Laboratorium 5 – Aproksymacja

04.04.2023 **Adam Trybus, Piotrek Olszak**

1. Opis i rozwiązania zadań

1.1 Zadanie 1a

Ekstrapolacje wielomianu do roku 1990 wykonujemy przy pomocy klasy numpy i funkcji polyfit i polyval

```
for m in range(7):
    coeffs = np.polyfit(years, population, m)
    tab_coeffs.append(coeffs)
    predicted_value = np.polyval(coeffs, 1990)
    error = (predicted_value - true_value) / true_value * 100
    print(f'm = {m}, Przewidywana wartość: {predicted_value}, Błąd względny: {error:.2f}%')
```

1.2 Zadanie 1b

AIC liczymy przy pomocy funkcji square, log, polyval

```
def AIC(population, coeffs, k):
    residual_sum_sq = 0
    for i in range(len(population)):
        residual_sum_sq += np.square(population[i] - np.polyval(coeffs, years[i]))
    AIC = 2*k - k*np.log(residual_sum_sq/k)
    return AIC
```

Następnie liczymy AICc ze wzoru podanego w treści i printujemy wyniki.

```
def AICc(AIC, k, n):
    return AIC + (2*k*(k+1))/(n-k-1)

for m in range(len(coeffs)):
    print(AICc(AIC(population,tab_coeffs[m],m+1),m+1,len(population)))
```

1.3 Zadanie 4

Aproksymację średniokwadratową ciągłą funkcji robimy przy pomocy klasy numpy. Funkcje Chebyszewa bierzemy ze wcześniejszych laboratoriów.

```python
def get_chebyshev(n: int) -> list[float]:
    get_theta = lambda x: ((2*x + 1)/(2*n + 2))*np.pi
    return np.cos([get_theta(j) for j in range(n+1)])

def transform_chebyshev(cheb, a, b):
    return [a + (b-a)*(x+1)/2 for x in cheb]

f = lambda x: np.sqrt(x)

x_inter = transform_chebyshev(get_chebyshev(2), 0, 2)
y_inter = f(x_inter)

coefs = poly.chebyshev.chebfit(x_inter, y_inter, 2)
approx = poly.Polynomial(coefs)
print(approx)
```

Następnie prezentujemy wykres porównując funkcję właściwą do aproksymacji za pomocą biblioteki pyplot.

```python
plt.scatter(x_inter, y_inter)
plt.plot(x_new, f(x_new), label='Funkcja właściwa')
plt.plot(x_new, approx(x_new), label='Aproksymacja')
plt.title('Aproksymacja funkcji sqrt(x)')
plt.xlabel('x')
plt.ylabel('y')
plt.legend()
plt.show()
```

2. Wykresy, wyniki liczbowe i wnioski

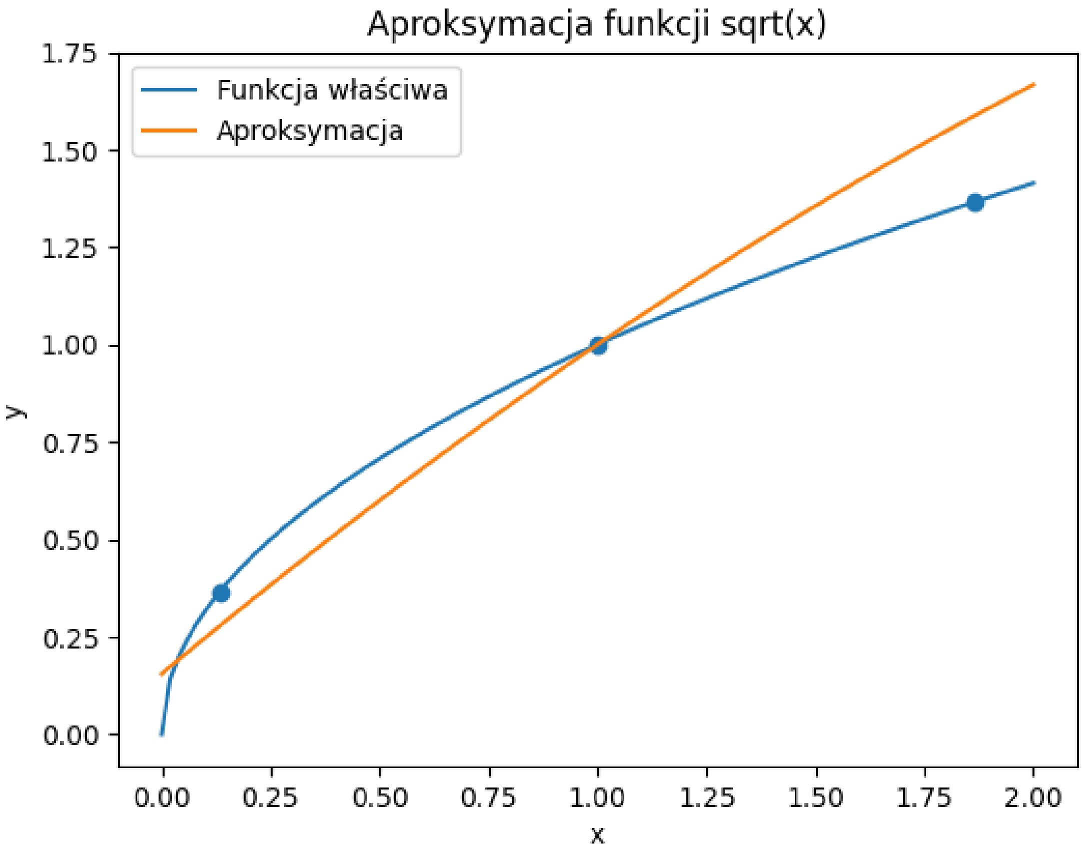
Tablica wyników z zadania 1a pokazuje, że błąd względny jest najmniejszy dla m = 4

```
m = 0, Przewidywana wartość: 143369177.4444445, Błąd względny: -42.35%
m = 1, Przewidywana wartość: 235808109.02777672, Błąd względny: -5.19%
m = 2, Przewidywana wartość: 254712944.6428604, Błąd względny: 2.41%
m = 3, Przewidywana wartość: 261439379.58825684, Błąd względny: 5.12%
m = 4, Przewidywana wartość: 243106971.640625, Błąd względny: -2.25%
m = 5, Przewidywana wartość: 220442975.0, Błąd względny: -11.37%
m = 6, Przewidywana wartość: 255058368.0, Błąd względny: 2.55%
```

Natomiast licząc z kryterium informacyjnego Akaikego najmniejsza wartość wynosi dla m = 5

```
m=  0 , wartość kryterium:  -35.00796783721549
m=  1 , wartość kryterium:  -59.90959437650341
m=  2 , wartość kryterium:  -82.04696149820107
m=  3 , wartość kryterium:  -103.85723272495633
m=  4 , wartość kryterium:  -117.89925638824025
m=  5 , wartość kryterium:  -119.93826309487412
m=  6 , wartość kryterium:  -74.29845202991285
```

Dodatkowo zamieszczam wykres porównujący funkcje właściwą oraz aproksymacje z zadania 4



3. Bibliografia

1) https://matplotlib.org/stable/tutorials/introductory/index.html#tutorials-introductory