

MOwNiT Lab3 - Funkcje sklejjane

Jakub Płowiec

12 Kwietnia 2023

1 Cel zadania

Wyznaczenie interpolacyjnej funkcji sklejjanej trzeciego oraz drugiego stopnia z różnymi warunkami brzegowymi dla zadanej funkcji:

$$e^{-3\sin(x)} + 3\sin(x) - 1, x \in [-3\pi, 4\pi]$$

2 Sprzęt

W zadaniu obliczeniowym posłużono się językiem Python 3.9.0 na systemie Windows 10 z procesorem Intel core i5-9600KF

3 Interpolacja 3-go stopnia

Do obliczenia wykorzystamy wzór:

$$S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3 \quad \text{dla } i \in [1, \dots, n - 1] \quad (1)$$

gdzie każdy odcinek $S_i(x)$ jest wielomianem interpolującym na odcinku $[x_i, x_{i+1}]$

Aby wybrany stopień interpolacji był możliwy muszą zachodzić następujące warunki:

1. $S_i(x_{i+1}) = f(x_{i+1})$
2. $S_i(x_{i+1}) = S_{i+1}(x_{i+1})$
3. $S'_i(x_{i+1}) = S'_{i+1}(x_{i+1})$
4. $S''_i(x_{i+1}) = S''_{i+1}(x_{i+1})$

Wiedząc, że $S(x)$ jest funkcją sześcienną, przez co $S''(x)$ liniowe - jesteśmy w stanie przekształcić wzór 1, na następującą formę:

$$S_i''(x) = S_i''(x_i) \frac{x_{i+1} - x_i}{h_i} + S_i''(x_{i+1}) \frac{x - x_i}{h_i} \quad (2)$$

gdzie $h_i = x_{i+1} - x_i$.

Po dwukrotnym scałkowaniu tej funkcji otrzymujemy:

$$S_i(x) = \frac{S_i''(x_i)}{6h_i}(x_{i+1} - x)^3 + \frac{S_i''(x_{i+1})}{6h_i}(x - x_i)^3 + C(x - x_i) + D(x_{i+1} - x) \quad (3)$$

gdzie C oraz D to stałe całkowania, które jesteśmy w stanie wyliczyć korzystając z 1 warunku na interpolację:

$$S_i(x_i) = f(x_i) \quad S_i(x_{i+1}) = f(x_{i+1})$$

przez co ostatecznie dostajemy:

$$S_i(x) = \frac{S_i''(x_i)}{6h_i}(x_{i+1} - x)^3 + \frac{S_i''(x_{i+1})}{6h_i}(x - x_i)^3 + \left(\frac{f(x_{i+1})}{h_i} - \frac{S_i''(x_{i+1})h_i}{6}\right)(x - x_i) + \left(\frac{f(x_i)}{h_i} - \frac{S_i''(x_i)h_i}{6}\right)(x_{i+1} - x) \quad (4)$$

W powyższym wzorze nie znamy $S_i''(x)$. Aby je wyliczyć korzystamy z warunku ciągłości pierwszej pochodnej. Różniczkujemy więc $S_i(x)$.

$$S_i'(x) = -\frac{h_i}{3}S_i''(x_i) - \frac{h_i}{3}S_i''(x_{i+1}) - \frac{f(x_i)}{h_i} + \frac{f(x_{i+1})}{h_i} \quad (5)$$

Dodatkowo dla przejrzystości zastosujemy symbole $\sigma_i = \frac{1}{6}S''(x_i)$ oraz $\Delta_i = \frac{f(x_{i+1}) - f(x_i)}{h_i}$. Wtedy z równania 5 otrzymujemy:

$$S'(x_i) = \Delta_i - h_i(\sigma_{i+1} + 2\sigma_i) \quad (6)$$

Wyznamy zarówno wzór w tym samym punkcie od drugiej strony:

$$s'_{i-1}(x_i) = \Delta_{i-1} + h_{i-1}(2\sigma_i + \sigma_{i-1}) \quad (7)$$

Korzystając z warunku ciągłości, a m.in 3-ciego warunku interpolacji korzystając z równan 6 i 7, tworzymy układ (n-2) równań liniowych dla punktów pośrednich:

$$\Delta_{i-1} + h_{i-1}(2\sigma_i + \sigma_{i-1}) = \Delta_i - h_i(\sigma_{i+1} + 2\sigma_i) \quad (8)$$

$$h_{i-1}\sigma_{i-1} + 2(h_{i-1} + h_i)\sigma_i + h_i\sigma_{i+1} = \Delta_i - \Delta_{i-1} \quad \text{dla } i = 2, 3, \dots, n-1 \quad (9)$$

Do rozwiązania powyższego układu potrzebujemy jednak 2 więcej równań, przez co przechodzimy do pojęcia jakim są warunki brzegowe.

3.1 Natural cubic spline

$$S''(x_1) = S''(x_n) = 0$$

Korzystając z podstawienia $\sigma_i = \frac{1}{6}S''_i(x_i)$ otrzymujemy:

$$\sigma_1 = 0 \quad \sigma_n = 0$$

Co końcowo prowadzi do utworzenia takiego układu równań:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ h_1 & 2(h_1 + h_2) & h_2 & 0 & 0 \\ 0 & h_2 & 2(h_2 + h_3) & h_3 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & h_{n-2} & 2(h_{n-2} + h_{n-1}) & h_{n-1} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \vdots \\ \sigma_{n-1} \\ \sigma_n \end{bmatrix} = \begin{bmatrix} 0 \\ \Delta_2 - \Delta_1 \\ \Delta_3 - \Delta_2 \\ \vdots \\ \Delta_{n-1} - \Delta_{n-2} \\ 0 \end{bmatrix}$$

3.2 Cubic function

Wprowadźmy takie oznaczenie:

$C_1(x)$ - f. sześcienna przez pierwsze 4 punkty

$C_n(x)$ - f. sześcienna przez ostatnie 4 punkty

co nam daje:

$$S'''(x_1) = C_1''' \quad S'''(x_n) = C_n'''$$

Stałe C_1''' i C_n''' mogą zostać określone bez znajomości $C_1(x)$ i $C_n(x)$:

$$\Delta_i^{(1)} = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}; \quad \Delta_i^{(2)} = \frac{\Delta_{i+1} - \Delta_i}{x_{i+2} - x_i}; \quad \Delta_i^{(3)} = \frac{\Delta_{i+1}^{(2)} - \Delta_i^{(2)}}{x_{i+3} - x_i}$$

Możemy zróżniczkować wzór 2 na $S''(x)$, aby otrzymać:

$$S'''(x_1) = C_1'''(x_1) = \frac{6}{h_1}(\sigma_2 - \sigma_1) = 6\Delta_1^{(3)}$$

$$S'''(x_n) = C_n'''(x_n) = \frac{6}{h_{n-1}}(\sigma_n - \sigma_{n-1}) = 6\Delta_{n-3}^{(3)}$$

Następnie otrzymujemy dwa równania:

$$-h_1\sigma_1 + h_1\sigma_2 = h_1^2\Delta_1^{(3)}$$

$$h_{n-1}\sigma_{n-1} - h_{n-1}\sigma_n = -h_{n-1}^2\Delta_{n-3}^{(3)}$$

Co końcowo prowadzi do utworzenia takiego układu równań:

$$\begin{bmatrix} -h_1 & h_1 & 0 & 0 & 0 \\ h_1 & 2(h_1 + h_2) & h_2 & 0 & 0 \\ 0 & h_2 & 2(h_2 + h_3) & h_3 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & h_{n-2} & 2(h_{n-2} + h_{n-1}) & h_{n-1} \\ 0 & 0 & 0 & h_{n-1} & -h_{n-1} \end{bmatrix} \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \vdots \\ \sigma_{n-1} \\ \sigma_n \end{bmatrix} = \begin{bmatrix} h_1^2 \Delta_1^{(3)} \\ \Delta_2 - \Delta_1 \\ \Delta_3 - \Delta_2 \\ \vdots \\ \Delta_{n-1} - \Delta_{n-2} \\ -h_{n-1}^2 \Delta_{n-3}^{(3)} \end{bmatrix}$$

3.3 Wyniki

Aby otrzymać gotową funkcję należy obliczyć macierz, a następnie można wyliczyć współczynniki do poniższej postaci, która da już nam gotowy wynik:

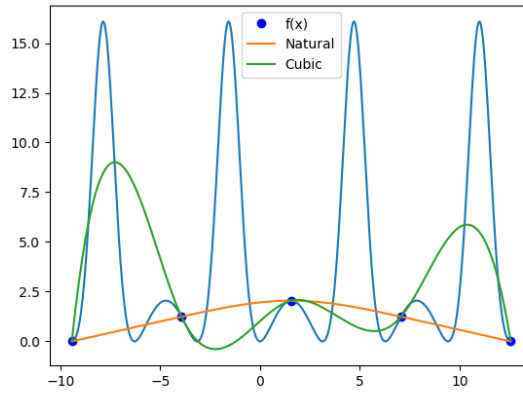
$$S_i(x) = f(x_i) + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3, \quad x \in [x_i, x_{i+1}]$$

$$b_i = \frac{f(x_{i+1}) - f(x_i)}{h_i} - h_i(\sigma_{i+1} + 2\sigma_i)$$

$$c_i = 3\sigma_i$$

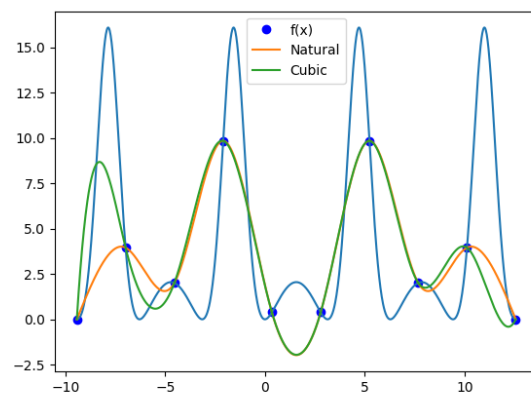
$$d_i = \frac{\sigma_{i+1} - \sigma_i}{h_i}$$

Wykresy były rysowane na podstawie 1000 punktów.



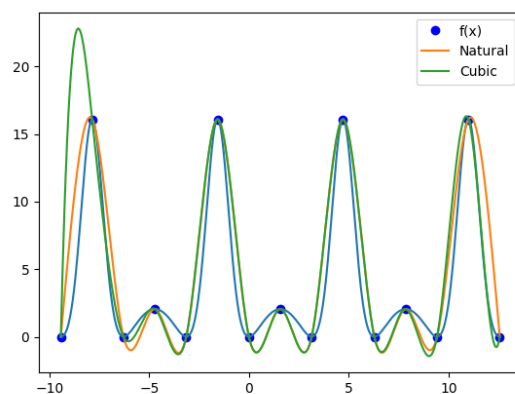
Wykres 1: N = 5 (liczba węzłów)

Jak na wykresie 1 przedstawiono, zauważalna jest dość spora różnica jeżeli chodzi o wygląd funkcji sklejanej przy małej liczbie węzłów zależnie od warunków brzegowych.



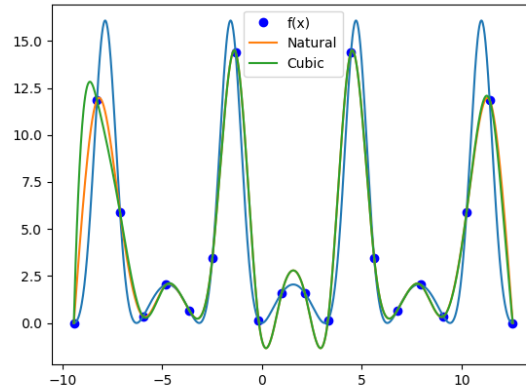
Wykres 2: $N = 10$

Na wykresie 2 można zauważyć, iż powoli funkcje sklejane kawałkami prezentują się identycznie.



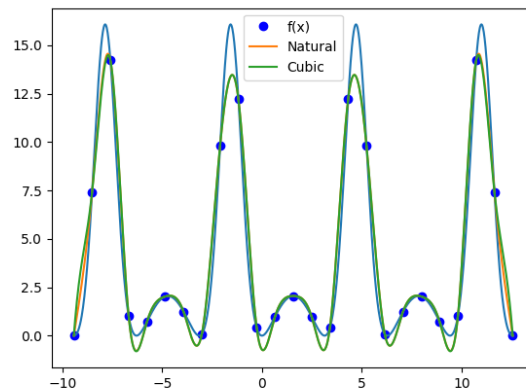
Wykres 3: $N = 15$

Ciekawym zjawiskiem już dla 15 węzłów jest, iż funkcje sklejane osiągną podobny kształt do funkcji zadanej w zadaniu, pomimo pierwszego wzniesienia w jednej z nich.



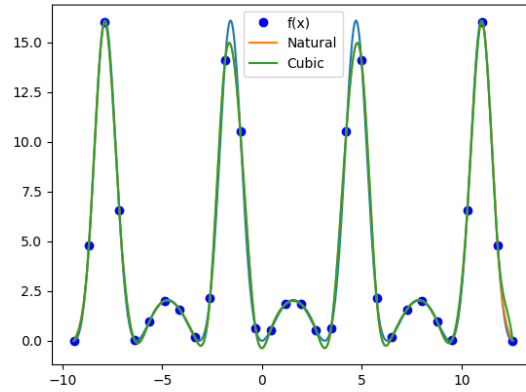
Wykres 4: $N = 20$

W porównaniu do Wykresu 3 - zauważamy jak funkcję mogą w mniejszym, bądź większym stopniu pogarszać swoje odwzorowanie pomimo zwiększaniu liczby węzłów.



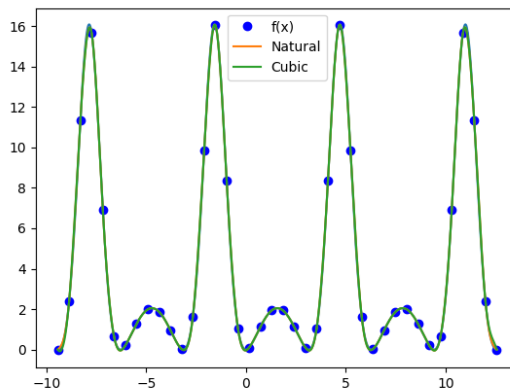
Wykres 5: $N = 25$

Funkcje sklejane ponownie starają się naśladować funkcję pierwotną.



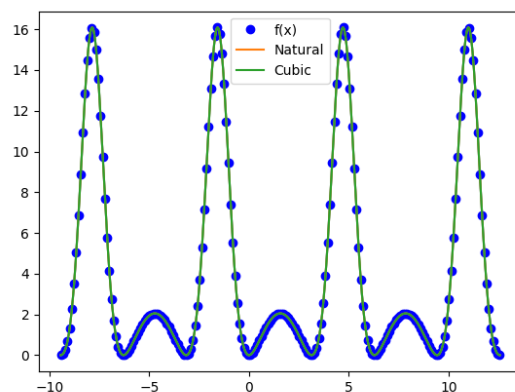
Wykres 6: $N = 30$

Na wykresie 6 używając 30 węzłów można śmiało powiedzieć, że funkcje sklejane w obu przypadkach mają identyczny kształt i niemalże identycznie wyglądają co do zadanej funkcji.



Wykres 7: $N = 40$

Ostatnim przystankiem jest 40 węzłów, gdzie idealnie funkcje sklejane w obu przypadkach niezależnie od warunków brzegowych - osiągają nasz upragniony kształt.



Wykres 8: $N = 200$

Ważnym zauważeniem jest na pewno, że w przypadku interpolacji za pomocą funkcji skleianej nie musimy się martwić o wszelakie błędy, problemy, które były często generowane w pozostałych metodach z powodu dużej ilości węzłów.

3.4 Błędy interpolacji

Liczba węzłów	Natural	Cubic
5	15.72	16.30
10	12.56	14.11
15	6.82	18.32
20	5.16	9.70
25	2.73	2.74
30	1.32	1.32
40	0.30	0.31

Tabela 1: Największy błąd interpolacyjny dla obu funkcji sklepanych

Na tabeli 1 można zauważyć, że obie funkcje w miarę utrzymywały największy błąd podobny do siebie.

Liczba węzłów	Natural	Cubic
5	3.73	3.70
10	3.52	3.49
15	1.90	2.16
20	1.38	1.56
25	0.78	0.85
30	0.23	0.25
40	0.05	0.05

Tabela 2: Średni błąd interpolacyjny dla obu funkcji sklepanych

Na tabeli 2 można zauważyć, że funkcje podobnie się zachowywały jeżeli chodzi o błąd a dla $N = 40$ przybliżenie już naprawdę jest zadowalające.

4 Interpolacja 2-go stopnia

(Niestety dość długo starałem się wyprowadzić wzór i doprowadzić tę interpolację do całości, ale koniec końców nie wychodziła pomimo starań. Parę kodów apropos interpolacji 2 stopnia jest dalej w pliku pythonowym i przedstawię niżej rozumowanie)

Do obliczenia wykorzystamy wzór:

$$g_i(x) = a_i x^2 + b_i x + c_i, \quad \text{dla } i \in [1, \dots, n-1]$$

W celu utworzenia układu równań posłużymy się następującymi równościami:

1. $g_i(x_i) = f(x_i)$
2. $g_i(x_{i+1}) = g_{i+1}(x_{i+1})$
3. $g'_i(x_{i+1}) = g'_{i+1}(x_{i+1})$

Zauważmy, iż w naszych równaniach będziemy mieli dokładnie $3 * (n - 1)$ niewiadomych. Korzystając zarówno z pierwszego jak i drugiego równania - jesteśmy w stanie utworzyć sobie następujące równania:

$$g_1(x_1) = a_1 x_1^2 + b_1 x_1 + c_1 = f(x_1)$$

$$g_1(x_2) = a_1 x_2^2 + b_1 x_2 + c_1 = f(x_2)$$

$$g_2(x_2) = a_2 x_2^2 + b_2 x_2 + c_2 = f(x_2)$$

$$\begin{aligned}
g_2(x_3) &= a_2x_3^2 + b_2x_3 + c_3 = f(x_3) \\
&\vdots \\
g_{n-1}(x_{n-1}) &= a_{n-1}x_{n-1}^2 + b_{n-1}x_{n-1} + c_{n-1} = f(x_{n-1}) \\
g_{n-1}(x_n) &= a_{n-1}x_n^2 + b_{n-1}x_n + c_{n-1} = f(x_n)
\end{aligned}$$

Zauważmy iż każdy wewnętrzny (nie na granicy) węzeł jest powtarzany dwukrotnie co nam daje $2*(n-2)$ równań, gdzie $n-2$ to liczba węzłów wewnętrznych, to dochodzą nam 2 pojedyncze równania dla x_1 oraz x_n - co końcowo nam daje już $2*(n-1)$ równań.

Następnie skorzystamy z równania numer 3, dzięki którym zbliżamy się coraz bardziej do liczby równań w celu rozwiązania układu. Wyglądają one tak:

$$\begin{aligned}
g'_1(x_2) &= g'_2(x_2) \\
g'_2(x_3) &= g'_3(x_3) \\
&\vdots \\
g'_{n-2}(x_{n-1}) &= g'_{n-1}(x_{n-1})
\end{aligned}$$

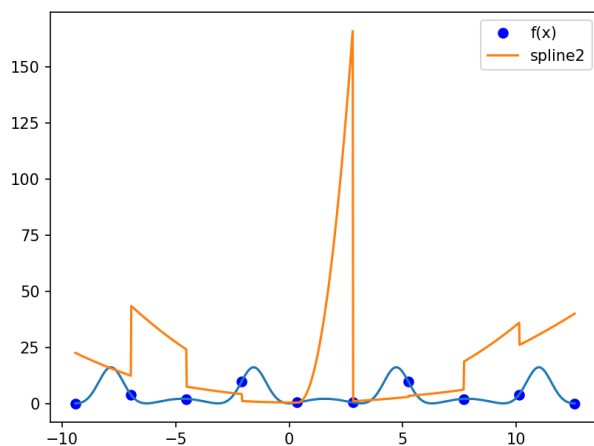
Daje nam to dodatkowo $n-2$ równań, ponieważ na każdy węzeł wewnętrzny przypada jedno równanie.

Zauważmy iż łącznie osiągnęliśmy $(2n-2) + (n-2) = 3n-4$ równań. Oznacza to, że brakuje nam ostatniego równania abyśmy mogli utworzyć układ równań i go rozwiązać.

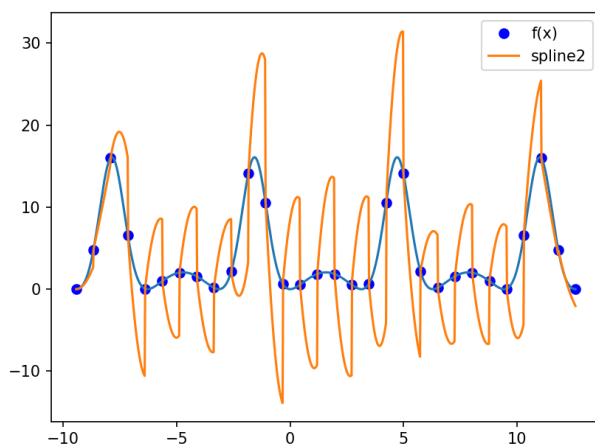
Ostatnim równaniem będzie wymyślony przez nas warunek brzegowy jakim może być $g'_1(x_1) = 0$, który da nam równanie:

$$2a_1x_1 + b_1 = 0$$

Niestety nie jestem pewien, czy jest to poprawne - mój algorytm generuje błędne wyniki, przez co nie jestem w stanie dokończyć sprawozdania.



Wykres 9: $N = 10$



Wykres 10: $N = 30$

Tutaj jeszcze kolejna próba interpolacji drugim algorytmem, lecz niestety jak widać też błędnym.

5 Wnioski

Interpolacja funkcją sklejaną jest dobrym wyborem w celu uniknięcia efektu Rungego przy zwiększaniu dokładności. Można zauważyć jak funkcja sklejana przyjmuje różne formy dla tej samej liczby węzłów, lecz z innymi warunkami brzegowymi - gdzie mimo innego kształtu błędy wychodzą podobne. Funkcje pomimo różnych warunków brzegowych osiągają perfekcję dla tej samej ilości węzłów.