# Задание по курсу "ВвЧМ 24/25": Приближение функций

Павел Васильев, 213 группа

Декабрь 2024

## 1 Постановка задачи

Интерполяция - это способ нахождения промежуточных значений величины по имеющемуся дискретному набору известных значений.

Интерполяция использует значения некоторой функции, заданные в ряде точек, чтобы предсказать значения функции между ними. Перечисленные ниже методы предназначены для создания ряда с более высокой частотой наблюдений на основе ряда с низкой частотой. Например, вычислить ряд с квартальной динамикой на основе ряда годовых данных.

Многие задачи машинного обучения можно сформулировать через интерполяцию "неизвестной" функции [1, 2]

Различные методы численного приближения и их теоретические обоснования можно найти в [3]

### 1.1 Условия задачи

Построить полином Лагранжа для следующих функций $f_i(x)$ на отрезке $x \in [-2, 0]$:

1. $f_1(x) = T_5(x)$, где $T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x)$, $T_1(x) = x$, $T_0(x) = 1$
2. $f_2(x) = |cos(5x)|e^{-x/2}$

В качестве узлов интерполяции выбрать узлы равномерной на [-2,0] сетки для количества узлов $n = 3, 5, 9, 17$. Исследовать сходимость интерполяции. Найти максимальное отклонение $\max |P_n(x) - f_i(x)|$ на равномерной сетке из 1001 узла. Построить графики исходных функций и их интерполянтов.

Подобрать более эффективный метод приближения функции для второй задачи.

## 2 Используемые численные методы

В решении были использованы методы приближения многочленами Лагранжа и кубическими сплайнами.

**Приближение многочленами Лагранжа.** В [3] представлено теоретическое обоснование данного метода, в частности, доказана теорема о приближении (n+1)-раз дифференцируемой функции.

**Теорема 1** *Пусть* $f \in C^{(n+1)}[a,b]$. *Тогда*

$$f(x) - L_n(x) = \frac{f^{(n+1)}(\xi(x))}{(n+1)!}\,\omega(x), \quad \omega(x) = \prod_{k=0}^{n}(x - x_k), \tag{1}$$

*где*

$$\min\{x, x_0, \dots, x_n\} < \xi(x) < \max\{x, x_0, \dots, x_n\}. \tag{2}$$

Как будет видно, требование дифференцируемости здесь существенно, потому что иначе функция может и вовсе не приближаться полиномом.

Если полиномы $l_0(x), \dots, l_n(x)$ удовлетворяют условиям:

$$l_j(x) = \{1, i = j0, i \neq j\} \tag{3}$$

то

$$L_n(x) = \sum_{j=0}^{n} f(x_j) l_j(x) \tag{4}$$

А многочлены $l_j(x)$ задаются единственным образом и являются элементарными полиномами Лагранжа. Тогда легко видеть:

$$l_j(x) = \prod_{k=0, k\neq j}^{n} \frac{x - x_k}{x_j - x_k} \tag{5}$$

Таким образом и вычисляется интерполяционный полином Лагранжа в реализации.

**Приближение кубическими сплайнами.** В [3] также представлен метод приближение сплайнами. Сплайн представляет собой гладкий кусочно заданный полином.

**Определение 2.1 (Естественный сплайн)** *Кубический сплайн, обладающий следующим свойством*

$$S''(x_0) = S''(x_n) = 0, \tag{6}$$

*называется естественным сплайном.*

**Теорема 2** *Естественный сплайн существует и единственен.*

Обозначим

$$h_k = x_k - x_{k-1} \tag{7}$$

**Теорема 3** *Пусть* $1 \le j \le 4$ *и* $f \in C^j[a,b]$. *Тогда*

$$\|f - S_n\|_{C[a,b]} = O(h^j), \quad h \equiv \max_k h_k. \tag{8}$$

Таким образом, мы можем приближать функцию сплайнами, тогда нам становится проще бороться с негладкими функциями, если мы выберем точки разбиения там, где производная не существует.

Пусть $f(x)$ является сплайном, а $f_i(x)$ элементарным полиномом $deg f_i(x) = 3$ на сегменте $[x_{i-1}, x_i]$. Пусть у нас имеется сетка из $n+1$ узлов: $x_0, \dots, x_n$.

Наложим ограничения на сплайн в точках смыканий.

$$f_i = y_i \tag{9}$$
$$f'(x_i - 0) = f'(x_i + 0) \tag{10}$$
$$f''(x_i - 0) = f''(x_i + 0) \tag{11}$$
$$i = 1, \dots, n \tag{12}$$

Будем искать элементарный полином вида:

$$f_i(x) = a_i + b_i(x - x_{i-1}) + c_i(x - x_{i-1})^2 + d_i(x - x_{i-1})^3, x_{i-1} < x < x_i \tag{13}$$

Если расписать $f_i'(x)$ и $f_i''(x)$ в виде (13) и поставить ограничения (10) и (11), то у нас получится $4n - 2$ уравнений относительно $4n$ неизвестных $a_i, b_i, c_i, d_i, i = 1, \dots, n$. Добавив условия $c_1 = 0$ и $c_n + 3d_n h_n$ получим полную систему.

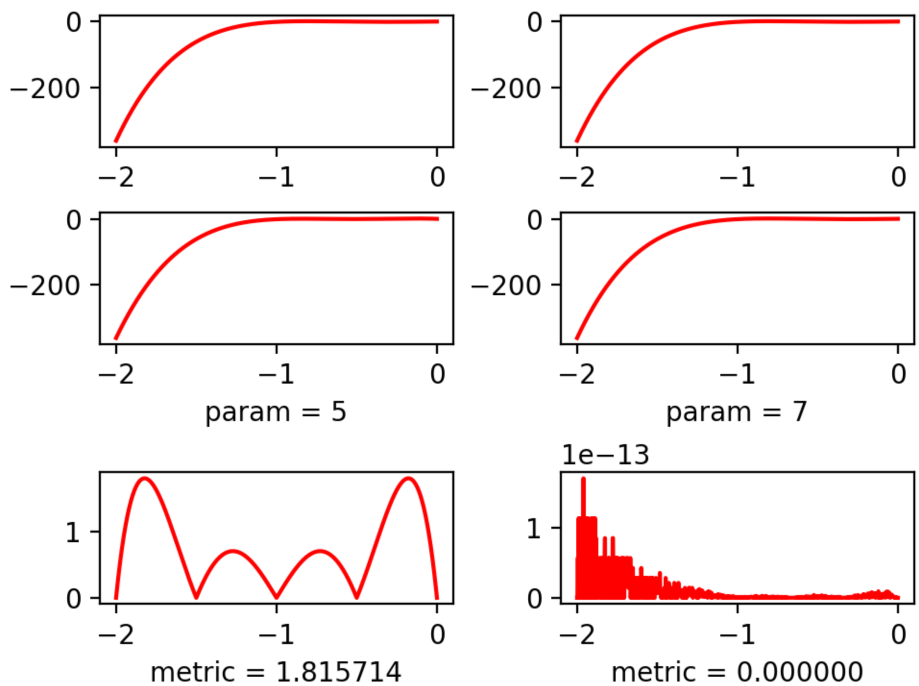
Можем выразить $a_i, b_i, d_i$ через $c_i$:

$$a_i = y_i \tag{14}$$
$$b_i = \frac{y_i - y_{i-1}}{h_i} - \frac{2c_i + c_{i+1}}{3} h_i \tag{15}$$
$$d_i = \frac{c_{i+1} - c_i}{3} \tag{16}$$
$$h_i c_i + 2(h_i + h_{i+1}) c_{i+1} + h_{i+1} c_{i+2} = 3\left(\frac{y_{i+1} - y_i}{h_{i+1}} + \frac{y_i - y_{i-1}}{h_i}\right) \tag{17}$$
$$i = 1, \dots, n-1 \tag{18}$$

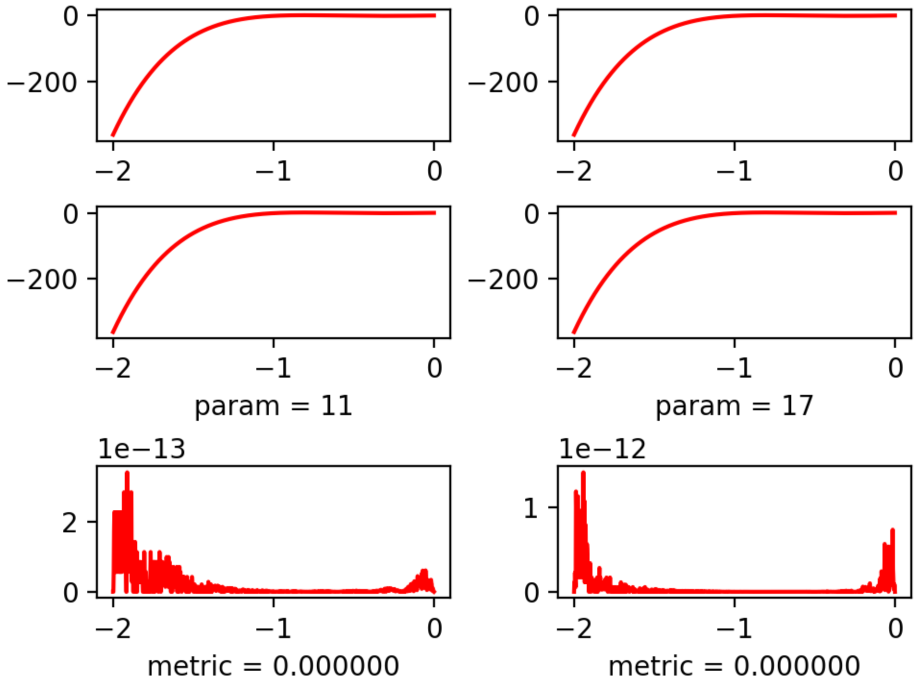
Мы имеем систему из $n-1$ уравнений относительно $n-1$ неизвестных $c_2, \dots, c_n$, данную систему можно решить методом прогонки. Этот метод был реализован в коде.

# 3 Результаты

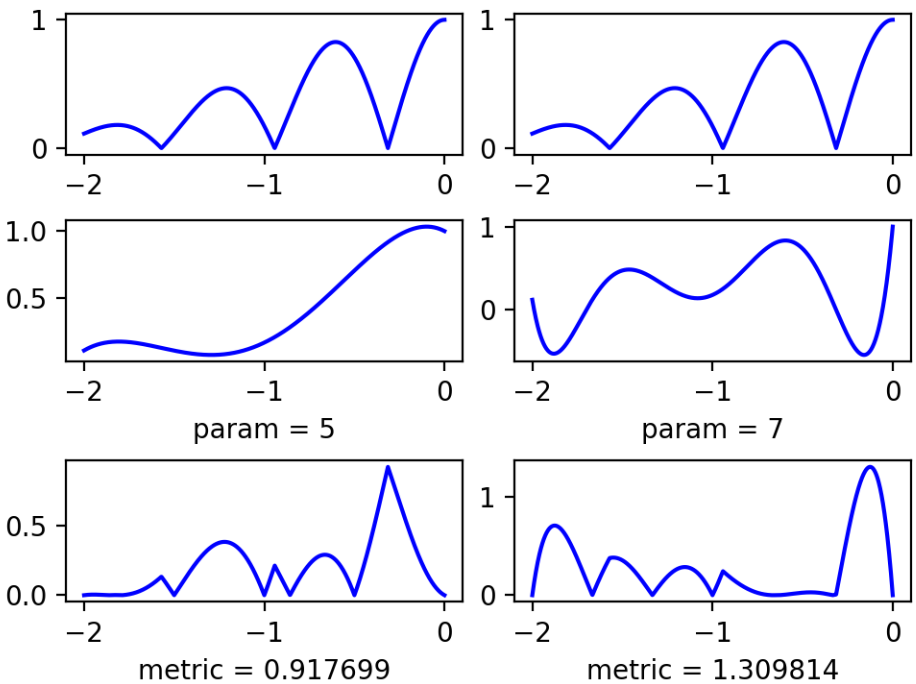
## 3.1 Приближение многочленами Лагранжа



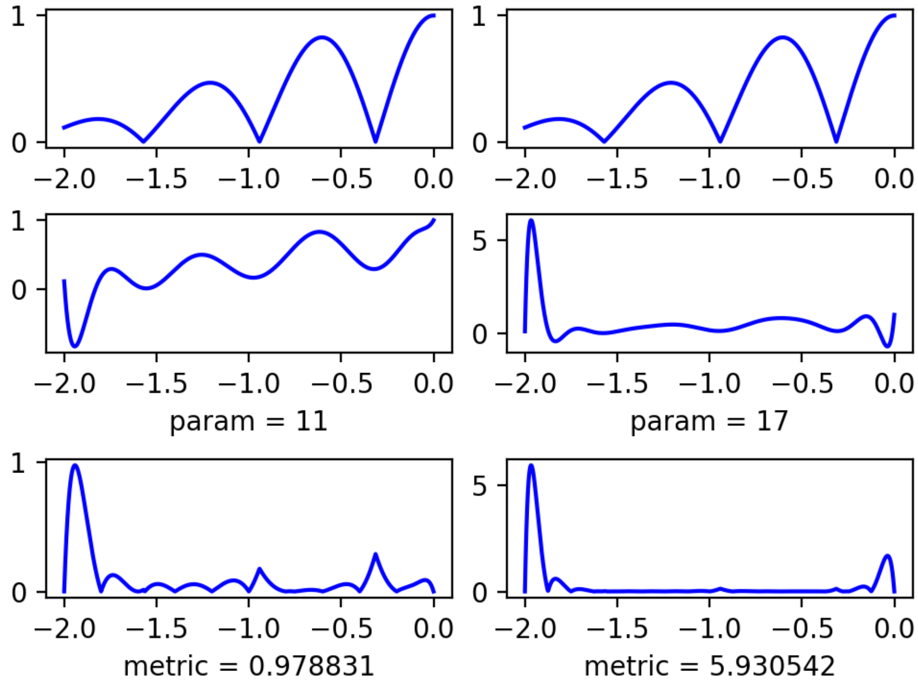

[Рис. 1] Интерполяция функции (1)




[Рис. 2] Интерполяция функции (1)




[Рис. 3] Интерполяция функции (2)

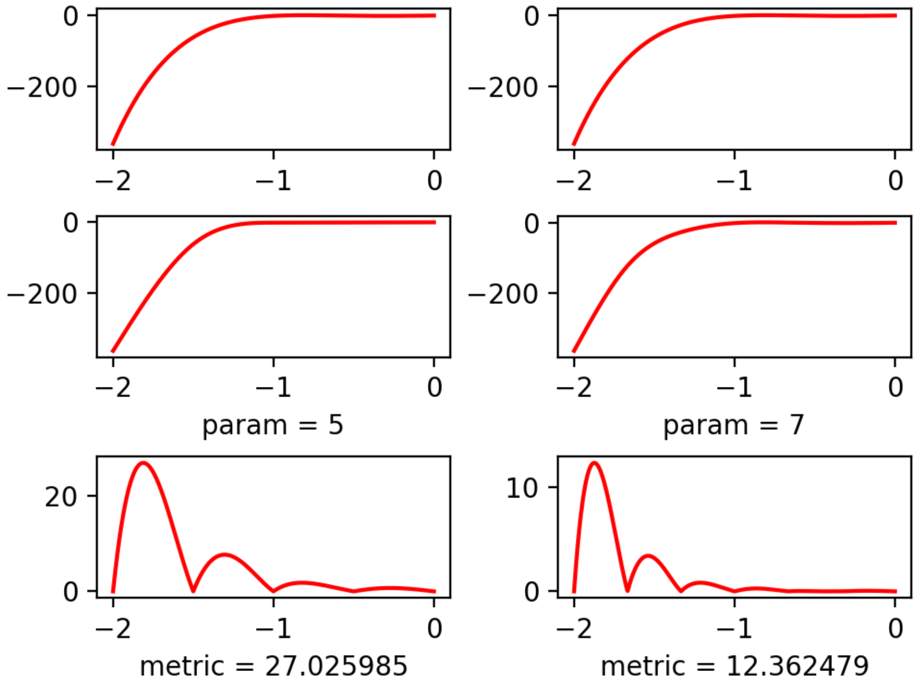



[Рис. 4] Интерполяция функции (2)

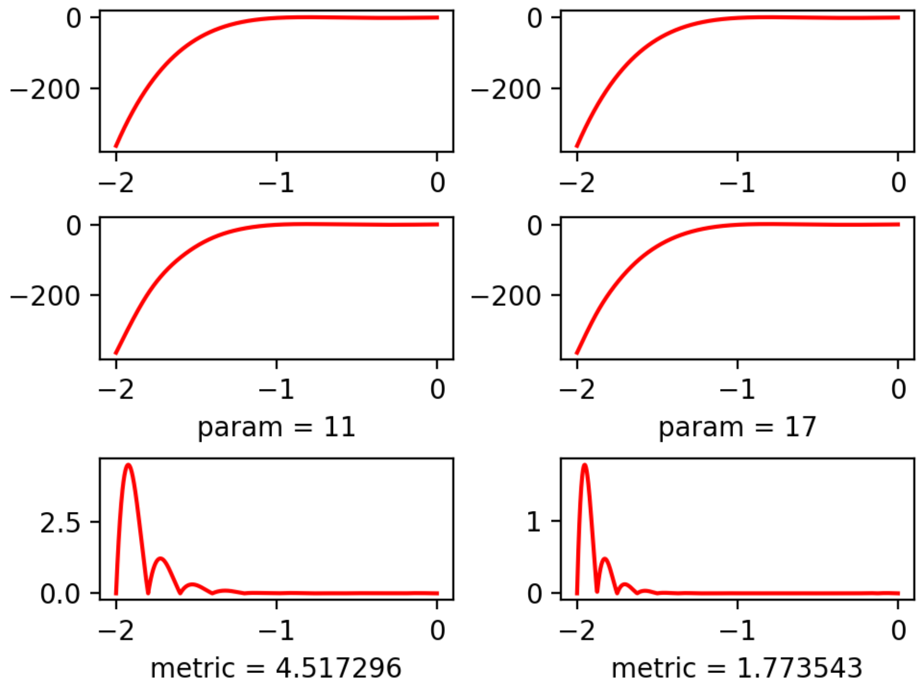
| | $f_1(x)$ | $f_2(x)$ |
|---|---|---|
| n=5 | 1.86 | 0.92 |
| n=7 | 0.00 | 1.31 |
| n=11 | 0.00 | 0.98 |
| n=17 | 0.00 | 5.93 |

[Таблица 1.] Максимальные отклонения интерполянтов от функций.

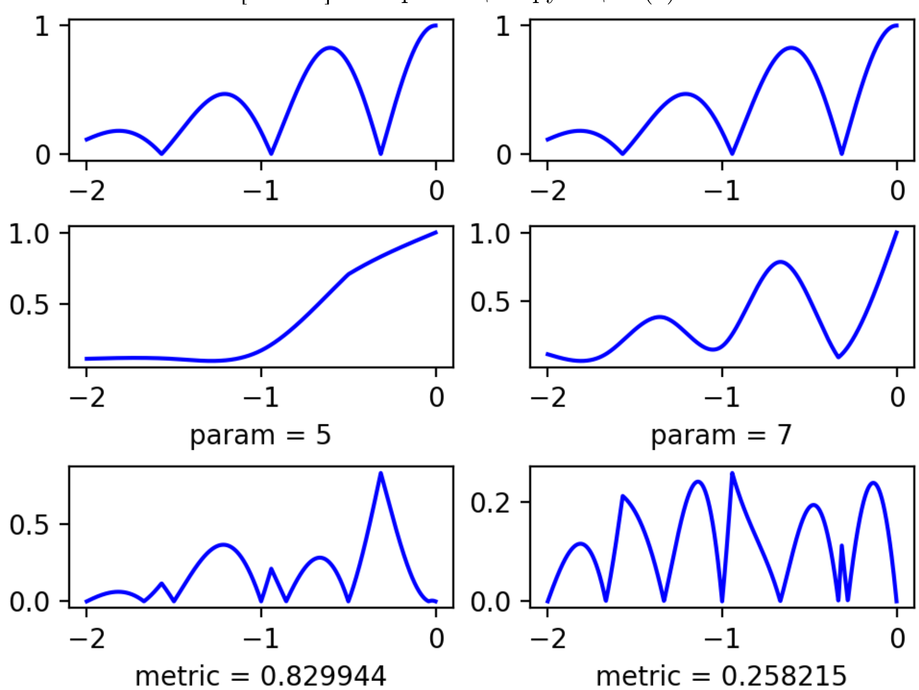
## 3.2 Приближение сплайнами



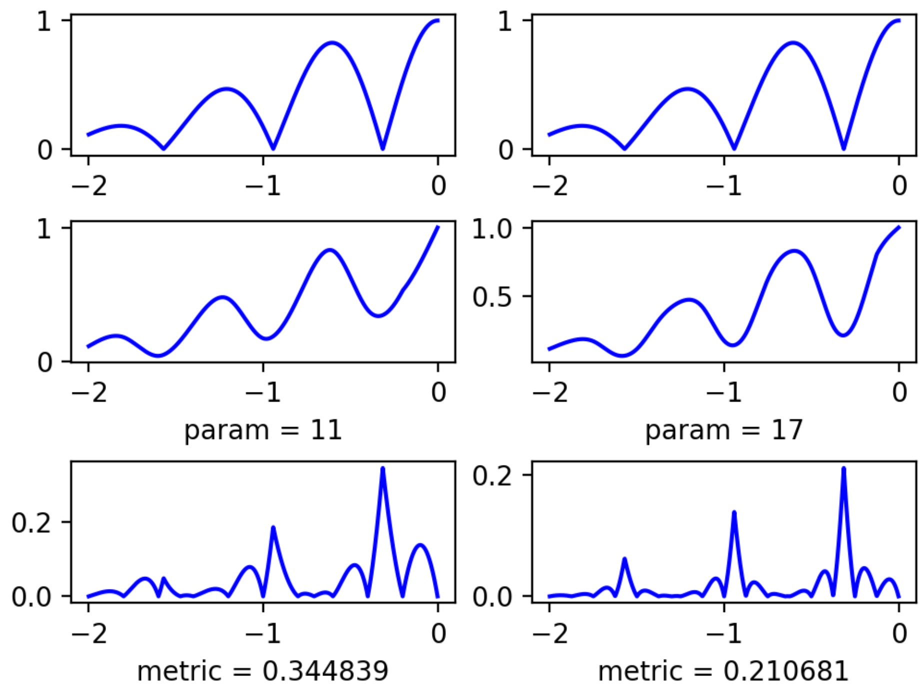

[Рис. 5] Интерполяция функции (1)




[Рис. 6] Интерполяция функции (1)




[Рис. 7] Интерполяция функции (2)




[Рис. 8] Интерполяция функции (2)

| | $f_1(x)$ | $f_2(x)$ |
|---|---|---|
| n=5 | 27.03 | 0.83 |
| n=7 | 12.36 | 0.26 |
| n=11 | 4.52 | 0.34 |
| n=17 | 1.77 | 0.21 |

[Таблица 2.] Максимальные отклонения интерполянтов от функций.

### 3.3 Комментарии к графикам

В 1 строке представлены интерполируемые функции $f_i(x)$. В 2 строке представлены интерполянты $F_i(x)$. В 3 строке представлена функция $g(x) = |f_i(x) - F_i(x)|$

На графиках параметр $\mathrm{param}$ означает количество узлов в равномерной сетке, по которой проходила интерполяция. Значение $\mathrm{metric}$ показывает максимальное отклонение интерполянта от функции на равномерной сетке из 1000 узлов.

### 3.4 Наблюдения

Метод интерполяции многочленами Лагранжа позволяет хорошо приближать многочлены, как видим, уже на сетке из 7 узлов, мы полностью вычислили функцию (1). На рис. 1 видно, что максимальное отклонение на сетке из 5 узлов равно 1.8147, а на сетке из 7 узлов равно 0.00, на рис. 2 на сетках из 11 и 17 узлов максимальное отклонение равно 0.00 и 0.00, соответственно. При этом мы совершенно не можем приблизить функцию (2). Видно, что на рис. 3 на сетке из 5 узлов ошибка составляет 0.9177, а на сетке из 7 узлов уже 1.3098, на рис. 4 на сетке из 11 и 17 узлов 0.9788 и 5.9305, соответственно.

Метод интерполяции сплайнами показывает себя лучше в задаче интерполяции функции (2), чем метод (1), к примеру, на рис. 8 на сетке из 17 узлов ошибка доходит до 0.2107, что улучшает результат метода (1) почти в 25 раз, при этом метод (2) даёт худший результат при интерполяции функции (2), так на рис. 6 видно, что на сетке из 17 узлов ошибка достигает 1.7735.

### 3.5 Программная реализация

Данные численные методы реализованы на языке Python с использованием NumPy, matplotlib.

Код хранится в репозитории: https://github.com/GoodDay-lab/practicum-numerical-method

## 4 Выводы

Таким образом, в ходе выполнения задания был реализован метод приближения полиномами Лагранжа и сплайнами. Было отмечено и теоретически доказано, что полиномы Лагранжа могут эффективно приближать функции $f \in C^n[a, b]$, давая достаточно хорошую оценку. При этом совершенно неопределено поведение для негладких функций, и как мы видим, их они могут и вовсе не приближать. В таком случае можно использовать приближение сплайнами - кусочно-полиномиальными функциями. В этом случае оценка становится в разы лучше для негладких функций.

## 5 Библиография


1. Хайкин С. "Нейронные сети. Полный курс", стр.82
2. (url: https://education.yandex.ru/handbook/ml/article/mashinnoye-obucheniye)
3. Тыртышников Е.Е. "Методы численного анализа", гл.12,13,14
4. http://www.machinelearning.ru/wiki/
5. А.А.Самарский, А.В.Гулин. "Численные методы"