Analiza numeryczna (L)
18.11.2020 r.

Sprawdzian

- niesobowiązkowy!
- 27.11.2020, godz. $13^{00}-15^{15}$
  - $\to 13^{00}$ – ogłoszenie zadań w SKOSie
  - $\to 13^{00}-15^{15}$ – nadsyłanie rozwiązań w systemie SKOS (pdf, jpeg)
- 6 zadań, każde za 5 pkt.
- wysłanie rozwiązania $\equiv$ deklaracja o samodzielnej pracy
- rozwiązania (pdf, jpeg): – każde na osobnej kartce A4 (zalecane użycie udostępnionego szablonu)
  – na górze kartki: imię nazwisko, numer indeksu, numer zadania,
  a potem rozwiązanie

$\Rightarrow$ **Wątpliwości:**
- komisja antyplagiatowa
- komisja dyscyplinarna
- prodziekan ds. dydaktyki informatyki

Powtórka

- postać potęgowa $\to$ schemat Hornera
- postać Newtona $\to$ uogólniony schemat Hornera
- postać Czebyszewa $\to$ algorytm Clenshawa

$$\begin{cases} T_0(x) \equiv 1, \quad T_1(x) \equiv x \\ T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x) \quad (k \geq 2) \end{cases}$$

$\hookrightarrow$ uwaga!

- zadanie interpolacyjne Lagrange'a

DANE $\begin{cases} x_0, x_1, \ldots, x_n \text{ – parami różne} \\ y_0, y_1, \ldots, y_n \end{cases}$

$\hookrightarrow L_n \in \Pi_n : \; L_n(x_k) = y_k \; (0 \leq k \leq n)$

postać Lagrange'a

$$\hookrightarrow L_n(x) = \sum_{k=0}^{n} \lambda_k(x) y_k$$

$$\lambda_k(x) := \prod_{\substack{i=0 \\ i \neq k}}^{n} \frac{x - x_i}{x_k - x_i} \in \Pi_n \setminus \Pi_{n-1}$$

Wykład 6. Interpolacja wielomianowa. Część II (postać Newtona, ilorazy różnicowe, błąd interpolacji, „dobry" wybór węzłów)

Niech dane będą parami różne węzły $x_0, x_1, \ldots, x_n$ oraz odpowiadające im wartości $y_0, y_1, \ldots, y_n$. Interesuje nas wielomian L_n stopnia co najwyżej $n \in \mathbb{N}$ spełniający następujące warunki:

$$L_n(x_k) = y_k \quad (0 \leq k \leq n).$$

Pomysł

Znajdźmy postać Newtona wielomianu interpolacyjnego, tzn. takie współczynniki $b_0, b_1, \ldots, b_n$,

$$L_n(x) = b_0 + b_1 \overbrace{(x-x_0)}^{p_1(x)} + b_2 \overbrace{(x-x_0)(x-x_1)}^{p_2(x)} + b_3 \underbrace{(x-x_0)(x-x_1)(x-x_2)}_{p_3(x)} + \ldots + b_n \underbrace{(x-x_0)(x-x_1)\ldots(x-x_{n-1})}_{p_n(x)},$$

aby spełnione były warunki interpolacji.

Pytanie

Jakimi wzorami wyrażają się współczynniki $b_0, b_1, \ldots, b_n$?

Aby je znaleźć rozwiązujemy następujący „trójkątny" układ równań liniowych:

$$\begin{cases} y_0 = L_n(x_0) = b_0 \\ y_1 = L_n(x_1) = b_0 + b_1(x_1 - x_0) \\ y_2 = L_n(x_2) = b_0 + b_1(x_2-x_0) + b_2(x_2-x_0)(x_2-x_1) \\ \vdots \\ y_k = L_n(x_k) = b_0 + b_1 p_1(x_k) + b_2 p_2(x_k) + b_3 p_3(x_k) + \ldots + b_k p_k(x_k) \\ \vdots \\ y_n = L_n(x_n) = b_0 + b_1 p_1(x_n) + b_2 p_2(x_n) + \ldots + b_n p_n(x_n) \end{cases}$$

$\to b_0 = y_0$

$\to b_1 = \frac{y_1 - y_0}{x_1 - x_0}$

$\vdots$

$\to b_k = \boxed{\text{ilorazy różnic}}$

Można łatwo sprawdzić, że

$$(*) \quad \boxed{b_k = \sum_{i=0}^{k} \frac{y_i}{\prod_{\substack{j=0 \\ j \neq i}}^{k} (x_i - x_j)}}$$

$\leftarrow$ wzór na współczynniki postaci Newtona wielomianu interpolacyjnego

Fakt

Nie zaleca się używania wprost wzoru $(*)$. Lepiej użyć pewnej zależności rekurencyjnej, o której powiemy za chwilę.

Definicja (ilorazy różnicowe)

Niech dane będą parami różne węzły $x_0, x_1, x_2, \ldots$ oraz funkcja f zdefiniowana w tych węzłach. Niech będzie

$$f(x_k) = y_k \quad (k = 0, 1, \ldots)$$

Ilorazy różnicowe funkcji f w węzłach $x_i, x_{i+1}, \ldots, x_j$ oznaczamy symbolem $f[x_i, x_{i+1}, \ldots, x_j]$. Ilorazy różnicowe definiujemy w następujący sposób rekurencyjny:

$$\begin{cases} f[x_i] := f(x_i) = y_i \\ f[x_i, x_{i+1}, \ldots, x_{j-1}, x_j] := \dfrac{f[x_{i+1}, \ldots, x_j] - f[x_i, \ldots, x_{j-1}]}{x_j - x_i} \end{cases}$$

Przykład

$$f[x_0, x_1] = \frac{\overset{=f(x_1)=y_1}{f[x_1]} - \overset{=f(x_0)=y_0}{f[x_0]}}{x_1 - x_0} = b_0 \qquad f[x_1, x_2] = \frac{\overset{=y_2}{f(x_2)} - \overset{=y_1}{f(x_1)}}{x_2 - x_1}$$

$$f[x_0, x_1, x_2] = \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0} = b_1$$

Twierdzenie

$$b_k = f[x_0, x_1, \ldots, x_k] \qquad (k = 0, 1, \ldots, n)$$

Wniosek

Postać Newtona wielomianu interpolacyjnego wyraża się wzorem

$$L_n(x) = \sum_{k=0}^{n} f[x_0, x_1, \ldots, x_k]\, p_k(x),$$

gdzie $p_0(x) \equiv 1, \quad p_k(x) = \prod_{i=0}^{k-1} (x - x_i) \quad (i \geq 1)$

$f(x_i) = y_i$

$$\begin{cases} L_n(x_k) = y_k \equiv f(x_k) \quad (k = 0, 1, \ldots, n) \\ L_n \in \Pi_n \end{cases}$$

Tablica ilorazów różnicowych

$$\begin{array}{lllllll}
 & x_0 & f(x_0) = y_0 & & & & \\
 & x_1 & f(x_1) = y_1 \to & f[x_0, x_1] & & & \\
 & x_2 & f(x_2) = y_2 \to & f[x_1, x_2] \to & f[x_0, x_1, x_2] & & \\
O(n^2) & x_3 & f(x_3) = y_3 \to & f[x_2, x_3] \to & f[x_1, x_2, x_3] \to & f[x_0, x_1, x_2, x_3] & \\
 & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\
 & x_n & f(x_n) = y_n \to & f[x_{n-1}, x_n] \to & f[x_{n-2}, x_{n-1}, x_n] \to & f[x_{n-3}, \ldots, x_n] \to \ldots & f[x_0, x_1, \ldots, x_n] \\
O(n) \to & x_{n+1} & f(x_{n+1}) = y_{n+1} \to & f[x_n, x_{n+1}] \to & f[x_{n-1}, x_n, x_{n+1}] \to & f[x_{n-2}, \ldots, x_{n+1}] \to \ldots & f[x_0, \ldots, x_{n+1}]
\end{array}$$

Fakt

$$\boxed{L_{n+1}(x) = L_n(x) + b_{n+1}\, p_{n+1}(x)}$$

L_{n+1}: wielomian interpolacyjny dla danych

x_0	x_1	...	x_n	x_{n+1}
y_0	y_1	...	y_n	y_{n+1}

x_{n+1}, y_{n+1}: nowa „obserwacja"

L_n: wielomian interpolacyjny dla danych

x_0	x_1	...	x_n
$y_0 = f(x_0)$	$y_1 = f(x_1)$	...	$y_n = f(x_n)$

Zadanie domowe

W jaki sposób dodać w czasie liniowym kolejną obserwację pamiętając jedynie przekątną tablicy ilorazów różnicowych?

Zalety postaci Newtona wielomianu interpolacyjnego

1° łatwo jest dodawać kolejne obserwacje (nie jest możliwe przy korzystaniu z postaci Lagrange'a)

2° Załóżmy, że ilorazy różnicowe są już znane $(O(n^2))$, wartość $L_n(x)$ dla danego x obliczamy w czasie $O(n)$ wykorzystując uogólniony schemat Hornera

$L_n(z_0), L_n(z_1), \ldots, L_n(z_M)$

$\Downarrow$

koszt: $O(n^2 + M \cdot n)$

Ciekawostki

1° Ze względów numerycznych przed utworzeniem tablicy ilorazów różnicowych zaleca się posortowanie węzłów

2° Podany algorytm wypełniania tablicy ilorazów różnicowych nie jest algorytmem numerycznie poprawnym.

3° Algorytm wypełniania tablicy ilorazów różnicowych szybko zawodzi, gdy węzłów jest więcej niż kilkadziesiąt ☹

Błąd interpolacji i problem doboru węzłów

Twierdzenie

Załóżmy, że funkcja f ma ciągłą $(n+1)$-szą pochodną. Niech L_n oznacza wielomian interpolacyjny funkcji f w parami różnych węzłach $x_0, x_1, \ldots, x_n$. Wtedy

$$\boxed{f(x) - L_n(x) = \frac{f^{(n+1)}(\eta_x)}{(n+1)!} \overbrace{(x-x_0)(x-x_1)\ldots(x-x_n)}^{p_{n+1}(x)}},$$

gdzie $\eta_x \in (a, b) \supset \{x_0, x_1, \ldots, x_n\}$.

Wniosek

$$\max_{a \leq x \leq b} |f(x) - L_n(x)| \leq \max_{a \leq x \leq b} \frac{|f^{(n+1)}(x)|}{(n+1)!} \cdot \max_{a \leq x \leq b} |(x-x_0)(x-x_1)\ldots(x-x_n)|$$

Na to nie mamy wpływu ☹

To możemy kontrolować! Jak? Dobierając węzły w taki sposób, aby zminimalizować maksimum ☺

Zadanie

$$(\Delta) \quad \min_{x_0, x_1, \ldots, x_n \in (a,b)} \; \max_{a \leq x \leq b} |\underbrace{(x-x_0)(x-x_1)\ldots(x-x_n)}_{1 \cdot x^{n+1} + \ldots}|$$

$\leftarrow$ wielomian stopnia dokładnie ~~n-tego~~ $(n+1)$-szego ze współczynnikiem wiodącym równym 1.

Fakt

Niech będzie $[a, b] = [-1, 1]$. Wtedy wyrażenie (Δ) przyjmuje najmniejszą wartość wtedy i tylko wtedy, gdy za węzły przyjmiemy miejsca zerowe $(n+1)$-szego wielomianu Czebyszewa, tj.

$$x_k = \cos\left(\frac{2k+1}{2n+2}\pi\right) \quad (k = 0, 1, \ldots, n),$$

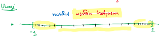
$\leftarrow$ węzły Czebyszewa w przedziale $[-1, 1]$

Mamy

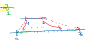
$$\max_{-1 \leq x \leq 1} |\underbrace{(x-x_0)(x-x_1)\ldots(x-x_n)}_{\frac{1}{2^n} T_{n+1}(x)}| = \frac{1}{2^n}$$

Uwagi

rozkład węzłów Czebyszewa



- Jak dobrać węzły w $[a, b] \neq [-1, 1]$?



Przykłady

Zobaczymy, że w ćwiczeniu taki specjalny dobór węzłów może znacznie polepszyć jakość interpolacji

a) $f(x) = \frac{1}{25x^2+1}$ $\leftarrow$ funkcja Rungego

rozważymy węzły Czebyszewa i równoodległe w przedziale $[-1, 1]$

b) podobnie postąpimy w wypadku funkcji $\boxed{f(x) = |x|}$.

Efekt Rungego!