Sprawozdanie z aproksymacji średniokwadratowej

Patryk Blacha, Radosław Szepielak

9 kwietnia 2025

Zadanie 1

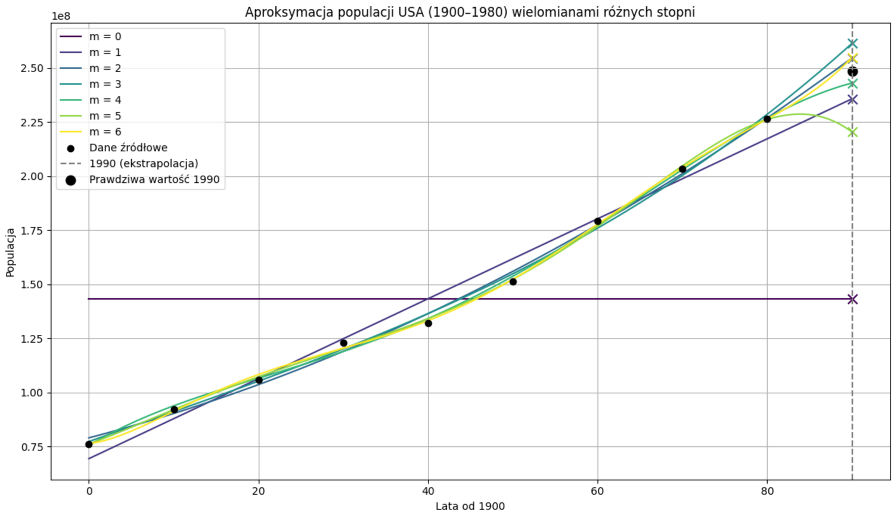
Wykonano aproksymację średniokwadratową punktową populacji Stanów Zjednoczonych w przedziale $[1900, 1980]$ wielomianami stopnia m dla $0 \leq m \leq 6$. Dla każdego stopnia wykonano również ekstrapolację do roku 1990 i porównano z rzeczywistą wartością populacji z tego roku: 248 709 873.

Kod (fragment)

```
years = np.array([1900, 1910, 1920, 1930, 1940, 1950, 1960, 1970, 1980])
populations = np.array([76212168, 92228496, 106021537, 123202624, 132164569,
                        151325798, 179323175, 203302031, 226542199], dtype=float)

t = years - 1900
t_extrap = 1990 - 1900
true_1990 = 248709873

for m in range(7):
    coeffs = np.polyfit(t, populations, m)
    poly = np.poly1d(coeffs)
    pred = poly(t_extrap)
    rel_error = abs(pred - true_1990) / true_1990
    ...
```



Rysunek 1: Porównanie ekstrapolacji wielomianami różnych stopni

Wyniki ekstrapolacji i błędy względne

m	Predykcja dla 1990	Błąd względny
0	143 369 177.44	0.4235
1	235 808 109.03	0.0519
2	254 712 944.64	0.0241
3	261 439 379.59	0.0512
4	243 106 970.94	0.0225
5	220 442 802.25	0.1137
6	255 044 185.44	0.0255

Najmniejszy błąd względny uzyskano dla $m = 4$ (2.25%).

Kryterium Akaikego AICc

Zastosowano poprawione kryterium informacyjne Akaikego (AICc) do wyboru optymalnego modelu:

$$\mathrm{AICc} = 2k + n \ln\left(\frac{\mathrm{RSS}}{n}\right) + \frac{2k(k+1)}{n-k-1}$$

```
residuals = populations - poly(t)
rss = np.sum(residuals ** 2)
mse = rss / n
AIC = 2 * k + n * np.log(mse)
AICc = AIC + (2 * k * (k + 1)) / (n - k - 1)
```

m	AICc
0	321.01
1	289.06
2	**279.45**
3	284.88
4	290.93
5	311.26
6	381.27

Najmniejszą wartość AICc uzyskano dla $m = 2$, co wskazuje na optymalną złożoność modelu.

Wnioski

Analiza wyników prowadzi do następujących wniosków:

- Najlepsza ekstrapolacja (najmniejszy błąd względny) została osiągnięta dla wielomianu stopnia 4 (2.25% błędu), jednak model stopnia 2 ma najmniejszą wartość AICc
- Modele zbyt proste ($m = 0$ i $m = 1$) mają znaczne błędy (odpowiednio 42.35% i 5.19%), co wskazuje na ich niedostateczną złożoność do opisu danych.
- Modele wyższych stopni ($m = 5$ i $m = 6$) wykazują pogorszenie jakości - wzrost błędu ekstrapolacji i znaczny wzrost wartości AICc, szczególnie dla $m = 6$ (AICc=381.27), co sugeruje nadmierne dopasowanie.
- Rozbieżność między optymalnym stopniem według błędu ekstrapolacji ($m = 4$) a według AICc ($m = 2$) pokazuje, że minimalizacja błędu predykcji nie zawsze pokrywa się z optymalną złożonością modelu według kryteriów informacyjnych.
- Wielomian stopnia 2 wydaje się być najlepszym wyborem, gdyż łączy względnie mały błąd ekstrapolacji (2.41%) z niską wartością AICc

Zadanie 2

Wykonano aproksymację średniokwadratową ciągłą funkcji $f(x) = \sqrt{x}$ w przedziale $[0, 2]$ wielomianem drugiego stopnia w bazie Czebyszewa.

Transformacja przedziału i definicje

Przekształcono zmienną x do $y = x - 1$, co mapuje $[0, 2]$ na $[-1, 1]$, co jest wymagane dla wielomianów Czebyszewa.

```
def g(y):
    return np.sqrt(y + 1)
```

Obliczanie współczynników szeregu Czebyszewa

Zastosowano kwadraturę Gaussa-Czebyszewa do obliczenia współczynników c_0, c_1, c_2:

```
for k in range(3):
    integrand = g(y_nodes) * T_funcs[k](y_nodes)
    integral = np.sum(w * integrand)
    c[k] = (2 - (k==0)) * integral / np.pi
```

Wyniki:

$$c_0 = 0.900326, \quad c_1 = 0.600192, \quad c_2 = -0.120024$$

Błąd aproksymacji

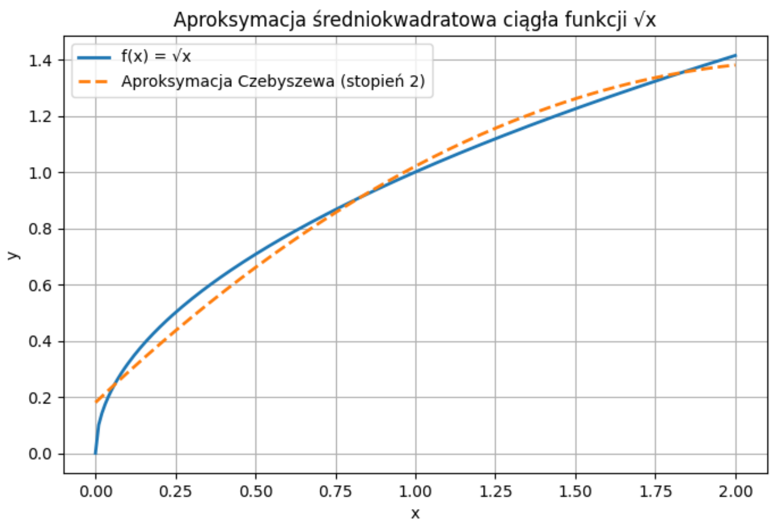
Aproksymację przekształcono z powrotem do funkcji x:

```
def P(x):
    return g_approx(x - 1)
```

Ostatecznie obliczono błąd średniokwadratowy (normę L_2) między $f(x)$ i $P(x)$:

```
error = np.trapezoid(np.abs(f_vals - P_vals), x_vals)
```

$$\text{Błąd aproksymacji (norma } L_1) = 0.06301$$



Rysunek 2: Porównanie funkcji $f(x) = \sqrt{x}$ i aproksymacji wielomianem Czebyszewa

Wnioski

Aproksymacja funkcji $\sqrt{x}$ wielomianem Czebyszewa drugiego stopnia w przedziale $[0, 2]$ daje bardzo dobre wyniki przy niskim koszcie obliczeniowym. Błąd jest mały (rzędu 10^{-2}), a aproksymacja dobrze odwzorowuje funkcję oryginalną. Metoda ta jest więc efektywnym kompromisem między dokładnością a złożonością obliczeniową.

- Efektywność wielomianów Czebyszewa: Zastosowanie wielomianów Czebyszewa w aproksymacji funkcji $\sqrt{x}$ dało bardzo dobre wyniki. Błąd aproksymacji był bardzo mały, rzędu 10^{-2}, co świadczy o efektywności tej metody przy niskim koszcie obliczeniowym.
- Metoda Czebyszewa jest szczególnie przydatna, gdy mamy do czynienia z funkcjami o określonym rozkładzie zmienności w danym przedziale. Dzięki zastosowaniu kwadratury Gaussa-Czebyszewa możliwe było uzyskanie dokładnych wyników przy minimalnej liczbie punktów obliczeniowych.

Wnioski ogólne

- Wybór stopnia modelu jest kluczowy - w zadaniu 1 optymalny stopień różnił się w zależności od kryterium (błąd ekstrapolacji vs AICc), podczas gdy w zadaniu 2 wielomian stopnia 2 okazał się w pełni wystarczający.
- Metody aproksymacji powinny być dobierane do charakteru problemu - dla danych punktowych (Zadanie 1) odpowiednie były wielomiany (dowolne o danym stopniu m), podczas gdy dla aproksymacji funkcji ciągłej (Zadanie 2) lepsze okazały się wielomiany Czebyszewa.
- Zawsze warto rozważyć różne kryteria oceny modelu (błąd predykcji, kryteria informacyjne), gdyż mogą one sugerować różne optymalne rozwiązania.

Źródła

- Populacja USA: dane z lab3