

# Sztuczna inteligencja

Dominik Lau

13 kwietnia 2023

## 1 Wstęp

Celem projektu było zaimplementowanie i przeanalizowanie dwóch algorytmów interpolacji na wybranych profilach wysokościowych - metody wykorzystującej wielomian Lagrange oraz metody z funkcjami sklejanymi trzeciego stopnia. Do implementacji wykorzystano język *Python* oraz biblioteki *matplotlib*, *pandas*.

## 2 Teoria

W obu przypadkach zakładamy, że mamy pewien zestaw  $n + 1$  punktów

$$\begin{aligned}(x_0, y_0) \\ (x_1, y_1) \\ \dots \\ (x_n, y_n)\end{aligned}$$

i chcemy znaleźć taką funkcję  $F(x)$ , że

$$\forall_{i=0..n} F(x_i) = y_i$$

dobrze określającą, jakie wartości przyjmują  $y$  w punktach  $x \notin \{x_0, \dots, x_n\}$

### 2.1 Metoda Lagrange

W metodzie tej funkcja  $F$  ma postać

$$F(x) = \sum_{i=0}^n y_i \phi_i(x)$$

gdzie

$$\phi_i(x) = \prod_{j=0, j \neq i}^{n+1} \frac{x - x_j}{x_i - x_j}$$

jest **bazą Lagrange'a**. Metoda ta zwraca takie same wyniki jak metoda Vandermonde, jednak nie musimy rozwiązywać układu równań liniowych

## 2.2 Metoda krzywych sklejanych 3. stopnia

w tej metodzie funkcja  $F$  ma postać

$$F(x) = S_i(x); x \in [x_i, x_{i+1}]$$

czyli przedstawiamy ją jako szereg połączonych wielomianów  $S_i(x)$  takich, że

$$\deg(S_i) = 3$$

w celu uzyskania układów równań, z których pozyskamy współczynniki  $S_i(x)$  przyjmujemy założenia

$$\begin{aligned} S_i(x_i) &= y_i \\ S_i(x_{i+1}) &= y_{i+1} \\ S'_{j-1}(x_i) &= S'_j(x_i); x = 1..n-1 \\ S''_{j-1}(x_i) &= S''_j(x_i); x = 1..n-1 \\ S''_0(x_0) &= 0 \\ S''_{n-1}(x_n) &= 0 \end{aligned}$$

znalezienie wielomianów  $S$  sprowadza się do rozwiązania powyższego układu równań

## 3 Wybrane profile wysokościowe

Do analizy wybrano następujące profile wysokościowe

- ścieżkę Yoshidy na górę Fuji - jedno duże wzniesienie
- trasę Al. Ujazdowskie-Łazienki-Solec - trasa głównie płaska
- trasę wokół centrum Słupska - wiele nagłych (ale niedużych) wzniesień

## 4 Trasa na górę Fuji

## 5 Trasa w Warszawie

## 6 Trasa wokół centrum Słupska

## 7 Źródła

- Wikipedia-Spline