

# Podstawy Obliczeń Komputerowych

## Laboratorium 2

Kamil Czop 259613

Łukasz Majchrzak 262761

Wtorek 11:15TN - Y02-37c - Dr inż. Konrad Kluwak

16 czerwca 2023

## 1 Metody interpolacji wielomianowej

### 1.1 Cel ćwiczeń

Celem wykonanego laboratorium było wykorzystanie dowolnego algorytmu interpolacji wielomianowej oraz interpolacji wykorzystującej funkcje sklejające.

### 1.2 Dane

$x$	10	20	30	40	50
$y = \cos(x)$	0.98	0.93	0.86	0.76	0.64

Szukane  $x$ :  $21^\circ$

## 2 Metoda Lagrange

Wielomianowa metoda interpolacyjna wykorzystuje zależność:

$$W(x) = \sum_{i=0}^n y_i = \frac{\varphi_i(x)}{\varphi_i(x_i)} = \sum_{i=0}^n \frac{\prod_{j \neq i} (x - x_j)}{\prod_{j \neq i} (x_i - x_j)}$$

---

### Algorithm 1 Wstępny algorytm lagrange

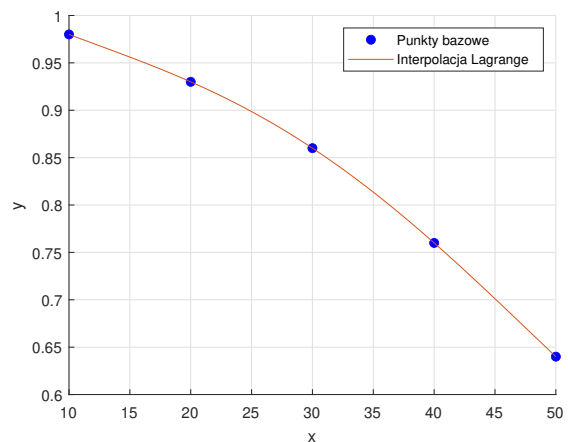
---

**Require:**  $x_1 \dots x_N, y_1 \dots y_N, x_s$

**Ensure:**  $y_s$

```
1: procedure LAGRANGE-OLD
2:    $res \leftarrow 0$ 
3:   for  $i \leftarrow 1$  to  $N$  do
4:      $a \leftarrow y(i)$ 
5:      $a_d \leftarrow 1$ 
6:      $res_{seg} \leftarrow 1$ 
7:     for  $j \leftarrow 1$  to  $n$  do
8:       if  $j \neq i$  then
9:          $a_d \leftarrow a_d * (x_i - x_j)$ 
10:         $res_{seg} \leftarrow res_{seg} * (x_s - x_j)$ 
11:     Next  $j$ 
12:     $a \leftarrow a / a_d$ 
13:     $res_{seg} = res_{seg} * a$ 
14:     $res = res + res_{seg}$ 
15:  Next  $i$ 
16:  Stop
```

---



Rysunek 1: Interpolacja metodą Lagrange'a

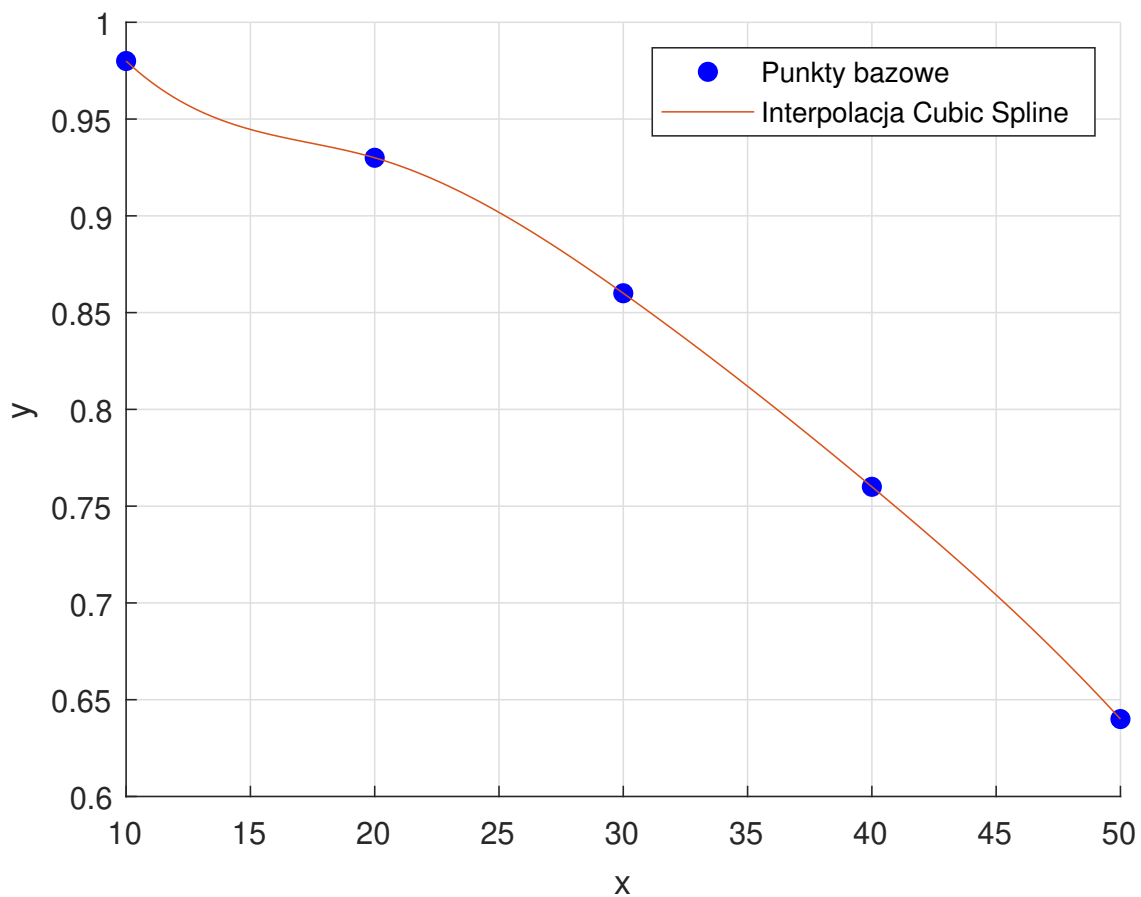
Dla  $x = 21$  wartość interpolowana wynosi: 0.9242.

Wykorzystywany w pierwszej wersji algorytm nie jest optymalny, ani natywny względem metod dostępnych w środowisku MATLAB. Zaprezentowany pseudokod korzysta z własności iteracyjnych algorytmu sumując kolejno wyznaczone elementy wielomianu.

Krzywa uzyskana jako całościowy wynik interpolacji przechodzi przez wszystkie punkty, co sugeruje prawidłowość działania implementacji metody.

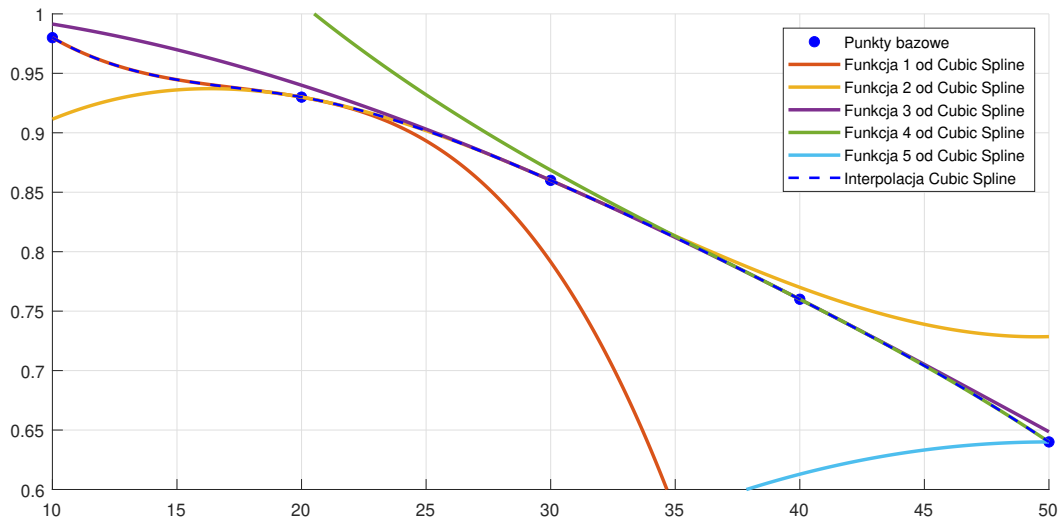
### 3 Metoda sześciennych funkcji sklejanych - Cubic Spline

Metoda funkcji sklejanych wskazuje nazwą iż do interpolacji wykorzystujemy więcej niż jedną funkcję. Cubic Spline partycjonuje oś  $x$  celem utworzenia sześciennych funkcji których krzywe łączą się w jedno w znanych nam punktach badanej funkcji bazowej.



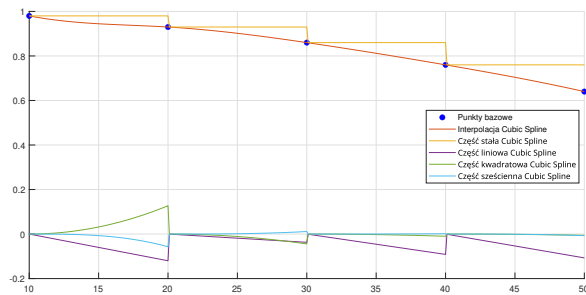
Rysunek 2: Interpolacja wykorzystująca Cubic Spline

Uzyskano wartość dla szukanego  $x$  :  $y \approx 0.9259$



Rysunek 3: Poszczególne "podfunkcje" sklejające

Na załączonym wykresie dokładnie widać "sklejanie funkcji", gdzie pierwsza funkcja odwzorowuje  $x \in [10 : 20)$ , druga  $x \in [20 : 30)$ , trzecia  $x \in [30 : 40)$ , czwarta  $x \in [40 : 50)$  oraz piąta nie wykorzystana w zakreślonym obszarze  $x \in [50 : \infty)$ .



Na załączonym wykresie można zauważyć jak utworzone funkcje się zmieniają, w szczególności jak przeżymy przebieg stałej części parametru  $a$ , gdzie widać "schodkowanie", wraz ze przejściem do wartości punktu bazowego.

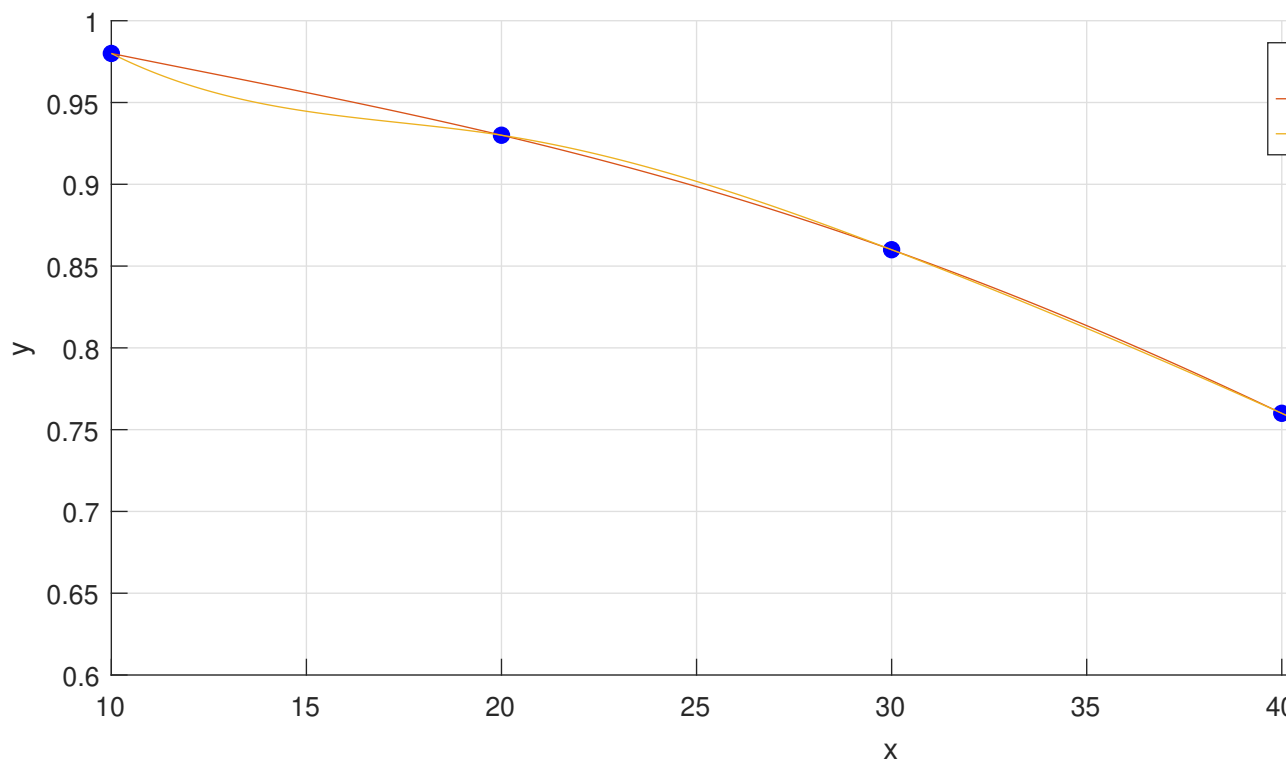
W interpolowanym przypadku, część liniowa o parametrze  $b$  była zawsze ujemna, wartość  $b$  była wystarczająco wysoka by ukazać kształt piłokształtny na wykresie.

Rysunek 4: Poszczególne wartości poziomów wartości w wielomianie

Wartości parametrów

$a$	0.9800	0.9300	0.8600	0.7600	0.6400
$b$	-0.0120	-0.0037	-0.0091	-0.0107	0
$c$	0.0013	$-4.46e-4$	-0.0001	$-5.71e-5$	$-2.71e-5$
$d$	$-5.71e-5$	$1.14e-5$	$1.42e-6$	$-7.14e-6$	0

### 3.1 Porównanie



Rysunek 5: Porównanie interpolacji Cubic Spline i metody Lagrange'a

## 4 Wektory własne i wartości własne macierzy

### 4.1 Dane

$$A = \begin{bmatrix} 3 & 1 & 2 \\ 1 & 8 & 1 \\ 2 & 1 & 1 \end{bmatrix}$$

### 4.2 Uzyskany wynik

- Wartość własna: 8.0393
- Wektor własny:  $\begin{bmatrix} 0.2447 \\ 0.9502 \\ 0.1930 \end{bmatrix}$

## 5 Github, bibliografia, dokumentacja

- <https://github.com/Myknakryu/pok-2023>
- <https://traf-barak.pwr.edu.pl/wp-content/uploads/2020/08/Raport.pdf>
- <https://www.youtube.com/watch?v=2YmCaxcKyBY>