МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)
ФАКУЛЬТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И ПРИКЛАДНОЙ МАТЕМАТИКИ
КАФЕДРА ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ И ПРОГРАММИРОВАНИЯ

КУРСОВАЯ РАБОТА
ПО ДИСЦИПЛИНЕ «ЧИСЛЕННЫЕ МЕТОДЫ»

# «**ИНТЕРПОЛЯЦИЯ ЭКСПОНЕНЦИАЛЬНЫМИ СПЛАЙНАМИ**»

Выполнил:
Студент группы М8О-405Б-20
Манташев А. У.
Преподаватель:
Иванов И. Э.

Оценка: ____

Дата: ______________

Москва
2023

## Постановка задачи

Аппроксимация некоторой таблично заданной функции с помощью экспоненциальных сплайнов и получение коэффициентов экспоненциальной функции.

## Теоретические сведения

Интерполяция экспоненциальными сплайнами - это метод аппроксимации данных, при котором каждый участок сплайна аппроксимируется экспоненциальной функцией. Этот метод позволяет более гибко и точно аппроксимировать данные, которые имеют экспоненциальный характер или требуют гладкой кривой аппроксимации.

На основе экспоненты представляются функции синус и косинус – гиперболические и тригонометрические. Такого рода обобщения называются экспоненциальными L-сплайнами и предлагаются для самого широкого использования. Однако проработанные теоретические основы оказываются весьма далекими от практического внедрения. Общие формулы, рассчитанные на все случаи жизни, нередко оказываются настолько сложными, что их не применяют ни в одном конкретном случае. Трудоемкость же вычислений часто оказывается существенно меньшей при применении рекуррентных процедур. Поэтому для практической деятельности по части формирования градуировочных зависимостей сенсорных устройств возникла необходимость выполнить интерполяцию экспонентами, хоть и не по обобщенной методике, но в значительной мере универсальной, ограничиваемой лишь условием монотонности возрастания (или убывания) требуемой функции.

Общий вид аппроксимационной фунции для трех произвольных точек имеет вид:

$$y = C + B * \exp(A * x).$$

Параметр C вычисляется по формуле: $\mathrm{C} = \frac{y_2^2 - y_1 y_3}{2y_2 - y_1 - y_3}$.

Параметр А вычисляется по формуле: $A = \frac{\ln\left(\frac{y_3-y_2}{y_2-y_1}\right)}{y_3-y_2}$.

Параметр В вычисляется по формуле:

$$B = (y_1 - C)\exp(-Ax_1) = (y_1 - \frac{y_2^2 - y_1 y_3}{2y_2 - y_1 - y_3})\exp\left(-\frac{\ln\left(\frac{y_3-y_2}{y_2-y_1}\right)}{y_3-y_2}x_1\right)$$

или $B = (y_2 - C)\exp(-Ax_2) = (y_2 - \frac{y_2^2 - y_1 y_3}{2y_2 - y_1 - y_3})\exp\left(-\frac{\ln\left(\frac{y_3-y_2}{y_2-y_1}\right)}{y_3-y_2}x_2\right)$

$$B = (y_3 - C)\exp(-Ax_3) = (y_3 - \frac{y_2^2 - y_1 y_3}{2y_2 - y_1 - y_3})\exp\left(-\frac{\ln\left(\frac{y_3-y_2}{y_2-y_1}\right)}{y_3-y_2}x_3\right).$$

На вход программы подается массив точек (x, y). Для каждых трех точек, начиная с $x_i, i = 1, \dots, n-2$, где $n$ – количество точек, должна быть построена экспоненциальная функция в виде $y_i = C_i + B_i * \exp(A_i * x),\ i = 1, \dots, n-2$.

## Листинг кода

```
import numpy as np
import matplotlib.pyplot as plt

def F(iA, iB, iC, x):
    return iC + iB * np.exp(iA * x)

def e_interp(x, y):

    A = np.zeros(len(x) - 2)
    B = np.zeros(len(x) - 2)
    C = np.zeros(len(x) - 2)

    for i in range(len(A)):
        dif = (y[i + 2] - y[i + 1]) * (x[i + 1] - x[i]) / ((y[i
+ 1] - y[i]) * (x[i + 2] - x[i + 1]))

        if abs(dif) == 1:
            A[i] = (y[i + 2] - y[i]) / (x[i + 2] - x[i])
            B[i] = y[i] - A[i] * x[i]
            print('Допустима только линейная интерполяция')
            C[i] = 0
```

```
        elif x[i + 2] + x[i] == x[i + 1] + x[i + 1]:
            z = (2 * y[i + 1] - y[i] - y[i + 2])
            r = (y[i + 2] - y[i + 1]) / (y[i + 1] - y[i])
            A[i] = np.log(r) / (x[i + 2] - x[i + 1])
            C[i] = (y[i + 1] * y[i + 1] - y[i] * y[i + 2]) / z
            B[i] = (y[i] - C[i]) * np.exp(-A[i] * x[i])

        else:

            Amin = np.log(dif) / (x[i + 2] - x[i])
            A0 = 2 * Amin
            while (1):
                u = np.exp(A0 * (x[i + 2] - x[i + 1]))
                v = np.exp(-A0 * (x[i + 1] - x[i]))
                F = (y[i + 1] - y[i]) * (u - 1) + (y[i + 2] -
y[i + 1]) * (v - 1)
                FF = (y[i + 1] - y[i]) * (x[i + 2] - x[i + 1]) *
u - (y[i + 2] - y[i + 1]) * (x[i + 1] - x[i]) * v
                dA = -F / FF
                A0 += dA
                if (abs(dA / Amin) < 0.000000001):
                    break

            A[i] = A0
            B[i] = (y[i] - y[i + 1]) / (np.exp(A0 * x[i]) -
np.exp(A0 * x[i + 1]))
            C[i] = y[i] - B[i] * np.exp(A0 * x[i])

    return A, B, C


def get_graphics(A, B, C, x, y):

    n = len(A)
    x1 = np.arange(x[0], x[1] + 0.01, 0.01)
    y1 = C[0] + B[0] * np.exp(A[0] * x1)
    plt.title("Экспоненциальная интерполяция:")
    plt.plot(x1, y1, color="red")
    plt.scatter(x, y, c='b')

    for i in range(1, n):
        x1 = np.arange(x[i], x[i + 1] + 0.01, 0.01)
        y1 = ((x[i + 1] - x1) * (C[i - 1] + B[i - 1] *
np.exp(A[i - 1] * x1)) + (x1 - x[i]) * (C[i] + B[i] *
np.exp(A[i] * x1))) / (x[i + 1] - x[i])
```

```
        plt.plot(x1, y1, color="red")

    x1 = np.arange(x[-2], x[-1] + 0.01, 0.01)
    y1 = C[n - 1] + B[n - 1] * np.exp(A[n - 1] * x1)
    plt.plot(x1, y1, color="red")
    plt.grid()
    plt.show()


x = [0, 1, 2, 3, 4, 5]
y = [1, 4, 6, 10, 12, 15]

x1 = [0, 4, 7, 9, 10]
y1 = [2, 4, 6, 15, 20]

x3 = [0, 1, 2, 3, 3.5, 5, 7, 10]
y3 = [500, 1000, 2230, 3040, 3974, 4587, 5000, 5500]


A, B, C  = e_interp(x, y)
for i in range(len(x)-2):
    print(f'Экспоненциальная интерп. функция {i+1}-го сегмента:
y = {C[i]} + {B[i]} * exp({A[i]} * x)')

get_graphics(A, B, C, x, y)
```

## Результаты

Экспоненциальная интерп. функция 1-го сегмента:
y = 10.0 + -9.0 * exp(-0.40546510810816444 * x)
Экспоненциальная интерп. функция 2-го сегмента:
y = 2.0 + 1.0 * exp(0.6931471805599453 * x)
Экспоненциальная интерп. функция 3-го сегмента:
y = 14.0 + -32.0 * exp(-0.6931471805599453 * x)
Экспоненциальная интерп. функция 4-го сегмента:
y = 6.0 + 1.1851851851851851 * exp(0.4054651081081644 * x)

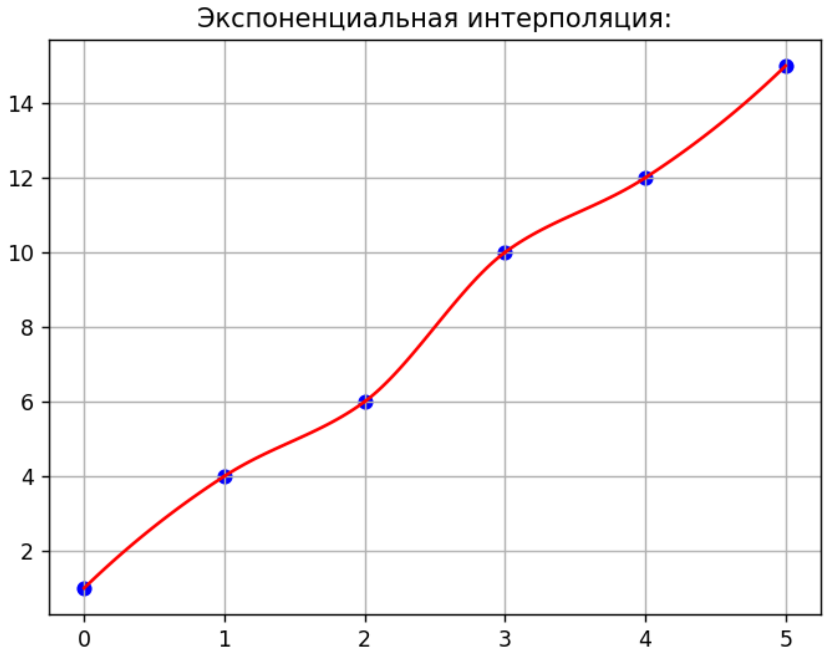



Рис. 1

Экспоненциальная интерп. функция 1-го сегмента:
y = -3.096284316193442 + 5.096284316193442 * exp(0.08276490064167358 * x)
Экспоненциальная интерп. функция 2-го сегмента:
y = 3.810479968492467 + 0.007256547616173988 * exp(0.8156476363319497 * x)
Экспоненциальная интерп. функция 3-го сегмента:
y = -53.29455265819017 + 36.158912455666105 * exp(0.07065628312399189 * x)

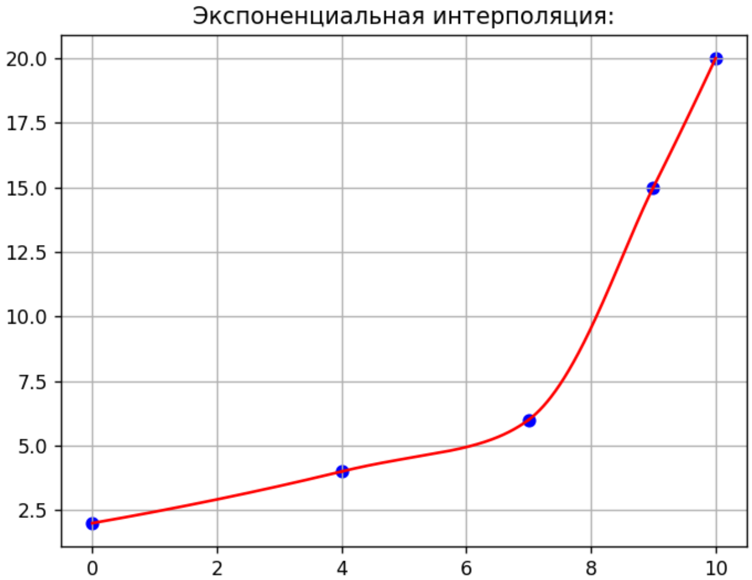
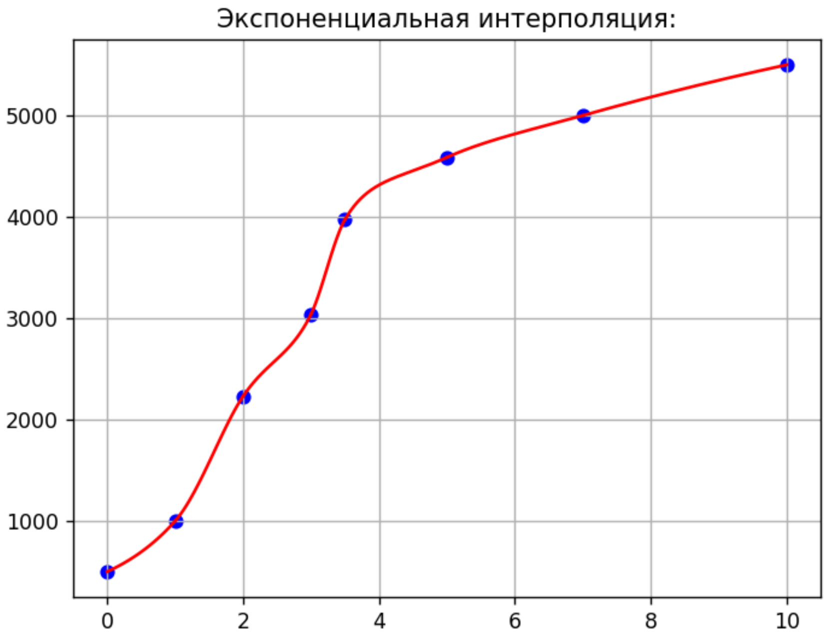



Рис. 2

Экспоненциальная интерп. функция 1-го сегмента: y = 157.53424657534248 + 342.4657534246575 * exp(0.9001613499442714 * x)
Экспоненциальная интерп. функция 2-го сегмента: y = 4602.142857142857 + -5469.920634920634 * exp(-0.41773520069997866 * x)
Экспоненциальная интерп. функция 3-го сегмента: y = 1865.6733234825624 + 35.06671332592055 * exp(1.1703992929479687 * x)
Экспоненциальная интерп. функция 4-го сегмента: y = 4633.813657188932 + -316617.56460504414 * exp(-1.7638549846230158 * x)
Экспоненциальная интерп. функция 5-го сегмента: y = 5341.247659286375 + -5478.033653720391 * exp(-0.39655614371350795 * x)
Экспоненциальная интерп. функция 6-го сегмента: y = 7191.075659186878 + -4009.998016310037 * exp(-0.08634259354421288 * x)




## Прикладные задачи

В настоящее время имеется значительное число работ, посвященных численным аспектам аппроксимации сплайнами функций одного переменного, связанных с локальным наследованием сплайном свойств монотонности и выпуклости исходных данных.

Однако сохранение монотонности и выпуклости, как правило, удается достичь только при дополнительных, довольно жестких ограничениях на исходные данные. При этом построенные локальные процессы приближенного восстановления кривых были либо интерполяционного типа, либо обеспечивали точность для алгебраических многочленов заданной степени.

Формосохраняющие свойства можно достичь с помощью аппроксимации экспоненциальными сплайнами.

Сплайн же можно использовать для уменьшения веса хранимых данных. Легче хранить коэффициенты аппроксимации, чем всю таблицу точек. Например, если нужно хранить форму крыла, которая имеет вид, похожий на функцию экспоненты.

Формулу просто интегрировать, находить производную. К ней можно применять математические методы для исследования. Это следует из свойств функции экспоненты.

Также важной областью применения алгоритма является робототехника, прежде всего – сенсорные устройства. Обработка сигналов от датчиков обычно происходит с применением градуировочных функций, определяющих зависимость электрического сигнала от измеряемых физических величин внешнего пространства. Известно, что для обработки экспериментальных данных выбор метода аппроксимации (интерполяции) целесообразно осуществлять с учетом условий, характерных для данного анализируемого эксперимента: характер изменения данных, количество точек, точность измерения. Для градуировочных функций типично, что природа нам предоставляет монотонно возрастающие зависимости. Однако возрастание может быть неравномерным и переходящим в асимптотическое приближение к некоторому предельному уровню. Если на таких участках получены редкие исходные данные, то обычные способы построения аппроксимационной (интерполяционной) кривой с помощью полиномов дадут неверную картину с выпуклостями. В связи с этим заслуживают внимание экспоненты.

## Обоснование выбора формулы

Построение суммы экспонент целесообразно в тех случаях, когда происхождение исходной таблично заданной функции имеет экспоненциальную природу (например, результаты измерений интенсивности излучения смеси радиоактивных материалов).

На основе экспоненты представляются функции синус и косинус – гиперболические и тригонометрические. Такого рода обобщения называются экспоненциальными L-сплайнами, однако проработанные теоретические основы оказываются далекими от практического внедрения. Они теоретически могут использоваться для массы задач, но практически настолько сложны, что этого никто не делает. Поэтому для практической деятельности по части формирования градуировочных зависимостей сенсорных устройств возникла необходимость выполнить интерполяцию экспонентами, хоть и не по обобщенной методике, но в значительной мере универсальной, ограничиваемой лишь условием монотонности возрастания (или убывания) требуемой функции. Это обосновывает выбор формулы для расчета функции в теоретической справке.

## Вывод

В ходе выполнения курсового проекта я познакомился с экспоненциальными сплайнами. Была разработана программная реализация алгоритма интерполяции экспоненциальными сплайнами, рассчитывающего коэффициенты A, B, и C в уравнении $y = C + B \exp(Ax)$. Интерполяция экспоненциальными сплайнами является мощным инструментом для аппроксимации экспоненциальных данных, что позволяет более точно моделировать экспоненциальные зависимости.

## Источники

1. Моделирование вычислительных, телекоммуникационных, управляющих и социально-экономических систем. Алгоритм интерполяции возрастающей функции экспоненциальными сплайнами.

А. С. Ильин. URL: Алгоритм интерполяции возрастающей функции экспоненциальными сплайнами (cyberleninka.ru).

2. Аппроксимация локальными экспоненциальными сплайнами с произвольными узлами Е.В. Шевалдина.