29.0    -37.0   45.0
25.0    332.0   -41.0   10.0    -8.0    8.0
30.0    291.0   -31.0   2.0 0.0
35.0    260.0   -29.0   2.0
40.0    231.0   -27.0
45.0    204.0
----------------------------------------
Các sai phân đã chọn: [291.0, -31.0, 10.0, -8.0, -37.0, 45.0]
Hệ số đa thức theo t: [ 2.9100e+02 -3.6250e+01  6.5417e+00 -1.2500e-01 -1.5417e+00  3.7500e-01]
Giá trị tại 34 là: 265.61408
----------------------------------------
Kiểm tra đa thức:
Predict: [354.0, 332.0, 291.0, 260.0, 231.0, 204.0]
True: [354.0, 332.0, 291.0, 260.0, 231.0, 204.0]
[Finished in 3.5s]
```

Hình 3.4: Kết quả chạy ví dụ 3.2 với Gauss 1.

```
Phương pháp Gauss 2
Bảng sai phân:
20.0    354.0   -22.0   -19.0   29.0    -37.0   45.0
25.0    332.0   -41.0   10.0    -8.0    8.0
30.0    291.0   -31.0   2.0 0.0
35.0    260.0   -29.0   2.0
40.0    231.0   -27.0
45.0    204.0
----------------------------------------
Các sai phân đã chọn: [260.0, -31.0, 2.0, -8.0, 8.0, 45.0]
Hệ số đa thức theo t: [260.     -27.8333   0.6667  -2.5417   0.3333   0.375 ]
Giá trị tại 34 là: 265.61408
----------------------------------------
Kiểm tra đa thức:
Predict: [354.0, 332.0, 291.0, 260.0, 231.0, 204.0]
True: [354.0, 332.0, 291.0, 260.0, 231.0, 204.0]
[Finished in 8.0s]
```

Hình 3.5: Kết quả chạy ví dụ 3.2 với Gauss 2.



Hình 3.6: Dáng điệu của đa thức nội suy trong đoạn nội suy trong ví dụ 3.2.

```
Phương pháp Gauss 2
Bảng sai phân:
20.0    354.0   -22.0   -19.0   29.0    -37.0   45.0
25.0    332.0   -41.0   10.0    -8.0    8.0
30.0    291.0   -31.0   2.0 0.0
35.0    260.0   -29.0   2.0
40.0    231.0   -27.0
45.0    204.0
-----------------------------------------
Các sai phân đã chọn: [291.0, -41.0, 10.0, 29.0, -37.0]
Hệ số đa thức theo t: [291.     -37.75     6.5417    1.75     -1.5417]
Giá trị tại 34 là: 265.2512
-----------------------------------------
Kiểm tra đa thức:
Predict: [354.0, 332.0, 291.0, 260.0, 231.0, 158.99999999999997]
True: [354.0, 332.0, 291.0, 260.0, 231.0, 204.0]
[Finished in 10.2s]
```

Hình 3.7: Kết quả chạy ví dụ 3.2 với Gauss 2 nhưng không dùng hết mốc.

Ví dụ 3.3. *Đảo ngược thứ tự bảng giá trị trong ví dụ 3.1.*

Ta sẽ xem xét trường hợp đảo ngược thứ tự của bảng giá trị trong ví dụ 3.1, khi đó kết quả chạy với phương pháp Gauss 1 cho trong hình 3.8. Có thể thấy thuật toán vẫn cho ra kết quả đúng trong ví dụ 3.1. Khi đảo ngược thứ tự, kết quả bảng sai phân thay đổi dẫn đến hệ số của đa thức theo t cũng thay đổi khi mà dấu của các hệ số ứng với số mũ lẻ sẽ đổi dấu so với ví dụ 3.1. Đa thức theo t là khác nhau giữa hai ví dụ, tuy nhiên do ta đã sử dụng hai phép đổi biến từ x sang t ngược dấu nhau nên khi quay lại đa thức theo x thì đa thức này vẫn là duy nhất, do đó giá trị hàm tại 345 vẫn không hề thay đổi.

```
Phương pháp Gauss 1
Bảng sai phân:
370.0   2.5687  -0.0124 0.0002  -0.0006 0.0006  -0.0005 0.0001
360.0   2.5563  -0.0122 -0.0004 0.0 0.0001  -0.0004
350.0   2.5441  -0.0126 -0.0004 0.0001  -0.0003
340.0   2.5315  -0.013  -0.0003 -0.0002
330.0   2.5185  -0.0133 -0.0005
320.0   2.5052  -0.0138
310.0   2.4914
----------------------------------------
Các sai phân đã chọn: [2.5315, -0.013, -0.0004, 0.0001, 0.0001, -0.0004, 0.0001]
Hệ số đa thức theo t: [ 2.5315e+00 -1.2800e-02 -2.0000e-04  0.0000e+00  0.0000e+00 -0.0000e+00
  0.0000e+00]
Giá trị tại 345 là: 2.53785771484375
----------------------------------------
Kiểm tra đa thức:
Predict: [2.5686999999999998, 2.5563, 2.5441, 2.5315, 2.5185, 2.5052, 2.4914]
True: [2.5687, 2.5563, 2.5441, 2.5315, 2.5185, 2.5052, 2.4914]
[Finished in 5.1s]
```

Hình 3.8: Kết quả chạy ví dụ 3.3 với Gauss 1.

Ví dụ 3.4. *Tính giá trị hàm số* $f(x) = x^7 - 4x^3 + x - 2$*, biết giá trị hàm số tại* $x = 1, 2, \ldots, 40$ *tại*

- $x = 2.5$ *với 7 mốc nội suy*
- $x = 2.5$ *với 6 mốc nội suy*
- $x = 21.4$ *với 8 mốc nội suy*
- $x = 21.4$ *với 9 mốc nội suy*

Trong ví dụ này, ta sẽ xem xét trường hợp với bộ dữ liệu lớn, cụ thể là bộ với 40 cặp dữ liệu được sinh ra từ một hàm số để kiểm nghiệm thuật toán Gauss với bộ dữ liệu lớn.

Trường hợp $x = 2.5$ với 7 mốc nội suy, kết quả trả ra là không đủ mốc nội suy trong hình 3.9 bởi ta không thể chọn được 7 mốc sao cho có một mốc gần giá trị $x = 2.5$ làm mốc trung tâm.

```
if __name__ == '__main__':
    print('Phương pháp Gauss với dữ liệu lớn')

    # ví dụ 4:
    # Hàm sinh dữ liệu
    # Dãy data lớn
    x1 = range(1, 40, 1)
    y1 = [i**7 - 4*i**3 + i - 2 for i in x1]
    value = 2.5
    num = 7
```

```
Phương pháp Gauss với dữ liệu lớn
----------------------------------------
Không đủ mốc
[Finished in 2.8s]
```

Hình 3.9: Kết quả chạy ví dụ 3.4 khi $x = 2.5$ với 7 mốc nội suy.

```
Phương pháp Gauss với dữ liệu lớn
----------------------------------------
Mốc được chọn: x = 3
Bảng sai phân:
1   -4   100  1884    10182   25200   31920
2   96   1984    12066   35382   57120
3   2080    14050   47448   92502
4   16130   61498   139950
5   77628   201448
6   279076
Các sai phân đã chọn theo Gauss 1: [2080, 14050, 12066, 35382, 25200, 31920]
Các sai phân đã chọn theo Gauss 2: [16130, 14050, 47448, 35382, 57120, 31920]


Gauss 1 Horner:
Hệ số: [2080.0, 5284.0, 4983.0, 2467.0, 1050.0, 266.0]
Giá trị tại 2.5 là: 432.6875
Đa thức theo t: 266.0*t**5 + 1050.0*t**4 + 2467.0*t**3 + 4983.0*t**2 + 5284.0*t + 2080.0
Đa thức theo x: 266.0*x**5 - 2940.0*x**4 + 13807.0*x**3 - 32340.0*x**2 + 36325.0*x - 15122.0


Gauss 2 Horner:
Hệ số: [16130.0, 28181.0, 21344.0, 9327.0, 2380.0, 266.0]
Giá trị tại 2.5 là: 432.6875
Đa thức theo t: 266.0*t**5 + 2380.0*t**4 + 9327.0*t**3 + 21344.0*t**2 + 28181.0*t + 16130.0
Đa thức theo x: 266.0*x**5 - 2940.0*x**4 + 13807.0*x**3 - 32340.0*x**2 + 36325.0*x - 15122.0
[Finished in 1.5s]
```

Hình 3.10: Kết quả chạy ví dụ 3.4 khi $x = 2.5$ với 6 mốc nội suy.

Tuy nhiên, với trường hợp $x = 2.5$ cùng 6 mốc nội suy, ta lại có thể chọn được 6 mốc đó là $1, 2, 3, 4, 5, 6$, khi đó mốc $x = 3$ gần 2.5 nhất là một mốc trung tâm. Khi đó, kết quả chạy cho trong hình 3.10. Lưu ý đây là trường hợp mốc số chẵn nên để dùng hết mốc theo Gauss 2, ta phải chọn mốc xuất phát trung tâm là $x = 4$. Có thể thấy hai đa thức theo t của hai phương pháp là khác nhau nhưng đa thức theo x là duy nhất.

Trong hai trường hợp trước, số mốc nội suy sử dụng là khá bé nên đa thức nội suy ta nhận được đều có bậc thấp hơn đa thức sinh ra bộ dữ liệu. Tiếp theo, ta xem xét trường hợp $x = 21.4$ với 8 mốc nội suy cho kết quả như trong hình 3.11. Có thể thấy với 8 mốc nội suy được sử dụng, ta sẽ có một đa thức bậc 7 và đúng bằng đa thức đã sinh ra bộ dữ liệu $x^7 - 4x^3 + x - 2$.

Vậy với trường hợp $x = 21.4$ với 9 mốc nội suy thì sao? Khi đó như trong hình 3.12, đa thức nội suy ta đạt được vẫn là đa thức bậc 7 (như trong đề bài) vì hệ số của bậc 8 bằng 0 (giá trị sai phân cấp 8 bằng 0). Điều này sẽ xảy ra với số lượng mốc chọn lớn hơn 8, bởi ta không thể nội suy đa thức theo phương pháp Gauss với cấp cao hơn cấp của đa thức sinh ra bộ dữ liệu.

```
Phương pháp Gauss với dữ liệu lớn
----------------------------------------
Mốc được chọn: x = 21
Bảng sai phân:
18  612196720   281647600   104476098   30483702    6736800 1060080 105840  5040
19  893844320   386123698   134959800   37220502    7796880 1165920 110880
20  1279968018  521083498   172180302   45017382    8962800 1276800
21  1801051516  693263800   217197684   53980182    10239600
22  2494315316  910461484   271177866   64219782
23  3404776800  1181639350  335397648
24  4586416150  1517036998
25  6103453148
Các sai phân đã chọn theo Gauss 1: [1801051516, 693263800, 172180302, 45017382, 7796880, 1165920, 105840, 5040]
Các sai phân đã chọn theo Gauss 2: [2494315316, 693263800, 217197684, 45017382, 8962800, 1165920, 110880, 5040]


Gauss 1 Horner:
Hệ số: [1801051516.0, 600357556.0, 85765869.0, 6806831.0, 324135.0, 9261.0, 147.0, 1.0]
Giá trị tại 21.4 là: 2055361107.9163895
Đa thức theo t: 1.0*t**7 + 147.0*t**6 + 9261.0*t**5 + 324135.0*t**4 + 6806831.0*t**3 + 85765869.0*t**2 + 600357556.0*t + 1801051516.0
Đa thức theo x: 1.0*x**7 - 4.0*x**3 + 1.0*x - 2.0


Gauss 2 Horner:
Hệ số: [2494315316.0, 793653521.0, 108226008.0, 8198956.0, 372680.0, 10164.0, 154.0, 1.0]
Giá trị tại 21.4 là: 2055361107.9163895
Đa thức theo t: 1.0*t**7 + 154.0*t**6 + 10164.0*t**5 + 372680.0*t**4 + 8198956.0*t**3 + 108226008.0*t**2 + 793653521.0*t + 2494315316.0
Đa thức theo x: 1.0*x**7 - 4.0*x**3 + 1.0*x - 2.0
[Finished in 2.8s]
```

Hình 3.11: Kết quả chạy ví dụ 3.4 khi $x = 21.4$ với 8 mốc nội suy.

```
Phương pháp Gauss với dữ liệu lớn
----------------------------------------
Mốc được chọn: x = 21
Bảng sai phân:
17  410319036   201877684   79769916    24706182    5777520 959280  100800  5040    0
18  612196720   281647600   104476098   30483702    6736800 1060080 105840  5040
19  893844320   386123698   134959800   37220502    7796880 1165920 110880
20  1279968018  521083498   172180302   45017382    8962800 1276800
21  1801051516  693263800   217197684   53980182    10239600
22  2494315316  910461484   271177866   64219782
23  3404776800  1181639350  335397648
24  4586416150  1517036998
25  6103453148
Các sai phân đã chọn theo Gauss 1: [1801051516, 693263800, 172180302, 45017382, 7796880, 1165920, 105840, 5040, 0]
Các sai phân đã chọn theo Gauss 2: [1801051516, 521083498, 172180302, 37220502, 7796880, 1060080, 105840, 5040, 0]


Gauss 1 Horner:
Hệ số: [1801051516.0, 600357556.0, 85765869.0, 6806831.0, 324135.0, 9261.0, 147.0, 1.0, 0.0]
Giá trị tại 21.4 là: 2055361107.9163895
Đa thức theo t: 1.0*t**7 + 147.0*t**6 + 9261.0*t**5 + 324135.0*t**4 + 6806831.0*t**3 + 85765869.0*t**2 + 600357556.0*t + 1801051516.0
Đa thức theo x: 1.0*x**7 - 4.0*x**3 + 1.0*x - 2.0


Gauss 2 Horner:
Hệ số: [1801051516.0, 600357556.0, 85765869.0, 6806831.0, 324135.0, 9261.0, 147.0, 1.0, 0.0]
Giá trị tại 21.4 là: 2055361107.9163895
Đa thức theo t: 1.0*t**7 + 147.0*t**6 + 9261.0*t**5 + 324135.0*t**4 + 6806831.0*t**3 + 85765869.0*t**2 + 600357556.0*t + 1801051516.0
Đa thức theo x: 1.0*x**7 - 4.0*x**3 + 1.0*x - 2.0
[Finished in 2.3s]
```

Hình 3.12: Kết quả chạy ví dụ 3.4 khi $x = 21.4$ với 9 mốc nội suy.

Ví dụ 3.5. *Cho bảng dữ liệu nhiệt độ Trái đất từ năm 1880 đến năm 2020 với bước nhảy 5 năm. Ước lượng nhiệt độ của Trái đất tại một thời điểm nào đó trong khoảng thời gian từ 1880 đến 2020.*

Ví dụ, cuối cùng ta xem xét là bộ dữ liệu thực của nhiệt độ Trái đất từ năm 1880 đến năm 2020, ta sẽ tính tại thời điểm năm 1933 với 7 mốc nội suy. Kết quả cho như trong hình 3.13. Nhiệt độ dự đoán rơi vào cỡ -0.225.

```
Phương pháp Gauss với dữ liệu lớn
----------------------------------------
Mốc được chọn: x = 1935.0
Bảng sai phân:
1920.0  -0.276  0.053   0.012   -0.118  0.588   -1.779  3.818
1925.0  -0.223  0.065   -0.106  0.47    -1.191  2.039
1930.0  -0.158  -0.041  0.364   -0.721  0.848
1935.0  -0.199  0.323   -0.357  0.127
1940.0  0.124   -0.034  -0.23
1945.0  0.09    -0.264
1950.0  -0.174
Các sai phân đã chọn theo Gauss 1: [-0.199, 0.323, 0.364, -0.721, -1.191, 2.039, 3.818]
Các sai phân đã chọn theo Gauss 2: [-0.199, -0.041, 0.364, 0.47, -1.191, -1.779, 3.818]


Gauss 1 Horner:
Hệ số: [-0.199, 0.16625, 0.2528361111111111, -0.026333333333333334, -0.0761388888888889, 0.0010833333333333355, 0.005302777777777778]
Giá trị tại 1933 là: -0.22529941760000002
Đa thức theo t: 0.0053*t**6 + 0.0011*t**5 - 0.0761*t**4 - 0.0263*t**3 + 0.2528*t**2 + 0.1662*t - 0.199
Đa thức theo x: 3.392e-7*x**6 - 0.00393776*x**5 + 19.04708944*x**4 - 49136.46946*x**3 + 71301558.873128*x**2 - 55181061794.7438*x + 17793732760779.5


Gauss 2 Horner:
Hệ số: [-0.199, 0.16625, 0.2528361111111111, -0.026333333333333347, -0.0761388888888889, 0.0010833333333333337, 0.005302777777777778]
Giá trị tại 1933 là: -0.22529941760000002
Đa thức theo t: 0.0053*t**6 + 0.0011*t**5 - 0.0761*t**4 - 0.0263*t**3 + 0.2528*t**2 + 0.1662*t - 0.199
Đa thức theo x: 3.392e-7*x**6 - 0.00393776*x**5 + 19.04708944*x**4 - 49136.46946*x**3 + 71301558.873128*x**2 - 55181061794.7438*x + 17793732760779.5
[Finished in 2.1s]
```

Hình 3.13: Kết quả chạy ví dụ 3.5 khi tại thời điểm năm 1933 với 7 mốc nội suy.

Chương 4

Kết luận

Báo cáo đã trình bày về hai phương pháp nội suy trung tâm Gauss I và Gauss II bao gồm cơ sở lý thuyết, thuật toán và ví dụ chạy chương trình. Đánh giá về phương pháp thì có thể thấy Gauss I và Gauss II là hai phương pháp nội suy trung tâm cơ bản và dựa trên phương pháp nội suy Newton nhưng đã cải thiện để phục vụ tính toán tại các điểm giữa đoạn nội suy. Hai thuật toán đã trình bày đã đưa ra được đa thức nội suy theo biến t, giá trị đa thức tại một điểm trong khoảng nội suy và đều có khối lượng tính toán cỡ $O\left(\frac{5n^2}{2}\right)$. Về mặt ứng dụng, hai phương pháp Gauss không được sử dụng nhiều trong các thực tiễn. Tuy nhiên, đây là công thức quan trọng trong việc phát triển ra các công thức nội suy như Sterling hay Bessel.

Phụ lục A

Hướng dẫn sử dụng chương trình

Folder chương trình gồm 4 file chương trình:

- **Hamphutro.py**: Đây là file chứa các hàm sử dụng trong các thuật toán chính.
- **Gauss1.py**: File chương trình của phương pháp Gauss I.
- **Gauss2.py**: File chương trình của phương pháp Gauss II.
- **GaussBigData.py**: File chương trình của phương pháp Gauss với bộ dữ liệu lớn.

Để sử dụng file **Gauss1.py** và **Gauss2.py**, ta sẽ truy cập file và thực hiện nhập bộ dữ liệu. Dữ liệu có thể nhập trực tiếp trên file hoặc đọc từ một file ".txt" nào đó. (Lưu ý, đóng comment cách nhập dữ liệu còn lại để tránh lỗi chương trình). Sau khi nhập dữ liệu, chạy file **Gauss1.py** và **Gauss2.py** tương ứng. Xem hình A.1 và A.2

Để sử dụng file **GaussBigData.py**, ta sẽ truy cập file và thực hiện điều chỉnh tên file dữ liệu ".txt" và thực hiện chạy file này.

```python


if __name__ == '__main__':
  print('Phương pháp Gauss 1')

  # đọc dữ liệu trực tiếp
  # x = [310, 320, 330, 340, 350, 360, 370]
  # y = [2.4914, 2.5052, 2.5185, 2.5315, 2.5441, 2.5563, 2.5687]
  # value = 345

  # đọc dữ liệu từ file
  f = open("vd1.txt", "r")
  x = []
  y = []
  for line in f:
    val1, val2 = line.split()
    x.append(float(val1))
    y.append(float(val2))

  # x.reverse()
  # y.reverse()
  value = 34

```

Hình A.1: Nhập dữ liệu trong file **Gauss1.py** và **Gauss2.py**.

```python

if __name__ == '__main__':
  print('Phương pháp Gauss với dữ liệu lớn')

  # ví dụ 4:
  # Hàm sinh dữ liệu
  # Dãy data lớn
  x = range(1, 40, 1)
  y = [i**7 - 4*i**3 + i - 2 for i in x]
  value = 21.4
  num = 10

  # # ví dụ 5:
  # # đọc dữ liệu từ file nhiệt độ trái đất
  # f = open("vd5.txt", "r")
  # x = []
  # y = []
  # for line in f:
  #   val1, val2 = line.split()
  #   x.append(float(val1))
  #   y.append(float(val2))

  # value = 1933 # tính tại năm 1933
  # num = 7
```

Hình A.2: Nhập dữ liệu trong file **GaussBigData.py**.

Tài liệu tham khảo

[1] G. de Vahl Davis. Numerical methods in engineering and science. *Springer, Dordrecht*, pages 274–338, 1986.

[2] L. T. Vinh. Giáo trình giải tích số. *NXB Khoa học và Kỹ thuật*, 2007.