Analiza numeryczna

1. Analiza błędów

Rafał Nowak

Błędy

Niech $\tilde{x}$ będzie przybliżoną wartością wielkości x.

- błąd bezwzględny
$$\Delta x := |\tilde{x} - x|;$$
- błąd względny
$$\delta x := |\tilde{x} - x|/|x| \qquad (x \neq 0).$$

Symbol $|\cdot|$ może oznaczać dowolną normę, tzn. $|x - y|$ jest odległością x od y.

$$x := (e_n \dots e_1 e_0 . e_{-1} e_{-2} \dots)_B = \pm \left(\sum_{i=0}^{n} e_i B^i + \sum_{j=1}^{\infty} e_{-j} B^{-j} \right).$$

- $B \geqslant 2$ - liczba całkowita - podstawa systemu; najczęściej $B = 2, 10$.
- $0 \leqslant e_i \leqslant B - 1$ - liczby całkowite - cyfry liczby x

Cyfry dokładne vs cyfry znaczące

Niech $B = 10$ ($B = 2$) oraz niech $\tilde{a}$ będzie przybliżoną wartością wielkości a.

- jeśli $|a - \tilde{a}| \leqslant \frac{1}{2} \cdot B^{-p}$ to $\tilde{a}$ ma p **dokładnych cyfr** dziesiętnych (dwójkowych) ułamkowych.
- ponadto, jeśli w reprezentacji liczby $\tilde{a}$ jest $e_n = e_{n-1} = \ldots = e_{q+1} = 0$, $e_q \neq 0$ to cyfry $e_q, e_{q-1}, \ldots, e_p$ nazywamy dziesiętnymi (dwójkowymi) cyframi znaczącymi liczby $\tilde{a}$.

Cyfry dokładne vs cyfry znaczące

Niech $B = 10$ ($B = 2$) oraz niech $\tilde{a}$ będzie przybliżoną wartością wielkości a.

- jeśli $|a - \tilde{a}| \leqslant \frac{1}{2} \cdot B^{-p}$ to $\tilde{a}$ ma p **dokładnych cyfr** dziesiętnych (dwójkowych) ułamkowych.
- ponadto, jeśli w reprezentacji liczby $\tilde{a}$ jest $e_n = e_{n-1} = \ldots = e_{q+1} = 0$, $e_q \neq 0$ to cyfry $e_q, e_{q-1}, \ldots, e_p$ nazywamy dziesiętnymi (dwójkowymi) cyframi znaczącymi liczby $\tilde{a}$.
  Przykład: niech będzie $a = 0.00045675$; liczba $\tilde{a} = 0.00045679$ ma 7 dokładnych cyfr ułamkowych oraz cztery cyfry znaczące: 4, 5, 6, 7.
- Przykład: liczba 0.001234 ± 0.000004 ma pięć cyfr dokładnych, z czego trzy są znaczące.
- Przykład: liczba 0.001234 ± 0.000006 ma cztery cyfry dokładne i tylko dwie cyfry znaczące.

- znormalizowana zmiennopozycyjna postać

$$x = s\, m\, B^c,$$

- $s = \operatorname{sgn} x$ — znak liczby x
- $1 \leqslant m < B$ — mantysa
- c - liczba całkowita — cecha

Reprezentacja dwójkowa

- $B = 2$, $x = s\, m\, 2^c$
- $m = (1.e_{-1}e_{-2}\ldots)_2 = 1 + \sum_{i=1}^{\infty} e_{-i}2^{-i} \in [1, 2)$
- $d + 1$ — długość słowa (32 = float, 64 = double w języku C)
- $t \in \mathbb{N}$ — liczba bitów na mantysę
- $m_t = (1.e^*_{-1}e^*_{-2}\ldots e^*_{-t})_2$, — zaokrąglenie mantysy

Definicja (Reguła zaokrąglenia)

Zaokrąglenie liczby x

$$\mathrm{rd}\,(x) := s\,\bar{m}\,2^c, \tag{1}$$

gdzie

$$\bar{m} = (1.e_{-1}e_{-2}\ldots e_{-t})_2 + (0.\underbrace{00\cdots0}_{t-1\ \text{razy}}\, e_{-t-1})_2$$

Twierdzenie

Liczbę $\mathrm{rd}\,(x)$ *można zapisać w postaci*

$$\mathrm{rd}\,(x) = s\, m_t\, 2^{c_t}, \tag{2}$$

gdzie mantysa $m_t = 1.e^*_{-1}e^*_{-2}\ldots e^*_{-t}$ *i cecha* $c_t \in \mathbb{Z}$ *są dane wzorami*

$$m_t := 1.0, \quad c_t := c + 1$$

jeśli

$$e_{-k} = 1 \quad \textit{dla} \quad k = 1, 2, \ldots, t+1,$$

lub wzorami

$$m_t := \bar{m}, \quad c_t := c$$

w przeciwnym wypadku.

Precyzja arytmetyki

Twierdzenie

Błąd bezwzględny zaokrąglenia spełnia nierówność

$$| \operatorname{rd}(x) - x | \leqslant 2^{-t-1} \cdot 2^{c}.$$

Twierdzenie

Błąd względny zaokrąglenia spełnia nierówność

$$\left| \frac{\operatorname{rd}(x) - x}{x} \right| \leqslant \frac{1}{2} 2^{-t}.$$

Definicja

Precyzją arytmetyki danego komputera nazywamy liczbę

$$\mathrm{u} := \frac{1}{2} 2^{-t}.$$

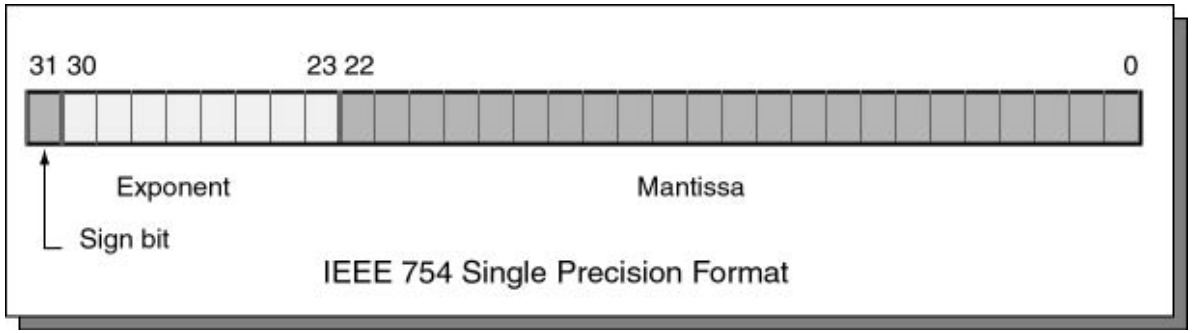


Tabela: Formaty liczb zmiennopozycyjnych (IEEE 754)

		single	double
$d+1$	długość słowa (w bitach)	32	64
t	długość mantysy (w bitach)	23	52
$d-t$	długość cechy (w bitach)	8	11
$c_{\max}$	największa cecha	127	1023
$c_{\min}$	najmniejsza cecha	-126	-1022
	największa liczba dod.	$3.4 \cdot 10^{38}$	$1.8 \cdot 10^{308}$
	najmniejsza liczba dod.	$1.2 \cdot 10^{-38}$	$2.2 \cdot 10^{-308}$
	najmn. dod. liczba subnorm.	$1.4 \cdot 10^{-45}$	$4.9 \cdot 10^{-324}$
u	precyzja arytmetyki	$5.96 \cdot 10^{-8}$	$1.11 \cdot 10^{-16}$

Zbiór reprezentacji arytmetyki zmiennopozycyjnej

$$X_{fl} := \mathrm{rd}\,(X) = \{\mathrm{rd}\,(x) : x \in X\}$$

Założenie (**Model standardowy arytmetyki**)

Niech będzie $a,\ b \in X_{fl}$, $\diamond \in \{+, -, \times, /\}$, $a \diamond b \in X'$,
$\mathrm{fl}(a \diamond b) := \mathrm{rd}\,(a \diamond b)$ — **obliczony** wynik spełnia

$$fl(a \diamond b) = (a \diamond b)(1 + \varepsilon_\diamond), \tag{3}$$

gdzie $\varepsilon_\diamond = \varepsilon_\diamond(a, b)$, $|\varepsilon_\diamond| \leqslant \mathrm{u}$.

Twierdzenie

Jeśli $|\alpha_j| \leqslant u$ *i* $\rho_j = \pm 1$ *dla* $j = 1, 2, \ldots, n$ *oraz* $nu < 1$, *to zachodzi równość*

$$\prod_{j=1}^{n} (1 + \alpha_j)^{\rho_j} = 1 + \theta_n, \tag{4}$$

gdzie θ_n *jest wielkością spełniającą nierówność*

$$|\theta_n| \leqslant \gamma_n,$$

gdzie z kolei

$$\gamma_n := \frac{nu}{1 - nu} \approx nu. \tag{5}$$

Twierdzenie

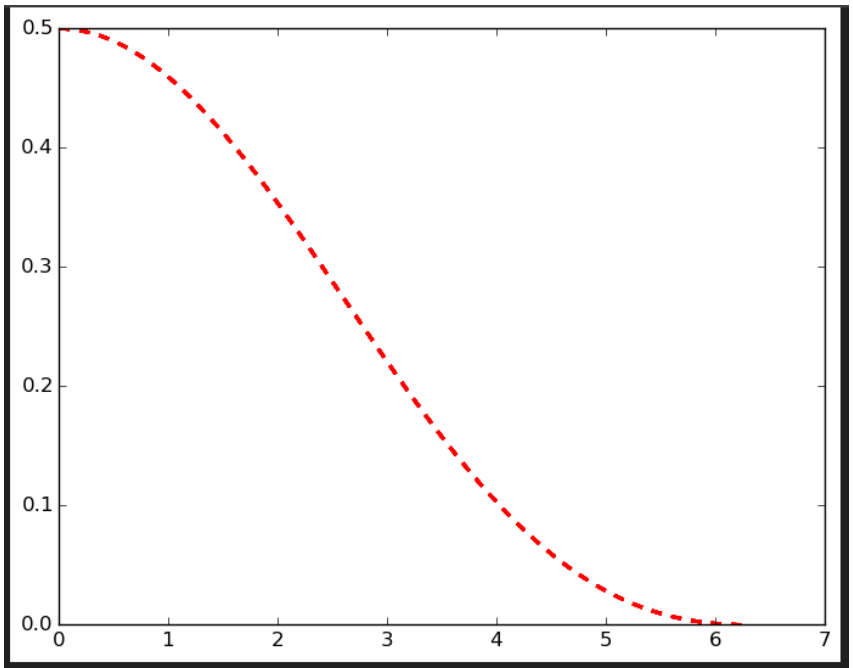
Jeśli $|\alpha_j| \leqslant u$ *dla* $j = 1, 2, \ldots, n$ *oraz* $nu < 0.01$, *to zachodzi równość*

$$\prod_{j=1}^{n}(1 + \alpha_j) = 1 + \eta_n, \tag{6}$$

gdzie $|\eta_n| \leqslant 1.01nu$.

Utrata cyfr znaczących

Utrata cyfr znaczących występuje wtedy, gdy odejmujemy dwie prawie równe liczby.
Przykład: $f(x) = (1 - \cos(x))/x^2$




```
f(4.00e-08) = 4.86e-01
f(3.00e-08) = 4.93e-01
f(2.00e-08) = 5.55e-01
f(1.50e-08) = 4.93e-01
f(1.30e-08) = 6.57e-01
f(1.15e-08) = 8.39e-01
f(1.10e-08) = 9.18e-01
f(1.05e-08) = 0.00e+00
f(1.01e-08) = 0.00e+00
f(1.00e-08) = 0.00e+00
```

Uwarunkowanie zadania

Definicja

Jeśli niewielkie względne zmiany danych zadania powodują duże względne zmiany jego rozwiązania, to zadanie takie nazywamy **źle uwarunkowanym**. Wielkości charakteryzujące wpływ zaburzeń danych na odkształcenia rozwiązania nazywamy **wskaźnikami uwarunkowania** zadania.

Przykład

Zadanie: obliczyć wartość funkcji f w punkcie $x \in \mathbb{R}$.

$$\frac{|f(x+h)-f(x)|}{|f(x)|} \approx \frac{|hf'(x)|}{|f(x)|} = \frac{|xf'(x)|}{|f(x)|}\frac{|h|}{|x|} = C_f(x)\cdot\frac{|h|}{|x|}.$$

Czynnik $C_f(x) = |xf'(x)|/|f(x)|$ można traktować jako *wskaźnik uwarunkowania* zadania.

Algorytmy numerycznie poprawne

Problem: jak dokładny może być dla wybranego zadania wynik obliczony w arytmetyce zmiennopozycyjnej?

Definicja

Algorytmem *numerycznie poprawnym* nazywamy taki algorytm, dla którego **obliczone rozwiązanie jest mało zaburzonym rozwiązaniem dokładnym dla mało zaburzonych danych.** Przez „małe zaburzenia" rozumiemy tu zaburzenia na poziomie błędu reprezentacji.

Analiza numeryczna

2. Równania nieliniowe

Rafał Nowak

Równania nieliniowe

Zajmiemy się numerycznym rozwiązaniem równania

$$f(x) = 0, \tag{1}$$

gdzie f jest znaną funkcją rzeczywistą.

$$x - \operatorname{tg} x = 0 \qquad \text{(równanie dyfrakcji światła)},$$
$$x - a \sin x = b \qquad \text{(równanie Keplera)},$$
$$x - e^{-\frac{1}{2}x} = 0,$$
$$2^{x^2} - 10x + 1 = 0,$$
$$\cosh\left(\sqrt{x^2+1} - e^x\right) + \log|\sin x| = 0,$$
$$3.24x^8 - 2.42x^7 + 10.34x^6 + 11.01x^2 + 47.98 = 0.$$

Niemal nigdy nie ma mowy o podaniu wzoru na rozwiązanie dokładne. Dotyczy to również ostatniego przykładu, w którym chodzi o wyznaczenie **zer wielomianu**.

Definicja (Zero wielokrotne)

Jeśli f można w otoczeniu punktu α przedstawić w postaci

$$f(x) = (x - \alpha)^m g(x),$$

gdzie g jest funkcją ciągłą i taką, że $g(\alpha) \neq 0$, to α nazywamy *zerem krotności* m. Jeśli $m = 1$, to α jest *zerem pojedynczym*.

Twierdzenie

Jeśli funkcja f jest m-krotnie różniczkowalna w otoczeniu punktu α oraz jeśli

$$f(\alpha) = f'(\alpha) = \ldots = f^{(m-1)}(\alpha) = 0, \quad f^{(m)}(\alpha) \neq 0,$$

to α jest m-krotnym zerem funkcji f.

Twierdzenie

Jeśli funkcja f jest m-krotnie różniczkowalna w otoczeniu punktu α oraz jeśli

$$f(\alpha) = f'(\alpha) = \ldots = f^{(m-1)}(\alpha) = 0, \quad f^{(m)}(\alpha) \neq 0,$$

to α jest m-krotnym zerem funkcji f.

W szczególności, $\alpha \in (a, b)$ jest **pojedynczym zerem** funkcji $f \in C^1[a, b]$, jeśli

$$f(\alpha) = 0 \quad \text{oraz} \quad f'(\alpha) \neq 0.$$

Krzywa $y = f(x)$ przecina oś x-ów pod niezerowym kątem.
Ogólniej, w wypadku zera o **nieparzystej krotności** funkcja zmienia znak w punkcie α; jeśli m jest **parzyste**, to f nie zmienia znaku w pewnym otoczeniu punktu α – oś x-ów jest styczna do wykresu funkcji f, ponieważ $f'(\alpha) = 0$.

Metoda bisekcji

Twierdzenie (Darboux)

Jeśli $f \in C[a,b]$, *a* K *jest dowolną liczbą leżącą pomiędzy* $f(a)$ *i* $f(b)$, *to istnieje taki punkt* $c \in [a, b]$, *że* $f(c) = K$.

Algorytm

Wychodząc od $[a_0, b_0] := [a, b]$ *budujemy zstępujący ciąg przedziałów*

$$[a_0, b_0] \supset [a_1, b_1] \supset [a_2, b_2] \supset \ldots,$$

taki że $\alpha \in [a_k, b_k]$ *dla* $k = 0, 1, \ldots$.
Dla $k = 0, 1, \ldots$

- *obliczamy* $m_k := \frac{1}{2}(a_k + b_k)$;
- *jeśli* $f(a_k)f(m_k) \leqslant 0$, *to* $[a_{k+1}, b_{k+1}] := [m_k, b_k]$,
  w przeciwnym razie $[a_{k+1}, b_{k+1}] := [a_k, m_k]$.

Zbieżność metody bisekcji

Zauważmy, że

- tylko przy założeniu ciągłości funkcji f (różniczkowalność nie jest wymagana) ciąg $\{m_k\}$ jest zbieżny do α ;
- przedział $[a_k, b_k]$ ma długość $b_k - a_k = (b_0 - a_0)/2^k$;
- zbieżność metody jest bardzo wolna (jedna cyfra dwójkowa na jeden krok) i nie zależy od f (!):

$$|m_k - \alpha| = (b_0 - a_0)/2^{k+1} \quad (k \geqslant 1);$$

zatem $|m_k - \alpha| < \varepsilon$ z pewnością zachodzi w następujących warunkach:

$k+1$	10	20	40	80
$\frac{\varepsilon}{b_0-a_0}$	10^{-3}	10^{-6}	10^{-12}	10^{-24}

Tak więc liczba kroków musi zostać podwojona, jeśli chcemy podwojenia liczby dokładnych cyfr dziesiętnych.

Przykład

Dla $f(x) = x^2/4 - \sin x$ i $I_0 = [1.8,\, 2]$ otrzymujemy wyniki podane w tabelce.

k	a_k	b_k	m_k	$f(m_k)$
0	1.8	2	**1.9**	< 0
1	1.9	2	**1.9**5	> 0
2	1.9	1.95	**1.9**25	< 0
3	1.925	1.95	**1.93**75	> 0
4	1.925	1.9375	**1.93**125	< 0
5	1.93125	1.9375	**1.93**4375	> 0

Dla porównania: $\alpha = \mathbf{1.933753\,762827}\ldots$, więc $|\alpha - m_5| = 6 \cdot 10^{-4}$.

Metoda Newtona

$$x_{n+1} = x_n + h_n, \qquad h_n := -\frac{f(x_n)}{f'(x_n)} \qquad (n = 0, 1, \ldots). \qquad (2)$$

Metoda Newtona

$$x_{n+1} = x_n + h_n, \qquad h_n := -\frac{f(x_n)}{f'(x_n)} \qquad (n = 0, 1, \ldots). \qquad (2)$$

Przykład

Dla $f(x) = \sin x - x^2/4$, $x_0 = 1.8$ i $\epsilon = 5 \cdot 10^{-9}$ otrzymujemy:

n	x_n	$f(x_n)$	$f'(x_n)$	h_n
0	**1**.8	−0.163847 630878	1.127202 094693	+0.145357
1	**1.9**45357 812631	0.015436 106659	1.338543 359427	−0.011532
2	**1.933**825 794225	0.000095 223283	1.322020 778469	−0.000072
3	**1.933753 76**5643	0.000000 003722	1.321917 429113	−0.000000 002816
4	**1.933753 762827021257**			

Zauważmy, że liczba cyfr dokładnych (wytłuszczone w drugiej kolumnie) podwaja się w każdym kroku iteracyjnym. Pomimo kiepskiego przybliżenia początkowego już x_4 ma 18 cyfr dokładnych!

Metoda Newtona

$$x_{n+1} = x_n + h_n, \qquad h_n := -\frac{f(x_n)}{f'(x_n)} \qquad (n = 0, 1, \ldots). \tag{3}$$

Metoda Newtona

$$x_{n+1} = x_n + h_n, \qquad h_n := -\frac{f(x_n)}{f'(x_n)} \qquad (n = 0, 1, \ldots). \tag{3}$$

$$e_n := x_n - \alpha$$

Metoda Newtona

$$x_{n+1} = x_n + h_n, \qquad h_n := -\frac{f(x_n)}{f'(x_n)} \qquad (n = 0, 1, \ldots). \tag{3}$$

$$e_n := x_n - \alpha$$

$$e_{n+1} = \frac{1}{2} F''(\eta_n)\, e_n^2, \quad F(x) := x - \frac{f(x)}{f'(x)}, \quad \eta_n \in \mathsf{interv}(x_n, \alpha)$$

Metoda Newtona

$$x_{n+1} = x_n + h_n, \qquad h_n := -\frac{f(x_n)}{f'(x_n)} \qquad (n = 0, 1, \ldots). \tag{3}$$

$$e_n := x_n - \alpha$$

$$e_{n+1} = \frac{1}{2} F''(\eta_n)\, e_n^2, \quad F(x) := x - \frac{f(x)}{f'(x)}, \quad \eta_n \in \mathsf{interv}(x_n, \alpha)$$

Twierdzenie

Jeśli przybliżenie x_0 jest dostatecznie bliskie pojedynczego zera α równania $f(x) = 0$ to metoda Newtona jest zbieżna kwadratowo do α.

Twierdzenie

Załóżmy, że $f'(x) > 0$ *i* $f''(x) > 0$ *dla* $x \in \mathbb{R}$. *Niech* α *będzie pierwiastkiem równania* $f(x) = 0$. *Wówczas jest to jedyny pierwiastek, a metoda Newtona daje ciąg do niego zbieżny dla dowolnego przybliżenia początkowego* x_0.

Twierdzenie

Załóżmy, że $f'(x) > 0$ *i* $f''(x) > 0$ *dla* $x \in \mathbb{R}$. *Niech* α *będzie pierwiastkiem równania* $f(x) = 0$. *Wówczas jest to jedyny pierwiastek, a metoda Newtona daje ciąg do niego zbieżny dla dowolnego przybliżenia początkowego* x_0.

Twierdzenie

Załóżmy, że $f \in C^2[a, b]$, $f'(x)f''(x) \neq 0$ *dla dowolnego* $x \in [a, b]$ *i że* $f(a)f(b) < 0$. *Jeśli*

$$\left|\frac{f(a)}{f'(a)}\right| < b - a, \qquad \left|\frac{f(b)}{f'(b)}\right| < b - a,$$

to metoda Newtona jest zbieżna dla dowolnego $x_0 \in [a, b]$.

Twierdzenie

Załóżmy, że $f \in C^2[a, b]$, $f'(x)f''(x) \neq 0$ *dla dowolnego* $x \in [a, b]$ *i że* $f(a)f(b) < 0$. *Jeśli* $f(x_0)f''(x_0) > 0$ *dla* $x_0 \in [a, b]$, *to ciąg* $\{x_0, x_1, \ldots\}$, *otrzymany metodą Newtona, jest zbieżny monotonicznie do pierwiastka* $\alpha \in (a, b)$.

Definicja

Niech ciąg a_k będzie zbieżny do g. Jeśli istnieją takie liczby rzeczywiste p i C $(C > 0)$, że

$$\lim_{n \to \infty} \frac{|a_{n+1} - g|}{|a_n - g|^p} = C,$$

to p nazywamy **wykładnikiem zbieżności ciągu**, a C **– stałą asymptotyczną błędu**.

Definicja

Niech ciąg a_k będzie zbieżny do g. Jeśli istnieją takie liczby rzeczywiste p i C $(C > 0)$, że

$$\lim_{n \to \infty} \frac{|a_{n+1} - g|}{|a_n - g|^p} = C,$$

to p nazywamy **wykładnikiem zbieżności ciągu**, a C **– stałą asymptotyczną błędu**. Dla $p = 1$ oraz $0 < C < 1$ zbieżność jest **liniowa**, dla $p = 2$ **– kwadratowa**, dla $p = 3$ **– sześcienna**.

Definicja

Niech ciąg a_k będzie zbieżny do g. Jeśli istnieją takie liczby rzeczywiste p i C $(C > 0)$, że

$$\lim_{n\to\infty} \frac{|a_{n+1} - g|}{|a_n - g|^p} = C,$$

to p nazywamy **wykładnikiem zbieżności ciągu**, a C **– stałą asymptotyczną błędu**. Dla $p = 1$ oraz $0 < C < 1$ zbieżność jest **liniowa**, dla $p = 2$ **– kwadratowa**, dla $p = 3$ **– sześcienna**.

Ta definicja nie obejmuje pewnych typów zbieżności. Ciąg może być zbieżny wolniej niż liniowo, co odpowiada warościom $p = 1$ i $C = 1$; mówimy wówczas o zbieżności **podliniowej**. Jeśli $p = 1$, a $C = 0$, to zbieżność nazywamy **nadliniową**. Np. ciągi $a_n = 1/n$, $a_n = 2^{-n}$, $a_n = n^{-n}$ są zbieżne odpowiednio podliniowo, liniowo i nadliniowo.

Metoda Newtona — modyfikacje

1. Metoda Newtona z nadzorem $\alpha \in [a, b]$
   Idea: pilnujemy, aby $x_{n+1} \in [a, b]$

Metoda Newtona — modyfikacje

1. Metoda Newtona z nadzorem $\alpha \in [a, b]$
   Idea: pilnujemy, aby $x_{n+1} \in [a, b]$
2. Wypadek r-krotnego pierwiastka $(r > 1)$

$$x_{n+1} = x_n + r h_n, \qquad h_n := -\frac{f(x_n)}{f'(x_n)} \qquad (n = 0, 1, \ldots). \tag{4}$$

Metoda Newtona — modyfikacje

1. Metoda Newtona z nadzorem $\alpha \in [a, b]$
   Idea: pilnujemy, aby $x_{n+1} \in [a, b]$
2. Wypadek r-krotnego pierwiastka $(r > 1)$

$$x_{n+1} = x_n + r h_n, \qquad h_n := -\frac{f(x_n)}{f'(x_n)} \qquad (n = 0, 1, \ldots). \tag{4}$$

3. Algorytm adaptacyjny

$$x_{n+1} = x_n - r_n \frac{f(x_n)}{f'(x_n)} \qquad (n \geqslant 2).$$

gdzie

$$r_n := \frac{x_{n-1} - x_{n-2}}{2x_{n-1} - x_n - x_{n-2}}.$$

Metoda Newtona

W dziedzinie liczb zespolonych

Niech $f : \mathbb{C} \to \mathbb{C}$ będzie funkcją holomorficzną

$$z = x + \mathrm{i}y, \qquad f(z) = u(x, y) + \mathrm{i}v(x, y)$$

Metoda Newtona

W dziedzinie liczb zespolonych

Niech $f\colon \mathbb{C} \to \mathbb{C}$ będzie funkcją holomorficzną

$$z = x + \mathrm{i}y, \qquad f(z) = u(x,y) + \mathrm{i}v(x,y)$$

Niech

$$\phi(x,y) := |f(x+\mathrm{i}y)| = \sqrt{u^2(x,y) + v^2(x,y)}$$

Wówczas

$$\mathrm{grad}(\phi) := (\phi_x, \phi_y) = \frac{uu_x + vv_x + \mathrm{i}(uu_y + vv_y)}{\phi},$$

Ze wzorów Cauchy'ego-Riemanna wiemy, że

$$u_x = v_y, u_y = -v_x.$$

W metodzie Newtona mamy zatem

$$\frac{f(z)}{f'(z)} = \frac{u + \mathrm{i}v}{u_x + \mathrm{i}v_x} = \frac{uu_x + vv_x + \mathrm{i}(uu_y + vv_y)}{u_x^2 + v_x^2}.$$

Metoda Newtona

W dziedzinie liczb zespolonych

Niech $f\colon \mathbb{C} \to \mathbb{C}$ będzie funkcją holomorficzną

$$z = x + \mathrm{i}y, \qquad f(z) = u(x,y) + \mathrm{i}v(x,y)$$

W metodzie Newtona mamy zatem

$$\frac{f(z)}{f'(z)} = \frac{u + \mathrm{i}v}{u_x + \mathrm{i}v_x} = \frac{uu_x + vv_x + \mathrm{i}(uu_y + vv_y)}{u_x^2 + v_x^2}.$$

$$\begin{aligned} z_{n+1} = x_{n+1} + \mathrm{i}y_{n+1} = \\ x_n - \frac{u(x_n,y_n)u_x(x_n,y_n) + v(x_n,y_n)v_x(x_n,y_n)}{u_x^2(x_n,y_n) + v_x^2(x_n,y_n)} \\ \mathrm{i}\left(y_n - \frac{u(x_n,y_n)u_y(x_n,y_n) + v(x_n,y_n)v_y(x_n,y_n)}{u_x^2(x_n,y_n) + v_x^2(x_n,y_n)}\right) \end{aligned} \tag{5}$$

Metoda siecznych

Zastępując we wzorze (2) pochodną $f'(x_n)$ **ilorazem różnicowym**

$$f[x_{n-1},\, x_n] := \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}$$

otrzymujemy *metodę siecznych*:

$$x_{n+1} := x_n + h_n, \quad h_n := -f_n \frac{x_n - x_{n-1}}{f_n - f_{n-1}} \tag{6}$$

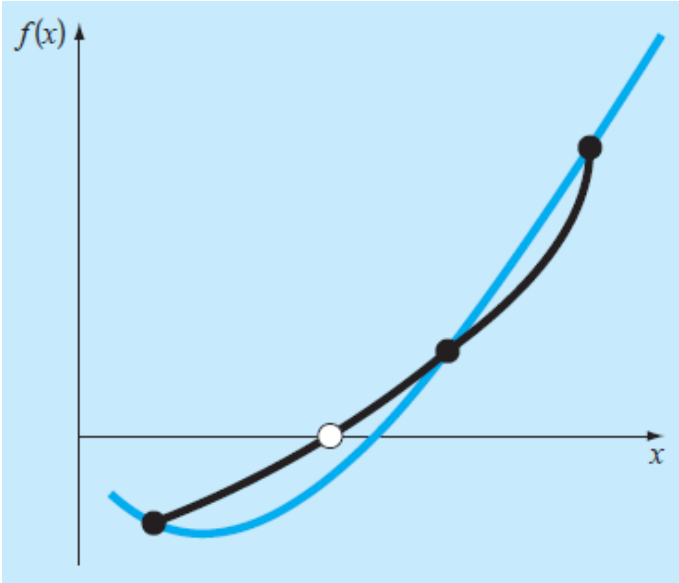
$$(f_n \neq f_{n-1};\ n = 1, 2, \ldots;\ x_0,\ x_1 \text{ – dane}),$$

gdzie $f_n := f(x_n)$.

Metoda regula falsi

Regula falsi jest wariantem metody siecznych, w którym – inaczej niż w tamtej metodzie – prowadzi się sieczną przez punkty $(x_n,\ f_n)$ i $(x_{n'},\ f_{n'})$, gdzie n' jest takim największym wskaźnikiem mniejszym od n, że $f_{n'} f_n < 0$. Początkowe przybliżenia x_0 i x_1 trzeba oczywiście wybrać tak, żeby $f_0 f_1 < 0$.

Odwrotna interpolacja kwadratowa



$$x_{n+1} = \frac{f_{n-1}f_n}{(f_{n-2}-f_{n-1})(f_{n-2}-f_n)}x_{n-2} + \frac{f_{n-2}f_n}{(f_{n-1}-f_{n-2})(f_{n-1}-f_n)}x_{n-1} + \frac{f_{n-2}f_{n-1}}{(f_n-f_{n-2})(f_n-f_{n-1})}x_n$$

Metoda Brenta

http://gams.nist.gov/cgi-bin/serve.cgi/Module/C/BRENT/11665

Układ równań nieliniowych

Twierdzenie (Wzór Taylora)

$$f(\boldsymbol{a}+\boldsymbol{h}) = \sum_{j=0}^{\infty} \left. \frac{\left(\boldsymbol{h}^T \nabla_x\right)^j f(\boldsymbol{x})}{j!} \right|_{\boldsymbol{x}=\boldsymbol{a}}$$

$$\nabla_x = \begin{bmatrix} \frac{\partial}{\partial x_1} \\ \frac{\partial}{\partial x_2} \\ .. \\ \frac{\partial}{\partial x_n} \end{bmatrix}$$

Układ równań nieliniowych

Twierdzenie (Wzór Taylora)

$$f(\boldsymbol{a}+\boldsymbol{h}) = \sum_{j=0}^{\infty} \frac{\left(\boldsymbol{h}^T \nabla_x\right)^j f(\boldsymbol{x})}{j!}\Bigg|_{\boldsymbol{x}=\boldsymbol{a}}$$

$$\nabla_x = \begin{bmatrix} \frac{\partial}{\partial x_1} \\ \frac{\partial}{\partial x_2} \\ .. \\ \frac{\partial}{\partial x_n} \end{bmatrix}$$

Na przykład dla funkcji dwu zmiennych $f(x_1, x_2)$ mamy

$$f(x_1+h1, x_2+h2) = f(x_1,x_2) + h_1 \frac{\partial}{\partial x_1}(x_1,x_2) + h_2 \frac{\partial}{\partial x_2} f(x_1,x_2) + \ldots$$

Metoda Newtona 2D

- punkt początkowy: $[x_1^{(0)}, x_2^{(0)}]^T$
- krok metody:

$$\begin{bmatrix} x_1^{(n+1)} \\ x_2^{(n+1)} \end{bmatrix} = \begin{bmatrix} x_1^{(n)} \\ x_2^{(n)} \end{bmatrix} + \begin{bmatrix} h_1^{(n)} \\ h_2^{(n)} \end{bmatrix},$$

gdzie

$$J^{(n)} \cdot \begin{bmatrix} h_1^{(n)} \\ h_2^{(n)} \end{bmatrix} = - \begin{bmatrix} f_1(x_1^{(n)}, x_2^{(n)}) \\ f_2(x_1^{(n)}, x_2^{(n)}) \end{bmatrix},$$

przy czym $J^{(n)}$ jest macierzą Jacobiego

$$J^{(n)} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(x_1^{(n)}, x_2^{(n)}) & \frac{\partial f_1}{\partial x_2}(x_1^{(n)}, x_2^{(n)}) \\ \frac{\partial f_2}{\partial x_1}(x_1^{(n)}, x_2^{(n)}) & \frac{\partial f_2}{\partial x_2}(x_1^{(n)}, x_2^{(n)}) \end{bmatrix}$$

Zadanie

Rozważamy wielomian

$$w_n(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_0, \tag{7}$$

gdzie wszystkie $a_j \in \mathbb{R}$ albo nawet $a_j \in \mathbb{C}$.

Zadanie

Rozważamy wielomian

$$w_n(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_0, \tag{7}$$

gdzie wszystkie $a_j \in \mathbb{R}$ albo nawet $a_j \in \mathbb{C}$.

Twierdzenie

Wielomian stopnia n ma dokładnie n pierwiastków na płaszczyźnie zespolonej, gdy każdy z nich liczony tyle razy, ile wynosi jego krotność.

Lokalizacja pierwiastków

Twierdzenie (I)

Wszystkie pierwiastki wielomianu (7) leżą w kole otwartym o środku w punkcie 0 płaszczyzny zespolonej i promieniu

$$R := 1 + |a_0|^{-1} \max_{1 \leqslant k \leqslant n} |a_k|.$$

Lokalizacja pierwiastków

Twierdzenie (I)

Wszystkie pierwiastki wielomianu (7) leżą w kole otwartym o środku w punkcie 0 płaszczyzny zespolonej i promieniu

$$R := 1 + |a_0|^{-1} \max_{1 \leqslant k \leqslant n} |a_k|.$$

Twierdzenie (II)

Niech będzie

$$u(x) := x^n w(1/x) = a_n + a_{n-1}x + \cdots + a_0 x^0 \tag{8}$$

Jeśli wszystkie pierwiastki wielomianu u leżą w kole $|x| < r$, to wszystkie niezerowe pierwiastki wielomianu w leżą poza kołem $|x| < 1/r$.

Przykład

Na przykład na mocy tw. I zera wielomianu

$$w(x) = x^4 - 4x^3 + 7x^2 - 5x - 2$$

leżą w kole o promieniu $R = 1 + |a_0|^{-1} \max_{1 \leqslant k \leqslant 4} |a_k| = 8$. Pierwiastki wielomianu

$$u(x) = -2x^4 - 5x^3 + 7x^2 - 4x + 1$$

leżą w kole o promieniu $r = 1 + |a_0|^{-1} \max_{1 \leqslant k \leqslant 4} |a_k| = \frac{9}{2}$. Na podstawie tw. II pierwiastki wielomianu w leżą poza kołem o promieniu $\frac{2}{9}$. Ostatecznie wszystkie pierwiastki wielomianu w leżą w pierścieniu

$$\frac{2}{9} < |x| < 8$$

na płaszczyźnie zespolonej.

Schemat Hornera

Algorytm (Schemat Hornera)

Obliczanie wartości $w_n(z)$*:*

1: $b \leftarrow a_n$
2: ***for*** $k = n-1$ ***to*** 0 ***do***
3: $\quad b \leftarrow a_k + zb$
4: ***end for***
5: ***return*** b

Dzielenie syntetyczne

Niech będzie

$$w_n(x) = (x - z)q_{n-1}(x; z) + b_{-1}$$

oraz

$$q_{n-1}(k; z) = b_0 + b_1 x + \ldots + b_{n-1}x^{n-1}.$$

Dzielenie syntetyczne

Niech będzie

$$w_n(x) = (x - z)q_{n-1}(x; z) + b_{-1}$$

oraz

$$q_{n-1}(k; z) = b_0 + b_1 x + \ldots + b_{n-1}x^{n-1}.$$

Łatwo zauważyć, że

$$b_{k-1} = a_k + zb_k \qquad 0 \leqslant k \leqslant n - 1.$$

Dzielenie syntetyczne

Niech będzie

$$w_n(x) = (x - z)q_{n-1}(x; z) + b_{-1}$$

oraz

$$q_{n-1}(k; z) = b_0 + b_1 x + \ldots + b_{n-1}x^{n-1}.$$

Łatwo zauważyć, że

$$b_{k-1} = a_k + zb_k \qquad 0 \leqslant k \leqslant n - 1.$$

Ponadto, mamy

$$w'_n(z) = q_{n-1}(z; z).$$

Dzielenie syntetyczne

Niech będzie

$$w_n(x) = (x - z)q_{n-1}(x; z) + b_{-1}$$

oraz

$$q_{n-1}(k; z) = b_0 + b_1 x + \ldots + b_{n-1}x^{n-1}.$$

Ponadto, mamy

$$w'_n(z) = q_{n-1}(z; z).$$

Algorytm (Schemat Hornera, raz jeszcze)

Obliczanie wartości $w_n(z)$ *i* $w'_n(z)$*:*

1: $b \leftarrow a_n$
2: $c \leftarrow 0$
3: ***for*** $k = n - 1$ ***to*** 0 ***do***
4: $\quad c \leftarrow b + zc$
5: $\quad b \leftarrow a_k + zb$
6: ***end for***
7: ***return*** b, c

Metoda Newtona

$$z_{k+1} := z_k - \frac{w_n(z_k)}{w_n'(z_k)}$$

Metoda Laguerre'a

$$z_{k+1} = z_k - \frac{n\, w(z_k)}{w'(z_k) \pm \sqrt{H(z_k)}}, \tag{9}$$

$$H(x) := (n-1)\left[(n-1)w'^2(x) - nw(x)\, w''(x)\right], \tag{10}$$

a n jest stopniem wielomianu. Znak w mianowniku wyrażenia (10) trzeba wybrać tak, aby $|z_{k+1} - z_k|$ było jak najmniejsze.

Metoda Laguerre'a

$$z_{k+1} = z_k - \frac{n\, w(z_k)}{w'(z_k) \pm \sqrt{H(z_k)}}, \tag{9}$$

$$H(x) := (n-1)\left[(n-1)w'^2(x) - n w(x)\, w''(x)\right], \tag{10}$$

a n jest stopniem wielomianu. Znak w mianowniku wyrażenia (10) trzeba wybrać tak, aby $|z_{k+1} - z_k|$ było jak najmniejsze.

1. Metoda Laguerre'a wymaga obliczania $w(z_k)$, $w'(z_k)$ i $w''(z_k)$ w każdym kroku.

Metoda Laguerre'a

$$z_{k+1} = z_k - \frac{n\, w(z_k)}{w'(z_k) \pm \sqrt{H(z_k)}}, \tag{9}$$

$$H(x) := (n-1)\left[(n-1)w'^2(x) - n w(x)\, w''(x)\right], \tag{10}$$

a n jest stopniem wielomianu. Znak w mianowniku wyrażenia (10) trzeba wybrać tak, aby $|z_{k+1} - z_k|$ było jak najmniejsze.

1. Metoda Laguerre'a wymaga obliczania $w(z_k)$, $w'(z_k)$ i $w''(z_k)$ w każdym kroku.
2. Można wykazać, że jest ona **zbieżna sześciennie dla pierwiastków pojedynczych**, rzeczywistych lub zespolonych.

Metoda Laguerre'a

$$z_{k+1} = z_k - \frac{n\, w(z_k)}{w'(z_k) \pm \sqrt{H(z_k)}}, \tag{9}$$

$$H(x) := (n-1)\left[(n-1)w'^2(x) - n w(x)\, w''(x)\right], \tag{10}$$

a n jest stopniem wielomianu. Znak w mianowniku wyrażenia (10) trzeba wybrać tak, aby $|z_{k+1} - z_k|$ było jak najmniejsze.

1. Metoda Laguerre'a wymaga obliczania $w(z_k)$, $w'(z_k)$ i $w''(z_k)$ w każdym kroku.
2. Można wykazać, że jest ona **zbieżna sześciennie dla pierwiastków pojedynczych**, rzeczywistych lub zespolonych.
3. Dla równań algebraicznych mających tylko pierwiastki rzeczywiste metoda Laguerre'a jest zbieżna niezależnie od wyboru przybliżeń początkowych.

Metoda Laguerre'a

$$z_{k+1} = z_k - \frac{n\, w(z_k)}{w'(z_k) \pm \sqrt{H(z_k)}}, \tag{9}$$

$$H(x) := (n-1)\left[(n-1)w'^2(x) - n w(x)\, w''(x)\right], \tag{10}$$

a n jest stopniem wielomianu. Znak w mianowniku wyrażenia (10) trzeba wybrać tak, aby $|z_{k+1} - z_k|$ było jak najmniejsze.

1. Metoda Laguerre'a wymaga obliczania $w(z_k)$, $w'(z_k)$ i $w''(z_k)$ w każdym kroku.
2. Można wykazać, że jest ona **zbieżna sześciennie dla pierwiastków pojedynczych**, rzeczywistych lub zespolonych.
3. Dla równań algebraicznych mających tylko pierwiastki rzeczywiste metoda Laguerre'a jest zbieżna niezależnie od wyboru przybliżeń początkowych.
4. Jeśli równanie algebraiczne ma pierwiastki zespolone, to już nie jest prawdą, że metoda Laguerre'a jest zbieżna dla dowolnych przybliżeń początkowych. Doświadczenie uczy jednak, że i w tym wypadku zbieżność globalna jest dobra.

Rafał Nowak

Notatka do wykładu analizy numerycznej

Kilka własności wielomianów Czebyszewa

Niech $\{T_n(x)\}_{n=0}^{\infty}$ oznacza ciąg wielomianów Czebyszewa I-go rodzaju:

$$T_0(x) \equiv 1, \qquad T_1(x) = x, \qquad T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x) \qquad (k \geqslant 2),$$

a $\{U_n(x)\}_{n=0}^{\infty}$ — ciąg wielomianów Czebyszewa II-go rodzaju:

$$U_0(x) \equiv 1, \qquad U_1(x) = 2x, \qquad U_k(x) = 2xU_{k-1}(x) - U_{k-2}(x) \qquad (k \geqslant 2).$$

Łatwo sprawdzić, że zera $t_k \equiv t_{n+1,k}$ wielomianu T_{n+1} wyrażają się wzorami

$$t_{n+1,k} \coloneqq \cos\frac{2k+1}{2n+2}\pi \qquad (k = 0, 1, \ldots, n).$$

Natomiast punkty ekstremalne $u_k \equiv u_{nk}$ wielomianu T_n wyrażają się wzorami

$$u_{nk} \coloneqq \cos(k\pi/n) \qquad (k = 0, 1, \ldots, n).$$

Lemat 1. *Wielomiany T_n są ortogonalne w sensie iloczynu skalarnego*

$$\langle f, g\rangle = \int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} f(x)g(x)\,\mathrm{d}x.$$

Zachodzi wzór

$$\langle T_i, T_j\rangle = \begin{cases} \pi, & i = j = 0, \\ \pi/2, & i = j \neq 0, \\ 0, & i \neq j. \end{cases} \tag{1}$$

Lemat 2. *Wielomiany U_n są ortogonalne w sensie iloczynu skalarnego*

$$\langle f, g\rangle = \int_{-1}^{1} \sqrt{1-x^2} f(x)g(x)\,\mathrm{d}x.$$

Zachodzi wzór

$$\langle U_i, U_j\rangle = \begin{cases} \pi/2, & i = j, \\ 0, & i \neq j. \end{cases} \tag{2}$$

Lemat 3. *Wielomiany $T_0, T_1, \ldots, T_n$ są ortogonalne w sensie dyskretnego iloczynu skalarnego*

$$\langle f, g\rangle = \sum_{k=0}^{n} f(t_k)g(t_k).$$

Zachodzi wzór

$$\langle T_i, T_j\rangle = \begin{cases} n+1, & i = j = 0, \\ (n+1)/2, & i = j \neq 0, \\ 0, & i \neq j. \end{cases} \tag{3}$$

Lemat 4. *Wielomiany $T_0, T_1, \ldots, T_n$ są ortogonalne w sensie dyskretnego iloczynu skalarnego*

$$\langle f, g\rangle = \sum_{k=0}^{n}{}'' f(u_k)g(u_k).$$

Zachodzi wzór

$$\langle T_i, T_j\rangle = \begin{cases} n, & i = j = 0 \text{ lub } i = j = n, \\ n/2, & i = j \neq 0, n \\ 0, & i \neq j. \end{cases} \tag{4}$$

Lemat 5. *Wielomian $I_n \in \Pi_n$ interpolujący funkcję f w węzłach t_k można zapisać w postaci*

$$I_n(x) = \sum_{i=0}^{n}{}' \alpha_i \, T_i(x), \tag{5}$$

gdzie

$$\alpha_i := \frac{2}{n+1} \sum_{j=0}^{n} f(t_j) T_i(t_j) \qquad (i = 0, 1, \ldots, n). \tag{6}$$

Ponadto, mamy

$$\int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} I_n(x) \, \mathrm{d}x = \frac{\pi}{n+1} \sum_{j=0}^{n} f(t_j).$$

Lemat 6. *Wielomian $J_n \in \Pi_n$ interpolujący funkcję f w węzłach u_k można zapisać wzorem*

$$J_n(x) = \sum_{j=0}^{n}{}'' \beta_j \, T_j(x), \tag{7}$$

gdzie

$$\beta_j := \frac{2}{n} \sum_{k=0}^{n}{}'' f(u_k) T_j(u_k) \qquad (j = 0, 1, \ldots, n). \tag{8}$$

Ponadto, mamy

$$\int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} J_n(x) \, \mathrm{d}x = \frac{\pi}{n} \sum_{j=0}^{n}{}'' f(u_j).$$

Analiza numeryczna

3. Interpolacja

Rafał Nowak

Interpolacja
Zadanie interpolacyjne Lagrange'a

1. dane: $[x_0, x_1, \ldots, x_n]$, $[y_0, y_1, \ldots, y_n]$
2. znaleźć wielomian $L_n(x)$ st. $\leqslant n$ o własnościach

$$L_n(x_i) = y_i, \qquad i = 0, 1, \ldots, n$$

3. rozwiązanie (postać Lagrange'a):

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

gdzie

$$\lambda_k(x) := \prod_{j=0,\, j\neq k}^{n} \frac{x - x_j}{x_k - x_j}$$

Inne postaci wielomianu interpolacyjnego

Niech

$$\sigma_k := \prod_{j=0,\, j\neq k}^{n} \frac{1}{x_k - x_j} \qquad (0 \leqslant k \leqslant n)$$

- postać barycentryczna

$$L_n(x) = \begin{cases} \displaystyle\sum_{k=0}^{n} \frac{\sigma_k}{x - x_k} y_k \Bigg/ \sum_{k=0}^{n} \frac{\sigma_k}{x - x_k}, & \text{gdy } x \notin \{x_0, x_1, \ldots, x_n\}, \\ y_k, & \text{gdy } x = x_k,\ 0 \leqslant k \leqslant n. \end{cases}$$

Algorytm Wernera

Algorytm (Werner, 1984)

1. *Obliczamy pomocnicze wielkości* $a_k^{(i)}$ *wg wzorów*

$$a_0^{(0)} := 1, \quad a_k^{(0)} := 0 \quad (k = 1, 2, \ldots, n),$$

$$\left.\begin{aligned} a_k^{(i)} &:= a_k^{(i-1)} / (x_k - x_i), \\ a_i^{(k+1)} &:= a_i^{(k)} - a_k^{(i)} \end{aligned}\right\} \quad (i = 1, 2, \ldots, n;\ k = 0, 1, \ldots, i-1),$$

2. *Wówczas*

$$\sigma_k := a_k^{(n)} \quad (k = 0, 1, \ldots, n).$$

Inne postaci wielomianu interpolacyjnego

Niech

$$p_0(x) \equiv 1, \quad p_k(x) := (x - x_0)(x - x_1) \dots (x - _{k-1}) \quad (1 \leqslant k \leqslant n+1)$$

oraz

$$b_k := \sum_{i=0}^{k} \frac{y_i}{p'_{k+1}(x_i)} = \sum_{i=0}^{k} \frac{y_i}{\prod_{j=0, j\neq i}^{k} (x_i - x_j)} \quad (k = 0, 1, \dots, n)$$

- postać Newtona

$$L_n(x) = \sum_{k=0}^{n} b_k p_k(x)$$

Uogólniony schemat Hornera

Algorytm (uogólniony algorytm Hornera)

$$
\begin{aligned}
&w_n := b_n;\\
&w_k := w_{k+1}(x - x_k) + b_k \qquad (k = n-1, n-2, \ldots, 0);\\
&w(x) = w_0.
\end{aligned}
$$

Ponieważ

$$\sigma_k = \frac{1}{p'_{n+1}(x_k)},$$

więc

- inny wariant wzoru Lagrange'a

$$L_n(x) = p_{n+1}(x) \sum_{k=0}^{n} y_k \frac{\sigma_k}{x - x_k}.$$

Ilorazy różnicowe

Definicja

Niech funkcja f będzie określona w parami różnych punktach $x_0, x_1, \ldots$
Iloraz różnicowy k-tego rzędu (krócej: *k-ty iloraz różnicowy*) $(k = 0, 1, \ldots)$ funkcji f w punktach $x_0, x_1, \ldots, x_k$ oznaczamy symbolem $f[x_0, x_1, \ldots, x_k]$ i określamy wzorem

$$f[x_0, x_1, \ldots, x_k] := \sum_{i=0}^{k} \frac{f(x_i)}{\prod_{j=0, j\neq i}^{k} (x_i - x_j)}. \tag{1}$$

Własności ilorazów różnicowych

1. Iloraz $f[x_0, x_1, \ldots, x_k]$ jest symetryczną funkcją zmiennych $x_0, x_1, \ldots, x_k$.
2. Iloraz różnicowy zależy liniowo od funkcji, dla której został utworzony, tj. jeśli $f = g + ch$ (c - stała), to $f[x_0, x_1, \ldots, x_k] = g[x_0, x_1, \ldots, x_k] + ch[x_0, x_1, \ldots, x_k]$.
3. Jeśli $w \in \Pi_m \setminus \Pi_{m-1}$, to $w[x, x_1, \ldots, x_k]$ jest wielomianem stopnia $(m-k)$-tego zmiennej x; w szczeg. iloraz $w[x, x_1, \ldots, x_m]$ jest stałą, a $w[x, x_1, \ldots, x_{m+1}]$ jest zerem.
4. Zachodzi wzór rekurencyjny

$$f[x_0, x_1, \ldots, x_k] = \frac{f[x_1, x_2, \ldots, x_k] - f[x_0, x_1, \ldots, x_{k-1}]}{x_k - x_0}$$

$$(k = 1, 2, \ldots).$$

Obliczanie ilorazów różnicowych

Schemat obliczeń

$$
\begin{array}{llllllll}
x_0 & \underline{f(x_0)} & \searrow & & & & & \\
x_1 & f(x_1) & \rightarrow & \underline{f[x_0,x_1]} & & & & \\
x_2 & f(x_2) & & f[x_1,x_2] & & & & \\
\multicolumn{8}{c}{\dotfill} \\
x_{n-1} & f(x_{n-1}) & \searrow & f[x_{n-2},x_{n-1}] & \cdots & \underline{f[x_0,x_1,\ldots,x_{n-1}]} & \searrow & \\
x_n & f(x_n) & \rightarrow & f[x_{n-1},x_n] & \cdots & f[x_0,x_1,\ldots,x_{n-1}] & \rightarrow & \underline{f[x_0,x_1,\ldots,x_n]}
\end{array}
$$

Obliczanie ilorazów różnicowych

Algorytm (Obliczanie współczynników postaci Newtona)

Ensure: $b_k = f[x_0, \ldots, x_k]$

1: ***for*** $k = 0$ ***to*** n ***do***
2: $\quad b_k \leftarrow f(x_k)$
3: ***end for***
4: ***for*** $j = 1$ ***to*** n ***do***
5: $\quad$ ***for*** $k = n$ ***downto*** j ***do***
6: $\quad\quad b_k \leftarrow (b_k - b_{k-1})/(x_k - x_{k-j})$
7: $\quad$ ***end for***
8: ***end for***
9: ***return*** b

Reszta wzoru interpolacyjnego

Twierdzenie

Niech f będzie funkcją określoną w przedziale $[a,b]$, niech $x_0, x_1, \ldots, x_n \in [a,b]$ będą parami różne i niech wielomian $L_n \in \Pi_n$ spełnia warunki

$$L_n(x_i) = f(x_i) \qquad (i = 0, 1, \ldots, n). \tag{2}$$

Wówczas dla każdego $x \in [a,b] \setminus \{x_0, x_1, \ldots, x_n\}$ zachodzi równość

$$f(x) - L_n(x) = f[x, x_0, x_1, \ldots, x_n] p_{n+1}(x), \tag{3}$$

gdzie

$$p_{n+1}(x) = (x - x_0)(x - x_1) \cdots (x - x_n).$$

Twierdzenie

Jeśli funkcja f ma w przedziale $[a, b]$ *ciągłą* $(n+1)$*-szą pochodną, a wielomian* $L_n \in \Pi_n$ *interpoluje tę funkcję w parami różnych punktach* $x_0, x_1, \ldots, x_n \in [a, b]$*, to dla każdego* $x \in [a, b]$ *zachodzi równość*

$$f(x) - L_n(x) = \frac{1}{(n+1)!} f^{(n+1)}(\xi_x) p_{n+1}(x), \tag{4}$$

gdzie ξ_x *jest pewną liczbą (zależną od* x*) z przedziału* (a, b)*.*

Twierdzenie

Jeśli funkcja f ma w przedziale $[a, b]$ *ciągłą* $(n+1)$*-szą pochodną, a wielomian* $L_n \in \Pi_n$ *interpoluje tę funkcję w parami różnych punktach* $x_0, x_1, \ldots, x_n \in [a, b]$*, to dla każdego* $x \in [a, b]$ *zachodzi równość*

$$f(x) - L_n(x) = \frac{1}{(n+1)!} f^{(n+1)}(\xi_x) p_{n+1}(x), \tag{4}$$

gdzie ξ_x *jest pewną liczbą (zależną od* x*) z przedziału* (a, b)*.*

Wniosek

Jeśli $f \in C^{n+1}[a, b]$*,* $x_0, x_1, \ldots, x_n, x_{n+1} \in [a, b]$*, to istnieje taki punkt* $\xi \in (a, b)$*, że*

$$f[x_0, x_1, \ldots, x_{n+1}] = \frac{1}{(n+1)!} f^{(n+1)}(\xi).$$

Wniosek

Jeśli funkcja f ma w przedziale $[-1, 1]$ ciągłą $(n+1)$-szą pochodną, to

$$\max_{-1 \leqslant x \leqslant 1} |f(x) - L_n(x)| \leqslant \frac{M_{n+1} P_{n+1}}{(n+1)!}, \tag{5}$$

gdzie

$$M_{n+1} := \max_{-1 \leqslant x \leqslant 1} |f^{(n+1)}(x)|,$$
$$P_{n+1} := \max_{-1 \leqslant x \leqslant 1} |p_{n+1}(x)|.$$

Wielomiany Czebyszewa

Definicja (Wielomiany Czebyszewa (pierwszego rodzaju) $T_k(x)$)

$$T_0(x) \equiv 1; \quad T_1(x) = x;$$
$$T_k(x) = 2xT_{k-1} - T_{k-2} \qquad (k = 2, 3, \ldots).$$

1. Współczynnik wielomianu T_k $(k \geqslant 1)$ przy x^k (zwany **współczynnikiem wiodącym**) jest równy 2^{k-1}.
2. Zachodzi równość $T_k(-x) = (-1)^k T_k(x)$ dla $k \geqslant 0$.
3. Dla dowolnego x z przedziału $[-1, 1]$ k-ty wielomian Czebyszewa $(k \geqslant 0)$ wyraża się wzorem

$$T_k(x) = \cos(k \arccos x).$$

   Zatem $|T_k(x)| \leqslant 1 \quad (-1 \leqslant x \leqslant 1;\ k \geqslant 0)$.
4. **Punkty ekstremalne** wielomianu $T_k(x)$ w przedziale $[-1, 1]$, czyli rozwiązania równania $|T_k(x)| = 1$, wyrażają się wzorem

$$u_{kj} = \cos \frac{j\pi}{k} \qquad (j = 0, 1, \ldots, k).$$

   Stąd, wobec poprzedniej własności, mamy

$$\|T_k\|_{[-1,1]} = 1 \qquad (k \geqslant 0).$$

5. Wielomian Czebyszewa $T_k(x)$ $(k \geqslant 1)$ ma k zer pojedynczych, leżących w przedziale $(-1, 1)$, równych

$$t_{kj} = \cos \frac{2j+1}{2k}\pi \qquad (j = 0, 1, 2, \ldots, k-1).$$

Twierdzenie (Postać Czebyszewa wielomianu)

Każdy wielomian $w \in \Pi_n$ *można jednoznacznie przedstawić w postaci*

$$w(x) = \sum_{k=0}^{n}{}' c_k T_k(x). \tag{6}$$

Twierdzenie (Postać Czebyszewa wielomianu)

Każdy wielomian $w \in \Pi_n$ można jednoznacznie przedstawić w postaci

$$w(x) = \sum_{k=0}^{n}{}' c_k T_k(x). \tag{6}$$

Algorytm (algorytm Clenshawa)

Aby obliczyć wartość wielomianu (6) w punkcie x określamy pomocniczo wielkości B_0, B_1, B_{n+2} wzorami

$$\begin{aligned} &B_{n+2} := B_{n+1} := 0; \\ &B_k := 2xB_{k+1} - B_{k+2} + c_k \qquad (k = n, n-1, \ldots, 0). \end{aligned}$$

Wówczas

$$w(x) = \frac{1}{2}(B_0 - B_2).$$

Twierdzenie

Dla danych $c_i \in X_{fl}$ i $x \in X_{fl}$ wartość wielomianu (6) obliczonego za pomocą algortymu Clenshawa wyraża sie wzorem

$$\mathrm{fl}(w(x)) = \sum_{k=0}^{n}{}' c_k(1+e_k)T_k(x),$$

gdzie $|e_k| \leqslant L(n)\,u$, przy czym $L(n)$ rośnie kwadratowo wraz z n.

Zatem algortym Clenshawa jest numerycznie poprawny.

Węzły Czebyszewa

1. zera wielomianu T_{n+1}:

$$x_k := t_{n+1,k} = \cos \frac{2k+1}{2n+2}\pi \qquad (k = 0, 1, \ldots, n)$$

2. punkty ekstremalne wielomianu T_n:

$$x_k := u_{n,k} = \cos \frac{k}{n}\pi \qquad (k = 0, 1, \ldots, n)$$

Lemat

Wielomian $\tilde{T}_n := 2^{1-n}T_n$ *ma najmniejszą normę w przedziale* $[-1, 1]$ *spośród wszystkich wielomianów stopnia* $\leqslant n$*, o współczynniku wiodącym równym 1.*

Lemat

Wielomian $\tilde{T}_n := 2^{1-n}T_n$ *ma najmniejszą normę w przedziale* $[-1, 1]$ *spośród wszystkich wielomianów stopnia* $\leqslant n$*, o współczynniku wiodącym równym 1.*

Wniosek

W poniższym oszacowaniu błędu interpolacji

$$\max_{-1\leqslant x\leqslant 1} |f(x) - L_n(x)| \leqslant \frac{M_{n+1}P_{n+1}}{(n+1)!}, \tag{7}$$

prawa strona jest najmniejsza i równa $\frac{M_{n+1}}{2^n(n+1)!}$ *wtedy i tylko wtedy, gdy*

$$p_{n+1}(x) = 2^{-n}T_{n+1}(x),$$

tj. gdy węzłami $x_0, x_1, \ldots, x_n$ *są zera wielomianu Czebyszewa* T_{n+1}*.*

Lemat

Wielomian $I_n \in \Pi_n$ *interpolujący funkcję* f *w węzłach*

$$t_j \equiv t_{n+1,j} = \cos \frac{2j+1}{2n+2}\pi \qquad (j = 0, 1, \ldots, n)$$

(zerach wielomianu T_{n+1}*) można zapisać w postaci*

$$I_n(x) = \sum_{k=0}^{n}{}' \alpha_k T_k(x), \tag{8}$$

gdzie

$$\alpha_k := \frac{2}{n+1} \sum_{j=0}^{n} f(t_j) T_k(t_j) \qquad (k = 0, 1, \ldots, n). \tag{9}$$

Lemat

Wielomian $J_n \in \Pi_n$ *interpolujący funkcję* f *w węzłach*

$$u_j \equiv u_{nj} = \cos(j\pi/n) \qquad (j = 0, 1, \ldots, n)$$

(punktach ekstremalnych wielomianu T_n*) można zapisać wzorem*

$$J_n(x) = \sum_{k=0}^{n}{}'' \beta_k \, T_k(x), \tag{10}$$

gdzie

$$\beta_k := \frac{2}{n} \sum_{j=0}^{n}{}'' f(u_j) T_k(u_j) \qquad (k = 0, 1, \ldots, n). \tag{11}$$

Wybór węzłów

- Wybór węzłów w przedziale $[a, b]$: Zauważmy, że funkcja

$$t \to \frac{b-a}{2}t + \frac{a+b}{2}$$

przekształca przedział $[-1, 1]$ w przedział $[a, b]$.

Wybór węzłów

- Wybór węzłów w przedziale $[a, b]$: Zauważmy, że funkcja

$$t \to \frac{b-a}{2}t + \frac{a+b}{2}$$

przekształca przedział $[-1, 1]$ w przedział $[a, b]$.

Oszacowanie reszty wzoru interpolacyjnego:

1. węzły równoodległe

$$x_k = -1 + 2k/n \qquad (k = 0, 1, \ldots, n);$$

Mamy

$$\frac{1}{n^{3/2}}\left(\frac{2}{e}\right)^{n+1} \leqslant P_{n+1}^{e} \leqslant n!\left(\frac{2}{n}\right)^{n+1},$$

przy czym lewa nierówność zachodzi dla dostatecznie dużego n.

2. węzły Czebyszewa:

$$P_{n+1} = 2^{-n},$$

DFT

Dyskretna transformata Fouriera (DFT)

DFT przekształca ciąg $\boldsymbol{x} = [x_0, x_1, \ldots, x_{N-1}]$ w ciąg $\boldsymbol{y} = [y_0, y_1, \ldots, y_{N-1}]$, gdzie

$$y_k = \sum_{j=0}^{N-1} x_j e^{2\pi \mathrm{i} jk/N} \qquad (0 \leqslant k < N). \tag{12}$$

Wyznaczanie wszystkich wartości y_k wprost ze wzoru (12) wymaga wykonania $\mathcal{O}(N^2)$ operacji arytmetycznych. Okazuje się, że można to zrobić w czasie $\mathcal{O}(N \log N)$ — wykorzystując technikę *dziel i zwyciężaj*.

Algorytm DFT (1/2)

Niech $\omega_N := e^{2\pi \mathrm{i}/N}$. Wówczas wzór (12) można zapisać w postaci

$$y_k = \sum_{j=0}^{N-1} x_j \omega_N^{jk} \qquad (0 \leqslant k < N). \tag{13}$$

Załóżmy, że $N = 2M$, a najlepiej niech N będzie potęgą dwójki.

Algorytm DFT (1/2)

Niech $\omega_N := e^{2\pi \mathrm{i}/N}$. Wówczas wzór (12) można zapisać w postaci

$$y_k = \sum_{j=0}^{N-1} x_j \omega_N^{jk} \qquad (0 \leqslant k < N). \tag{13}$$

Załóżmy, że $N = 2M$, a najlepiej niech N będzie potęgą dwójki. Łatwo sprawdzić, że dla $k = 0, 1, \ldots, M-1$ mamy

$$y_{2k} = \sum_{j=0}^{M-1} \underbrace{(x_j + x_{M+j})}_{a_j} \omega_M^{jk},$$

$$y_{2k+1} = \sum_{j=0}^{M-1} \left[\underbrace{(x_j - x_{M+j})\omega_N^j}_{b_j} \right] \omega_M^{jk}.$$

Algorytm DFT (1/2)

Niech $\omega_N := e^{2\pi \mathrm{i}/N}$. Wówczas wzór (12) można zapisać w postaci

$$y_k = \sum_{j=0}^{N-1} x_j \omega_N^{jk} \qquad (0 \leqslant k < N). \tag{13}$$

Załóżmy, że $N = 2M$, a najlepiej niech N będzie potęgą dwójki. Łatwo sprawdzić, że dla $k = 0, 1, \ldots, M-1$ mamy

$$y_{2k} = \sum_{j=0}^{M-1} (\underbrace{x_j + x_{M+j}}_{a_j}) \omega_M^{jk},$$

$$y_{2k+1} = \sum_{j=0}^{M-1} \left[\underbrace{(x_j - x_{M+j})\omega_N^j}_{b_j} \right] \omega_M^{jk}.$$

Oznacza to, że wektor $\boldsymbol{y}$ można obliczyć wywołując DFT$(\boldsymbol{a})$ i DFT$(\boldsymbol{b})$ czyli dwukrotnie DFT, ale dla wektorów o połowę krótszych.

Algorytm DFT (2/2)

Algorytm (Szybka transformata Fouriera)

DFT$(\boldsymbol{x})$

1: $N \leftarrow length(\boldsymbol{x})$
2: **if** $N = 1$ **then**
3: **return** x
4: **end if**
5: $M \leftarrow N/2$
6: $\boldsymbol{x}_{left} \leftarrow \boldsymbol{x}[1 : M]$
7: $\boldsymbol{x}_{right} \leftarrow \boldsymbol{x}[M + 1 : N]$
8: $\boldsymbol{y}_{even} \leftarrow DFT(\boldsymbol{x}_{left} + \boldsymbol{x}_{right})$
9: $\boldsymbol{y}_{odd} \leftarrow DFT((\boldsymbol{x}_{left} - \boldsymbol{x}_{right}) .* [\omega_N^j \textbf{ for } j = 0 : M - 1])$
10: $\boldsymbol{y}[1 : 2 : N - 1] \leftarrow \boldsymbol{y}_{even}$
11: $\boldsymbol{y}[2 : 2 : N] \leftarrow \boldsymbol{y}_{odd}$
12: **return** $\boldsymbol{y}$

Uwaga: operator $.*$ *oznacza mnożenie wektorów po współrzędnych.*

Zbieżność ciągu wielomianów interpolacyjnych

Twierdzenie (Bernstein)

Niech będzie $f(x) = |x|$, $[a, b] = [-1, 1]$, $x_{nk} = -1 + \frac{2k}{n}$ $(k = 0, 1, \ldots, n;\ n > 0)$. *Wówczas dla* $x \notin \{-1, 0, 1\}$ *ciąg* $\{L_n(x)\}$ *nie jest zbieżny do* $f(x)$*!*

Zbieżność ciągu wielomianów interpolacyjnych

Twierdzenie (Bernstein)

Niech będzie $f(x) = |x|$, $[a, b] = [-1, 1]$, $x_{nk} = -1 + \frac{2k}{n}$ $(k = 0, 1, \ldots, n;\ n > 0)$. *Wówczas dla* $x \notin \{-1, 0, 1\}$ *ciąg* $\{L_n(x)\}$ *nie jest zbieżny do* $f(x)$*!*

Twierdzenie (Runge)

Niech będzie $f(x) = 1/(1 + 25x^2)$, $[a, b] = [-1, 1]$, $x_{nk} = -1 + \frac{2k}{n}$ $(k = 0, 1, \ldots, n;\ n > 0)$. *Ciąg* $\{L_n(x)\}$ *jest zbieżny do* $f(x)$ *tylko dla* $|x| \leqslant 0.72668\ldots$ *i rozbieżny dla* $|x| > 0.72668\ldots$

Zbieżność ciągu wielomianów ...

Twierdzenie (Faber)

Dla każdej tablicy węzłów $\{x_{nk}\}$ *istnieje taka funkcja ciągła w przedziale* $[a, b]$, *do której ciąg wielomianów interpolacyjnych nie jest zbieżny jednostajnie (tj. taka, że* $\max_{a \leqslant x \leqslant b} |f(x) - L_n(x)| \not\to 0$*).*

Zbieżność ciągu wielomianów ...

Twierdzenie (Faber)

Dla każdej tablicy węzłów $\{x_{nk}\}$ *istnieje taka funkcja ciągła w przedziale* $[a, b]$, *do której ciąg wielomianów interpolacyjnych nie jest zbieżny jednostajnie (tj. taka, że* $\max_{a \leqslant x \leqslant b} |f(x) - L_n(x)| \not\to 0$*).*

Twierdzenie (Kryłow)

Niech dana będzie funkcja $f \in C^1[-1, 1]$ *i niech* $\{L_n\}$ *będzie ciągiem wielomianów interpolujących funkcję* f *w węzłach Czebyszewowskich. Wówczas dla każdego* $x \in [-1, 1]$ *jest*

$$\lim_{n \to \infty} L_n(x) = f(x).$$

Funkcja sklejana interpolująca III stopna

Definicja

Dla danej liczby naturalnej n, danych węzłów $x_0, x_1, \ldots, x_n$ $(a = x_0 < x_1 < \ldots < x_n = b)$ i danej funkcji f funkcją sklejaną interpolującą III stopnia nazywamy funkcję s, określoną w przedziale $[a, b]$ i spełniającą następujące warunki:

1° s, s' i s'' są ciągłe w $[a, b]$,

2° w każdym przedziałów $[x_{k-1}, x_k]$ $(k = 1, 2, \ldots, n)$ s jest identyczna z pewnym wielomianem p_k, stopnia co najwyżej trzeciego,

3° $s(x_k) = f(x_k) \qquad (k = 0, 1, \ldots, n).$

Jeśli dodatkowe (tzw. brzegowe) dwa warunki mają postać

4°_{nat} $s''(a) = s''(b) = 0$

4°_{comp} $s'(a) = f'(a),\ s'(b) = f'(b)$

4°_{per} $s'(a) = s'(b),\ s''(a) = s''(b)$ (jeśli f jest funkcją okresową o okresie $b - a$)

to s nazywamy odpowiednio funkcją naturalną, zupełną lub okresową.

Naturalna funkcja sklejana interpolująca III stopnia

Twierdzenie 1.

Dla dowolnych danych: $n \in \mathbf{N}$, $a = x_0 < x_1 < \ldots < x_n = b$ i funkcji f istnieje dokładnie jedna naturalna funkcja sklejana interpolacyjna III stopnia s. Wartości $M_k := s''(x_k)$ $(k = 0, 1, \ldots, n;\ M_0 = M_n = 0)$ spełniają układ równań liniowych

$$\lambda_k M_{k-1} + 2M_k + (1-\lambda_k) M_{k+1} = 6f[x_{k-1}, x_k, x_{k+1}] \quad (k = 1, \ldots, n-1), \tag{14}$$

gdzie $\lambda_k := h_k/(h_k + h_{k+1})$, $h_k := x_k - x_{k-1}$.
W każdym z przedziałów $[x_{k-1}, x_k]$ $(k = 1, 2, \ldots, n)$ jest

$$\begin{aligned} s(x) \quad = \quad & h_k^{-1} \left[\frac{1}{6} M_{k-1} (x_k - x)^3 + \frac{1}{6} M_k (x - x_{k-1})^3 \right. \\ & + \left(f(x_{k-1}) - \frac{1}{6} M_{k-1} h_k^2 \right) (x_k - x) \\ & \left. + \left(f(x_k) - \frac{1}{6} M_k h_k^2 \right) (x - x_{k-1}) \right]. \end{aligned} \tag{15}$$

Dowód 1/2

s'' jest funkcją kawałkami liniową; w przedziale $[x_{k-1}, x_k]$ wyraża się wzorem:

$$s''(x) = h_k^{-1}[M_{k-1}(x_k - x) + M_k(x - x_{k-1})]. \tag{16}$$

Całkując dwukrotnie otrzymujemy

$$s'(x) = (2h_k)^{-1}[-M_{k-1}(x_k - x)^2 + M_k(x - x_{k-1})^2] + A_k, \tag{17}$$

$$s(x) = (6h_k)^{-1}[M_{k-1}(x_k - x)^3 + M_k(x - x_{k-1})^3] + A_k x + B_k. \tag{18}$$

Stałe A_k i B_k wyznaczamy kładąc w (18) $x = x_{k-1}, x_k$ i uwzględniając równości $s(x_{k-1}) = f(x_{k-1}), s(x_k) = f(x_k)$. Otrzymujemy

$$A_k = \frac{f(x_k) - f(x_{k-1})}{h_k} - \frac{1}{6}h_k(M_k - M_{k-1}),$$

$$B_k = \frac{x_k f(x_{k-1}) - f(x_k)x_{k-1}}{h_k} - \frac{1}{6}h_k(M_{k-1}x_k - M_k x_{k-1}).$$

Wówczas wzór (17) jest równoważny wzorowi (15).

Dowód 2/2

Należy jeśli tylko dobrać tak M_k, aby zapewnić ciągłość s'. Ciągłość s i s'' wynika bowiem odpowiednio z (18) i (16).

Dowód 2/2

Należy jeśli tylko dobrać tak M_k, aby zapewnić ciągłość s'. Ciągłość s i s'' wynika bowiem odpowiednio z (18) i (16).
Ze wzoru (17) otrzymujemy, że

$$s'(x_{k-1}+0) = -\frac{1}{3}h_k M_{k-1} - \frac{1}{6}h_k M_k + f[x_{k-1}, x_k],$$
$$s'(x_k-0) = \frac{1}{3}h_k M_k + \frac{1}{6}h_k M_{k-1} + f[x_{k-1}, x_k].$$

Żądamy, aby było $s'(x_k-0) = s'(x_k+0)$ dla $k = 1, 2, \ldots, n-1$, czyli

$$\frac{1}{3}h_k M_k + \frac{1}{6}h_k M_{k-1} + f[x_{k-1}, x_k] =$$
$$= -\frac{1}{3}h_{k+1} M_k - \frac{1}{6}h_{k+1} M_{k+1} + f[x_k, x_{k+1}] \qquad (k = 1, 2, \ldots, n-1).$$

Po łatwych przekształceniach otrzymuje się stąd układ (14), tj. układ $n-1$ równań z $n-1$ niewiadomymi o nieosobliwej macierzy współczynników, który ma jedyne rozwiązanie $M_0, M_1, \ldots, M_{n-1}$, jednoznacznie określające funkcję s. □

Dalsze własności

Twierdzenie (Holladay)

W klasie funkcji F mających ciągłą drugą pochodną w przedziale $[a, b]$ i takich, że

$$F(x_k) = y_k \qquad (k = 0, 1, \ldots, n) \tag{19}$$

najmniejszą wartość całki

$$\int_a^b \left[F''(x)\right]^2 \, dx \tag{20}$$

daje naturalna funkcja sklejana s z twierdzenia 1. Przy tym

$$\int_a^b \left[s''(x)\right]^2 \, dx = \sum_{k=1}^{n-1} \left(f[x_k, x_{k+1}] - f[x_{k-1}, x_k]\right) M_k. \tag{21}$$

Algorytm obliczania wielkości M_k

Algorytm

Obliczamy pomocnicze wielkości $p_1, p_2, \ldots, p_{n-1}$, $q_0, q_1, \ldots, q_{n-1}$, $u_0, u_1, \ldots, u_{n-1}$ w następujący sposób rekurencyjny:

$$q_0 := u_0 := 0, \tag{22}$$

$$\left.\begin{aligned} p_k &:= \lambda_k q_{k-1} + 2, \\ q_k &:= (\lambda_k - 1)/p_k, \\ u_k &:= (d_k - \lambda_k u_{k-1})/p_k \end{aligned}\right\} \quad (k = 1, 2, \ldots, n-1), \tag{23}$$

gdzie

$$d_k := 6f[x_{k-1}, x_k, x_{k+1}] \qquad (k = 1, 2, \ldots, n-1). \tag{24}$$

Wówczas

$$M_{n-1} = u_{n-1}, \tag{25}$$

$$M_k = u_k + q_k M_{k+1} \qquad (k = n-2, n-3, \ldots, 1). \tag{26}$$

Twierdzenie

Niech będzie dana funkcja $f \in C^4[a, b]$. *Dla danej liczby naturalnej* n *niech* s *będzie naturalną funkcją sklejaną III stopnia interpolującą funkcję* f *w danych węzłach* $x_0, x_1, \ldots, x_n$ $(a = x_0 < x_1 < \ldots < x_n = b)$. *Wówczas*

$$\max_{a \leqslant x \leqslant b} |f^{(r)}(x) - s^{(r)}(x)| \leqslant C_r \, h^{4-r} \max_{a \leqslant x \leqslant b} |f^{(r)}(x)| \qquad (r = 0, 1, 2, 3),$$

gdzie $C_0 := 5/384$, $C_1 := 1/24$, $C_2 := 3/8$, $C_3 := (\beta + \beta^{-1})/2$,

$$h := \max_i h_i, \qquad \beta := h / \min_i h_i, \qquad h_i := x_i - x_{-1} \qquad (i = 1, 2, \ldots, n).$$

Przykład

W wypadku funkcji Rungego $f(x) = 1/(25x^2 + 1)$ $(-1 \leqslant x \leqslant 1)$ i równoodległych węzłów uzyskano następujące wyniki:

Tabela: Przykład Rungego: interpolacja za pomocą funkcji sklejanych III stopnia

n	10	20	40	80	160
h	0.2	0.1	0.05	0.025	0.0125
$\|f - s\|_\infty^{[-1,1]}$	$2.20 \cdot 10^{-2}$	$3.18 \cdot 10^{-3}$	$2.78 \cdot 10^{-4}$	$1.61 \cdot 10^{-5}$	$1.61 \cdot 10^{-}$

Analiza numeryczna

4. Aproksymacja

Rafał Nowak

Definicja

Wzór

$$\langle f, g \rangle := \int_a^b p(x) f(x) g(x)\, dx \tag{1}$$

definiuje **iloczyn skalarny** funkcji $f,\ g \in C_p[a,b]$. Sprawdza się że dla dowolnych f, g, h i $\alpha \in \mathbb{R}$

(i) $\langle f, f \rangle \geqslant 0$; $\langle f, f \rangle = 0 \ \Leftrightarrow f = 0$;
(ii) $\langle f, g \rangle = \langle g, f \rangle$,
(iii) $\langle \alpha f, g \rangle = \alpha \langle f, g \rangle$,
(iv) $\langle f + g, h \rangle = \langle f, h \rangle + \langle g, h \rangle$.

Twierdzenie (Ortogonalizacja Grama-Schmidta)

Dla dowolnego układu $f_1, f_2, \ldots, f_m$ *funkcji liniowo niezależnych układ* $g_1, g_2, \ldots, g_m$, *określony wzorami*

$$\begin{cases} g_1 := f_1, \\ g_k := f_k - \sum_{i=1}^{k-1} \frac{\langle f_k, g_i \rangle}{\langle g_i, g_i \rangle} g_i \quad (k = 2, 3, \ldots, m), \end{cases} \tag{2}$$

jest ortogonalny.

Wielomiany ortogonalne $\{\bar{P}_k\}$ nazwiemy **standardowymi**, jeśli dla każdego k wielomian $\bar{P}_k$ ma współczynnik 1 przy x^k. Zauważmy, że jeśli $\{P_k\}$ jest dowolnym ciągiem wielomianów ortogonalnych w tej przestrzeni i $P_k(x) = a_k x^k + \ldots$ $(k \geqslant 0)$, to $P_k = a_k \bar{P}_k$ $(k \geqslant 0)$.

Twierdzenie

Wielomiany ortogonalne $\{\bar{P}_k\}$ spełniają związek rekurencyjny

$$\bar{P}_0(x) = 1, \tag{3}$$
$$\bar{P}_1(x) = x - c_1, \tag{4}$$
$$\bar{P}_k(x) = (x - c_k)\bar{P}_{k-1}(x) - d_k \bar{P}_{k-2}(x) \qquad (k = 2, 3, \ldots), \tag{5}$$

gdzie

$$c_k = \langle x\bar{P}_{k-1}, \bar{P}_{k-1}\rangle / \langle \bar{P}_{k-1}, \bar{P}_{k-1}\rangle \qquad (k = 1, 2, \ldots), \tag{6}$$
$$d_k = \langle \bar{P}_{k-1}, \bar{P}_{k-1}\rangle / \langle \bar{P}_{k-2}, \bar{P}_{k-2}\rangle \qquad (k = 2, 3, \ldots). \tag{7}$$

Twierdzenie

Standardowe wielomiany ortogonalne względem parzystej funkcji wagowej $p(x)$ *w przedziale* $[-a, a]$ $(a > 0)$ *spełniają związek rekurencyjny*

$$\bar{P}_0(x) = 1, \tag{8}$$
$$\bar{P}_1(x) = x, \tag{9}$$
$$\bar{P}_k(x) = x\bar{P}_{k-1}(x) - d_k\bar{P}_{k-2}(x) \qquad (k = 2, 3, \ldots), \tag{10}$$

gdzie

$$d_k = \langle \bar{P}_{k-1}, \bar{P}_{k-1} \rangle / \langle \bar{P}_{k-2}, \bar{P}_{k-2} \rangle \qquad (k = 1, 2, \ldots). \tag{11}$$

Wniosek

Jeśli funkcja wagowa $p(x)$ *jest parzysta, to dla* $m = 0, 1, \ldots$ *wielomiany* $\bar{P}_{2m}(x)$ *są funkcjami parzystymi,* $\bar{P}_{2m+1}(x)$ *– funkcjami nieparzystymi.*

Twierdzenie

Jeśli ciąg $\{P_k\} \subset C_p[a,b]$ *jest ortogonalny, to* n*-ty wielomian optymalny* w_n^* *określony w zadaniu 1 istnieje, jest określony jednoznacznie i wyraża się wzorem*

$$w_n^* = \sum_{k=0}^{n} \frac{\langle f, P_k \rangle}{\langle P_k, P_k \rangle} P_k, \tag{12}$$

a n*-ty błąd aproksymacji optymalnej funkcji* f *jest równy*

$$||f - w_n^*||_2 = \sqrt{||f||_2^2 - \sum_{k=0}^{n} \frac{\langle f, P_k \rangle^2}{\langle P_k, P_k \rangle}}. \tag{13}$$

Twierdzenie

Niech $\{P_k\}$ będzie ciągiem wielomianów, określonych w następujący sposób rekurencyjny:

$$P_0(x) = \alpha_0, \qquad P_1(x) = (\alpha_1 x - \beta_1)P_0(x),$$
$$P_k(x) = (\alpha_k x - \beta_k)P_{k-1}(x) - \gamma_k P_{k-2}(x) \qquad (k = 2, 3, \ldots),$$

gdzie α_k, β_k, γ_k są danymi stałymi. Wartość wielomianu

$$s_n := a_0 P_0 + a_1 P_1 + \ldots + a_n P_n$$

o danych współczynnikach $a_0, a_1, \ldots, a_n$ można obliczyć stosując następujący uogólniony algorytm Clenshawa:
Obliczamy pomocnicze wielkości V_k $(k = 0, 1, \ldots, n+2)$ według wzorów

$$V_k = a_k + (\alpha_{k+1} x - \beta_{k+1})V_{k+1} - \gamma_{k+2} V_{k+2} \quad (k = n, n-1, \ldots, 0),$$

gdzie $V_{n+1} = 0$, $V_{n+2} = 0$. *Wynik: $s_n(x) = \alpha_0 V_0$.*

Definicja

Aproksymacją jednostajną nazywamy aproksymację w przestrzeni $C(T)$ funkcji rzeczywistych ciągłych na zbiorze zwartym (tj. domkniętym i ograniczonym) $T \subset \mathbb{R}1$, z normą

$$||f||_\infty \equiv ||f||_\infty^T := \max_{x \in T} |f(x)|,$$

zwaną *normą jednostajną* (albo *normą Czebyszewa*).

Twierdzenie

Dla dowolnej funkcji $f \in C(T)$ i dla dowolnego $n \in \mathbb{N}$ istnieje dokładnie jeden n-ty wielomian optymalny.

Twierdzenie (twierdzenie Czebyszewa o alternansie)

Niech T będzie dowolnym podzbiorem domkniętym przedziału $[a, b]$. Na to, by wielomian w_n był n-tym wielomianem optymalnym dla funkcji $f \in C(T)$ (tj. by dla każdego $u_n \in \Pi_n$ zachodziła nierówność $||f - w_n||_\infty^T \leqslant ||f - u_n||_\infty^T$) potrzeba i wystarcza, żeby istniały takie punkty $x_0, x_1, \ldots, x_{n+1} \in T$ ($x_0 < x_1 < \ldots < x_{n+1}$), że dla $e_n := f - w_n$ jest

$$e_n(x_k) = -e_n(x_{k-1}) \qquad (k = 0, 1, ..., n+1), \tag{14}$$

$$|e_n(x_j)| = ||e_n||_\infty^T \qquad (j = 0, 1, ..., n+1). \tag{15}$$

Zbiór punktów $x_0, x_1, \ldots, x_{n+1}$, w których różnica e_n przyjmuje wartość $||e_n||_\infty^T = \max\limits_{x \in T} |e_n(x)|$ z naprzemiennymi znakami, nazywamy (n-tym) alternansem funkcji f (związanym ze zbiorem T).

Przykład

Niech będzie $f(x) = \sqrt{1-x^2}$, $T = [a, b] = [-1, 1]$, $Y = \Pi_1$ (zauważmy, że $\dim Y = 2$), $g(x) \equiv \frac{1}{2}$. Wykresem różnicy $e := f - g$ w przedziale $[-1, 1]$ jest półokrąg o promieniu 1, przechodzący przez punkty $(-1, -\frac{1}{2})$, $(0, \frac{1}{2})$ i $(1, -\frac{1}{2})$. Norma tej różnicy w przedziale $[-1, 1]$ jest równa $\frac{1}{2}$, a w trzech punktach -1, 0, 1 różnica e ma na przemian wartości $-\frac{1}{2}$, $\frac{1}{2}$ i $-\frac{1}{2}$. Punkty te spełniają zatem równości (14), (15). Stąd i z Twierdzenia 2.3 wynika, że stała $\frac{1}{2}$ jest pierwszym wielomianem optymalnym dla funkcji $\sqrt{1-x^2}$ w przedziale $[-1, 1]$. Inaczej mówiąc, nie istnieje żaden wielomian postaci $a_0 + a_1 x$, o własności

$$||\sqrt{1-x^2} - (a_0 + a_1 x)||_\infty^{[-1,\,1]} < ||\sqrt{1-x^2} - \frac{1}{2}||_\infty^{[-1,\,1]}.$$

Twierdzenie

Niech s oznacza dowolną funkcję określoną zbiorze $T = \{x_0, x_1, \ldots, x_{n+1}\}$ (gdzie $x_0 < x_1 < \ldots < x_{n+1}$) i taką, że $s(x_k) = (-1)^k$ $(k = 0, 1, \ldots, n+1)$. n-ty wielomian optymalny dla funkcji f na zbiorze T wyraża się wzorem

$$\begin{aligned} w_n(x) &= d(x_0) + (d[x_0, x_1](x - x_0) + \\ &\quad \ldots + d[x_0, x_1, \ldots, x_n](x - x_0) \ldots (x - x_{n-1}), \end{aligned} \tag{16}$$

gdzie

$$d := f - \varepsilon s, \qquad \varepsilon = \frac{f[x_0, x_1, \ldots, x_{n+1}]}{s[x_0, x_1, \ldots, x_{n+1}]}. \tag{17}$$

Twierdzenie

n-tym wielomianem optymalnym dla jednomianu x^{n+1} w przedziale $[-1,1]$ jest wielomian $x^{n+1} - 2^{-n}T_{n+1}(x)$, a n-ty błąd aproksymacji optymalnej tej funkcji jest równy 2^{-n} . Spośród wszystkich wielomianów postaci $x^{n+1} + a_1 x^n + \ldots + a_{n+1}$ (z dowolnymi współczynnikami $a_1, a_2, \ldots, a_{n+1}$) najmniejszą normę $||\cdot||_\infty^{[-1,1]}$, równą 2^{-n}, ma wielomian $\tilde{T}_{n+1} := 2^{-n}T_{n+1}$.

Algorytm Remeza I

Dane są: zbiór domknięty T zawierający co najmniej $n+2$ punkty i funkcja f, która na T jest ciągła i nie jest tam wielomianem klasy Π_n. Niech $w_n^{(m)}$ będzie n-tym wielomianem optymalnym dla funkcji f na podzbiorze

$$D_m = \{x_{m0}, x_{m1}, \ldots, x_{m,n+1}\} \qquad (x_{m0} < x_{m1} < \ldots < x_{m,n+1})$$

zbioru T, określonym w następujący sposób:

1. Podzbiór D_0 jest dowolny.
2. Jeśli $E_n(f; D_{m-1}) = 0$ $(m \geqslant 1)$, to wybieramy taki punkt $\xi \in T \setminus D_{m-1}$, że $f(\xi) - w_n^{(m-1)}(\xi) \neq 0$ i przyjmujemy $D_m := D_{m-1} \setminus \{x_{m-1,j}\} \cup \{\xi\}$ (j – dowolne).

Algorytm Remeza II

3. Jeśli zbiór D_{m-1} nie tworzy $(n+2)$-punktowego alternansu dla funkcji $f - w^{(m-1)}$ (tzn. wielomian $w_n^{(m-1)}$ nie jest n-tym wielomianem optymalnym dla funkcji f) na zbiorze T, to podzbiór D_m wybieramy tak, żeby

   (R1) różnice $f(x_{mk}) - w_n^{(m-1)}(x_{mk})$ $(k = 0, 1, \ldots, n+1)$ są na przemian dodatnie i ujemne,

   (R2) $|f(x_{mk}) - w_n^{(m-1)}(x_{mk})| \geqslant E_n(f; D_{m-1})$ $(k = 0, 1, ..., n+1)$,

   (R3) $\max\limits_{0 \leqslant k \leqslant n+1} |f(x_{mk}) - w_n^{(m-1)}(x_{mk})| = ||f - w_n^{(m-1)}||_\infty^T$.

Jeśli powyższy algorytm określa ciąg nieskończony $\{w_n^{(m-1)}\}$, to jest on zbieżny do n-tego wielomianu optymalnego w_n dla funkcji f na zbiorze T. W przeciwnym razie ostatni skonstruowany element ciągu jest równy w_n.

Wielomiany prawie optymalne

Przykładem wielomianu prawie optymalnego jest wielomian

$$S_n(x) := \sum_{k=0}^{n}{}' a_k\, T_k(x), \quad a_k := \frac{2}{\pi}\int_{-1}^{1} f(x)T_k(x)(1-x^2)^{-1/2}\mathrm{d}x \quad (k \geqslant 0),$$

czyli n-ty wielomian optymalny w sensie aproksymacji w przestrzeni $L^2(-1,1,(1-x^2)^{-1/2})$, z normą

$$||f||_2 = \left(\int_{-1}^{1} f^2(x)(1-x^2)^{-1/2}\,dx\right).$$

Udowodniono, że dla dowolnej funkcji $f \in C[-1,1]$ zachodzi

$$||f - S_n||_\infty \leqslant K_n E_n(f), \tag{18}$$

gdzie

$$K_n := \frac{2n+2}{2n+1} + \frac{2}{\pi}\sum_{k=1}^{n} k^{-1}\,\mathrm{tg}\,\frac{k\pi}{2n+1} \sim \frac{4}{\pi^2}\ln n. \tag{19}$$

np. $K_5 = 2.961$, $K_{10} = 3.223$, $K_{20} = 3.494$, $K_{100} = 4.139$.

Wielomian interpolacyjny I_n z węzłami $t_{n+1,j}$, wyraża się wzorem

$$I_n(x) = \frac{2}{n+1} \sum_{i=0}^{n}{}' \left(\sum_{j=0}^{n} f(t_{n+1,j}) T_i(t_{n+1,j}) \right) T_i(x). \qquad (20)$$

Można wykazać, że

$$||f - I_n||_\infty \leqslant L_n E_n(f),$$

gdzie

$$L_n := 1 + \frac{1}{n+1} \sum_{k=0}^{n} \operatorname{tg} \frac{2k+1}{4n+4} \pi \sim \ln n.$$

Zatem czynnik L_n rośnie wolno wraz z n. Np. $L_5 = 3.104$, $L_{10} = 3.489$, $L_{20} = 3.901$, $L_{100} = 4.901$.

1. *Wielomiany Legendre'a* $\{P_k\}$, ortogonalne w przedziale $[-1,1]$ z wagą $p(x)\equiv 1$. Przyjmiemy, że
$$P_k(1)=1 \qquad (k=0,1,\dots).$$
Uwaga. Niech $c_k := \bar{P}_k(1)$. Wówczas $P_k = c_k^{-1}\bar{P}_k$ dla $k=0,1,\dots$.
Związek rekurencyjny :
$$\begin{aligned} &P_0(x)\equiv 1, \quad P_1(x)=x,\\ &P_k(x)=\frac{2k-1}{k}xP_{k-1}(x)-\frac{k-1}{k}P_{k-2}(x) \qquad (k\geqslant 2).\end{aligned}$$
Wielomiany standardowe Legendre'a $\{\bar{P}_k\}$:
$$\begin{aligned} &\bar{P}_0(x)\equiv 1, \quad \bar{P}_1(x)=x,\\ &\bar{P}_k(x)=x\bar{P}_{k-1}(x)-\frac{(k-1)^2}{(2k-1)(2k-3)}\bar{P}_{k-2}(x) \qquad (k\geqslant 2).\end{aligned}$$

2. *Wielomiany Czebyszewa I rodzaju* $\{T_k\}$, ortogonalne w przedziale $[-1,1]$ z wagą $p(x)=(1-x^2)^{-1/2}$:
$$\begin{aligned} &\int_{-1}^{1}(1-x^2)^{-1/2}T_k(x)T_l(x)\,dx=0 \qquad (k\neq l),\\ &\int_{-1}^{1}(1-x^2)^{-1/2}[T_k(x)]^2\,dx=\begin{cases}\frac{1}{2}\pi & (k\geqslant 1),\\ \pi & (k=0).\end{cases}\end{aligned}$$
Przyjmiemy, że
$$T_k(1)=1 \qquad (k=0,1,\dots).$$
Związek rekurencyjny :
$$\begin{aligned} &T_0(x)\equiv 1, \quad T_1(x)=x,\\ &T_k(x)=2xT_{k-1}(x)-T_{k-2}(x) \qquad (k\geqslant 2).\end{aligned}$$
Wielomiany standardowe Czebyszewa $\{\bar{T}_k\}$:
$$\begin{aligned} &\bar{T}_0(x)\equiv 1, \quad \bar{T}_1(x)=x,\\ &\bar{T}_k(x)=x\bar{T}_{k-1}(x)-\gamma_k\bar{T}_{k-2}(x) \qquad (k\geqslant 2),\end{aligned}$$
gdzie $\gamma_2=\frac{1}{2}$ i $\gamma_k=\frac{1}{4}$ dla $k>2$.

3. *Wielomiany Czebyszewa II rodzaju* $\{U_k\}$, ortogonalne w przedziale $[-1,1]$ z wagą $p(x)=(1-x^2)^{1/2}$. Przyjmiemy, że
$$U_k(1)=k+1 \qquad (k=0,1,\dots).$$
Związek rekurencyjny :
$$\begin{aligned} &U_0(x)\equiv 1, \quad U_1(x)=2x,\\ &U_k(x)=2xU_{k-1}(x)-U_{k-2}(x) \qquad (k\geqslant 2).\end{aligned}$$
Wielomiany standardowe Czebyszewa II rodzaju $\{\bar{U}_k\}$:
$$\begin{aligned} &\bar{U}_0(x)\equiv 1, \quad \bar{U}_1(x)=x,\\ &\bar{U}_k(x)=x\bar{U}_{k-1}(x)-\frac{1}{4}\bar{U}_{k-2}(x) \qquad (k\geqslant 2).\end{aligned}$$

4. *Wielomiany Gegenbauera* $\{C_k^\lambda\}$, ortogonalne w przedziale $[-1,1]$ z wagą $p(x)=(1-x^2)^{\lambda-1/2}$, $\lambda > -\frac{1}{2}$. Przyjmiemy, że

$$C_k^\lambda(1) = \frac{(2\lambda)(2\lambda+1)\cdots(2\lambda+k-1)}{k!} \qquad (k=0,1,\dots).$$

ZWIĄZEK REKURENCYJNY :

$$\begin{aligned} &C_0^\lambda(x) \equiv 1, \quad C_1^\lambda(x) = 2\lambda x, \\ &C_k^\lambda(x) = 2\frac{k+\lambda-1}{k} x C_{k-1}^\lambda(x) - \frac{k+2\lambda-3}{k} C_{k-2}^\lambda(x) \qquad (k \geqslant 2). \end{aligned}$$

5. *Wielomiany Jacobiego* $\{P_k^{(\alpha,\beta)}\}$, ortogonalne w przedziale $[-1,1]$ z wagą $p(x)=(1-x)^\alpha(1+x)^\beta$, α, $\beta > -1$.

$$P_0^{(\alpha,\beta)}(x) \equiv 1, \qquad P_1^{(\alpha,\beta)}(x) = \left(\frac{a+b}{2}+1\right)x + \frac{a-b}{2},$$

$$\begin{aligned} &2k(k+\alpha+\beta)(2k+\alpha+\beta-2)P_k^{(\alpha,\beta)}(x) \\ &= (2k+\alpha+\beta-1)\left[(2k+\alpha+\beta)(2k+\alpha+\beta-2)x + \alpha^2 - \beta^2\right] P_{k-1}^{(\alpha,\beta)}(x) \\ &\qquad - 2(k+\alpha-1)(k+\beta-1)(2k+\alpha+\beta)P_{k-2}^{(\alpha,\beta)}(x) \end{aligned}$$

6. *Wielomiany Laguerre'a* $\{L_k^\alpha\}$, ortogonalne w przedziale $[0,\infty)$ z wagą $p(x)=e^{-x}x^\alpha$. Przyjmiemy, że

$$L_k^\alpha(x) = \frac{(-1)^k}{k!}x^k + \dots \qquad (k \geqslant 0).$$

ZWIĄZEK REKURENCYJNY :

$$\begin{aligned} &L_0^\alpha(x) \equiv 1, \quad L_1^\alpha(x) = \alpha + 1 - x, \\ &L_k^\alpha(x) = \frac{2k+\alpha-1-x}{k} L_{k-1}^\alpha(x) - \frac{k+\alpha-1}{k} L_{k-2}^\alpha(x) \qquad (k \geqslant 2). \end{aligned}$$

Wielomiany standardowe Laguerre'a $\{\bar{L}_k^\alpha\}$:

$$\begin{aligned} &\bar{L}_0^\alpha(x) \equiv 1, \quad \bar{L}_1^\alpha(x) = x - \alpha - 1, \\ &\bar{L}_k^\alpha(x) = (x-2k-\alpha+1)\bar{L}_{k-1}^\alpha(x) - (k-1)(k+\alpha-1)\bar{L}_{k-2}^\alpha(x) \qquad (k \geqslant 2). \end{aligned}$$

7. *Wielomiany Hermite'a* $\{H_k\}$, ortogonalne na prostej $(-\infty,\infty)$ z wagą $p(x)=e^{-x^2}$. Przyjmiemy, że

$$H_k(x) = 2^k x^k + \dots \qquad (k \geqslant 0).$$

ZWIĄZEK REKURENCYJNY :

$$\begin{aligned} &H_0(x) \equiv 1, \quad H_1(x) = 2x, \\ &H_k(x) = 2xH_{k-1}(x) - 2(k-1)H_{k-2}(x) \qquad (k \geqslant 2). \end{aligned}$$

Wielomiany standardowe Hermite'a $\{\bar{H}_k\}$

$$\begin{aligned} &\bar{H}_0(x) \equiv 1, \quad \bar{H}_1(x) = x, \\ &\bar{H}_k(x) = x\bar{H}_{k-1}(x) - \frac{1}{2}(k-1)\bar{H}_{k-2}(x) \qquad (k \geqslant 2). \end{aligned}$$

Rafał Nowak

Notatka do wykładu analizy numerycznej

Kilka własności wielomianów Czebyszewa

Niech $\{T_n(x)\}_{n=0}^{\infty}$ oznacza ciąg wielomianów Czebyszewa I-go rodzaju:

$$T_0(x) \equiv 1, \qquad T_1(x) = x, \qquad T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x) \qquad (k \geqslant 2),$$

a $\{U_n(x)\}_{n=0}^{\infty}$ — ciąg wielomianów Czebyszewa II-go rodzaju:

$$U_0(x) \equiv 1, \qquad U_1(x) = 2x, \qquad U_k(x) = 2xU_{k-1}(x) - U_{k-2}(x) \qquad (k \geqslant 2).$$

Łatwo sprawdzić, że zera $t_k \equiv t_{n+1,k}$ wielomianu T_{n+1} wyrażają się wzorami

$$t_{n+1,k} \coloneqq \cos\frac{2k+1}{2n+2}\pi \qquad (k = 0, 1, \ldots, n).$$

Natomiast punkty ekstremalne $u_k \equiv u_{nk}$ wielomianu T_n wyrażają się wzorami

$$u_{nk} \coloneqq \cos(k\pi/n) \qquad (k = 0, 1, \ldots, n).$$

Lemat 1. *Wielomiany T_n są ortogonalne w sensie iloczynu skalarnego*

$$\langle f, g\rangle = \int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} f(x)g(x)\,\mathrm{d}x.$$

Zachodzi wzór

$$\langle T_i, T_j\rangle = \begin{cases} \pi, & i = j = 0, \\ \pi/2, & i = j \neq 0, \\ 0, & i \neq j. \end{cases} \tag{1}$$

Lemat 2. *Wielomiany U_n są ortogonalne w sensie iloczynu skalarnego*

$$\langle f, g\rangle = \int_{-1}^{1} \sqrt{1-x^2} f(x)g(x)\,\mathrm{d}x.$$

Zachodzi wzór

$$\langle U_i, U_j\rangle = \begin{cases} \pi/2, & i = j, \\ 0, & i \neq j. \end{cases} \tag{2}$$

Lemat 3. *Wielomiany $T_0, T_1, \ldots, T_n$ są ortogonalne w sensie dyskretnego iloczynu skalarnego*

$$\langle f, g\rangle = \sum_{k=0}^{n} f(t_k)g(t_k).$$

Zachodzi wzór

$$\langle T_i, T_j\rangle = \begin{cases} n+1, & i = j = 0, \\ (n+1)/2, & i = j \neq 0, \\ 0, & i \neq j. \end{cases} \tag{3}$$

Lemat 4. *Wielomiany $T_0, T_1, \ldots, T_n$ są ortogonalne w sensie dyskretnego iloczynu skalarnego*

$$\langle f, g\rangle = \sum_{k=0}^{n}{}'' f(u_k)g(u_k).$$

Zachodzi wzór

$$\langle T_i, T_j\rangle = \begin{cases} n, & i = j = 0 \text{ lub } i = j = n, \\ n/2, & i = j \neq 0, n \\ 0, & i \neq j. \end{cases} \tag{4}$$

Lemat 5. *Wielomian $I_n \in \Pi_n$ interpolujący funkcję f w węzłach t_k można zapisać w postaci*

$$I_n(x) = \sum_{i=0}^{n}{}' \alpha_i \, T_i(x), \tag{5}$$

gdzie

$$\alpha_i := \frac{2}{n+1} \sum_{j=0}^{n} f(t_j) T_i(t_j) \qquad (i = 0, 1, \ldots, n). \tag{6}$$

Ponadto, mamy

$$\int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} I_n(x) \, \mathrm{d}x = \frac{\pi}{n+1} \sum_{j=0}^{n} f(t_j).$$

Lemat 6. *Wielomian $J_n \in \Pi_n$ interpolujący funkcję f w węzłach u_k można zapisać wzorem*

$$J_n(x) = \sum_{j=0}^{n}{}'' \beta_j \, T_j(x), \tag{7}$$

gdzie

$$\beta_j := \frac{2}{n} \sum_{k=0}^{n}{}'' f(u_k) T_j(u_k) \qquad (j = 0, 1, \ldots, n). \tag{8}$$

Ponadto, mamy

$$\int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} J_n(x) \, \mathrm{d}x = \frac{\pi}{n} \sum_{j=0}^{n}{}'' f(u_j).$$

Analiza numeryczna

5. Kwadratury liniowe

Rafał Nowak

Rozważmy zbiór $\mathbb{F} \equiv \mathbb{F}[a,b]$ funkcji całkowalnych (ograniczonych i ciągłych prawie wszędzie w $[a,b]$). Funkcjonał liniowy I_p odwzorowujący $\mathbb{F}$ w zbiór liczb rzeczywistych $\mathbb{R}$ określamy następująco:

$$I_p(f) := \int_a^b p(x)f(x)\,dx \qquad (f \in \mathbb{F}), \tag{1}$$

gdzie *funkcja wagowa* $p \in \mathbb{F}$ jest nieujemna w $[a,b]$, znika w skończonej liczbie punktów tego przedziału.

Definicja

Kwadraturą liniową nazywamy funkcjonał Q_n określony następująco

$$Q_n(f) := \sum_{k=0}^{n} A_k f(x_k) \qquad (n > 0). \tag{2}$$

gdzie liczby $A_k \equiv A_k^{(n)}, (k = 0, 1, \ldots, n)$ – nazywamy *współczynnikami (wagami)*, a liczby $x_k \equiv x_k^{(n)}, (k = 0, 1, \ldots, n)$ – *węzłami kwadratury* Q_n. *Resztą kwadratury* Q_n nazywamy funkcjonał

$$R_n(f) := I_p(f) - Q_n(f).$$

Definicja

Mówimy, że kwadratura Q_n jest *rzędu* r, jeśli

(i) $R_n(f) = 0$ dla każdego wielomianu $f \in \Pi_{r-1}$,

(ii) istnieje taki wielomian $w \in \Pi_r \setminus \Pi_{r-1}$, że $R_n(w) \neq 0$.

Lemat

Jeśli kwadratura Q_n jest określona wzorem (2), to jej rząd nie przekracza $2n + 2$.

Rozważamy kwadratury

$$Q_n(f) := I_p(L_n[f]), \tag{3}$$

gdzie $L_n[f]$ jest wielomianem interpolacyjnym dla funkcji f w punktach x_k. Wprowadźmy oznaczenie na *wielomian węzłowy*:

$$\omega(x) := (x - x_0)(x - x_1)\dots(x - x_n).$$

Współczynniki kwadratury interpolacyjnej wyrażają się wzorem

$$A_k := I_p(\lambda_k) := \int_a^b p(x)\lambda_k(x)\mathrm{d}x = \int_a^b p(x) \prod_{j=0,\, j\neq k}^{n} \frac{x - x_j}{x_k - x_j}\mathrm{d}x, \tag{4}$$

a reszta – wzorem

$$\begin{aligned} R_n(f) &= \int_a^b p(x)\omega(x)f[x, x_0, x_1, \dots, x_n]\mathrm{d}x \\ &= \frac{1}{(n+1)!}\int_a^b p(x)\omega(x)f^{(n+1)}(\xi_x)\mathrm{d}x \qquad (k = 0, 1, \dots, n). \end{aligned}$$

Ostatni wzór zachodzi przy założeniu, że $f \in C^{n+1}[a, b]$.

Kwadratury Newtona to kwadratury interpolacyjne z węzłami równoodległymi

$$x_k \equiv x_k^{(n)} := a + kh \qquad (k = 0, 1, \ldots, n;\ h := (b-a)/n), \tag{5}$$

stosowane do obliczenia całki (1) dla $p \equiv 1$, czyli całki

$$I(f) := \int_a^b f(x)\mathrm{d}x.$$

Zatem

$$Q_n(f) := \sum_{k=0}^{n} A_k f(a + kh),$$

gdzie zgodnie z wzorem (4)

$$A_k \equiv A_k^{(n)} = I(\lambda_k) = \int_a^b \lambda_k(x)\mathrm{d}x = \frac{h(-1)^{n-k}}{k!(n-k)!} \int_0^n \prod_{j=0,\, j\neq k}^{n} (t-j)\, dt \qquad (k = 0, 1, \ldots$$

Twierdzenie

Reszta R_n kwadratury Newtona-Cotesa wyraża się wzorem

$$R_n(f) = \begin{cases} \frac{f^{(n+1)}(\xi)}{(n+1)!} \int_a^b \omega(x)\mathrm{d}x & (n = 1, 3, \ldots), \\ \frac{f^{(n+2)}(\eta)}{(n+2)!} \int_a^b x\,\omega(x)\mathrm{d}x & (n = 2, 4, \ldots), \end{cases} \tag{6}$$

gdzie ξ, $\eta \in (a, b)$.

Wypadek $n = 1, 2$

W wypadku $n = 1$ kwadratura Newtona-Cotesa nosi nazwę *wzoru trapezów*. Mamy $h = b - a$, $x_0 = a$, $x_1 = b$, $A_0 = A_1 = h/2$,

$$Q_1(f) := \frac{b-a}{2}[f(a) + f(b)], \tag{7}$$

$$R_1(f) = \frac{f''(\xi)}{2!}\int_a^b (x-a)(x-b)\mathrm{d}x = -\frac{(b-a)^3}{12}f''(\xi) = -\frac{h^3}{12}f''(\xi). \tag{8}$$

Dla $n = 2$ otrzymujemy *wzór Simpsona*:

$$h = (b-a)/2, \quad x_0 = a, \quad x_1 = (a+b)/2, \quad x_1 = b,$$
$$A_0 = A_2 = h/3, \quad A_1 = 4h/3,$$

$$Q_2(f) := \frac{b-a}{6}[f(a) + 4f((a+b)/2) + f(b)], \tag{9}$$

$$\begin{aligned} R_2(f) &= \frac{f^{(4)}(\eta)}{4!}\int_a^b x(x-a)(x-\frac{a+b}{2})(x-b)\mathrm{d}x \qquad (10)\\ &= -\frac{1}{90}\left(\frac{b-a}{2}\right)^5 f^{(4)}(\eta) = -\frac{h^5}{90}f^{(4)}(\eta). \end{aligned}$$

Złożony wzór trapezów

$$T_n(f) := h\sum_{k=0}^{n}{}''f(t_k) \qquad (h := \frac{b-a}{n},\ t_k := a + kh), \tag{11}$$

Reszta R_n^T jest równa

$$R_n^T(f) = -n\frac{h^3}{12}f''(\xi) = -(b-a)\frac{h^2}{12}f''(\xi). \tag{12}$$

dla pewnego $\xi \in (a, b)$.

Złożony wzór trapezów

$$T_n(f) := h\sum_{k=0}^{n}{}''f(t_k) \qquad (h := \frac{b-a}{n},\ t_k := a + kh), \tag{11}$$

Reszta R_n^T jest równa

$$R_n^T(f) = -n\frac{h^3}{12}f''(\xi) = -(b-a)\frac{h^2}{12}f''(\xi). \tag{12}$$

dla pewnego $\xi \in (a, b)$.

Twierdzenie (Euler-Maclaurin)

Jeśli funkcja f jest klasy $C^{2m+2}[a, b]$, to

$$R_n^T(f) = \frac{c_1}{n^2} + \frac{c_2}{n^4} + \ldots + \frac{c_m}{n^{2m}} + \frac{d(n)}{n^{2m+2}}, \tag{13}$$

gdzie

$$c_k := \frac{(b-a)^{2k}B_{2k}}{(2k)!}\left[f^{(2k-1)}(a) - f^{(2k-1)}(b)\right] \qquad (k = 1, 2, \ldots, m),$$

$d(n)$ jest ograniczoną funkcją zmiennej n: istnieje taka stała M, że dla każdego n zachodzi nierówność $|d(n)| \leqslant M$, a B_{2k} są tzw. liczbami Bernoulliego. (Np. $B_0 = 1$, $B_2 = 1/6$, $B_4 = -1/30$, $B_6 = 1/42$, $B_8 = -1/30$, $B_{10} = 5/66$).

Złożony wzór Simpsona

$$\begin{aligned} S_n(f) &:= \frac{h}{3}\{f(t_0)+4f(t_1)++2f(t_2)+4f(t_3)+2f(t_4)+\ldots \\ &\quad \ldots+2f(t_{2m-2})+4f(t_{2m-1})+f(t_{2m})\} \\ &= \frac{h}{3}\left\{2\sum_{k=0}^{m}{}''f(t_{2k})+4\sum_{k=1}^{m}f(t_{2k-1})\right\} \\ &= \frac{1}{3}(4T_n-T_m) \qquad (n=2m,\ h=\frac{b-a}{n}). \end{aligned} \tag{14}$$

Rzeszta $R_n^S(f)$ jest równa

$$R_n^S(f) = -m\frac{h^5}{90}f(\eta) = -(b-a)\frac{h^4}{180}f^{(4)}(\eta), \tag{15}$$

gdzie $\eta \in (a,b)$.

$$h_k := (b-a)/2^k,$$
$$x_i^{(k)} := a + ih_k \qquad (i = 0, 1, .., 2^k),$$
$$T_{0k} := T_{2^k}(f) = h_k \sum\nolimits_{i=0}^{2^k}{}'' f(x_i^{(k)}).$$

$$T_{mk} = \frac{4^m T_{m-1,k+1} - T_{m-1,k}}{4^m - 1} \qquad (k = 0, 1, \ldots;\ m = 1, 2, \ldots).$$

Tak więc, zaczynając od złożonych wzorów trapezów T_{00}, T_{01}, $T_{02}, \ldots$ budujemy trójkątną **tablicę Romberga** przybliżeń całki.

$$\begin{array}{cccccc}
T_{00} & & & & & \\
T_{01} & T_{10} & & & & \\
T_{02} & T_{11} & T_{20} & & & \\
T_{03} & T_{12} & T_{21} & T_{30} & & \\
\vdots & \vdots & \vdots & \vdots & \ddots & \\
T_{0m} & T_{1,m-1} & T_{2,m-2} & T_{3,m-3} & \ldots & T_{m0} \\
\vdots & \vdots & \vdots & \vdots & & \ddots
\end{array}$$

Można wykazać, że

1^o $T_{mk} = I - c_m^* h_k^{2m+2} - \ldots \qquad (k \geqslant 0;\ m \geqslant 1);$

2^o $T_{mk} = \sum_{j=0}^{2^{m+k}} A_j^{(m)} f(x_j^{(m+k)}) \qquad (k \geqslant 0;\ m \geqslant 1)$
(elementy k-tego wiersza tablicy Romberga zawierają te same węzły, co T_{0k}), gdzie $A_j^{(m)} > 0$ $(j = 0, 1, \ldots, 2^{m+k})$;

3^o dla każdej pary k, m T_{mk} jest sumą Riemanna;

4^o każdy z wzorów T_{m0}, T_{m1}, $\ldots$ jest kwadraturą rzędu $2m + 2$;

5^o (wniosek z 2^o, 3^o, 4^o i z twierdzenia o zbieżności ciągu kwadratur o dodatnich współczynnikach) niech $I = I(f)$, gdzie f jest dowolną funkcją ciągłą w $[a, b]$; wówczas

$$\lim_{k \to \infty} T_{mk} = I \qquad (m = 1, 2, \ldots);$$

$$\lim_{m \to \infty} T_{mk} = I \qquad (k = 0, 1, \ldots).$$

Przykład

$$I = \int_1^3 \frac{dx}{x} = \ln 3 = 1.098612\ldots$$

$$\begin{array}{lllll}
T_{00} = 1.333333 & & & & \\
T_{01} = 1.166667 & T_{10} = 1.111111 & & & \\
T_{02} = 1.116667 & T_{11} = 1.100000 & T_{20} = 1.099259 & & \\
T_{03} = 1.103211 & T_{12} = 1.098726 & T_{21} = 1.098641 & T_{30} = 1.098631 & \\
T_{04} = 1.099768 & T_{13} = 1.098620 & T_{22} = 1.098613 & T_{31} = 1.098613 & T_{40} = 1.098613 \\
T_{05} = 1.098902 & T_{14} = 1.098613 & T_{23} = 1.098613 & T_{32} = 1.098613 & T_{41} = 1.098613 \\
T_{06} = 1.098685 & T_{15} = 1.098613 & & & \\
T_{07} = 1.098630 & T_{16} = 1.098612 & & &
\end{array}$$

Kwadratury Gaussa

$$I_p(f) := \int_a^b p(x) f(x)\,\mathrm{d}x \qquad (f \in \mathbb{F})$$

$$Q_n(f) := \sum_{k=0}^{n} A_k^{(n)} f(x_k^{(n)})$$

$$A_k^{(n)} = \int_a^b p(x)\lambda_k(x)\,\mathrm{d}x, \qquad \lambda_k(x) = \frac{\omega(x)}{\omega'(x_k)(x - x_k)}$$

$$\omega(x) = \bar{P}_{n+1}(x) = (x - x_0)(x - x_1)\cdots(x - x_n)$$

Kwadratury Gaussa

$$I_p(f) := \int_a^b p(x) f(x)\,\mathrm{d}x \qquad (f \in \mathbb{F})$$

$$Q_n(f) := \sum_{k=0}^{n} A_k^{(n)} f(x_k^{(n)})$$

$$A_k^{(n)} = \int_a^b p(x)\lambda_k(x)\,\mathrm{d}x, \qquad \lambda_k(x) = \frac{\omega(x)}{\omega'(x_k)(x - x_k)}$$

$$\omega(x) = \bar{P}_{n+1}(x) = (x - x_0)(x - x_1)\cdots(x - x_n)$$

Lemat

$$A_k^{(n)} = \frac{a_{n+1}}{a_n} \cdot \frac{\|P_n\|^2}{P'_{n+1}(x_k)\,P_n(x_k)}$$

$$P_j(x) = a_j x^k + \ldots$$

Kwadratury Gaussa

Lemat

Współczynniki kwadratury Gaussa są dodatnie, tzn.

$$A_k^{(n)} > 0, \qquad k = 0, 1, \ldots, n.$$

Kwadratury Gaussa

Lemat

Współczynniki kwadratury Gaussa są dodatnie, tzn.

$$A_k^{(n)} > 0, \qquad k = 0, 1, \ldots, n.$$

Lemat

Jeśli $f \in C^{2n+2}[a,b]$. to reszta kwadratury Gaussa wyraża się wzorem

$$R_n(f) := I_p(f) - Q_n(f) = \frac{f^{(2n+2)}(\xi)}{(2n+2)!\, a_{n+1}^2} \int_a^b p(x)[P_{n+1}(x)]^2 \,\mathrm{d}x.$$

Lemat

Jeśli $f \in C[a,b]$, to $\lim_{n\to\infty} Q_n(f) = I_p(f)$.

Kwadratury Gaussa

Związek rekurencyjny dla wielomianów ortogonalnych P_k

$$P_k(x) = (b_k x + c_k)P_{k-1}(x) - d_k P_{k-2}(x),$$

można zapisać macierzowo

$$x\boldsymbol{p}(x) = A\boldsymbol{p}(x) + \frac{1}{b_{n+1}} P_{n+1}(x)\boldsymbol{e}_{n+1}$$

$$\boldsymbol{p}(x) = [P_0(x), P_1(x), \ldots, P_n(x)]^T, \quad \boldsymbol{e}_{n+1} = [0, 0, \ldots, 0, 1]^T \in \mathbb{R}^{n+1}$$

$$A = \{a_{ij}\}, \quad a_{ij} = \begin{cases} -c_i/b_i, & i = j, \\ d_i/b_i, & i = j+1, \\ 1/b_i, & i = j-1, \\ 0, & \text{w p.p.} \end{cases}$$

Kwadratury Gaussa

Lemat

Węzły $x_k^{(n)}$ kwadratury Czebyszewa są wartościami własnymi macierzy A, a współczynniki wyrażają się wzorami

$$A_k^{(n)} = [\boldsymbol{v}_1^{(k)}]^2 \int_a^b p(x)\,\mathrm{d}x,$$

gdzie $\boldsymbol{v}^{(k)}$ jest wektorem własnym odpowiadającym wartości $x_k^{(n)}$.

Kwadratury Gaussa

Lemat

Węzły $x_k^{(n)}$ kwadratury Czebyszewa są wartościami własnymi macierzy A, a współczynniki wyrażają się wzorami

$$A_k^{(n)} = [\boldsymbol{v}_1^{(k)}]^2 \int_a^b p(x)\,\mathrm{d}x,$$

gdzie $\boldsymbol{v}^{(k)}$ jest wektorem własnym odpowiadającym wartości $x_k^{(n)}$.

Lemat

Macierz A jest podobna do macierzy symetrycznej trójprzekątniowej $T = \{t_{ij}\}$, gdzie

$$t_{ii} = -\frac{c_i}{b_i}, \qquad t_{i+1,i} = t_{i,i+1} = \left(\frac{d_{i+1}}{b_i b_{i+1}}\right)^{1/2}.$$

Kwadratury Gaussa-Legendre'a

$$P_k(x) = \frac{2k-1}{k} x P_{k-1}(x) - \frac{k-1}{k} P_{k-2}(x)$$

Kwadratury Gaussa-Legendre'a

$$P_k(x) = \frac{2k-1}{k} x P_{k-1}(x) - \frac{k-1}{k} P_{k-2}(x)$$

$$t_{ii} = 0, \qquad t_{i,i+1} = t_{i+1,i} = (4 - 1/i^2)^{-1/2}$$

Kwadratury Gaussa-Legendre'a

$$P_k(x) = \frac{2k-1}{k} x P_{k-1}(x) - \frac{k-1}{k} P_{k-2}(x)$$

$$t_{ii} = 0, \qquad t_{i,i+1} = t_{i+1,i} = (4 - 1/i^2)^{-1/2}$$

Implementacja kwadratury GL w Juli

```
# Funkcja oblicza całkę \int_{-1}^{1} f(x) dx
# za pomocą (n+1)-punktowej  kwadratury Gaussa-Legendre'a
function GaussLegendre(f,n)
  β = 1 ./ sqrt(4.0 -(1:n).^(-2.0));
  T = SymTridiagonal(zeros(n+1),β);     # Macierz podobna do macierzy
  x,V = eig(T);                         # x - wartości własne (pierwia
  w = 2.0*vec(V[1,:]).^2;               # w - współczynniki kwadratury
  return dot(w,f(x));
end;

# Przykładowe użycie
GaussLegendre(x -> sqrt(1-x.^2), 100);
```

Kwadratura Gaussa-Czebyszewa

$$I_p(f) = \int_{-1}^{1} p(x) f(x)\,\mathrm{d}x, \qquad p(x) = \frac{1}{\sqrt{1-x^2}}$$

$$Q_n^{GC}(f) := \int_{-1}^{1} p(x) I_n(x)\,\mathrm{d}x, \quad I_n(t_k) = f(t_k), \quad t_k = \cos\left(\frac{2k+1}{2n+2}\pi\right)$$

Lemat

$$I_n(x) = {\sum_{i=0}^{n}}' \alpha_i T_i(x), \qquad \alpha_i = \frac{2}{n+1} \sum_{k=0}^{n} f(t_k) T_i(t_k)$$

Kwadratura Gaussa-Czebyszewa

$$I_p(f) = \int_{-1}^{1} p(x) f(x)\,\mathrm{d}x, \qquad p(x) = \frac{1}{\sqrt{1-x^2}}$$

$$Q_n^{GC}(f) := \int_{-1}^{1} p(x) I_n(x)\,\mathrm{d}x, \quad I_n(t_k) = f(t_k), \quad t_k = \cos\left(\frac{2k+1}{2n+2}\pi\right)$$

Lemat

$$I_n(x) = {\sum_{i=0}^{n}}' \alpha_i T_i(x), \qquad \alpha_i = \frac{2}{n+1} \sum_{k=0}^{n} f(t_k) T_i(t_k)$$

Wniosek

$$Q_n^{GC}(f) = \sum_{k=0}^{n} A_k f(t_k), \qquad A_k = \frac{\pi}{n+1}.$$

Kwadratura Lobatto

$$I_p(f) = \int_{-1}^{1} p(x) f(x)\,\mathrm{d}x, \qquad p(x) = \frac{1}{\sqrt{1-x^2}}$$

$$Q_n^L(f) := \int_{-1}^{1} p(x) J_n(x)\,\mathrm{d}x, \quad J_n(u_k) = f(u_k), \quad u_k = \cos\left(\frac{k}{n}\pi\right)$$

Kwadratura Lobatto

$$I_p(f) = \int_{-1}^{1} p(x) f(x)\,\mathrm{d}x, \qquad p(x) = \frac{1}{\sqrt{1-x^2}}$$

$$Q_n^L(f) := \int_{-1}^{1} p(x) J_n(x)\,\mathrm{d}x, \quad J_n(u_k) = f(u_k), \quad u_k = \cos\left(\frac{k}{n}\pi\right)$$

Lemat

$$J_n(x) = \sum_{j=0}^{n}{}'' \beta_j T_j(x), \qquad \beta_j = \frac{2}{n} \sum_{k=0}^{n}{}'' f(u_k) T_j(u_k)$$

Kwadratura Lobatto

$$I_p(f) = \int_{-1}^{1} p(x) f(x)\,\mathrm{d}x, \qquad p(x) = \frac{1}{\sqrt{1-x^2}}$$

$$Q_n^L(f) := \int_{-1}^{1} p(x) J_n(x)\,\mathrm{d}x, \quad J_n(u_k) = f(u_k), \quad u_k = \cos\left(\frac{k}{n}\pi\right)$$

Lemat

$$J_n(x) = \sum_{j=0}^{n}{}'' \beta_j T_j(x), \qquad \beta_j = \frac{2}{n} \sum_{k=0}^{n}{}'' f(u_k) T_j(u_k)$$

Wniosek

$$Q_n^L(f) = \sum_{k=0}^{n}{}'' A_k f(u_k), \qquad A_k = \frac{\pi}{n}.$$

Kwadratura Lobatto

Lemat

Wzór

$$\int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} f(x)\,\mathrm{d}x \approx \frac{\pi}{n} \sum_{k=0}^{n}{}'' f(u_k)$$

jest dokładny dla $f \in \Pi_{2n-1}$.

Implementacja kwadratury Gaussa-Czebyszewa i Lobatto w Juli

```
function GaussChebyshev(f,n)
  x = cos(collect(1:2:2*n+1)*π/(2*n+2)); # węzły kwadratury Gaussa-
  return π/(n+1)*sum(f(x));
end;

function Lobatto(f,n)
  x = cos(collect(0:n)*π/n); # węzły kwadratury Lobatto (punkty eks
  y = f(x); y[1] *= 0.5; y[n+1] *= 0.5;
  return π/n*sum(y);
end;


# Przykładowe użycie
GaussChebyshev(x -> (1-x.^2), 1000);
Lobatto(x -> (1-x.^2), 1000);
```

Szereg Czebyszewa

Niech $f \in C^1[-1,1]$. Wówczas funkcję f można rozwinąć w szereg Czebyszewa

$$f(x) = {\sum_{k=0}^{\infty}}' a_k T_k(x) \qquad (-1 \leqslant x \leqslant 1),$$

$$a_k \equiv a_k[f] := \frac{2}{\pi} \int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} f(x) T_k(x)\,\mathrm{d}x.$$

Szereg Czebyszewa

Niech $f \in C^1[-1,1]$. Wówczas funkcję f można rozwinąć w szereg Czebyszewa

$$f(x) = {\sum_{k=0}^{\infty}}' a_k T_k(x) \qquad (-1 \leqslant x \leqslant 1),$$

$$a_k \equiv a_k[f] := \frac{2}{\pi} \int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} f(x) T_k(x)\,\mathrm{d}x.$$

Współczynniki Czebyszewa a_k obliczamy w sposób przybliżony

$$a_k \approx \alpha_k^n := \frac{2}{\pi} Q_n^{GC}(f \cdot T_k) = \frac{2}{n+1} \sum_{j=0}^{n} f(t_j) T_k(t_j),$$

$$a_k \approx \beta_k^n := \frac{2}{\pi} Q_n^{L}(f \cdot T_k) = \frac{2}{n} {\sum_{j=0}^{n}}'' f(u_j) T_k(u_j),$$

Szereg Czebyszewa

Lemat

Jeśli $f = {\sum_{k=0}^{\infty}}' a_k T_k$, *to zachodzą wzory*

$$\beta_k^n = a_k + \sum_{i=1}^{\infty} (a_{2in-k} + a_{2in+k}) \qquad (k = 0, 1, \ldots, n-1), \tag{16}$$

$$\beta_n^n = a_n + \sum_{i=1}^{\infty} a_{(2i+1)n}. \tag{17}$$

Wzory te mówią, że jeśli ciąg $\{a_k\}$ dąży dostatecznie regularnie do zera, to równość przybliżona $\beta_k^n \approx a_k$ jest obarczona niewielkim błędem, wyrażającym się przez współczynniki $a_m[f]$ dla $m > n$:

$$\begin{aligned}
\beta_0^n &= a_0 + 2a_{2n} + 2a_{4n} + \ldots && \approx a_0 + 2a_{2n},\\
\beta_{n-1}^n &= a_{n-1} + a_{n+1} + a_{3n-1} + \ldots && \approx a_{n-1} + a_{n+1},\\
&\ldots\ldots\ldots\ldots\ldots\ldots\\
\beta_{n-2}^n &= a_{n-2} + a_{n+2} + a_{3n-2} + \ldots && \approx a_{n-2} + a_{n+2},\\
\beta_n^n &= a_n + a_{3n} + a_{5n} + \ldots && \approx a_n + a_{3n}.
\end{aligned}$$

Kwadratura Clenshawa-Curtisa

$$I(f) = \int_{-1}^{1} f(x)\,\mathrm{d}x, \qquad f(x) = {\sum_{k=0}^{\infty}}' a_k T_k(x)$$

$$a_k = \frac{2}{\pi} \int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} f(x) T_k(x)\,\mathrm{d}x, \qquad k \geqslant 0$$

Kwadratura Clenshawa-Curtisa

$$I(f) = \int_{-1}^{1} f(x)\,\mathrm{d}x, \qquad f(x) = {\sum_{k=0}^{\infty}}' a_k T_k(x)$$

$$a_k = \frac{2}{\pi} \int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} f(x) T_k(x)\,\mathrm{d}x, \qquad k \geqslant 0$$

$$Q_n(f) := \int_{-1}^{1} \left({\sum_{k=0}^{n}}' a_k T_k(x) \right) \mathrm{d}x$$

Kwadratura Clenshawa-Curtisa

$$I(f) = \int_{-1}^{1} f(x)\,\mathrm{d}x, \qquad f(x) = {\sum_{k=0}^{\infty}}' a_k T_k(x)$$

$$a_k = \frac{2}{\pi} \int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} f(x) T_k(x)\,\mathrm{d}x, \qquad k \geqslant 0$$

$$Q_n(f) := \int_{-1}^{1} \left({\sum_{k=0}^{n}}' a_k T_k(x) \right) \mathrm{d}x = 2 {\sum_{k=0}^{\lfloor n/2 \rfloor}}' \frac{a_{2k}}{1-4k^2}$$

Interpolacyjna kwadratura Clenshawa-Curtisa

$$I(f) = \int_{-1}^{1} f(x)\,\mathrm{d}x$$

Interpolacyjna kwadratura Clenshawa-Curtisa

$$I(f) = \int_{-1}^{1} f(x)\,\mathrm{d}x$$

$$Q_n^{CC}(f) := \int_{-1}^{1} J_n(x)\,\mathrm{d}x, \qquad J_n = {\sum_{j=0}^{n}}'' \beta_j T_j$$

Interpolacyjna kwadratura Clenshawa-Curtisa

$$I(f) = \int_{-1}^{1} f(x)\,\mathrm{d}x$$

$$Q_n^{CC}(f) := \int_{-1}^{1} J_n(x)\,\mathrm{d}x, \qquad J_n = \sum_{j=0}^{n}{}'' \beta_j T_j$$

$$Q_n^{CC}(f) = \sum_{k=0}^{n}{}'' A_k^{(n)} f(u_k), \qquad A_k^{(n)} := \frac{4}{n} \sum_{j=0}^{n/2}{}'' \frac{T_{2j}(u_k)}{1-4j^2}$$

Uwaga: w powyższym wzorze symbol $\sum_{j=0}^{n/2}{}''$ oznacza sumę, w której pierwszy składnik jest pomnożony przez $1/2$, zaś ostatni jest mnożony przez $1/2$ tylko, gdy n jest parzyste.

Interpolacyjna kwadratura Clenshawa-Curtisa

Z własności wielomianów Czebyszewa

$$T_j(u_k) = T_k(u_j),$$

mamy

$$\beta_j = \frac{2}{n} \sum_{k=0}^{n}{}'' f(u_k) T_j(u_k) = \frac{2}{n} \sum_{k=0}^{n}{}'' f(u_k) T_k(u_j),$$

a więc przybliżenia β_j współczynników Czebyszyszewa $a_j[f]$ można obliczać za pomocą algorytmu Clenshawa.

Interpolacyjna kwadratura Clenshawa-Curtisa

Z własności wielomianów Czebyszewa

$$T_j(u_k) = T_k(u_j),$$

mamy

$$\beta_j = \frac{2}{n}\sum_{k=0}^{n}{}''f(u_k)T_j(u_k) = \frac{2}{n}\sum_{k=0}^{n}{}''f(u_k)T_k(u_j),$$

a więc przybliżenia β_j współczynników Czebyszyszewa $a_j[f]$ można obliczać za pomocą algorytmu Clenshawa. Obliczając raz wektor współczynników $[f(u_0), f(u_1), \ldots, f(u_n)]^T$ wywołujemy $n+1$ razy algorytm Clenshawa, mianowicie z parametrem $t = u_0, u_1, \ldots, u_n$, w celu obliczenia współczynników $\beta_0, \beta_1, \ldots, \beta_n$.

Interpolacyjna kwadratura Clenshawa-Curtisa

Z własności wielomianów Czebyszewa

$$T_j(u_k) = T_k(u_j),$$

mamy

$$\beta_j = \frac{2}{n}\sum_{k=0}^{n}{}''f(u_k)T_j(u_k) = \frac{2}{n}\sum_{k=0}^{n}{}''f(u_k)T_k(u_j),$$

a więc przybliżenia β_j współczynników Czebyszyszewa $a_j[f]$ można obliczać za pomocą algorytmu Clenshawa. Obliczając raz wektor współczynników $[f(u_0), f(u_1), \ldots, f(u_n)]^T$ wywołujemy $n+1$ razy algorytm Clenshawa, mianowicie z parametrem $t = u_0, u_1, \ldots, u_n$, w celu obliczenia współczynników $\beta_0, \beta_1, \ldots, \beta_n$.Ostatecznie otrzymujemy metodę o złożoności $\mathcal{O}(n^2)$.

Interpolacyjna kwadratura Clenshawa-Curtisa

Wszystkie współczynniki

$$\beta_j = \frac{2}{n} \sum_{k=0}^{n}{}'' f(u_k) \cos(kj\pi/n), \qquad j = 0, 1, \ldots, n$$

można wyznaczyć w czasie $\mathcal{O}(n \log n)$ za pomocą szybkiej transformaty Fouriera (FFT).

Interpolacyjna kwadratura Clenshawa-Curtisa

Wszystkie współczynniki

$$\beta_j = \frac{2}{n} \sum_{k=0}^{n}{}'' f(u_k) \cos(kj\pi/n), \qquad j = 0, 1, \ldots, n$$

można wyznaczyć w czasie $\mathcal{O}(n \log n)$ za pomocą szybkiej transformaty Fouriera (FFT).

- algorytm FFT powstał w 1965 roku.

Interpolacyjna kwadratura Clenshawa-Curtisa

Wszystkie współczynniki

$$\beta_j = \frac{2}{n} \sum_{k=0}^{n}{}'' f(u_k) \cos(kj\pi/n), \qquad j = 0, 1, \ldots, n$$

można wyznaczyć w czasie $\mathcal{O}(n \log n)$ za pomocą szybkiej transformaty Fouriera (FFT).

- algorytm FFT powstał w 1965 roku.
- Clenshaw i Curtis opublikowali swoją metodę w 1960 roku.

Interpolacyjna kwadratura Clenshawa-Curtisa

Wszystkie współczynniki

$$\beta_j = \frac{2}{n} \sum_{k=0}^{n}{}'' f(u_k) \cos(kj\pi/n), \qquad j = 0, 1, \ldots, n$$

można wyznaczyć w czasie $\mathcal{O}(n \log n)$ za pomocą szybkiej transformaty Fouriera (FFT).

- algorytm FFT powstał w 1965 roku.
- Clenshaw i Curtis opublikowali swoją metodę w 1960 roku.
- Uwaga: Jeśli $n = 2^j$, to wszystkie węzły $u_k = \cos(k\pi/n)$ można wyznaczyć obliczając tylko $\mathcal{O}(\log n)$ wywołań funkcji cosinus.

Szybka transformata Fouriera

Problem $\boldsymbol{y} = DCT(N, \boldsymbol{x})$

Dane: $\boldsymbol{x} = [x_0, x_1, \ldots, x_{N-1}]$
Wynik: $\boldsymbol{y} = [y_0, y_1, \ldots, y_{N-1}]$, gdzie

$$y_k = \sum_{j=0}^{N-1} x_j \theta_N^{jk} \qquad (\theta_N = \exp(2\pi \mathrm{i}/N), \quad \mathrm{i} = \sqrt{-1}).$$

Szybka transformata Fouriera

Problem $\boldsymbol{y} = DCT(N, \boldsymbol{x})$

Dane: $\boldsymbol{x} = [x_0, x_1, \ldots, x_{N-1}]$
Wynik: $\boldsymbol{y} = [y_0, y_1, \ldots, y_{N-1}]$, gdzie

$$y_k = \sum_{j=0}^{N-1} x_j \theta_N^{jk} \qquad (\theta_N = \exp(2\pi \mathrm{i}/N), \quad \mathrm{i} = \sqrt{-1}).$$

Zakładamy, że $N = 2M$. Mamy

$$y_k = \sum_{j=0}^{N-1} x_j \theta_N^{jk} = \sum_{j=0}^{M-1} x_j \theta_N^{jk} + \sum_{j=M}^{N-1} x_j \theta_N^{jk}$$

$$= \sum_{j=0}^{M-1} x_j \theta_N^{jk} + \sum_{j=0}^{M-1} x_{M+j} \theta_N^{(M+j)k} = \sum_{j=0}^{M-1} x_j \theta_N^{jk} + \sum_{j=0}^{M-1} (-1)^k x_{M+j} \theta_N^{jk}$$

$$= \sum_{j=0}^{M-1} \left(x_j + (-1)^k x_{M+j} \right) \theta_N^{jk} \qquad (18)$$

$$y_k = \sum_{j=0}^{M-1} \left(x_j + (-1)^k x_{M+j} \right) \theta_N^{jk}, \qquad k = 0, 1, \ldots, N-1$$

$$y_k = \sum_{j=0}^{M-1} \left(x_j + (-1)^k x_{M+j} \right) \theta_N^{jk}, \qquad k = 0, 1, \ldots, N-1$$

Dla parzystych i nieparzystych wskaźników otrzymujemy wzory

$$y_{2k} = \sum_{j=0}^{M-1} \left(x_j + x_{M+j} \right) \theta_M^{jk}, \qquad k = 0, 1, \ldots, M-1,$$

$$y_{2k+1} = \sum_{j=0}^{M-1} \left(x_j - x_{M+j} \right) \theta_N^{j} \theta_M^{jk}, \qquad k = 0, 1, \ldots, M-1$$

Stąd widzimy, że wektory $[y_0, y_2, \ldots, y_{N-2}] = DCT(M, \bar{\boldsymbol{x}})$, $[y_1, y_3, \ldots, y_{N-1}] = DCT(M, \tilde{\boldsymbol{x}})$ możemy obliczyć rekurencyjnie rozwiązując dwa podproblemy DCT o rozmiarze $M = N/2$.

$$y_k = \sum_{j=0}^{M-1} \left(x_j + (-1)^k x_{M+j} \right) \theta_N^{jk}, \qquad k = 0, 1, \ldots, N-1$$

Dla parzystych i nieparzystych wskaźników otrzymujemy wzory

$$y_{2k} = \sum_{j=0}^{M-1} \left(x_j + x_{M+j} \right) \theta_M^{jk}, \qquad k = 0, 1, \ldots, M-1,$$

$$y_{2k+1} = \sum_{j=0}^{M-1} \left(x_j - x_{M+j} \right) \theta_N^{j} \theta_M^{jk}, \qquad k = 0, 1, \ldots, M-1$$

Stąd widzimy, że wektory $[y_0, y_2, \ldots, y_{N-2}] = DCT(M, \bar{\boldsymbol{x}})$, $[y_1, y_3, \ldots, y_{N-1}] = DCT(M, \tilde{\boldsymbol{x}})$ możemy obliczyć rekurencyjnie rozwiązując dwa podproblemy DCT o rozmiarze $M = N/2$.
Dla uzasadnienia złożoności obliczeniowej algorytmu FFT, wystarczy skorzystać z tego, że rozwiązaniem związku rekurencyjnego

$$T(N) = 2T(N/2) + \mathcal{O}(N)$$

jest $T(N) = \mathcal{O}(N \log N)$.

Szybka transformata Fouriera
Implementacja w Juli

```
# Dyskretna transformacja cosinusowa
# Implementacja naiwna, wprost ze wzroru.
# Złożoność: O(n^2)
function slowFFT(x)
  N = length(x);
  θ = [exp(Complex(0,2π*j/N)) for j=0:N-1];
  y = Complex(0,0)*zeros(N);
  for k=0:N-1
    y[k+1] = dot(x,θ.^k);
  end;
  return y;
end;

# Dyskretna transformacja cosinusowa
# Implementacja za pomocą "dziel i zwyciężaj"
# Złożoność: O(n log n)
function myFFT(x) # N = 2^k
  N = length(x);
  if (N==1) return x; end;
  M = Int( floor(N/2) );
  xL = x[1:M];
  xR = x[M+1:N];
  ye = myFFT(xL+xR);
  yo = myFFT((xL-xR).*[exp(Complex(0,2π*j/N)) for j=0:M-1]);
  y = Complex(0,0)*zeros(N);
  y[1:2:N-1] = ye;
  y[2:2:N] = yo;
  return y;
end;
```

Rafał Nowak

Notatki do wykładu analizy numerycznej

Kwadratury

18 grudnia 2018

Rozważmy zbiór $\mathbb{F} \equiv \mathbb{F}[a,b]$ funkcji całkowalnych (ograniczonych i ciągłych prawie wszędzie w $[a,b]$). Funkcjonał liniowy I_p odwzorowujący $\mathbb{F}$ w zbiór liczb rzeczywistych $\mathbf{R}^1$ określamy następująco:

$$I_p(f) := \int_a^b p(x)f(x)\,dx \qquad (f \in \mathbb{F}), \tag{1}$$

gdzie *funkcja wagowa* $p \in \mathbb{F}$ jest nieujemna w $[a,b]$, znika w skończonej liczbie punktów tego przedziału.

Definicja 1. Kwadraturą liniową *nazywamy funkcjonał* Q_n *określony następująco*

$$Q_n(f) := \sum_{k=0}^{n} A_k f(x_k) \qquad (n > 0). \tag{2}$$

gdzie liczby

$$A_k \equiv A_k^{(n)} \qquad (k = 0, 1, \ldots, n)$$

– *nazywamy* współczynnikami (wagami)*, a liczby*

$$x_k \equiv x_k^{(n)} \qquad (k = 0, 1, \ldots, n)$$

– węzłami kwadratury Q_n. Resztą kwadratury Q_n *nazywamy funkcjonał*

$$R_n(f) \coloneqq I_p(f) - Q_n(f).$$

Definicja 2. *Mówimy, że kwadratura* Q_n *jest* rzędu r*, jeśli*
(i) $R_n(f) = 0$ *dla każdego wielomianu* $f \in \Pi_{r-1}$ *i*
(ii) istnieje taki wielomian $w \in \Pi_r \setminus \Pi_{r-1}$*, że* $R_n(w) \neq 0$.

Lemat 1. *Jeśli kwadratura* Q_n *jest określona wzorem (2), to jej rząd nie przekracza* $2n + 2$.

Kwadratury interpolacyjne

Rozważamy kwadratury

$$Q_n(f) \coloneqq I_p(L_n[f]), \tag{3}$$

gdzie $L_n[f]$ jest wielomianem interpolacyjnym dla funkcji f w punktach x_k. Wprowadźmy oznaczenie na *wielomian węzłowy*:

$$\omega(x) \coloneqq (x - x_0)(x - x_1)\ldots(x - x_n).$$

Współczynniki kwadratury interpolacyjnej wyrażają się wzorem

$$A_k := I_p(\lambda_k) := \int_a^b p(x)\lambda_k(x)\,\mathrm{d}x = \int_a^b p(x) \prod_{j=0,\, j\neq k}^{n} \frac{x - x_j}{x_k - x_j}\,\mathrm{d}x \qquad (k = 0, 1, \ldots, n), \tag{4}$$

a reszta – wzorem

$$\begin{aligned} R_n(f) &= \int_a^b p(x)\omega(x) f[x, x_0, x_1, \ldots, x_n]\,\mathrm{d}x \\ &= \frac{1}{(n+1)!}\int_a^b p(x)\omega(x) f^{(n+1)}(\xi_x)\,\mathrm{d}x. \end{aligned} \tag{5}$$

Ostatni wzór zachodzi przy założeniu, że $f \in C^{n+1}[a,\, b]$.

Twierdzenie 1 (Jacobi)**.** *Kwadratura Q_n określona wzorem (2) ma rząd $\geqslant n+1+m$ $(1 \leqslant m \leqslant n+1)$ wtedy i tylko wtedy, gdy spełnione są następujące dwa warunki:*
(i) Q_n jest kwadraturą interpolacyjną,
(ii) dla każdego wielomianu $u \in \Pi_{m-1}$ zachodzi równość $I_p(\omega u) = 0$.

Kwadratury Newtona-Cotesa

Kwadratury Newtona to kwadratury interpolacyjne z węzłami równoodległymi

$$x_k \equiv x_k^{(n)} := a + kh \qquad (k = 0, 1, \ldots, n;\ h := (b-a)/n), \tag{6}$$

stosowane do obliczenia całki (1) dla $p \equiv 1$, czyli całki

$$I(f) := \int_a^b f(x)\,\mathrm{d}x.$$

Zatem

$$Q_n(f) := \sum_{k=0}^{n} A_k f(a+kh),$$

gdzie zgodnie z wzorem (4)

$$A_k \equiv A_k^{(n)} = I(\lambda_k) = \int_a^b \lambda_k(x)\,\mathrm{d}x = \frac{h(-1)^{n-k}}{k!(n-k)!}\int_0^n \prod_{j=0,\, j\neq k}^{n} (t-j)\,dt \qquad (k = 0, 1, \ldots, n).$$

Twierdzenie 2. *Reszta R_n kwadratury Newtona-Cotesa wyraża się wzorem*

$$R_n(f) = \begin{cases} \frac{f^{(n+1)}(\xi)}{(n+1)!}\int_a^b \omega(x)\,\mathrm{d}x & (n = 1, 3, \ldots), \\ \frac{f^{(n+2)}(\eta)}{(n+2)!}\int_a^b x\,\omega(x)\,\mathrm{d}x & (n = 2, 4, \ldots), \end{cases} \tag{7}$$

gdzie ξ, $\eta \in (a, b)$.

W wypadku $n = 1$ kwadratura Newtona-Cotesa nosi nazwę *wzoru trapezów*. Mamy $h = b - a$, $x_0 = a$, $x_1 = b$, $A_0 = A_1 = h/2$,

$$Q_1(f) := \frac{b-a}{2}[f(a) + f(b)], \tag{8}$$

$$R_1(f) = \frac{f''(\xi)}{2!}\int_a^b (x-a)(x-b)\,\mathrm{d}x = -\frac{(b-a)^3}{12}f''(\xi) = -\frac{h^3}{12}f''(\xi). \tag{9}$$

Dla $n = 2$ otrzymujemy *wzór Simpsona*:

$$h = (b-a)/2, \quad x_0 = a, \quad x_1 = (a+b)/2, \quad x_1 = b,$$
$$A_0 = A_2 = h/3, \quad A_1 = 4h/3,$$

$$Q_2(f) := \frac{b-a}{6}[f(a) + 4f((a+b)/2) + f(b)], \tag{10}$$

$$R_2(f) = \frac{f^{(4)}(\eta)}{4!}\int_a^b x(x-a)(x-\frac{a+b}{2})(x-b)\,\mathrm{d}x \tag{11}$$

$$= -\frac{1}{90}\left(\frac{b-a}{2}\right)^5 f^{(4)}(\eta) = -\frac{h^5}{90}f^{(4)}(\eta).$$

Złożone kwadratury Newtona-Cotesa

Złożony wzór trapezów

$$T_n(f) := h\sum_{k=0}^{n}{}'' f(t_k), \tag{12}$$

Reszta R_n^T jest równa

$$R_n^T(f) = -n\frac{h^3}{12}f''(\xi) = -(b-a)\frac{h^2}{12}f''(\xi). \tag{13}$$

dla pewnego $\xi \in (a,b)$.

Złożony wzór Simpsona

$$\begin{aligned} S_n(f) &:= \frac{h}{3}\{f(t_0)+4f(t_1)++2f(t_2)+4f(t_3)+2f(t_4)+\ldots \\ &\quad \ldots+2f(t_{2m-2})+4f(t_{2m-1})+f(t_{2m})\} \\ &= \frac{h}{3}\left\{2\sum_{k=0}^{m}{}''f(t_{2k})+4\sum_{k=1}^{m}f(t_{2k-1})\right\} \\ &= \frac{1}{3}(4T_n - T_m) \qquad (n=2m). \end{aligned} \tag{14}$$

Rzeszta $R_n^S(f)$ jest równa

$$R_n^S(f) = -m\frac{h^5}{90}f(\eta) = -(b-a)\frac{h^4}{180}f^{(4)}(\eta), \tag{15}$$

gdzie $\eta \in (a,b)$.

Metoda Romberga

$$\begin{gathered} h_k := (b-a)/2^k, \\ x_i^{(k)} := a + ih_k \qquad (i=0,1,..,2^k), \\ T_{0k} := T_{2^k}(f) = h_k \textstyle\sum_{i=0}^{2^k}{}''f(x_i^{(k)}). \end{gathered}$$

$$T_{mk} = \frac{4^m T_{m-1,k+1} - T_{m-1,k}}{4^m - 1} \qquad (k=0,1,\ldots;\ m=1,2,\ldots).$$

Tak więc, zaczynając od złożonych wzorów trapezów $T_{00}, T_{01}, T_{02}, \ldots$ budujemy trójkątną **tablicę Romberga** przybliżeń całki (zob. tablicę 1).

Tabela 1. Tablica Romberga

T_{00}					
T_{01}	T_{10}				
T_{02}	T_{11}	T_{20}			
T_{03}	T_{12}	T_{21}	T_{30}		
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	
T_{0m}	$T_{1,m-1}$	$T_{2,m-2}$	$T_{3,m-3}$	$\ldots$	T_{m0}
$\vdots$	$\vdots$	$\vdots$	$\vdots$		$\ddots$

Można wykazać, że

1^o $T_{mk} = I - c_m^* h_k^{2m+2} - \ldots \qquad (k \geqslant 0;\ m \geqslant 1)$;

2^o $T_{mk} = \sum_{j=0}^{2^{m+k}} A_j^{(m)} f(x_j^{(m+k)}) \qquad (k \geqslant 0;\ m \geqslant 1)$

(elementy k-tego wiersza tablicy Romberga zawierają te same węzły, co T_{0k}), gdzie $A_j^{(m)} > 0$ $(j = 0,1,\ldots,2^{m+k})$;

3^o dla każdej pary k, m T_{mk} jest sumą Riemanna;

4^o każdy z wzorów $T_{m0}, T_{m1}, \ldots$ jest kwadraturą rzędu $2m+2$;

5^o (wniosek z 2^o, 3^o, 4^o i z twierdzenia o zbieżności ciągu kwadratur o dodatnich współczynnikach) niech $I = I(f)$, gdzie f jest dowolną funkcją ciągłą w $[a,b]$; wówczas

$$\lim_{k\to\infty} T_{mk} = I \qquad (m=1,2,\ldots);$$
$$\lim_{m\to\infty} T_{mk} = I \qquad (k=0,1,\ldots).$$

Rafał Nowak

Analiza numeryczna

12 stycznia 2021

1. Metody Rungego-Kutty

Ogólna postać s-etapowej metody Rungego-Kutty, dla parametrów a_i, c_i, b_{ij} $(i, j = 1, \ldots, s)$, dana jest wzorem

$$y_{n+1} = y_n + \Phi_f(h; x_n, y_n, y_{n+1}), \tag{1}$$

gdzie

$$\Phi_f(h; x_n, y_n, y_{n+1}) = h \sum_{i=1}^{s} c_i k_i,$$

zaś

$$k_i \equiv k_i(h; x_n, y_y) = f\Big(x_n + a_i h, y_n + h \sum_{j=1}^{s} b_{ij} k_j\Big).$$

Wygodna forma zapisu tej metody dana jest w tabeli, w której często pomija się wyrazy zerowe:

$$\begin{array}{c|ccc} a_1 & b_{11} & \cdots & b_{1s} \\ a_2 & b_{21} & \cdots & b_{2s} \\ \vdots & & \vdots & \\ a_s & b_{s1} & \cdots & b_{ss} \\ \hline & c_1 & \cdots & c_s \end{array}$$

Definicja 1. Powiemy, że metoda opisana wzorem (1) jest *rzędu* p, jeśli po podstawieniu w nim $y_n \coloneqq y(x_n)$ otrzymujemy y_{n+1} o własności

$$y_{n+1} - y(x_n + h) = \mathcal{O}(h^{p+1}).$$

Przykłady

— metody rzędu pierwszego
  — metoda jawna Eulera

$$\begin{array}{c|c} 0 & \\ \hline & 1 \end{array}$$

  — metoda niejawna Eulera (wzór wsteczny)

$$\begin{array}{c|c} 1 & 1 \\ \hline & 1 \end{array}$$

  — ulepszony (jawny) wzór wsteczny Eulera

$$\begin{array}{c|cc} 0 & & \\ 1 & 1 & \\ \hline & 0 & 1 \end{array}$$

— metody rzędu drugiego
  — wzór trapezów

0		
1	0	1
	$\frac{1}{2}$	$\frac{1}{2}$

  — jawny wzór trapezów (metoda Heuna)

0		
1	1	
	$\frac{1}{2}$	$\frac{1}{2}$

  — jawna metoda punktu środkowego

0		
$\frac{1}{2}$	$\frac{1}{2}$	
	0	1

— metody rzędu trzeciego
  — metoda Heuna (3)

0			
$\frac{1}{3}$	$\frac{1}{3}$		
$\frac{2}{3}$	0	$\frac{2}{3}$	
	$\frac{1}{4}$	0	$\frac{3}{4}$

  — metoda Rungego-Kutty (3)

0				
$\frac{1}{2}$	$\frac{1}{2}$			
1	0	1		
1	0	0	1	
	$\frac{1}{6}$	$\frac{2}{3}$	0	$\frac{1}{6}$

— metody rzędu czwartego
  — metoda Rungego-Kutty

0				
$\frac{1}{2}$	$\frac{1}{2}$			
$\frac{1}{2}$	0	$\frac{1}{2}$		
1	0	0	1	
	$\frac{1}{6}$	$\frac{2}{6}$	$\frac{2}{6}$	$\frac{1}{6}$

  — reguła 3/8

0				
$\frac{1}{3}$	$\frac{1}{3}$			
$\frac{2}{3}$	$-\frac{1}{3}$	1		
1	1	-1	1	
	$\frac{1}{8}$	$\frac{3}{8}$	$\frac{3}{8}$	$\frac{1}{8}$

  — metoda Mersona (4,5)

0					
1/3	1/3				
1/3	1/6	1/6			
1/2	1/8	0	3/8		
1	1/2	0	$-3/2$	2	
	1/6	0	0	2/3	1/6

— metoda Scratona (4,5)

0					
2/9	2/9				
1/3	1/12	1/4			
3/4	69/128	$-243/128$	270/128		
0.9	$-9 * 0.0345$	$9 * 0.2025$	$-9 * 0.1224$	$9 * 0.0544$	
	17/162	0	81/170	32/135	250/1377

1.1. Analityczne badanie rzędu metody

Aby sprawdzić jakiego rzędu jest dana metoda należy rozwinąć w szereg Taylora wartości $y(x_n+h)$ oraz y_{n+1}, a następnie porównać współczynniki stojące przy kolejnych potęgach h. Do tego przydatne okazują się następujące wzory Taylora:

$$y(x_n + h) = y(x_n) + hy'(x_n) + \frac{1}{2!}h^2y''(x_n) + \frac{1}{3!}h^3y'''(x_n) + \dots$$

Ponieważ $y'(x) = f(x, y(x))$, więc

$$y(x_n + h) = y_n + hf + \frac{1}{2!}h^2(f_x + f_yf) + \frac{1}{3!}h^3\left[f_{xx} + f_{xy}f + (f_{xy} + f_{yy}f)f + f_y(f_x + f_yf)\right] + \dots,$$

gdzie $f \equiv f(x_n, y_n)$, $f_x \equiv \frac{\partial}{\partial x}f(x_n, y_n)$, $f_y \equiv \frac{\partial}{\partial y}f(x_n, y_n)$, $f_{xx} \equiv \frac{\partial^2}{\partial x^2}f(x_n, y_n)$, ...

Z drugiej strony, aby znaleźć rozwinięcie wartości y_{n+1}, należy rozwinąć wszystkie wartości k_i we wzorze (1). Do tego celu stosujemy wzór Taylora dla funkcji wielu zmiennych:

$$f(x + ah, y + bh) = f(x, y) + \mathrm{d}f(x, y)(ah, bh) + \frac{1}{2!}\mathrm{d}^2f(x, y)(ah, bh) + \frac{1}{3!}\mathrm{d}^3f(x, y)(ah, bh) + \dots,$$

gdzie $\mathrm{d}f(x, y)(ah, bh)$ oznacza różniczkę zupełną funkcji f w punkcie (x, y) dla argumentu (ah, bh):

$$\mathrm{d}^jf(x, y)(ah, bh) = \left(\frac{\partial}{\partial x}ah + \frac{\partial}{\partial y}bh\right)^j f(x, y).$$

Na przykład

$$f(x_n + ah, y_n + bh) = f + h(af_x + bf_y) + \frac{1}{2}h^2(a^2f_{xx} + 2abf_{xy} + b^2f_{yy}) + \dots,$$

gdzie symbole $f, f_x, f_y, \dots$ mają takie samo znaczenie, jak wcześniej.

Analiza numeryczna

Algebra liniowa

Rafał Nowak

Macierzą nazywamy prostokątną tablicę $m \times n$ liczb rzeczywistych, ustawionych w m wierszach i n kolumnach:

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix}. \tag{1}$$

Sumą macierzy $A = [a_{ij}] \in \mathbb{R}^{m\times n}$ i $B = [b_{ij}] \in \mathbb{R}^{m\times n}$ jest macierz $C = [c_{ij}]$ tego samego rozmiaru:

$$C = A + B, \qquad c_{ij} = a_{ij} + b_{ij}.$$

Iloczyn macierzy $A = [a_{ij}]$ przez liczbę α jest macierz

$$B = \alpha A, \qquad b_{ij} = \alpha a_{ij}.$$

Iloczyn macierzy A i B jest określony tylko wtedy, gdy liczba kolumn macierzy A jest równa liczbie wierszy macierzy B. Iloczyn $C = AB$ macierzy $A = [a_{ij}] \in \mathbb{R}^{m\times n}$ i $B = [b_{ij}] \in \mathbb{R}^{n\times p}$ jest macierzą

$$C = [c_{ij}], \qquad c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}.$$

Mnożenie macierzy jest łączne i rozdzielne względem dodawania:

$$A(BC) = (AB)C, \qquad A(B + C) = AB + AC,$$

jednak nie jest przemienne.

Wektory $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k \in \mathbb{R}^n$ są **liniowo niezależne**, jeśli żaden z nich nie jest liniową kombinacją pozostałych, tj. jeśli

$$\sum_{i=1}^{k} \alpha_i \boldsymbol{x}_i = \boldsymbol{\theta} \quad \Rightarrow \quad \alpha_i = 0 \quad (i = 1, 2, \ldots, k).$$

Wektory $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k \in \mathbb{R}^n$ są **liniowo niezależne**, jeśli żaden z nich nie jest liniową kombinacją pozostałych, tj. jeśli

$$\sum_{i=1}^{k} \alpha_i \boldsymbol{x}_i = \boldsymbol{\theta} \quad \Rightarrow \quad \alpha_i = 0 \quad (i = 1, 2, \ldots, k).$$

Rząd macierzy A jest liczbą jej liniowo niezależnych kolumn (wierszy). Macierz kwadratowa $A \in \mathbb{R}^{n \times n}$ jest **nieosobliwa** wtedy i tylko wtedy, gdy jej rząd jest równy n. Wówczas istnieje **macierz odwrotna** oznaczana symbolem A^{-1}, o własności

$$A^{-1}A = AA^{-1} = I.$$

Jeśli A i B są nieosobliwe, a iloczyn AB jest określony, to

$$(AB)^{-1} = B^{-1}A^{-1},$$

tj. macierz odwrotna do iloczynu macierzy jest równa iloczynowi odwrotności czynników w odwrotnym porządku. Macierz jest nieosobliwa wtedy i tylko wtedy, gdy $\det A \neq 0$.

Definicja

Normą wektorową nazywamy nieujemną funkcję rzeczywistą $\|\cdot\|$, określoną w przestrzeni $\mathbb{R}^n$, o następujących własnościach:

$$\bigwedge_{\boldsymbol{x}\in\mathbb{R}^n\setminus\{\theta\}} \{\|\boldsymbol{x}\| > 0\};$$

$$\bigwedge_{\boldsymbol{x}\in\mathbb{R}^n}\ \bigwedge_{\alpha\in\mathbf{R}} \{\|\alpha\boldsymbol{x}\| = |\alpha|\|\boldsymbol{x}\|\};$$

$$\bigwedge_{\boldsymbol{x},\boldsymbol{y}\in\mathbb{R}^n} \{\|\boldsymbol{x}+\boldsymbol{y}\| \leqslant \|\boldsymbol{x}\| + \|\boldsymbol{y}\|\}.$$

Definicja

Normą wektorową nazywamy nieujemną funkcję rzeczywistą $\|\cdot\|$, określoną w przestrzeni $\mathbb{R}^n$, o następujących własnościach:

$$\bigwedge_{\boldsymbol{x}\in\mathbb{R}^n\setminus\{\theta\}} \{\|\boldsymbol{x}\| > 0\};$$

$$\bigwedge_{\boldsymbol{x}\in\mathbb{R}^n}\bigwedge_{\alpha\in\mathbf{R}} \{\|\alpha\boldsymbol{x}\| = |\alpha|\|\boldsymbol{x}\|\};$$

$$\bigwedge_{\boldsymbol{x},\boldsymbol{y}\in\mathbb{R}^n} \{\|\boldsymbol{x}+\boldsymbol{y}\| \leqslant \|\boldsymbol{x}\| + \|\boldsymbol{y}\|\}.$$

Najczęściej używane są trzy normy wektorów, zwane *normami Hoeldera*, definiowane następująco ($\boldsymbol{x} = (x_1, \ldots, x_n)$):

$$\|\boldsymbol{x}\|_1 := |x_1| + \ldots, |x_n|,$$
$$\|\boldsymbol{x}\|_2 := (x_1^2 + \ldots, x_n^2)^{1/2},$$
$$\|\boldsymbol{x}\|_\infty := \max_{1<k<n} |x_k|.$$

Definicja

Normą macierzy nazywamy nieujemną funkcję rzeczywistą $\|\cdot\|$, określoną w przestrzeni liniowej $\mathbb{R}^{n\times n}$ wszystkich macierzy kwadratowych stopnia n, o następujących własnościach:

$$\bigwedge_{A\in\mathbb{R}^{n\times n}\setminus\{\Theta\}} \{\|A\|>0\};$$

$$\bigwedge_{A\in\mathbb{R}^{n\times n}} \bigwedge_{\alpha\in\mathbf{R}} \{\|\alpha A\|=|\alpha|\|A\|\};$$

$$\bigwedge_{A\,,B\in\mathbb{R}^{n\times n}} \{\|A+B\|\leqslant\|A\|+\|B\|\};$$

$$\bigwedge_{A\,,B\in\mathbb{R}^{n\times n}} \{\|AB\|\leqslant\|A\|\|B\|\}.$$

Przyjęcie jakiejś normy wektora pozwala na wprowadzenie odpowiedniej normy macierzy, zdefiniowanej równością

$$\|A\| := \sup_{\boldsymbol{x}\in\mathbb{R}^n\setminus\{\theta\}} \frac{\|A\mathbf{x}\|}{\|\boldsymbol{x}\|}.$$

Mówimy, że ta norma macierzy jest indukowana przez normę wektora. Można sprawdzić, że

$$\|A\|_1 = \max_{1\leqslant j\leqslant n} \sum_{i=1}^{n} |a_{ij}|,$$

$$\|A\|_\infty = \max_{1\leqslant i\leqslant n} \sum_{j=1}^{n} |a_{ij}|,$$

$$\|A\|_2 = \left(\text{największa wartość własna macierzy} A^T A\right)^{1/2},$$

gdzie A^T oznacza macierz transponowaną do A. Normę $\|\cdot\|_2$ nazywamy niekiedy normą spektralną. Zauważmy, że $\|I\| = 1$ dla dowolnej normy macierzy, indukowanych przez normy wektorów. Symbol I oznacza macierz jednostkową, $I = \operatorname{diag}(1,\ldots,1)$.

Definicja

Będziemy mówili, że normy macierzy i wektora są *zgodne*, jeśli

$$\bigwedge_{A\in\mathbb{R}^{n\times n}} \bigwedge_{\boldsymbol{x}\in\mathbb{R}^n} \left\{\|A\boldsymbol{x}\| \leqslant \|A\|\|\boldsymbol{x}\|\right\}.$$

Definicja (Macierz trójkątna dolna)

Macierz $L = [l_{ij}] \in \mathbb{R}^{n \times n}$ nazywamy *trójkątną dolną*, jeśli $l_{ij} = 0$ dla $i < j$:

$$L = \begin{bmatrix} l_{11} & & & & & \\ l_{21} & l_{22} & & & & \\ l_{31} & l_{32} & l_{33} & & & \\ \cdots & \cdots & \cdots & \cdots & \cdots & \\ l_{n1} & l_{n2} & l_{n3} & \cdots & l_{n,n-1} & l_{nn} \end{bmatrix}.$$

Zbiór wszystkich macierzy trójkątnych dolnych stopnia n oznaczamy symbolem $\mathbb{L}_n$. Podzbiór zbioru $\mathbb{L}_n$, zawierający macierze o elementach $l_{ii} = 1$ $(i = 1, 2, \ldots, n)$, oznaczamy symbolem $\mathbb{L}_n^{(1)}$.

Definicja (Macierz trójkątna górna)

Macierz $U = [u_{ij}] \in \mathbb{R}^{n \times n}$ nazywamy *trójkątną górną*, jeśli $u_{ij} = 0$ dla $i > j$:

$$U = \begin{bmatrix} u_{11} & u_{12} & u_{13} & \dots & u_{1n} \\ & u_{22} & u_{23} & \dots & u_{2n} \\ & & u_{33} & \dots & u_{3n} \\ & & & \dots\dots\dots & \\ & & & & u_{nn} \end{bmatrix}.$$

Zbiór wszystkich macierzy trójkątnych górnych stopnia n oznaczamy symbolem $\mathbb{U}_n$.

$$U\boldsymbol{x} = \boldsymbol{b}, \qquad \mathbb{U}_n \ni U = [u_{ij}];$$

$$\sum_{j=i}^{n} u_{ij} x_j = b_i \qquad (i = 1, 2, \ldots, n).$$

$$x_i = \frac{1}{u_{ii}} \left\{ b_i - \sum_{j=i+1}^{n} u_{ij} x_j \right\} \qquad (i = n, n-1, \ldots, 2, 1).$$

$$U\boldsymbol{x} = \boldsymbol{b}, \qquad \mathbb{U}_n \ni U = [u_{ij}];$$

$$\sum_{j=i}^{n} u_{ij} x_j = b_i \qquad (i = 1, 2, \ldots, n).$$

$$x_i = \frac{1}{u_{ii}} \left\{ b_i - \sum_{j=i+1}^{n} u_{ij} x_j \right\} \qquad (i = n, n-1, \ldots, 2, 1).$$

$$L\boldsymbol{x} = \boldsymbol{b}, \qquad \mathbb{L}_n \ni L = [l_{ij}];$$

$$\sum_{j=1}^{i} l_{ij} x_j = b_i \qquad (i = 1, 2, \ldots, n).$$

$$x_i = \frac{1}{l_{ii}} \left\{ b_i - \sum_{j=1}^{i-1} l_{ij} x_j \right\} \qquad (i = 1, 2, \ldots, n)$$

Rozkłady trójkątne

Twierdzenie (Rozkład trójkątny macierzy)

Niech macierz $A = [a_{ij}] \in \mathbb{R}^{n \times n}$ *będzie taka, że*

$$\det A_k \neq 0 \qquad (k = 1, 2, \ldots, n),$$

gdzie

$$A_k := \begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1k} \\ a_{21} & a_{22} & \ldots & a_{2k} \\ \ldots & \ldots & \ldots & \ldots \\ a_{k1} & a_{k2} & \ldots & a_{kk} \end{bmatrix} \qquad (k = 1, 2, \ldots, n).$$

Wówczas istnieje dokładnie jedna para macierzy $L \in \mathbb{L}_n^{(1)}$, $U \in \mathbb{U}_n$, *spełniających równość* $LU = A$. *Ponadto,* $\det A = u_{11} u_{22} \cdots u_{nn}$.

Faktoryzacja LU

Dla $i = 1, 2, \ldots, n$ obliczamy

$$u_{ij} = a_{ij} - \sum_{k=1}^{i-1} l_{ik} u_{kj} \qquad (j = i, i+1, \ldots, n),$$

$$l_{ji} = \left(a_{ji} - \sum_{k=1}^{i-1} l_{jk} u_{ki} \right) / u_{ii} \qquad (j = i+1, i+2, \ldots, n).$$

Jeśli znany jest rozkład macierzy układu równań

$$A\boldsymbol{x} = \boldsymbol{b}$$

na czynniki trójkątne:

$$A = LU,$$

to zadanie sprowadza się do rozwiązania kolejno dwóch układów o macierzy trójkątnej:

$$\begin{cases} L\boldsymbol{y} = \boldsymbol{b}, \\ U\boldsymbol{x} = \boldsymbol{y} \end{cases}$$

Eliminacja Gaussa

Rozważmy układ równań

$$A\boldsymbol{x} = \boldsymbol{b} \qquad (A \in \mathbb{R}^{n\times n}, \boldsymbol{b} \in \mathbb{R}^n). \tag{2}$$

Eliminacja Gaussa

Rozważmy układ równań

$$A\boldsymbol{x} = \boldsymbol{b} \qquad (A \in \mathbb{R}^{n\times n}, \boldsymbol{b} \in \mathbb{R}^n). \tag{2}$$

Niech

$$A^{(1)} = [a_{ij}^{(1)}] := A, \quad \boldsymbol{b^{(1)}} = [b_1^{(1)}, \ldots, b_n^{(1)}]^T := \boldsymbol{b}.$$

Eliminacja Gaussa

Rozważmy układ równań

$$A\boldsymbol{x} = \boldsymbol{b} \qquad (A \in \mathbb{R}^{n\times n}, \boldsymbol{b} \in \mathbb{R}^n). \tag{2}$$

Niech

$$A^{(1)} = [a_{ij}^{(1)}] := A, \quad \boldsymbol{b^{(1)}} = [b_1^{(1)}, \ldots, b_n^{(1)}]^T := \boldsymbol{b}.$$

Układ (2) przekształcamy w sposób równoważny do układu

$$\sum_{j=k}^{n} a_{kj}^{(k)} x_j = b_k^{(k)} \qquad (k = 1, 2, \ldots, n), \tag{3}$$

gdzie $a_k^{(k)} \neq 0$ $(k = 1, 2, \ldots, n)$ oraz

$$\begin{cases} a_{ij}^{(k)} = a_{ij}^{(k-1)} + m_{i,k-1} a_{k-1,j}^{(k-1)}, \\ b_i^{(k)} = b_i^{(k-1)} + m_{i,k-1} b_{k-1}^{(k-1)}, \\ m_{i,k-1} = -\dfrac{a_{i,k-1}^{(k-1)}}{a_{k-1,k-1}^{(k-1)}} \end{cases} \qquad \begin{array}{l} (k = 2, 3, \ldots, n; \\ \quad i, j = k, k+1, \ldots, n). \end{array}$$

Współczynniki $a_{kk}^{(k)}$ $(k = 1, 2, \ldots, n)$ nazywamy elementami głównymi.

Współczynniki $a_{kk}^{(k)}$ $(k = 1, 2, \ldots, n)$ nazywamy elementami głównymi. Z układu (3) łatwo otrzymać rozwiązanie wg wzorów

$$x_k = \left(b_k^{(k)} - \sum_{i=k+1}^{n} a_{kj}^{(k)} x_j \right) \Big/ a_{kk}^{(k)} \qquad (k = n, n-1, \ldots, 2, 1) \qquad (4)$$

Zdefiniujmy stałą g_n, zwaną **współczynnikiem wzrostu**, wzorem

$$g_n := \max_{1 \leqslant i,j,r \leqslant n} |a_{ij}^{(r)}| / \max_{1 \leqslant i,j \leqslant n} |a_{ij}|.$$

Dla eliminacji z częściowym wyborem elem. gł. zachodzi nierówność

$$g_n \leqslant 2^{n-1}.$$

Zdefiniujmy stałą g_n, zwaną **współczynnikiem wzrostu**, wzorem

$$g_n := \max_{1 \leqslant i,j,r \leqslant n} |a_{ij}^{(r)}| / \max_{1 \leqslant i,j \leqslant n} |a_{ij}|.$$

Dla eliminacji z częściowym wyborem elem. gł. zachodzi nierówność

$$g_n \leqslant 2^{n-1}.$$

Najlepsze ze znanych oszacowań dla pełnego wyboru elem. gł.,

$$g_n \leqslant \varphi(n),$$

gdzie $\varphi(n) := n^{1/2} \left(2^1 3^{1/2} 4^{1/3} \dots n^{1/(n-1)}\right)^{1/2} < 1.8 n^{1/2+\log n/4}$,
wydaje się natomiast poważnie zawyżone. Np. $\varphi(10) = 19$, $\varphi(50) = 530$, $\varphi(100) = 3570$,

Twierdzenie

Niech $\tilde{\boldsymbol{x}}$ oznacza rozwiązanie układu $A\boldsymbol{x} = \boldsymbol{b}$, obliczone w t-cyfrowej arytmetyce fl za pomocą metody eliminacji z wyborem (częściowym lub pełnym) elementów głównych. Wówczas istnieje macierz $\delta A \in \mathbb{R}^{n\times n}$, spełniająca nierówność

$$\|\delta A\|_\infty \leqslant Cn^3 g_n 2^{-t} \|A\|_\infty \qquad (C - \text{const}) \tag{5}$$

i taka, że

$$(A + \delta A)\tilde{\boldsymbol{x}} = \boldsymbol{b}.$$

Wniosek

Metoda eliminacji z wyborem elem. gł. jest algorytmem numerycznie poprawnym (o ile współczynnik g_n nie jest zbyt duży).

Twierdzenie

Niech $\boldsymbol{x}$ będzie rozwiązaniem układu równań liniowych

$$A\boldsymbol{x} = \boldsymbol{b} \tag{6}$$

i niech wektor $\boldsymbol{x} + \delta\boldsymbol{x}$ spełnia zaburzony układ

$$(A + \delta A)(\boldsymbol{x} + \delta\boldsymbol{x}) = \boldsymbol{b} + \delta\boldsymbol{b}, \tag{7}$$

gdzie $\delta A \in \mathbb{R}^{n\times n}$ i $\delta\boldsymbol{b} \in \mathbb{R}^n$ są zaburzeniami macierzy A i wektora b. Załóżmy, że $\eta = \|\delta A\|\|A^{-1}\| = \operatorname{cond}(A)\|\delta A\|/\|A\| < 1$ i $\|I\| = 1$. Wówczas dla dowolnej pary norm zgodnych zachodzi nierowność

$$\frac{\|\delta\boldsymbol{x}\|}{\|\boldsymbol{x}\|} \leqslant \frac{\operatorname{cond}(A)}{1-\eta}\left(\frac{\|\delta\boldsymbol{b}\|}{\|\boldsymbol{b}\|} + \frac{\|\delta A\|}{\|A\|}\right), \tag{8}$$

gdzie

$$\operatorname{cond}(A) := \|A\| \cdot \|A^{-1}\|.$$

$$\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots \in \mathbb{R}^n.$$

Niech będzie $\boldsymbol{x}^{(k)} = [x_1^{(k)}, x_2^{(k)}, \ldots, x_n^{(k)}]^T$.

Definicja

Ciąg wektorów $\{\boldsymbol{x}^{(k)}\}$ jest zbieżny do wektora $\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{x} = [x_1, x_2, \ldots, x_n]^T$, gdy $k \to \infty$ (tj. $\boldsymbol{x}^{(k)} \to \boldsymbol{x}$ $(k \to \infty)$ lub $\lim_{k\to\infty} \boldsymbol{x}^{(k)} = \boldsymbol{x}$) wtedy i tylko wtedy gdy dla $i = 1, 2, \ldots, n$ jest

$$x_i^{(k)} \to x_i \qquad (k \to \infty).$$

Analogicznie, jeśli

$$\{A^{(k)}\} = A^{(1)}, A^{(2)}, \ldots$$

jest ciągiem macierzy klasy $\mathbb{R}^{n\times n}$, $A^{(k)} = [a_{ij}^{(k)}]$, to

Definicja

Ciąg macierzy $\{A^{(k)}\}$ jest zbieżny do macierzy $A = [a_{ij}] \in \mathbb{R}^{n\times n}$ (tj. $A^{(k)} \to A$ $(k \to \infty)$ lub $\lim_{k\to\infty} A^{(k)} = A$) wtedy i tylko wtedy gdy dla $i, j = 1, 2, \ldots, n$ jest

$$a_{ij}^{(k)} \to a_{ij} \qquad (k \to \infty).$$

Lemat

1. $\boldsymbol{x}^{(k)} \to \boldsymbol{\theta} \quad (k \to \infty) \iff \|\boldsymbol{x}^{(k)}\| \to 0 \quad (k \to \infty)$ *dla każdej normy wektorowej.*
2. $\boldsymbol{x}^{(k)} \to \boldsymbol{x} \quad (k \to \infty) \iff \|\boldsymbol{x}^{(k)} - \boldsymbol{x}\| \to 0 \quad (k \to \infty)$ *dla każdej normy wektorowej.*
3. $A^{(k)} \to \Theta \quad (k \to \infty) \iff \|A^{(k)}\| \to 0 \quad (k \to \infty)$ *dla każdej normy macierzowej.*
4. $A^{(k)} \to A \quad (k \to \infty) \iff \|A^{(k)} - A\| \to 0 \quad (k \to \infty)$ *dla każdej normy macierzowej.*
5. *Jeśli* $\|A\| < 1$ *dla co najmniej jednej normy, to* $A^k \to \Theta \quad (k \to \infty)$.

Metoda Richardsona

Metodę iteracyjną Richardsona definiuje wzór

$$\boldsymbol{x}^{(k+1)} = B_\tau \boldsymbol{x}^{(k)} + \boldsymbol{c} \qquad (k \geqslant 0).$$

gdzie

$$B_\tau := I - \tau A, \qquad c := \tau \boldsymbol{b}. \tag{9}$$

Równoważnie, dla $k = 0, 1, \ldots$ obliczamy

$$x_i^{(k+1)} = x_i^{(k)} + \tau \left(b_i - \sum_{j=1}^{n} a_{ij} x_j^{(k)} \right). \tag{10}$$

Metoda Jacobiego

W metodzie Jacobiego mamy następującą macierz przekształcenia:

$$B \equiv B_J := -D^{-1}(L+U). \tag{11}$$

Wersję skalarną metody opisują wzory

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1, j\neq i}^{n} a_{ij} x_j^{(k)} \right)$$

$$= x_i^{(k)} + \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{n} a_{ij} x_j^{(k)} \right). \tag{12}$$

Twierdzenie

Jeśli A jest macierzą ze ściśle dominującą przekątną, tj.

$$|a_{ii}| > \sum_{j=1,\, j\neq i}^{n} |a_{ij}| \qquad (i = 1, 2, \ldots, n),$$

to $||B_J||_\infty < 1$ i metoda Jacobiego jest zbieżna.

Metoda Gaussa-Seidela

W metodzie (Gaussa-)Seidela mamy następującą macierz przekształcenia:

$$B \equiv B_S := -(D+L)^{-1}U. \tag{13}$$

Wersję skalarną metody opisują wzory

$$\begin{aligned} x_i^{(k+1)} &= \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)} \right) \\ &= x_i^{(k)} + \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i}^{n} a_{ij} x_j^{(k)} \right). \end{aligned} \tag{14}$$

Metoda relaksacji

W metodzie relaksacji mamy następującą macierz przekształcenia:

$$B_\omega := (I - \omega M)^{-1} (\omega N + (1 - \omega) I) \tag{15}$$

gdzie

$$M := -D^{-1} L, \qquad N := -D^{-1} U.$$

Wersję skalarą metody relaksacji można zapisać w następujący sposób:

$$x_i^{(k+1)} = x_i^{(k)} + \frac{\omega}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i}^{n} a_{ij} x_j^{(k)} \right) \tag{16}$$

Metoda relaksacji

W metodzie relaksacji mamy następującą macierz przekształcenia:

$$B_\omega := (I - \omega M)^{-1} (\omega N + (1 - \omega) I) \tag{15}$$

gdzie

$$M := -D^{-1} L, \qquad N := -D^{-1} U.$$

Wersję skalarą metody relaksacji można zapisać w następujący sposób:

$$x_i^{(k+1)} = x_i^{(k)} + \frac{\omega}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i}^{n} a_{ij} x_j^{(k)} \right) \tag{16}$$

Twierdzenie (Kahan)

Dla dowolnej nieosobliwej macierzy A i dowolnej liczby ω zachodzi nierówność

$$\varrho(B_\omega) \geqslant |\omega - 1|. \tag{17}$$

Metoda relaksacji

W metodzie relaksacji mamy następującą macierz przekształcenia:

$$B_\omega := (I - \omega M)^{-1} (\omega N + (1 - \omega) I) \tag{15}$$

gdzie

$$M := -D^{-1} L, \qquad N := -D^{-1} U.$$

Wersję skalarą metody relaksacji można zapisać w następujący sposób:

$$x_i^{(k+1)} = x_i^{(k)} + \frac{\omega}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i}^{n} a_{ij} x_j^{(k)} \right) \tag{16}$$

Twierdzenie (Ostrowski, 1954)

Jeśli macierz A jest symetryczna i dodatnio określona, to metoda relaksacji jest zbieżna dla każdego $\omega \in (0, 2)$.

Twierdzenie

Niech A będzie macierzą symetryczną, dodatnio określoną i niech ma postać blokowo-trójprzekątniową:

$$A = \begin{bmatrix} D_1 & U_1 & & & \\ L_2 & D_2 & U_2 & & \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ & & L_{m-1} & D_{m-1} & U_{m-1} \\ & & & L_m & D_m \end{bmatrix},$$

gdzie D_i są kwadratowymi macierzami przekątniowymi. Wtedy $\varrho(B_S) = \varrho^2(B_J)$ i optymalny czynnik relaksacji wyraża się wzorem

$$\omega_{\text{opt}} = \frac{2}{1 + \sqrt{1 - \varrho(B_S)}}.$$

Optymalną wartością $\varrho(B_\omega)$ jest

$$\varrho(B_{\omega_{\text{opt}}}) = \omega_{\text{opt}} - 1.$$

Twierdzenie

Jeśli $A \in \mathbb{R}^{n \times n}$ *oraz* $A = A^T$, *to istnieje macierz ortogonalna* $U \in \mathbb{R}^{n \times n}$, *że*

$$U^T A U = \Lambda = \operatorname{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n) \in \mathbb{R}^{n \times n},$$

przy czym $\lambda_1, \lambda_2, \ldots, \lambda_n$ *są wartościami własnymi macierzy* A.

Twierdzenie

Jeśli $A \in \mathbb{R}^{n\times n}$ *oraz* $A = A^T$, *to istnieje macierz ortogonalna* $U \in \mathbb{R}^{n\times n}$, *że*

$$U^T A U = \Lambda = \operatorname{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n) \in \mathbb{R}^{n\times n},$$

przy czym $\lambda_1, \lambda_2, \ldots, \lambda_n$ *są wartościami własnymi macierzy* A.

Twierdzenie (Schur)

Jeli $A \in \mathbb{R}^{n\times n}$, *to istnieje macierz unitarna* $U \in \mathbb{C}^{n\times n}$, *że*

$$U^H A U = R \in \mathbb{C}^{n\times n},$$

gdzie $R \in \mathbb{C}^{n\times n}$ *jest macierzą górną trójkątną. Jeśli wszystkie wartości własne macierzy* A *są rzeczywiste, to* $U, R \in \mathbb{R}^{n\times n}$.

Twierdzenie (o rozkładzie SVD)

Dla dowolnej macierzy $A \in \mathbb{R}^{m\times n}$ *istnieją takie macierze ortogonalne* $U \in \mathbb{R}^{m\times m}$ *i* $V \in \mathbb{R}^{n\times n}$, *że*

$$U^T A V = \Sigma = \operatorname{diag}(\sigma_1, \sigma_2, \ldots, \sigma_\ell) \in \mathbb{R}^{m\times n}, \tag{17}$$

gdzie $\ell = \min(m, n)$. *Ponadto, jeśli* $\operatorname{rank}(A) = r$, *to*

$$\sigma_1 \geqslant \sigma_2 \geqslant \ldots \geqslant \sigma_r > 0, \qquad \sigma_{r+1} = \sigma_{r+2} = \ldots = \sigma_\ell = 0.$$

Algorytm 1 Ortogonalizacja Grama-Schmidta

Require: $A = [\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_n] \in \mathbb{R}^{m \times n}$, $\operatorname{rank}(A) = n$

Ensure: $A = QR$, $Q = [\boldsymbol{q}_1, \boldsymbol{q}_2, \ldots, \boldsymbol{q}_n] \in \mathbb{R}^{m \times n}$, $Q^T Q = I_n$, $R = [r_{ij}] \in \mathbb{R}^{n \times n}$, $r_{ij} = 0 \quad (i > j)$

for $k = 1, 2, \ldots, n$ **do** ▷ Obliczamy kolejne wektory $\boldsymbol{q}_k$

  for $i = 1, 2, \ldots, k-1$ **do**

    $r_{ik} \leftarrow \boldsymbol{q}_i^T \boldsymbol{a}_k$

  end for

  $\boldsymbol{p}_k \leftarrow \boldsymbol{a}_k - \sum_{i=1}^{k-1} \boldsymbol{q}_i r_{ik}$ ▷ Można sprawdzić, że $\langle \boldsymbol{p}_k, \boldsymbol{q}_j \rangle = 0$ dla $j = 1, 2, \ldots, k-1$

  $r_{kk} \leftarrow \|\boldsymbol{p}_k\|$

  $\boldsymbol{q}_k \leftarrow \boldsymbol{p}_k / r_{kk}$ ▷ W ten sposób mamy $\|\boldsymbol{q}_k\| = 1$

end for

Algorytm 2 Zmodyfikowany algorytm ortogonalizacji Grama-Schmidta

Require: $A = [\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_n] \in \mathbb{R}^{m\times n}$, $\operatorname{rank}(A) = n$
Ensure: $A = QR$, $Q = [\boldsymbol{q}_1, \boldsymbol{q}_2, \ldots, \boldsymbol{q}_n] \in \mathbb{R}^{m\times n}$, $Q^TQ = I_n$,
$R = [r_{ij}] \in \mathbb{R}^{n\times n}$, $r_{ij} = 0 \quad (i > j)$

for $k = 1, 2, \ldots, n$ **do** ▷ Obliczamy kolejne wektory $\boldsymbol{q}_k$
  $\boldsymbol{q}_k \leftarrow \boldsymbol{a}_k$
  for $i = 1, 2, \ldots, k-1$ **do**
    $r_{ik} \leftarrow \boldsymbol{q}_i^T \boldsymbol{q}_k$
    $\boldsymbol{q}_k \leftarrow \boldsymbol{q}_k - \boldsymbol{q}_i r_{ik}$
  end for
  $r_{kk} \leftarrow \|\boldsymbol{q}_k\|$
  $\boldsymbol{q}_k \leftarrow \boldsymbol{q}_k / r_{kk}$ ▷ W ten sposób mamy $\|\boldsymbol{q}_k\| = 1$
end for