CS381: Numerical Computation & Softwares

Interpolation

Jakramate Bootkrajang

Department of Computer Science

Chiang Mai University

Rev.2 – 2019

Quote of the day

"Nothing in life is to be feared, it is only to be understood. Now is the time to understand more, so that we may fear less."

- Marie Curie -

เป้าหมาย

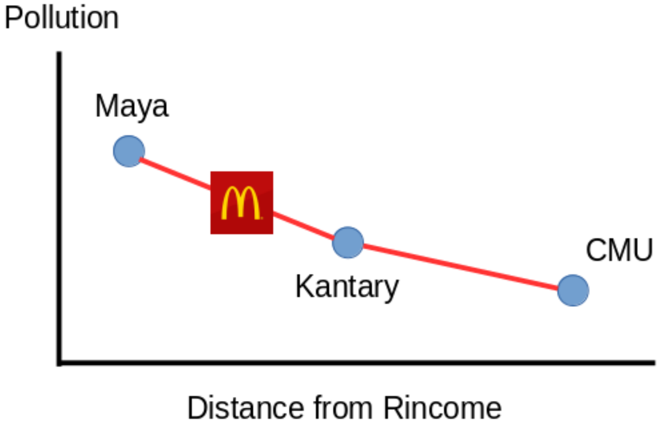
- Motivation & Problem definition
- Polynomial interpolation
- Spline interpolation

Motivation

- สมมติว่าเราต้องการติดตั้งเครื่องดักจำฝุ่นละอองในอากาศ เพื่อตรวจวัดคุณภาพอากาศ ตลอดแนวถนนนิมมาน
- เครื่องมีราคาสูงมาก เราซื้อได้แค่ 3 เครื่อง โดยติดตั้งไว้ที่ ห้างเมญ่า โรงแรมแคนทารี่ และหอประชุมมช.
- คุณภาพอากาศ ณ 3 ตำแหน่งดังกล่าวสามารถอ่านได้จากเครื่อง แต่เราจะประมาณคุณภาพอากาศตรง McDonald ได้อย่างไร

Motivation

วิธีหนึ่งเพื่อใช้ประมาณค่าคุณภาพอากาศที่ McDonald คือการลากเส้นเชื่อมจุด 3 จุดเข้าด้วยกัน และใช้ค่าบนเส้นเชื่อมดังกล่าวเป็นค่าประมาณ



Problem definition

กำหนดข้อมูล $n+1$ ตัว ในรูปคู่อันดับ $(x_0, y_0), \ldots (x_n, y_n)$ การทำ interpolation คือการหาฟังก์ชัน $F(x)$ หรือเซตของฟังก์ชัน K, $f_k(x) \in K$ ที่ลากผ่านข้อมูลทุกตัว

$$F(x_i) = y_i$$

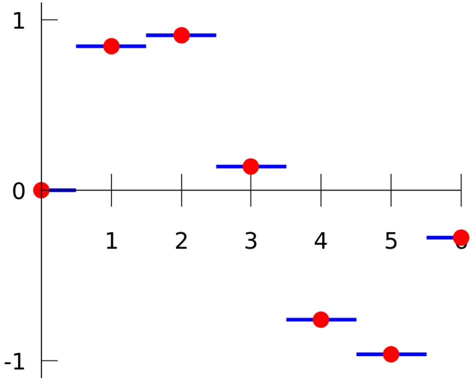
สำหรับ $i \in \{0, \ldots, n\}$ หรือ

$$f_k(x_{p_i}) = y_{p_i}$$

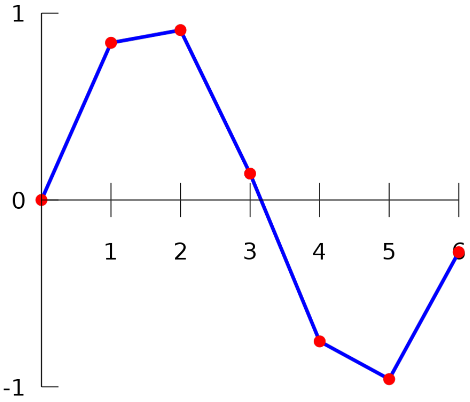
โดยที่ p_i คือ partition ที่ i ของเซต $\{0, \ldots, n\}$ และ $k \in \{1, \ldots, |K|\}$

ฟังก์ชันที่มีคุณสมบัติข้างต้นจะถูกเรียกว่า Interpolant

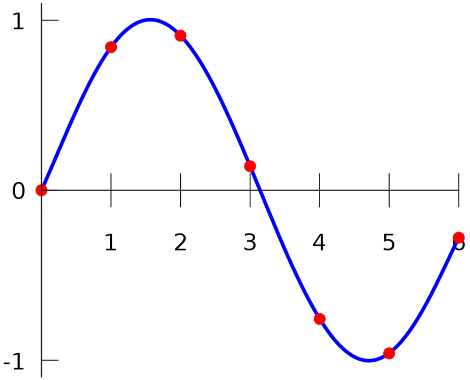
Constant interpolants



Linear interpolants



Polynomial interpolants



Polynomial Interpolation

The Interpolation Problem

กำหนดข้อมูล $n+1$ ตัว ในรูปคู่อันดับ $(x_0, y_0), \ldots (x_n, y_n)$. เราสนใจหาพหุนาม (Polynomial) ดีกรี p ให้ชื่อว่า $Q_p(x)$ ที่มีดีกรีต่ำที่สุด ที่ยังทำให้ข้อกำหนดของการทำ interpolation

$$Q_p(x_i) = y_i$$

สำหรับ $i \in \{0, \ldots, n\}$ เป็นจริง

Can we do that ?

มีทฤษฎีบทบทหนึ่งที่บอกว่า เราสามารถหา $Q_p(x)$ ดังกล่าวได้

Theorem

If $x_0, \ldots x_n$ *are distinct, then for any* $f(x_0), \ldots, f(x_n)$ *there exists a unique polynomial* $Q_n(x)$ *with sequence of coefficients* $a_0, \ldots, a_n$ *such that* $Q_n(x_i) = y_i$ *for all* $i \in \{0, \ldots, n\}$*, where*

$$Q_n(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n$$

Lagrange polynomials

Lagrange เสนอแนวคิดในการสร้างพหุนามดังกล่าวในรูปของ

$$Q_n(x) = \sum_{j=0}^{n} f(x_j)L_j(x) = f(x_0)L_0(x) + f(x_1)L_1(x) + \ldots f(x_n)L_n(x)$$

เพื่อให้เงื่อนไขของ Interpolant เป็นจริง นั่นคือ

$$Q_n(x_i) = f(x_i) \quad , 0 \leq i \leq n$$

สิ่งที่เราต้องทำก็คือหา $L_j(x)$ ที่

$$L_j(x) = \begin{cases} 1 & \text{if} \quad x = x_j \\ 0 & \text{otherwise} \end{cases}$$

A good choice of $L_j(x)$

เราสามารถสร้างพนุนามที่เราต้องการได้โดยกำหนดให้

$$
\begin{aligned}
L_j(x) &= \frac{(x-x_0)(x-x_1)\dots(x-x_{j-1})(x-x_{j+1})\dots(x-x_n)}{(x_j-x_0)(x_j-x_1)\dots(x_j-x_{j-1})(x_j-x_{j+1})\dots(x_j-x_n)} \\
&= \frac{\prod_{i=0,i\neq j}^{n}(x-x_i)}{\prod_{i=0,i\neq j}^{n}(x_j-x_i)}
\end{aligned}
\tag{1}
$$

for $j = 0, \dots, n$

Example: Polynomial Interpolation for 2 points

สมมติว่าเรามีข้อมูลสองตัว (x_0, y_0) and (x_1, y_1) การ interpolate ข้อมูลสองตัวด้วย polynomial interpolation คือการหาพหุนามดีกรี 1 ที่เชื่อมจุดทั้งสอง

เราพบว่าเราสามารถกำหนด Lagrange polynomial ให้เป็น $L_0(x) = \frac{x - x_1}{x_0 - x_1}$ and $L_1(x) = \frac{x - x_0}{x_1 - x_0}$ ดังนั้น พหุนามที่เราตามหาซึ่ง interpolate ข้อมูลทั้งสองได้คือ

$$
\begin{aligned}
Q(x) &= f(x_0)L_0(x) + f(x_1)L_1(x) = f(x_0)\frac{x - x_1}{x_0 - x_1} + f(x_1)\frac{x - x_0}{x_1 - x_0} \\
&= \frac{f(x_0) - f(x_1)}{x_0 - x_1}x + \frac{f(x_1)x_0 - f(x_0)x_1}{x_0 - x_1} = a_0 + a_1 x
\end{aligned}
$$

Quiz: ลองตรวจสอบดูว่าพหุนามดังกล่าว interpolate ข้อมูลทั้งสองจริงหรือไม่

ข้อจำกัดของ Lagrange Polynomial

ในทางปฏิบัติแล้ว เรามักไม่ค่อยใช้ Lagrange polynomial ในการ interpolate ข้อมูล ด้วยเหตุผลสองประการคือ

- พหุนามในรูปของ $Q(x) = \sum_{i=0}^{n} y_i \prod_{j=0, j\neq i}^{n} \frac{x - x_j}{x_i - x_j}$ มีลักษณะ ill-conditioned: cancellation errors สามารถเกิดขึ้นได้ !
- ในกรณีที่เรามีข้อมูลใหม่ (x_{n+1}, y_{n+1}) เราจำเป็นต้องหา Lagrange polynomial ใหม่หมดตั้งแต่ต้น

Lab: Implement Lagrange polynomial

Exercise

- Interpolate the following points $(1, 4.5), (1.5, 6), (3, 1)$
- What's the value of y at $x = 2.5$?

Newton polynomial

- Newton เสนอการสร้าง polynomial แบบ bottom-up
- โดยเริ่มจากการลองหา polynomial อย่างง่ายที่ interpolate ข้อมูล 1 ตัว
  - $Q_0(x) = a_0$
  - หากกำหนดค่าสัมประสิทธิ์ $a_0 = y_0 = f(x_0)$
- เราจะได้ $Q_0(x) = y_0$ ซึ่ง interpolate ข้อมูลได้ถูกต้อง

Newton polynomial

- ในกรณีที่ข้อมูลมีข้อมูลเพิ่มขึ้นอีก 1 ตัว เราอาจลองสร้างพหุนาม
  - $Q_1(x) = y_0 + a_1 x$
- เราต้องการปรับปรุงพหุนามข้างต้นเพื่อให้เป็นตามเงื่อนไข โดย
  - หาก $x = x_0$, $Q_1(x)$ ควรคืนค่า y_0
  - หาก $x = x_1$, $Q_1(x)$ ควรคืนค่า y_1

Newton polynomial

- $Q_1(x) = y_0 + y_1(x - x_0)$?
  - ผ่านจุด (x_0, y_0) อย่างเดียว

Newton polynomial

- $Q_1(x) = y_0 + \frac{y_1}{(x_1 - x_0)}(x - x_0)$?
    - ผ่านจุด (x_0, y_0) และเกือบผ่านจุด (x_1, y_1)

Newton polynomial

- ต้องนำ y_0 ไปลบออกจาก y_1 จึงจะได้ค่าที่ถูกต้อง
- $Q_1(x) = y_0 + \frac{y_1 - y_0}{(x_1 - x_0)}(x - x_0)$?

Newton polynomial

- หากมีข้อมูลมาเพิ่มอีก 1 ตัว
- $Q_2(x) = y_0 + \frac{y_1 - y_0}{(x_1 - x_0)}(x - x_0) + a_2 x^2$?

Newton polynomial

- First guess using the same trick,
- $Q_2(x) = y_0 + \frac{y_1 - y_0}{(x_1 - x_0)}(x - x_0) + \frac{y_2}{(x_2 - x_1)(x_2 - x_0)}(x - x_0)(x - x_1)$?
- and let's check

Newton polynomial

- เมื่อ $x = x_2$ เราจะพบว่า, $y_0 + \frac{y_1 - y_0}{(x_1 - x_0)}(x - x_0) := Q_1(x)$ จะเกินมา
- $Q_2(x) = y_0 + \frac{y_1 - y_0}{(x_1 - x_0)}(x - x_0) + \frac{y_2 - Q_1(x)}{(x_2 - x_1)(x_2 - x_0)}(x - x_0)(x - x_1)$?
- let's check

Newton polynomial in general

- สมมติว่าเราสามารถหา $Q_{n-1}(x)$ ที่เป็นไปตามเงื่อนไขของการทำ interpolation ได้
- เราสามารถสร้าง $Q_n(x) = Q_{n-1}(x) + c(x - x_0) \cdots (x - x_{n-1})$
- โดยที่ $c = \frac{f(x_n) - Q_{n-1}(x_n)}{\prod_{j=0}^{n-1}(x_n - x_j)}$

Newton polynomial in general

กล่าวโดยสรุปแล้ว Newton polynomial สามารถสร้างได้โดย

$$
\begin{aligned}
Q_n(x) &= a_0 + a_1(x - x_0) + \ldots + a_n(x - x_0) \cdot \ldots \cdot (x - x_n) \\
&= a_0 + \sum_{j=1}^{n} a_j \prod_{k=0}^{j-1} (x - x_k)
\end{aligned}
$$

โดยที่

$$
\begin{aligned}
a_0 &= f(x_0) \\
a_j &= \frac{f(x_j) - Q_{j-1}(x_j)}{\prod_{k=0}^{j-1} (x_j - x_k)}, \quad 1 \leq j \leq n
\end{aligned}
$$

ตัวอย่าง

ให้หา polynomial ที่ interpolate ข้อมูลสองตัว $(x_0, f(x_0)), (x_1, f(x_1))$

$$
\begin{aligned}
Q_1(x) &= a_0 + a_1(x - x_0) \\
&= f(x_0) + \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_0)
\end{aligned}
$$

Lab: Implement Newton polynomial

Divided differences

- การเขียนโปรแกรมเพื่อสร้าง Newton polynomial อาจทำได้ง่ายขึ้นโดยอาศัยการคำนวณ a_j ที่ใช้อัลกอริทึมที่เรียกว่า divided difference
- เราจะเรียก $a_j = \frac{f(x_j) - Q_{j-1}(x_j)}{\prod_{k=0}^{j-1}(x_j - x_k)}$ ว่า j^{th}-order divided difference
- สังเกตว่าการคำนวณค่า a_j ขึ้นอยู่กับจุด $x_0, \ldots x_j$ และค่าของฟังก์ชัน ณ จุดนั้นๆ $f(x_0) \ldots f(x_j)$

Divided differences

- เราจะเขียน $a_j = f[x_0, \ldots, x_j]$ เพื่อเน้นว่าค่าของ a_j ขึ้นอยู่กับจุด $x_0, \ldots x_j$ และค่าของฟังก์ชัน ณ จุดนั้นๆ
- สำหรับ $a_0 = f[x_0] = f(x_0)$
- เมื่อใช้ notation ใหม่ของ divided difference เราสามารถเขียน Newton polynomial ได้ใหม่ว่า

$$Q_n(x) = a_0 + f[x0, x1](x - x_0) + \ldots + f[x_0, \ldots, x_n](x - x_0) \cdot \ldots \cdot (x - x_n)$$

Divided differences

- Divided difference ของพจน์สูงๆ สามารถคำนวณได้โดยง่าย โดยใช้หลักการ recursive (การเรียกซ้ำ)

$$f[x_0, \ldots, x_n] = \frac{f[x_1, \ldots, x_n] - f[x_0, \ldots, x_{n-1}]}{x_n - x_0}$$

Lab: Implement divided difference algorithm

Exercise

- Find the polynomial of degree ≤ 2 that interpolates (-1,9), (0,5), (1,3)

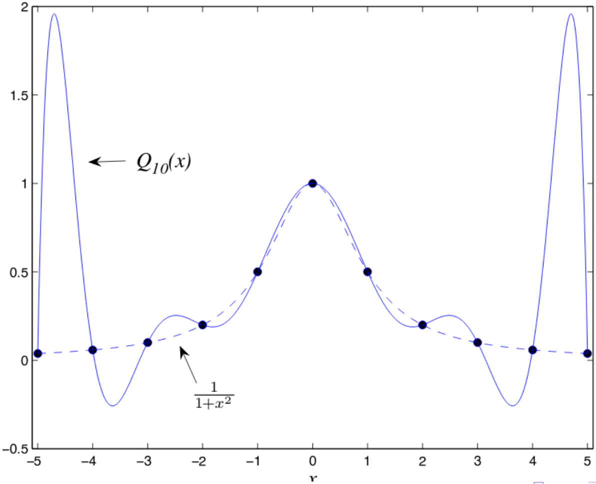
Spline Interpolation

Spline interpolation

- ข้างต้นเราสร้างฟังก์ชันหนุนาม 1 ตัวที่ interpolate ชุดข้อมูลทั้งหมด
- คาบนี้เราจะลองพิจารณาการทำ interpolation ที่แบ่งช่วงของข้อมูลออกเป็นช่วงย่อย แล้วหา polynomial ที่ interpolate ข้อมูลในช่วงย่อยนั้น
- เราเรียกพหุนามที่ interpolate ข้อมูลในช่วงย่อยว่า piece-wise polynomial หรือ spline
- การใช้ spline หลายๆตัวในการ interpolate ชุดข้อมูลจึงเรียกว่า spline interpolation

Why do we need spline interpolation ?

ในกรณีที่มีข้อมูลหลายตัว polynomial interpolation มักให้พหุนามที่ซับซ้อนเกินไป



นิยามของ Spline interpolation

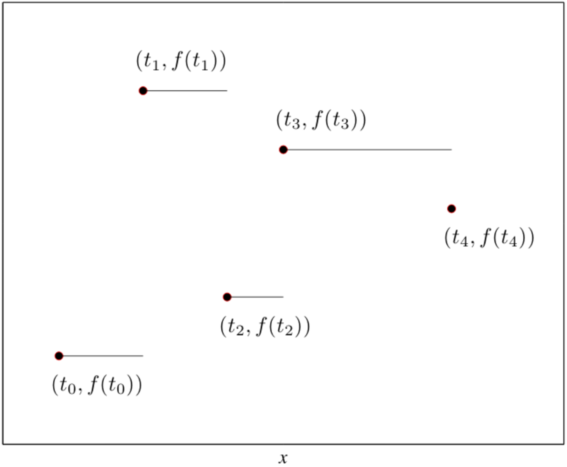
- กำหนดชุดข้อมูล $(x_0, y_0), \ldots, (x_n, y_n)$
- กำหนดปม (knots) t_i จำนวน $n+1$ ตัว โดยที่ $t_0 < t_1 < \cdots < t_n$ และ $t_0 \leq x_0, t_n \geq x_n$
- Spline interpolant, $s(x)$ ดีกรี k บนช่วง $[t_0, t_n]$ คือเซตของฟังก์ชัน $p_i(x)_{i=1}^{n}$ ที่
  1) ดีกรีของพนุหนาม $p_i(x)$ บนช่วงเปิด $[t_{i-1}, t_i)$ มีค่าเท่ากับ k
  2) พหุนาม $s(x)$ มีอนุพันธ์ถึงลำดับที่ $k-1$ บน $[t_0, t_n]$

k=0

a.k.a. piecewise-constant function.

$$s^0(x) = \begin{cases} p_0(x) = a_0 \quad , \quad x \in [t_o, t_1) \\ p_1(x) = a_1 \quad , \quad x \in [t_1, t_2) \\ \quad\quad\vdots \\ p_n(x) = a_n \quad , \quad x \in [t_{n-1}, t_n] \end{cases}$$

k=0: illustration

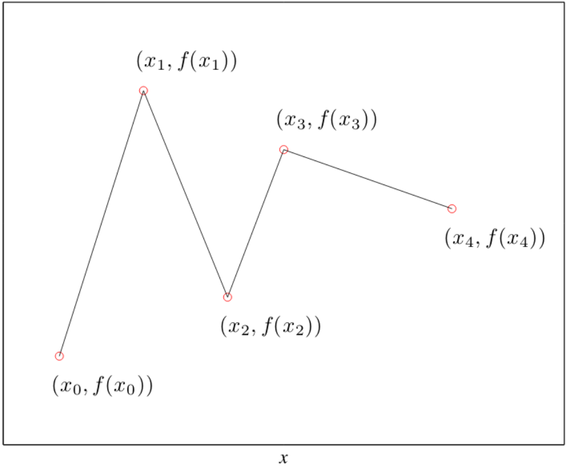


a.k.a. linear spline

$$s^1(x) = \begin{cases} p_0(x) = a_0 x + b_0 \quad , \quad x \in [t_o, t_1) \\ p_1(x) = a_1 x + b_1 \quad , \quad x \in [t_1, t_2) \\ \qquad \vdots \\ p_n(x) = a_n x + b_n \quad , \quad x \in [t_{n-1}, t_n] \end{cases}$$

Spline degree 1: illustration

ตัวอย่างของ Spline degree 1 หรือที่เรียกว่า piecewise-linear function.



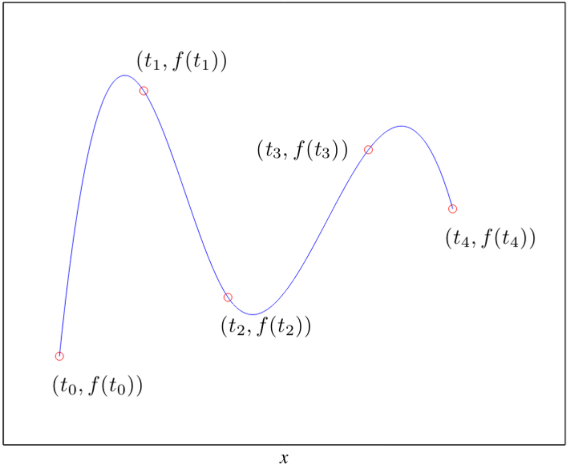
Cubic splines

Spline ที่น่าจะถูกนำมาใช้มากที่สุดอาจเป็น cubic spline ที่เกิดจากการนำพนุนามดีกรี 3 มาเชื่อมต่อกัน (cubic = กำลังสาม)

$$s(x) = \begin{cases} p_0(x) = a_0x^3 + b_0x^2 + c_0x + d_0, & x \in [t_o, t_1) \\ p_1(x) = a_1x^3 + b_1x^2 + c_1x + d_1, & x \in [t_1, t_2) \\ \qquad\qquad \vdots & \\ p_n(x) = a_nx^3 + b_nx^2 + c_nx + d_n, & x \in [t_{n-1}, t_n] \end{cases}$$

ค่า $a_0, b_0, c_0, d_0, \ldots, a_n, b_n, c_n, d_n$ คือสัมประสิทธิ์ที่ต้องตามหา

ตัวอย่างของ Cubic Spline



ข้อกำหนดเพื่อช่วยหาค่าสัมประสิทธิ์ของ cubic spline

- เพื่อให้ง่ายต่อการทำความเข้าใจ กำหนดให้ข้อมูลทำหน้าที่เป็นปมไปเลย $t_i = x_i$

$$p_i(t_i) = p_i(x_i) = y_i, \quad 0 \leq i \leq n$$

- จากข้อกำหนดในการทำ interpolation พบว่า

$$p_{i-1}(t_i) = y_i = p_i(t_i), \quad 1 \leq i \leq n-1$$

- เพื่อให้ $p_i(x)$ เชื่อมต่อกัน ณ จุดปม เราอยากจะให้
  1. $p_i'(t_{i+1}) = p_{i+1}'(t_{i+1}), \quad 0 \leq i \leq n-2$ (ความชันเท่ากันที่จุดปม)
  2. $p_i''(t_{i+1}) = p_{i+1}''(t_{i+1}), \quad 0 \leq i \leq n-2$ (อัตราเปลี่ยนแปลงความชันเท่ากันที่จุดปม)

[ต้องการ] อัตราเปลี่ยนแปลงความชันเท่ากันที่จุดปม

- หาอนุพันธ์ลำดับที่สองของ $p_i(x)$ ณ ทุกๆจุดปม กำหนดให้ผลลัพธ์แทนด้วย $z_i := p_i''(t_i)$

- เนื่องจากอนุพันธ์ลำดับที่สองของ $p_i(x)$ เป็นฟังก์ชันเชิงเส้น เราพบว่าความสัมพันธ์ระหว่าง z_i ของปมสองปมควรจะเป็นเชิงเส้นด้วย

- Cubic spline ใช้ piece-wise Lagrange polynomial เพื่อ interpolate (t_i, z_i) และ (t_{i+1}, z_{i+1})

$$p_i''(x) = \frac{x - t_i}{h_i} z_{i+1} - \frac{x - t_{i+1}}{h_i} z_i$$

โดยที่ $h_i := t_{i+1} - t_i$

Computing the spline

จาก $p_i''(x) = \frac{x-t_i}{h_i} z_{i+1} - \frac{x-t_{i+1}}{h_i} z_i$

- อินทิเกรตเส้นตรงหนึ่งครั้งได้

$$p_i'(x) = \frac{1}{2}(x - t_i)^2 \frac{z_{i+1}}{h_i} - \frac{1}{2}(x - t_{i+1})^2 \frac{z_i}{h_i} + \tilde{c}$$

- อินทิเกรตอีกหนึ่งครั้งจะได้

$$p_i(x) = \frac{z_{i+1}}{6h_i}(x - t_i)^3 + \frac{z_i}{6h_i}(t_{i+1} - x)^3 + C(x - t_i) + D(t_{i+1} - x)$$

- How to find C and D ?

ใช้เงื่อนไขการทำ interpolation เพื่อช่วยหา C และ D

- เงื่อนไขของการทำ interpolation กำหนดไว้ $p_i(t_i) = y_i$ ทำให้เราได้ว่า

$$y_i = \frac{z_i}{6h_i}h_i^3 + Dh_i \text{ นั่นคือ } D = \frac{y_i}{h_i} - \frac{z_i h_i}{6}$$

- เช่นเดียวกัน เงื่อนไข $p_i(t_{i+1}) = y_{i+1}$ ทำให้เราได้ว่า

$$y_{i+1} = \frac{z_{i+1}}{6h_i}h_i^3 + Ch_i \text{ นั่นคือ } C = \frac{y_{i+1}}{h_i} - \frac{z_{i+1} h_i}{6}$$

(Almost) done

เมื่อทราบค่าของ C และ D แล้ว

$$
\begin{aligned}
p_i(x) = {} & \frac{z_{i+1}}{6h_i}(x - t_i)^3 + \frac{z_i}{6h_i}(t_{i+1} - x)^3 \\
& + \left(\frac{y_{i+1}}{h_i} - \frac{z_{i+1}h_i}{6}\right)(x - t_i) \\
& + \left(\frac{y_i}{h_i} - \frac{z_i h_i}{6}\right)(t_{i+1} - x)
\end{aligned}
$$

Note: $p_i(x)$ ข้างต้น interpolate ข้อมูลและ มีอัตราเปลี่ยนแปลงความชันที่จุดปมเท่ากัน ตามต้องการ

[ต้องการ] ความชันเท่ากันที่จุดปม

- บังคับเงื่อนไขที่ความชันของ $p_i(x)$ ต้องเท่ากันที่จุดปม โดยอาศัยการบังคับอนุพันธ์ลำดับที่ 1

  $p'_i(t_i) = p'_{i-1}(t_i)$

- แนวทาง
  - หาอนุพันธ์อันดับ 1 ของ $p_i(x)$ ณ จุด t_i
  - หาอนุพันธ์อันดับ 1 ของ $p_{i-1}(x)$ ณ จุด t_i
  - จับสองสมการมาเท่ากัน เพื่อแก้สมการหาค่าสัมประสิทธิ์

The 1^{st} derivative of $p_i(x)$

$$
\begin{aligned}
p_i'(x) = &\frac{z_{i+1}}{2h_i}(x - t_i)^2 - \frac{z_i}{2h_i}(t_{i+1} - x)^2 \\
&+ \frac{y_{i+1}}{h_i} - \frac{z_{i+1}h_i}{6} \\
&- \frac{y_i}{h_i} + \frac{z_i h_i}{6}
\end{aligned}
$$

The 1^{st} derivative of $p_{i-1}(x)$

$$
\begin{aligned}
p'_{i-1}(x) = \frac{z_i}{2h_{i-1}}(x - t_{i-1})^2 - \frac{z_{i-1}}{2h_{i-1}}(t_i - x)^2 \\
+ \frac{y_i}{h_{i-1}} - \frac{z_i h_{i-1}}{6} \\
- \frac{y_{i-1}}{h_{i-1}} + \frac{z_{i-1} h_{i-1}}{6}
\end{aligned}
$$

อนุพันธ์ที่หาได้

$$
\begin{aligned}
p_i'(t_i) &= -\frac{z_i}{2h_i}h_i^2 + \frac{y_{i+1}}{h_i} - \frac{z_{i+1}h_i}{6} - \frac{y_i}{h_i} + \frac{z_ih_i}{6} \\
&= -\frac{h_i}{3}z_i - \frac{h_i}{6}z_{i+1} - \frac{y_i}{h_i} + \frac{y_{i+1}}{h_i} \qquad (1) \\
p_{i-1}'(t_i) &= \frac{z_i}{2h_{i-1}}h_{i-1}^2 + \frac{y_i}{h_{i-1}} - \frac{z_ih_{i-1}}{6} - \frac{y_{i-1}}{h_{i-1}} + \frac{z_{i-1}h_{i-1}}{6} \\
&= \frac{h_{i-1}}{6}z_{i-1} + \frac{h_{i-1}}{3}z_i - \frac{y_{i-1}}{h_{i-1}} + \frac{y_i}{h_{i-1}} \qquad (2)
\end{aligned}
$$

สืบเนื่องจากเงื่อนไขที่กำหนดว่าความชันของ spline ทั้งสองต้องเท่ากัน ณจุด t_i (แทนค่า t_i ในสมการ (1) และ (2))

$$p_i'(t_i) = p_{i-1}'(t_i)$$
$$-\frac{h_i}{3}z_i - \frac{h_i}{6}z_{i+1} - \frac{y_i}{h_i} + \frac{y_{i+1}}{h_i} = \frac{h_{i-1}}{6}z_{i-1} + \frac{h_{i-1}}{3}z_i - \frac{y_{i-1}}{h_{i-1}} + \frac{y_i}{h_{i-1}}$$

ทำซ้ำกระบวนการข้างต้น สำหรับคู่ spline อื่นๆทุกคู่ เราจะได้ระบบสมการ $n-1$ สมการ

$$-\frac{h_i}{6}z_{i-1} - \frac{h_i + h_{i-1}}{3}z_i + \frac{h_i}{6}z_{i+1} = \frac{1}{h_i}(y_{i+1} - y_i) - \frac{1}{h_{i-1}}(y_i - y_{i-1})$$

โดยที่ $1 \leq i \leq n$ และมี ตัวแปรที่ไม่ทราบค่า (unknown) $n+1$ ตัว ได้แก่ $z_0, \ldots, z_n$

ระบบสมการที่เกิดขึ้น มีตัวแปรไม่ทราบค่ามากกว่าจำนวนสมการ ทำให้เราต้องกำจัดตัวแปรไม่ทราบค่า บางตัวไป โดยทั่วไปแล้ว เรามักให้ $z_0 = z_n = 0$ แล้วเขียนระบบสมการในรูปของเมทริกซ์

$$\begin{bmatrix} \frac{h_0+h_1}{3} & \frac{h_1}{6} & & & \\ \frac{h_1}{6} & \frac{h_1+h_2}{3} & \frac{h_2}{6} & & \\ & \ddots & \ddots & \ddots & \\ & & \frac{h_{n-3}}{6} & \frac{h_{n-3}+h_{n-2}}{3} & \frac{h_{n-2}}{6} \\ & & & \frac{h_{n-2}}{6} & \frac{h_{n-2}+h_{n-1}}{3} \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_{n-2} \\ z_{n-1} \end{bmatrix} = \begin{bmatrix} \frac{y_2-y_1}{h_1} - \frac{y_1-y_0}{h_0} \\ \frac{y_3-y_2}{h_2} - \frac{y_2-y_1}{h_1} \\ \vdots \\ \frac{y_{n-1}-y_{n-2}}{h_{n-2}} - \frac{y_{n-2}-y_{n-3}}{h_{n-3}} \\ \frac{y_n-y_{n-1}}{h_{n-1}} - \frac{y_{n-1}-y_{n-2}}{h_{n-2}} \end{bmatrix}$$

$$H * Z = Y$$

Finally

- เนื่องจาก matrix H เป็น positive definite matrix ($a^T Ha \geq 0$)
- เราสามารถหา inverse ของเมทริกซ์ดังกล่าวได้
- คำตอบของระบบสมการดังกล่าวก็สามารถหาได้โดย $Z = H^{-1}Y$
- เมื่อเราทราบ Z เราสามารถกลับไปหา $p_i(x)$ แต่ละตัวได้

แบบฝึกหัด

- Implement Spline Interpolation.
- Derive the system of equations for the case where $h_i = h, \forall i$.

References

- Introduction to numerical analysis by Doron Levy (บทที่ 3)