

Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»

Факультет Программной Инженерии и Компьютерной Техники

**Лабораторная работа по дисциплине Вычислительная Математика  
№4**

Вариант 12

Преподаватель: Машина Екатерина Алексеевна

Выполнил: Печкуров Данила Алексеевич

Группа: P3208

Санкт-Петербург, 2024г

## Цель работы:

### Вычислительная часть лабораторной работы:

## Линейная аппроксимация

[illegible]

$$\begin{cases} a \sum_{i=1}^n x_i^2 + b \sum_{i=1}^n x_i = \sum_{i=1}^n x_i y_i \\ a \sum_{i=1}^n x_i + bn = \sum_{i=1}^n y_i \end{cases}$$

Решив систему получим:  $a=0.1578$ ;  $b=-0.0793$ ;

Искомое уравнение линейной регрессии имеет вид:

$$y = 0.1578x - 0.0793$$

-2	-1,8	-1,6	-1,4	-1,2	-1	-0,8	-0,6	-0,4	-0,2	0	$x_i$
-0,2857	-0,3200	-0,3449	-0,3535	-0,3411	-0,3077	-0,2579	-0,1979	-0,1330	-0,0667	0,0000	$y_i$
-0,3949	-0,36334	-0,33178	-0,30022	-0,26866	-0,2371	-0,20554	-0,17398	-0,14242	-0,11086	-0,0793	$\phi_i$
0,1092	0,0433	-0,0132	-0,0533	-0,0724	-0,0706	-0,0523	-0,0239	0,0094	0,0442	0,0793	$\epsilon$

$$\sigma = 0,0593$$

## Квадратичная аппроксимация

[illegible]

$$\begin{cases} a_0 n + a_1 \sum_{i=1}^n x_i + a_2 \sum_{i=1}^n x_i^2 = \sum_{i=1}^n y_i \\ a_0 \sum_{i=1}^n x_i + a_1 \sum_{i=1}^n x_i^2 + a_2 \sum_{i=1}^n x_i^3 = \sum_{i=1}^n x_i y_i \\ a_0 \sum_{i=1}^n x_i^2 + a_1 \sum_{i=1}^n x_i^3 + a_2 \sum_{i=1}^n x_i^4 = \sum_{i=1}^n x_i^2 y_i \end{cases}$$

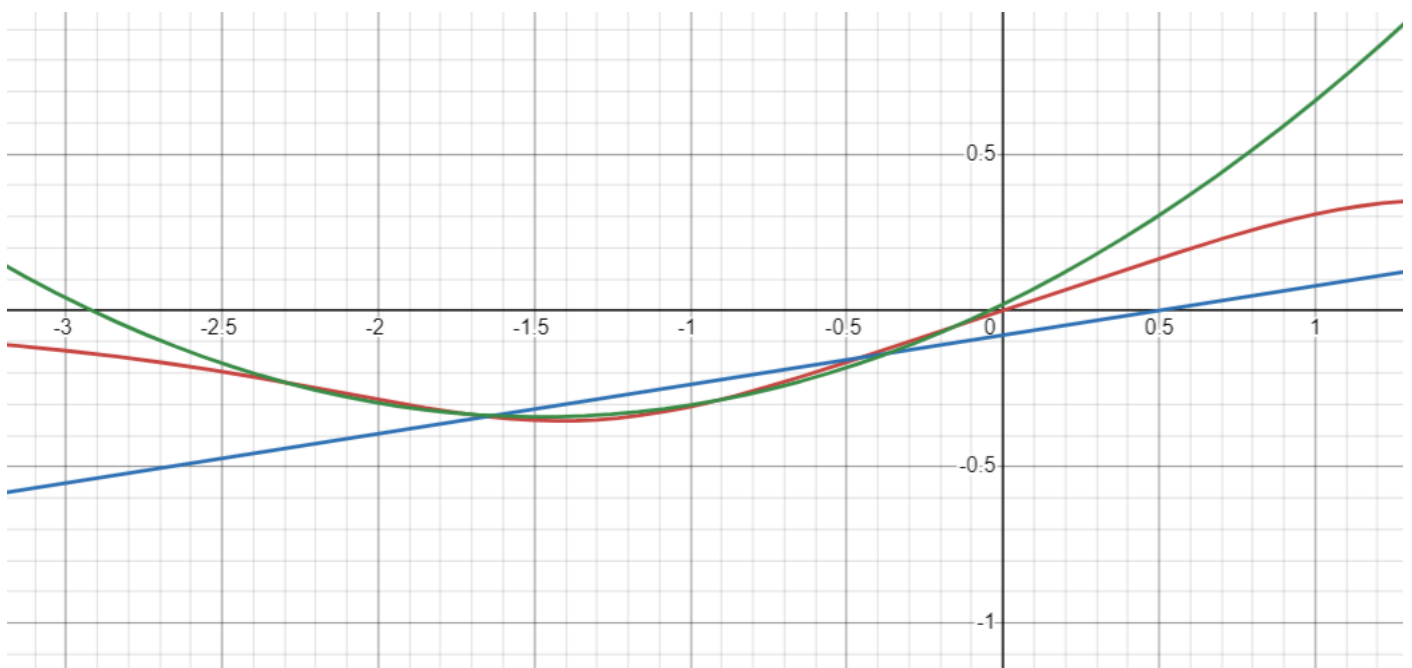
Решив систему получим:  $a_0=0.1646$ ;  $a_1=0.487$ ;  $a_2=0.0194$

Искомое уравнение квадратичной регрессии имеет вид:

$$y=0.1646x^2+0.487x+0.0194$$

-2	-1,8	-1,6	-1,4	-1,2	-1	-0,8	-0,6	-0,4	-0,2	0	$x_i$
-0,2857	-0,3200	-0,3449	-0,3535	-0,3411	-0,3077	-0,2579	-0,1979	-0,1330	-0,0667	0,0000	$y_i$
-0,2962	-0,323896	-0,338424	-0,339784	-0,327976	-0,303	-0,264856	-0,213544	-0,149064	-0,071416	0,0194	$\phi_i$
0,0105	0,0039	-0,0065	-0,0137	-0,0131	-0,0047	0,0070	0,0157	0,0160	0,0048	-0,0194	epsilon

$$\sigma = 0,0117$$



### Листинг программы:

```
def linear_least_squares(x, y):
    n = len(x)
    sum_x = sum(x)
    sum_y = sum(y)
    sum_xx = sum(xi ** 2 for xi in x)
    sum_xy = sum(xi * yi for xi, yi in zip(x, y))

    a = (n * sum_xy - sum_x * sum_y) / (n * sum_xx - sum_x ** 2)
    b = (sum_y - a * sum_x) / n

    return a, b

def polynomial_least_squares(x, y, degree):
    n = len(x)
    X = [[xi ** j for j in range(degree, -1, -1)] for xi in x]
    XT = [[row[i] for row in X] for i in range(degree + 1)]
```

```

A = [[sum(XT[i][k] * XT[j][k] for k in range(n)) for j in
range(degree + 1)] for i in range(degree + 1)]

B = [sum(yi * XT[i][k] for k, yi in enumerate(y)) for i in
range(degree + 1)]

for i in range(degree + 1):
    div = A[i][i]
    A[i] = [a_i / div for a_i in A[i]]
    B[i] /= div
    for j in range(i + 1, degree + 1):
        mult = A[j][i]
        A[j] = [a_j - mult * a_i for a_i, a_j in zip(A[i], A[j])]
        B[j] -= mult * B[i]

coeffs = [0] * (degree + 1)
for i in range(degree, -1, -1):
    coeffs[i] = B[i]
    for j in range(i + 1, degree + 1):
        coeffs[i] -= A[i][j] * coeffs[j]

return coeffs

```

```

def exponential_least_squares(x, y):
    if any(yi <= 0 for yi in y):
        raise ValueError("Отрицательные или нулевые значения y
недопустимы для экспоненциальной аппроксимации")

    y_log = [math.log(yi) for yi in y]
    a, b = linear_least_squares(x, y_log)
    A = math.exp(b)
    B = a

    return A, B

```

```

def logarithmic_least_squares(x, y):
    if any(yi <= 0 for yi in y):
        raise ValueError("Отрицательные или нулевые значения y
недопустимы для логарифмической аппроксимации")

    x_log = [math.log(xi) for xi in x]
    a, b = linear_least_squares(x_log, y)

    return a, b

```

```

def power_least_squares(x, y):
    if any(yi <= 0 for yi in y):
        raise ValueError("Отрицательные или нулевые значения y
недопустимы для степенной аппроксимации")

    x_log = [math.log(xi) for xi in x]
    y_log = [math.log(yi) for yi in y]

```

```

a, b = linear_least_squares(x_log, y_log)
A = math.exp(b)
B = a

return A, B

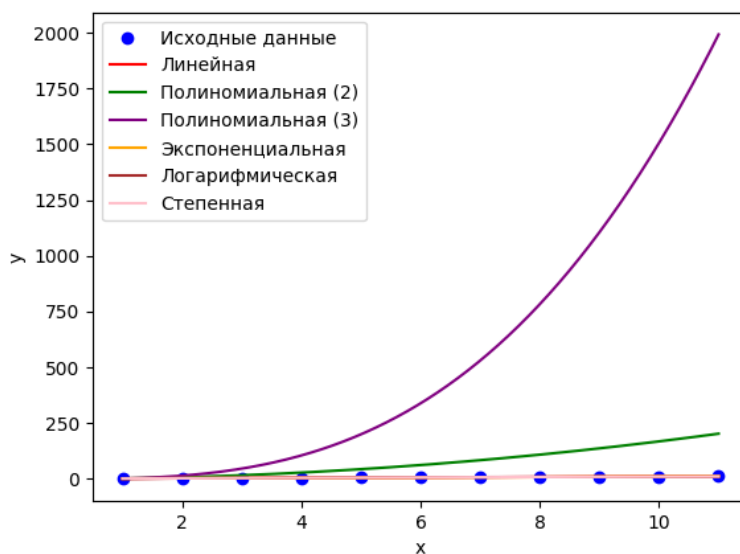
```

## Пример и результаты работы программы:

```

Хотите ли вы ввести данные из файла? (y/n): y
Введите имя файла: input.txt
Коэффициенты линейной аппроксимации: a = 1.011818181818182, b = 1.183636363636362
Коэффициенты полиномиальной аппроксимации (2-й степени): [0.016200466200464, 0.8174125874126146, 1.604848484848423]
Коэффициенты полиномиальной аппроксимации (3-й степени): [0.001806526806529886, -0.016317016317073052, 0.9803613053615989, 1.4075757575753867]
Коэффициенты экспоненциальной аппроксимации: a = 2.505834990426823, b = 0.15777355944606403
Коэффициенты логарифмической аппроксимации: a = 4.22333461745666, b = 0.534717944467046
Коэффициенты степенной аппроксимации: a = 2.084541990051234, b = 0.7106409248221797
Среднеквадратичное отклонение (линейная): 0.19351041571898797
Среднеквадратичное отклонение (полиномиальная 2-й степени): 94.56135219258299
Среднеквадратичное отклонение (полиномиальная 3-й степени): 872.9178779471284
Среднеквадратичное отклонение (экспоненциальная): 0.6756845236484322
Среднеквадратичное отклонение (логарифмическая): 1.1281775076959666
Среднеквадратичное отклонение (степенная): 0.49981569232364204
Коэффициент корреляции Пирсона (линейная): 0.9999999999999992
Коэффициент детерминации (линейная): 0.9999999999999984
Коэффициент детерминации (полиномиальная 2-й степени): 0.9525211201745355
Коэффициент детерминации (полиномиальная 3-й степени): 0.8660734568630386
Коэффициент детерминации (экспоненциальная): 0.9547368353321967
Коэффициент детерминации (логарифмическая): 0.9001140052744704
Коэффициент детерминации (степенная): 0.9929140081373027
Наилучшая аппроксимирующая функция: linear

```



## Вывод:

В результате выполнения данной лабораторной работы были изучены методы для нахождения аппроксимирующих функций, приближающим функцию, заданную множеством её точек.

