УДК 517.518.8

Студ. Д.И. Подшиваленко
Науч. рук. доц. Л. Д. Яроцкая (кафедра высшей математики, БГТУ)

АППРОКСИМАЦИЯ ФУНКЦИЙ С ПОМОЩЬЮ ОРТОГОНАЛЬНЫХ МНОГОЧЛЕНОВ ЛЕЖАНДРА

Процесс аппроксимации включает замену одних объектов более простыми, хорошо изученными, близкими в некотором смысле к исходным. Ортогональные многочлены играют важную роль в теории аппроксимации функций, главным образом в связи с возможностью разложения произвольных функций, принадлежащих к весьма широким функциональным классам, в ряды по ортогональным системам.

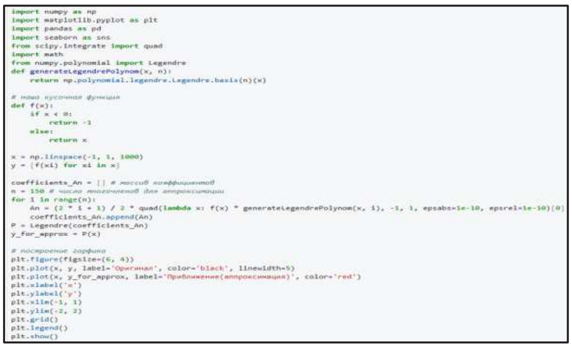
В курсе анализа показано, что многочлены Лежандра $P_n(x)$ для любых x определенные по формуле

$$P_n(x) = \frac{1}{2^n n!}\frac{d^n}{dx^n}(x^2-1)^n, \qquad n = 0, 1, 2, \ldots,$$

являются ортогональными в промежутке $[-1,1]$ и для любой кусочно-гладкой функции $f(x)$ на $[-1,1]$ справедливо разложение

$$\mathrm{f(x)} = \sum_{\mathrm{n}=0}^{\infty} \mathrm{c_n P_n(x)}, \qquad \mathrm{c_n} = \frac{2\mathrm{n}+1}{2}\int_{-1}^{1} \mathrm{f(x)P_n(x)dx}.$$

В данной работе для аппроксимации кусочно-гладких функций на [−1,1 многочленами Лежандра написан код на языке Python с использованием специальных библиотек (рис. 1).



```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
from scipy.integrate import quad
import math
from numpy.polynomial import Legendre
def generateLegendrePolynom(x, n):
    return np.polynomial.legendre.Legendre.basis(n)(x)

# наша кусочная функция
def f(x):
    if x < 0:
        return -1
    else:
        return x

x = np.linspace(-1, 1, 1000)
y = [f(xi) for xi in x]

coefficients_An = [] # массив коэффициентов
n = 150 # число многочленов для аппроксимации
for i in range(n):
    An = (2 * i + 1) / 2 * quad(lambda x: f(x) * generateLegendrePolynom(x, i), -1, 1, epsabs=1e-10, epsrel=1e-10)[0]
    coefficients_An.append(An)
P = Legendre(coefficients_An)
y_for_approx = P(x)

# построение графика
plt.figure(figsize=(6, 4))
plt.plot(x, y, label='Оригинал', color='black', linewidth=5)
plt.plot(x, y_for_approx, label='Приближение(аппроксимация)', color='red')
plt.xlabel('x')
plt.ylabel('y')
plt.xlim(-1, 1)
plt.ylim(-2, 2)
plt.grid()
plt.legend()
plt.show()
```

Рисунок 1 – Фрагмент кода

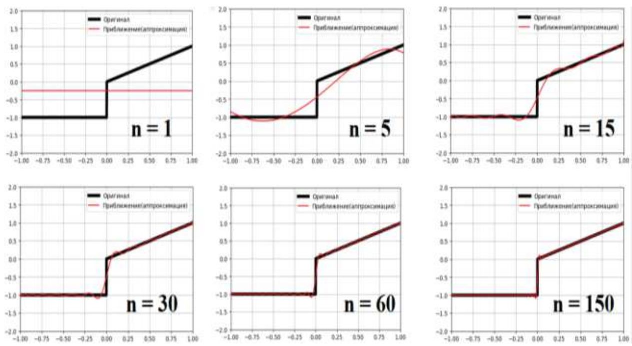


Рисунок 2 – Аппроксимация разрывной функции

Пример аппроксимации разрывной функции в зависимости от степени n многочленов Лежандра, представлены на рис. 2.

Таким образом, получено непрерывное приближение для функции, имеющей разрыв первого рода.

С увеличением степени многочленов наблюдается значительное улучшение точности представления, хотя абсолютная точность не достигается.