Akademia Górniczo-Hutnicza
Wydział Informatyki
Kierunek Informatyka

Metody Obliczeniowe w Nauce i Technice

Laboratorium 5

Aproksymacja

Wojciech Michaluk, Kyrylo Iakymenko

Kraków, 5 kwietnia 2024

1 Wprowadzenie

Podczas tego laboratorium zrealizujemy dwa zadania, w których poznamy i przeanalizujemy różne techniki aproksymacji, która pozwala na przybliżanie lub zastępowanie pewnej funkcji poprzez inną - podobnie jak interpolacja, ale jest ogólniejsza niż interpolacja ([2]).

W zadaniu pierwszym wykonujemy aproksymację średniokwadratową punktową (dyskretną) populacji Stanów Zjednoczonych, bazując na tych samych danych, co w *laboratorium 3* - oczywiście ponownie je tu przytoczymy w odpowiednim miejscu. Zbadamy empirycznie, dla jakiego stopnia wielomianu uzyskamy najlepszą jakość interpolacji i porównamy to z wartościa teoretyczną.

W zadaniu drugim również mamy do czynienia z aproksymacją średniokwadratową, ale tym razem w wersji ciągłej. Spowoduje to, że używamy innych funkcji bazowych - mianowicie będą to wielomiany Czebyszewa. W ten sposób uzyskamy podobny rezultat co dla aproksymacji jednostajnej, ale korzystamy z "tańszego" obliczeniowo jej zamiennika.

2 Zadanie 1

2.1 Opis zadania

Poniżej podajemy informacje o populacji Stanów Zjednoczonych w wybranych latach:

Rok	1900	1910	1920	1930	1940	1950	1960	1970	1980
Populacja	76212168	92228496	106021537	123202624	132164569	151325798	179323175	203302031	226542199

Tabela 1: Dane dotyczące populacji USA w latach 1900 - 1980

Na podstawie danych w tabeli powyżej, wyznaczymy wielomian aproksymacyjny stopnia m dla $m = 0, 1, ..., 6$. Dla każdej wartości m dokonamy ekstrapolacji wielomianu do roku 1990 i znajdziemy błąd względny ekstrapolacji, wiedząc, że prawdziwa wartość populacji dla 1990 roku wynosi 248 709 873. Następnie wskażemy wartość m, dla której ten błąd był najmniejszy oraz sprawdzimy, czy ta wartość pokrywa się z wynikiem uzyskanym przy wykorzystaniu tzw. *kryterium informacyjnego Akaikego*.

2.1.1 Kryterium informacyjne Akaikego

Po angielsku *Akaike Information Criterion* - w skrócie **AIC** - jest to zaproponowane przez Hirotugu Akaikego kryterium wyboru pomiędzy modelami statystycznymi, jeden ze wskaźników dopasowania modelu [1].

Stopień wielomianu m to tzw. hiperparametr modelu. Potrzebujemy znaleźć jego optymalną wartość i w tym celu wykorzystamy to kryterium.

- jeżeli wartość m jest zbyt mała, to model jest zbyt obciążony i nie jest w stanie odpowiednio uwzględnić zmienności danych.
- zbyt wysoka wartość m powoduje dużą wariancję (było to także widoczne przy interpolacji) oraz podatność na szumy i błędy w danych.
- wartość optymalnego m wyznaczamy, posiłkując się wzorem:

$$\mathrm{AIC} = 2k + n \ln\left(\frac{\sum_{i=1}^{n}[y_i - \hat{y}(x_i)]^2}{n}\right), \tag{1}$$

  gdzie $k = m+1$ to liczba parametrów, y_i dla $i = 0, 1, ..., n$ to prawdziwa wartość liczby osób w danym roku oraz $\hat{y}(x_i)$ to wartości wielomianu aproksymacyjnego w punktach x_i (w naszym przypadku są to wybrane lata) - czyli to wartości przewidywane przez model. Im mniejsza wartość kryterium, tym lepszy model.

[1] Fragment definicji z Wikipedii: `https://pl.wikipedia.org/wiki/Kryterium_informacyjne_Akaikego`

Ponieważ rozmiar próbki $n = 9$ jest stosunkowo "niewielki" w porównaniu do wartości k (zwykle $n \gg k$), używamy zmodyfikowanego wzoru (1) ze składnikiem korygującym. Nasz wzór przyjmuje zatem postać:

$$\mathrm{AIC}_c = \mathrm{AIC} + \frac{2k(k+1)}{n-k-1}. \tag{2}$$

To z tego wzoru na kryterium będziemy korzystać przy szukaniu optymalnego m.

2.2 Opracowanie zadania

W ramach tego zadania wykorzystujemy technikę aproksymacji *średniokwadratowej, dyskretnej*.

1. W tym podejściu szukamy wielomianu $p(x)$, dla którego minimalna jest wartość wyrażenia $\sum_{i=0}^{n} |y_i - p(x_i)|$, przy czym $p(x)$ to wielomian aproksymacyjny.
2. Wielomian aproksymacyjny da się zapisać w postaci $p(x) = \sum_{j=0}^{m} c_j \phi_j(x)$. Występujące w tym wzorze funkcje $\phi_j(x)$ to funkcje bazowe. Tutaj są to proste jednomiany, tzn. $\phi_j(x) = x^j$.

Pozostaje wyznaczyć wartości współczynników c_j stojących przed funkcjami bazowymi. Aby wielomian $p(x)$ spełniał podany powyżej warunek, można je wyliczyć, korzystając z równania normalnego:

$$A \cdot c = y, \tag{3}$$

gdzie A to macierz Vandermonde'a, c to szukana kolumna współczynników, natomiast y to kolumna z rzeczywistymi wartościami populacji.

Przekształcając odpowiednio równanie (3), tzn. mnożąc obie strony od lewej przez A^T, możemy równoważnie je zapisać jako

$$S \cdot c = T. \tag{4}$$

Tutaj S zawiera wyrazy postaci $S_k = \sum_{i=0}^{n} x_i^k$ dla $k = 0, 1, ..., 2m$ (k pełni funkcję sumy indeksu wiersza i kolumny). Z kolei T to kolumna z wyrazami postaci $T_k = \sum_{i=0}^{n} x_i^k y_i$ dla $k = 0, 1, ..., m$.

Dla każdej z rozważanych wartości m obliczamy najpierw kolumnę współczynników c - w kodzie wygląda to tak:

```
m = np.arange(0, 7)

for i in range(len(m)):
        #year zawiera lata, population liczbę mieszkańców USA w tych latach
        A = np.vander(year, m[i] + 1, increasing = True)
        S = np.dot(np.transpose(A), A)
        T = np.dot(np.transpose(A), population)

        #obliczamy kolumnę współczynników
        c = np.linalg.solve(S, T)
```

Listing 1: Obliczanie kolumny współczynników c dla różnych wartości m

W powyższej sekcji kodu, podobnie jak w następnych, `np` to alias dla biblioteki `numpy`. Ponadto, wartości c dla każdego m zapamiętujemy w tabeli, ale tego tu nie zamieściliśmy - jest to trywialne.

Możemy nastepnie przejść do ekstrapolacji uzyskanych wielomianów do roku 1990. Przy okazji, po obliczeniu wartości od razu wyznaczamy błąd względny ekstrapolacji.

Poniżej kod za to odpowiedzialny:

```
correct_val = 248709873 #rzeczywista populacja
extrapolate_year = 1990 #rok, do którego ekstrapolujemy

for i in range(len(m)):
        c = np.flip(C[i]) #odwracamy kolumnę, wynika to z funkcji np.polyval
        approx_val = np.polyval(c, extrapolate_year)
        relative_error = np.abs(approx_val - correct_val) / correct_val
```

Listing 2: Ekstrapolacja wielomianów do roku 1990

Wyniki przedstawiamy w poniższej tabeli.

Stopień wielomianu m	Ekstrapolowana wartość dla 1990 roku	Błąd względny
0	143369177	0.423549
1	235808109	0.051875
2	254712945	0.024137
3	261439962	0.051184
4	263612281	0.059919
5	274500666	0.103698
6	240280933	0.033891

Tabela 2: Zestawienie wartości ekstrapolowanych i błędów względnych

Warto wziąć pod uwagę, że wartości ekstrapolowane sa zaokrąglone do całości, natomiast wartości błędu - do 6 cyfr po przecinku. Z powyższej tabeli można odczytać, że najmniejszą wartość błędu względnego uzyskujemy dla stopnia wielomianu $m = 2$. Porównajmy to z rezultatami obliczonymi wedle kryterium AIC. Korzystamy z poniższego kodu:

```
def AIC_c(m, n): #w tym przypadku n = 9
        c = np.flip(C[m]) #podobnie jak wcześniej
        k = m + 1 #hiperparametr
        y_hat_values = np.polyval(c, year)

        AIC = 2 * k + n * np.log(np.sum((population - y_hat_values) ** 2) / n)
        correction = 2 * k * (k + 1) / (n - k - 1) #wspomniana poprawka

        return AIC + correction
```

Listing 3: Obliczanie kryterium informacyjnego Akaikego

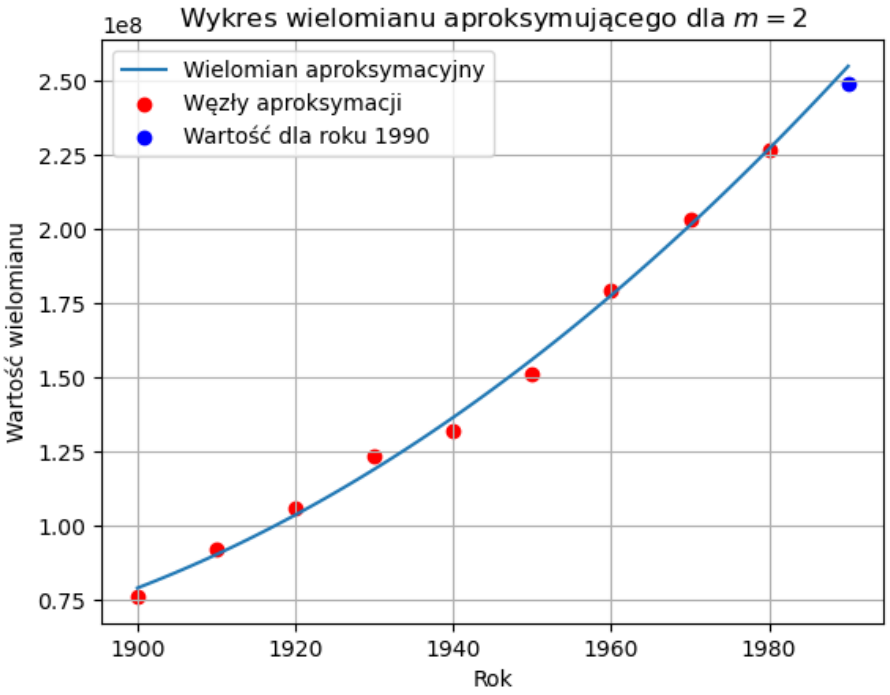
Wyniki prezentują się następująco.

Stopień wielomianu m	Wartość kryterium Akaikego
0	321.010975
1	289.056478
2	279.453374
3	284.880402
4	297.927536
5	326.842462
6	387.009551

Tabela 3: Wartości kryterium AIC dla różnych stopni wielomianu

Widzimy, że najmniejsza wartość kryterium jest osiągana dla $m = 2$. Jest to zgodne z rezultatem osiągniętym wcześniej, kiedy obliczaliśmy wartości ekstrapolowane dla roku 1990 dla wszystkich wielomianów i błędy względne.

Nie było to wprost napisane w poleceniu zadania, ale uznajemy, że warto zamieścić wykres wielomianu aproksymacyjnego dla wyznaczonej wartości $m = 2$, dla której jakość aproksymacji była najlepsza.



Rysunek 1: Wykres wielomianu aproksymacyjnego o stopniu $m = 2$

3 Zadanie 2

3.1 Opis zadania

Celem zadania jest wykonanie aproksymacji średniokwadratowej *ciągłej*. Funkcją aproksymowaną jest $f(x) = \sqrt{x}$ dla $x \in [0; 2]$. Przyjmujemy stopień wielomianu $m = 2$. W tym zadaniu wykorzystujemy wielomiany Czebyszewa jako funkcje bazowe.

3.1.1 Wielomiany Czebyszewa

Wielomiany Czebyszewa są zdefiniowane jako $T_k(x) = cos(k \cdot arccos(x))$ dla $x \in [-1; 1]$, $cos(\phi) = x$ oraz $k \in \mathbb{N}$ - reprezentacja trygonometryczna.

Korzystając z tożsamości trygonometrycznej ([2]):

$$cos(n\phi) + cos((n-2)\phi) = 2cos((n-1)\phi) \cdot cos(\phi)$$
$$cos(n\phi) = 2cos((n-1)\phi) \cdot cos(\phi) - cos((n-2)\phi)$$

uzyskujemy pomocną zależność rekurencyjną $T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x)$ dla $k \geq 2$. Korzystając z niej oraz z definicji (dla $k = 0, 1$) można łatwo obliczyć kolejne wielomiany Czebyszewa: $T_0(x) = 1, T_1(x) = x, T_2(x) = 2x^2 - 1$ itd.

Wielomiany te mają szereg własności, które sprawiają, że są bardzo dobrym wyborem w problemie aproksymacji czy też interpolacji, m. in.:

1. Miejsca zerowe wielomianów Czebyszewa to tzw. węzły Czebyszewa - o ich użyteczności mieliśmy okazję dowiedzieć się przy rozwiązywaniu zadania z *laboratorium 4*.
2. Właśność *minimaksu* ([2]): wielomian $2^{1-n} \cdot T_n(x)$ ma najmniejszą normę maksymalną ze wszystkich wielomianów stopnia $n \geq 1$ z czynnikiem wiodącym równym 1, wynosi ona właśnie 2^{1-n}.

3. I chyba najważniejsza w kontekście aproksymacji, czyli:
   Wielomiany Czebyszewa są ortogonalne. Oznacza to, że dla iloczynu skalarnego funkcji f i g zdefiniowanego jako $\langle f, g\rangle = \int_{-1}^{1} w(x)f(x)g(x)$, gdzie $w(x)$ to funkcja wagowa, jeżeli weźmiemy za f wielomian Czebyszewa ϕ_i oraz analogicznie za g weźmiemy ϕ_j to:
   - dla $i \neq j$ wartość iloczynu skalarnego wynosi 0
   - dla $i = j \neq 0$ wartość ta wynosi $\frac{\pi}{2}$
   - dla $i = j = 0$ wartość ta wynosi π

3.2 Opracowanie zadania

Porównując do **zadania 1**, postępujemy podobnie, ale jest kilka różnic. Przede wszystkim mamy do czynienia z aproksymacją ciągłą, zatem będziemy szukać wielomianu aproksymacyjnego $p(x)$, dla którego jest minimalna wartość całki $\int_a^b |f(x) - p(x)|^2$. Ponadto, wielomian $p(x)$ również reprezentujemy w postaci $p(x) = \sum_{j=0}^{m} c_j\phi_j(x)$, natomiast tak jak już wspomnieliśmy wcześniej - rolę funkcji bazowych pełnią wielomiany Czebyszewa.

W tym podejściu do wyliczenia współczynników c_j należy rozwiązać równanie macierzowe

$$\begin{bmatrix} \langle\phi_0,\phi_0\rangle & \langle\phi_0,\phi_1\rangle & \cdots & \langle\phi_0,\phi_m\rangle \\ \langle\phi_1,\phi_0\rangle & \langle\phi_1,\phi_1\rangle & \cdots & \langle\phi_1,\phi_m\rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle\phi_m,\phi_0\rangle & \langle\phi_m,\phi_1\rangle & \cdots & \langle\phi_m,\phi_m\rangle \end{bmatrix} \cdot \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_m \end{bmatrix} = \begin{bmatrix} \langle f,\phi_0\rangle \\ \langle f,\phi_1\rangle \\ \vdots \\ \langle f,\phi_m\rangle \end{bmatrix}. \tag{5}$$

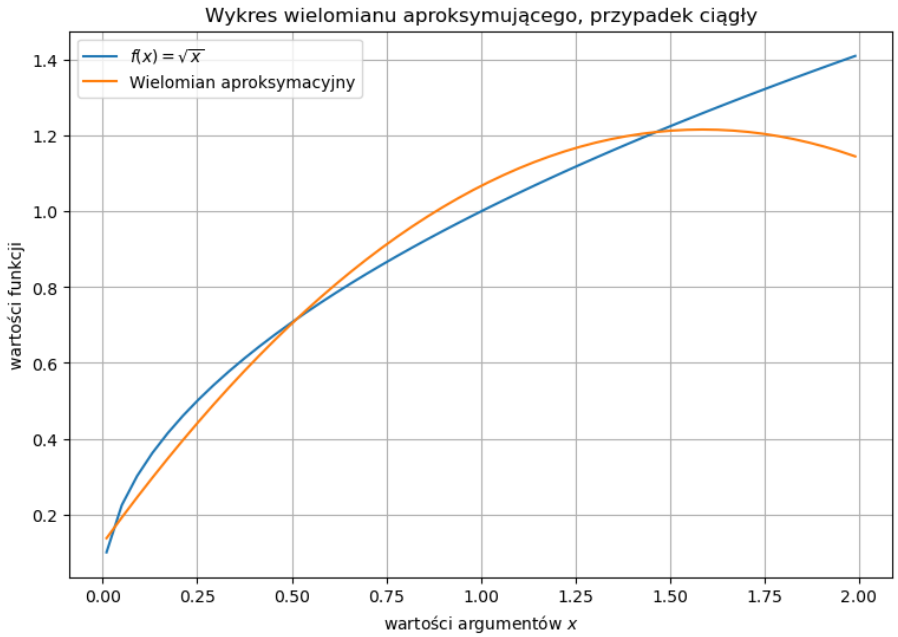
Dzięki wielomianom Czebyszewa macierz kwadratowa po lewej stronie jest diagonalna. Rozwiązanie takiego równania staje się trywialne i uzyskujemy $c_k = \frac{\langle f,\phi_k\rangle}{\langle\phi_k,\phi_k\rangle}$, przy czym $k = 0, 1, ..., m$. Funkcją wagową jest $w(x) = \frac{1}{\sqrt{1-x^2}}$ dla $x \in [-1; 1]$.

Przedstawmy nasz kod do tych obliczeń:

```
def weight(x): #stosujemy przesunięcie ze względu na dziedzinę
    return (1 - (x - 1) ** 2) ** (-0.5)

def dot_product_chebyshev(n): #wartości dla i == j
    if n == 0: return np.pi
    return np.pi / 2

def dot_product(n):
    def f_integrate(x):
        return weight(x) * f(x) * phi(x, n)

    #nie bierzemy całego przedziału [0;2] ze względu na funkcję wagową
    xs = np.linspace(0.01, 1.99, 50)
    ys = f_integrate(xs)

    return np.trapz(ys, xs)

def c_i(i):
    return dot_product(i) / dot_product_chebyshev(i)

def eval_poly(coeffs, x):
    res = 0
    for i, c in enumerate(coeffs):
        res += c * phi(x, i)
    return res
```

Listing 4: Funkcje pomocnicze do aproksymacji wielomianami Czebyszewa

Poniżej prezentujemy uzyskany wykres naszej aproksymacji.



Rysunek 2: Wykres wielomianu aproksymacyjnego z wielomianami bazowymi Czebyszewa o stopniu $m = 2$

Na oko widzimy, że nasza metoda w miarę dokładnie aproksymuje podaną funkcję, co może świadczyć o tym, że została zaimplementowana poprawnie. Żeby porównać tę aproksymację z testowaną w zadaniu pierwszym, wyznaczmy błąd względny. W tym celu policzymy całkę $\int_0^2 |\sqrt{x} - p(x)|dx$, gdzie $p(x)$ jest otrzymanym wielomianem aproksymacyjnym. Dalej podzielimy tę całkę przez całkę z $g(x) = x$ na podanym przedziale, żeby uzyskać błąd względny - jest ona równa długości przedziału, czyli 2. Te obliczenia są wykonane w poniższym fragmencie kodu:

```
def err():
    def f_err(x):
        return np.abs(eval_poly(cs, x) - f(x)) #cs to kolumna współczynników
    ys = f_err(xs) #xs = np.linspace(0.1, 1.99, 50)
    integral = np.trapz(ys, xs)

    return integral / 2

relative_error = err()
print("Wartość błędu względnego:", np.around(relative_error, decimals = 6))
```

Listing 5: Obliczanie błędu względnego aproksymacji ciągłej

W wyniku dostajemy wartość błędu względnego naszej metody, równą 0.062147. Można ją porównać do błędu względnego aproksymacji wielomianem czwartego stopnia z zadania pierwszego, który wynosił 0.059919, ale na ogół widzimy, że aproksymacja średniokwadratowa ciągła nie wykazuje się dużą skutecznością w porównaniu do odpowiednika dyskretnego, testowanego w zadaniu pierwszym. Dochodzimy do takich wniosków, gdyż przedstawiony błąd względny w wariancie ciągłym jest czasem 2-3 razy większy w porównaniu do aproksymacji dyskretnej.

Na podstawie tych spostrzeżeń możemy stwierdzić, że metoda aproksymacji średniokwadratowej ciągłej sprawdziła się gorzej w naszym konkretnym zastosowaniu. Warto jednak zwrócić uwagę na to, że aproksymacje nie były badane na jednej funkcji, więc tak naprawdę różnice w zbiorach wejściowych mogą być odpowiedzialne za różnice w wynikach. Zwracamy również uwagę na to, że nie były badane aproksymacje ciągłe stopni różnych od 2, co nie pozwala nam przeprowadzić analizy i porównania zachowań błędu względnego w obu przypadkach oraz uniemożliwia dokładne porównanie obu metod.

4 Podsumowanie i wnioski

Nasze pierwsze zadanie polegało na analizie populacji Stanów Zjednoczonych w wybranych latach, wykorzystując technikę aproksymacji średniokwadratowej dyskretnej. W ramach tego eksperymentu dokonaliśmy aproksymacji populacji za pomocą wielomianu stopnia m dla różnych wartości m. Następnie wykorzystaliśmy ekstrapolacje otrzymanych wielomianów w celu przewidzenia populacji dla roku 1990. Wartości ekstrapolowane porównaliśmy z rzeczywistymi danymi populacji dla 1990 roku, na tej podstawie wyznaczyliśmy najbardziej optymalny stopień wielomianu dla naszego problemu. Dodatkowo, wykorzystaliśmy kryterium informacyjne Akaikego (AIC), aby potwierdzić optymalność wybranego stopnia wielomianu.

W drugim zadaniu zaimplementowaliśmy aproksymację funkcji $f(x) = \sqrt{x}$ w przedziale $[0, 2]$, wykorzystując metodę aproksymacji średniokwadratowej ciągłej. W tym przypadku użyliśmy wielomianów Czebyszewa jako funkcji bazowych. Analizując błędy względne aproksymacji, przedstawiliśmy pewne wnioski odnośnie skuteczności tej metody oraz porównaliśmy ją z aproksymacją dyskretną z zadania pierwszego.

Wnioski z obu eksperymentów nie pozwalają nam konkretnie wybrać lepszą metodę spośród dwóch testowanych, ani nawet nie umożliwiają skostruowania porówniania, które by mogło nas doprowadzić do obiektywnej oceny obu metod, gdyż metody były testowane na różnych zbiorach i w różnych warunkach (między innymi w metodzie dyskretnej testowaliśmy kilka wielomianów aproksymacyjnych różnych stopni, a w metodzie ciągłej tylko jeden). Ostatecznie przeprowadzone testy sugerują, że wybór odpowiedniej metody aproksymacji zależy od specyfiki problemu i charakterystyki danych. Obie metody powinny być testowane przy problemie, który pozwala na skorzystanie zarówno z metody ciągłej, jak i dyskretnej (ciężko byłoby aproksymować ciągle funkcję zadaną zbiorem punktów, jak w zadaniu pierwszym), żeby zapewnić najlepsze wyniki.

Literatura

[1] Materiały pomocnicze do laboratorium zamieszczone na platformie Teams w katalogu *lab05/lab5-intro.pdf*.

[2] Treść przedstawiona na wykładzie o aproksymacji.