

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
Высшего образования
Факультет Программной Инженерии и Компьютерной Техники

Лабораторная работа №4 по Вычислительной Математике

Аппроксимация функции методом наименьших квадратов

Вариант №5

Группа: Р3214

Выполнил:

Минкова Алина Андреевна

Проверил:

Малышева Татьяна Алексеевна

Г. Санкт-Петербург

2024

Оглавление

Цель работы	3
Порядок выполнения работы	4
Рабочие формулы используемых методов	5
Вычислительная часть	8
Листинг программы	11
Графики аппроксимирующих функций.....	18
Результат выполнения программы.....	13
Вывод.....	19

Цель работы

Найти функцию, являющуюся наилучшим приближением заданной табличной функции по методу наименьших квадратов.

Порядок выполнения работы

Вычислительная реализация задачи:

1. Сформировать таблицу табулирования заданной функции на указанном интервале;
2. Построить линейное и квадратичное приближения по 11 точкам заданного интервала;
3. Найти среднеквадратические отклонения для каждой аппроксимирующей функции. Ответы дать с тремя знаками после запятой;
4. Выбрать наилучшее приближение;
5. Построить графики заданной функции, а также полученные линейное и квадратичное приближения.

Программная реализация задачи:

Для исследования использовать:

- линейную функцию,
- полиномиальную функцию 2-й степени,
- полиномиальную функцию 3-й степени,
- экспоненциальную функцию,
- логарифмическую функцию,
- степенную функцию.

Методика проведения исследования:

1. Вычислить меру отклонения для всех исследуемых функций;
2. Уточнить значения коэффициентов эмпирических функций, минимизируя функцию S ;
3. Сформировать массивы предполагаемых эмпирических зависимостей;
4. Определить среднеквадратичное отклонение для каждой аппроксимирующей функции. Выбрать наименьшее значение и, следовательно, наилучшее приближение;
5. Построить графики полученных эмпирических функций.

Задание:

1. Предусмотреть ввод исходных данных из файла/консоли (таблица $y = f(x)$ должна содержать от 8 до 12 точек);
2. Реализовать метод наименьших квадратов, исследуя все указанные функции;
3. Предусмотреть вывод результатов в файл/консоль: коэффициенты аппроксимирующих функций, среднеквадратичное отклонение, массивы значений x_i , y_i , $f(x_i)$, e_i ;
4. Для линейной зависимости вычислить коэффициент корреляции Пирсона;
5. Вычислить коэффициент детерминации, программа должна выводить соответствующее сообщение в зависимости от полученного значения R^2 ;
6. Программа должна отображать наилучшую аппроксимирующую функцию;
7. Организовать вывод графиков функций, графики должны полностью отображать весь исследуемый интервал (с запасом);
8. Программа должна быть протестирована при различных наборах данных, в том числе и некорректных.

Рабочие формулы используемых методов

Линейная аппроксимация

Рассмотрим в качестве эмпирической формулы линейную функцию:

$$\varphi(x, a, b) = ax + b$$

Сумма квадратов отклонений запишется следующим образом:

$$S = S(a, b) = \sum_{i=1}^n \varepsilon_i^2 = \sum_{i=1}^n [\varphi(x_i) - y_i]^2 = \sum_{i=1}^n (ax_i + b - y_i)^2 \rightarrow \min$$

Для нахождения a и b необходимо найти минимум функции $S(a, b)$.

Необходимое условие существования минимума для функции S :

$$\begin{cases} \frac{\partial S}{\partial a} = 0 \\ \frac{\partial S}{\partial b} = 0 \end{cases} \quad \text{или} \quad \begin{cases} 2 \sum_{i=1}^n (ax_i + b - y_i)x_i = 0 \\ 2 \sum_{i=1}^n (ax_i + b - y_i) = 0 \end{cases}$$

Упростим полученную систему:

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

Введем обозначения:

$$SX = \sum_{i=1}^n x_i, \quad SXX = \sum_{i=1}^n x_i^2, \quad SY = \sum_{i=1}^n y_i, \quad SXY = \sum_{i=1}^n x_i y_i$$

Получим систему уравнений для нахождения параметров a и b :

$$\begin{cases} aSXX + bSX = SXY \\ aSX + bn = SY \end{cases},$$

из которой находим (правило Крамера):

$$\Delta = SXX \cdot n - SX \cdot SX$$

$$\Delta_1 = SXY \cdot n - SX \cdot SY$$

$$\Delta_2 = SXX \cdot SY - SX \cdot SXY$$

$$a = \frac{\Delta_1}{\Delta}, \quad b = \frac{\Delta_2}{\Delta}$$

КВАДРАТИЧНАЯ АППРОКСИМАЦИЯ

Рассмотрим в качестве эмпирической формулы квадратичную функцию:

$$\varphi(x, a_0, a_1, a_2) = a_0 + a_1x + a_2x^2$$

Сумма квадратов отклонений запишется следующим образом:

$$S = S(a_0, a_1, a_2) = \sum_{i=1}^n (a_0 + a_1x_i + a_2x_i^2 - y_i)^2 \rightarrow \min$$

Приравниваем к нулю частные производные S по неизвестным параметрам, получаем систему линейных уравнений:

$$\left\{ \begin{array}{l} \frac{\partial S}{\partial a_0} = 2 \sum_{i=1}^n (a_0 + a_1x_i + a_2x_i^2 - y_i) = 0 \\ \frac{\partial S}{\partial a_1} = 2 \sum_{i=1}^n (a_0 + a_1x_i + a_2x_i^2 - y_i)x_i = 0 \\ \frac{\partial S}{\partial a_2} = 2 \sum_{i=1}^n (a_0 + a_1x_i + a_2x_i^2 - y_i)x_i^2 = 0 \end{array} \right. \quad \left\{ \begin{array}{l} a_0n + a_1 \sum_{i=1}^n x_i + a_2 \sum_{i=1}^n x_i^2 = \sum_{i=1}^n y_i \\ a_0 \sum_{i=1}^n x_i + a_1 \sum_{i=1}^n x_i^2 + a_2 \sum_{i=1}^n x_i^3 = \sum_{i=1}^n x_i y_i \\ a_0 \sum_{i=1}^n x_i^2 + a_1 \sum_{i=1}^n x_i^3 + a_2 \sum_{i=1}^n x_i^4 = \sum_{i=1}^n x_i^2 y_i \end{array} \right.$$

Аппроксимация с помощью других функций

Помимо линейных зависимостей для описания результатов эксперимента используют также показательные, степенные, логарифмические функции. Эти функции легко могут быть приведены к линейному виду, после чего для определения коэффициентов аппроксимирующей функции можно использовать описанный выше алгоритм.

Аппроксимирующая функция задана степенной функцией вида:

$$\varphi(x) = ax^b$$

Для применения метода наименьших квадратов степенная функция **линеаризуется**:

$$\ln(\varphi(x)) = \ln(ax^b) = \ln(a) + b\ln(x)$$

Введем обозначения: $Y = \ln(\varphi(x))$; $A = \ln(a)$; $B = b$; $X = \ln(x)$

Получаем линейную зависимость: $Y = A + BX$.

После определения коэффициентов A и B вернемся к принятым ранее обозначениям:

$$a = e^A \quad b = B$$

Аппроксимация с помощью других функций

Аппроксимирующая функция задана экспоненциальной функцией вида:

$$\varphi(x) = ae^{bx}$$

Для применения метода наименьших квадратов экспоненциальная функция линеаризуется:

$$\ln(\varphi(x)) = \ln(ae^{bx}) = \ln a + bx$$

Введем обозначения: $Y = \ln(\varphi(x))$; $A = \ln(a)$; $B = b$

Получаем линейную зависимость: $Y = A + Bx$.

После определения коэффициентов A и B вернемся к принятым ранее обозначениям:

$$a = e^A \quad b = B$$

Аппроксимирующая функция задана логарифмической функцией вида:

$$\varphi(x) = a \ln(x) + b$$

Вычислительная часть

Функция: $y = \frac{6x}{x^4+5}$, интервал $x \in [0,2], h = 0,2$

№	X	Y
1	0,0	0,0000
2	0,2	0,2399
3	0,4	0,4776
4	0,6	0,7018
5	0,8	0,8873
6	1,0	1,0000
7	1,2	1,0179
8	1,4	0,9501
9	1,6	0,8309
10	1,8	0,6969
11	2,0	0,5714

Линейная аппроксимация:

$$\sum_{i=1}^n x_i = 11; \quad \sum_{i=1}^n x_i^2 = 15,4; \quad \sum_{i=1}^n y_i = 7,3738; \quad \sum_{i=1}^n x_i y_i = 8,7916$$

Решим систему:

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

$$\begin{cases} 15,4a + 11b = 8,7916 \\ 11a + 11b = 7,3738 \end{cases} \Rightarrow \begin{cases} a = 0,3222 \\ b = 0,3481 \end{cases}$$

Полученная линейная аппроксимация: $P(x) = 0,3222x + 0,3481$

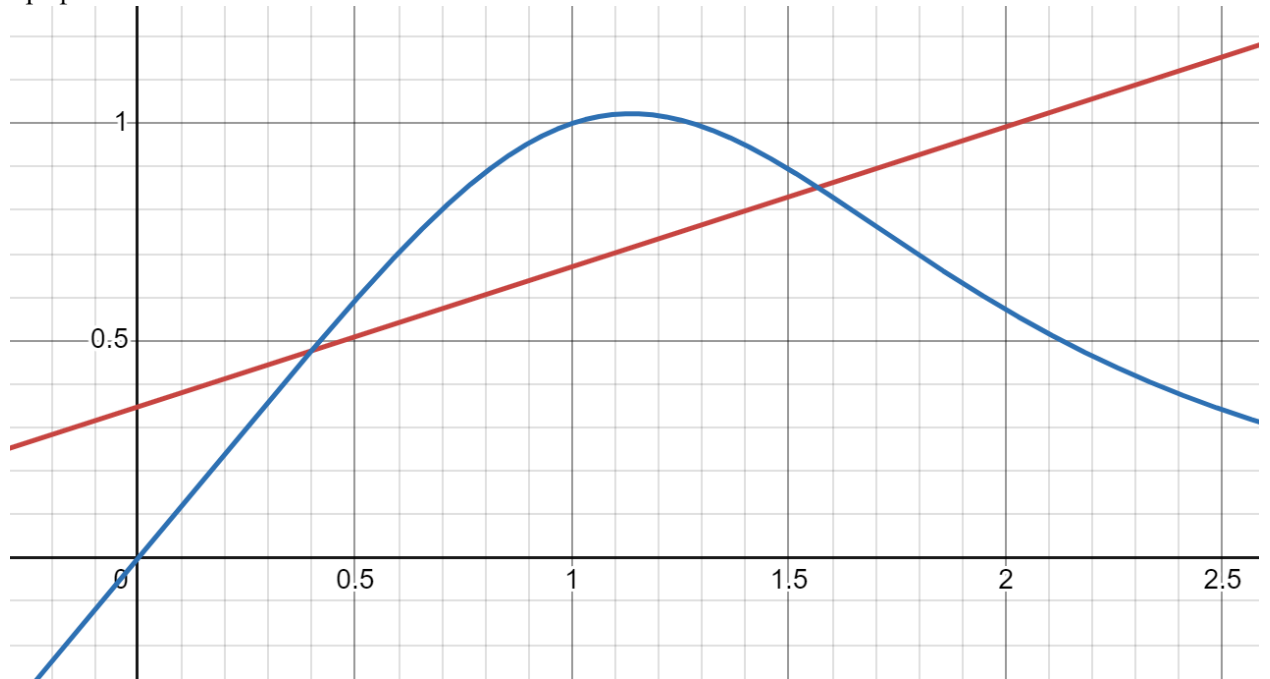
№	X	Y	P(x)	ε_i
1	0,0	0,0000	0,3481	0,3481
2	0,2	0,2399	0,4125	0,1726
3	0,4	0,4776	0,4770	-0,0006
4	0,6	0,7018	0,5414	-0,1604
5	0,8	0,8873	0,6059	-0,2814
6	1,0	1,0000	0,6703	-0,3297
7	1,2	1,0179	0,7347	-0,2832
8	1,4	0,9501	0,7992	-0,1509
9	1,6	0,8309	0,8636	0,0327
10	1,8	0,6969	0,9281	0,2312
11	2,0	0,5714	0,9925	0,4211

Мера отклонения: $S = \sum_{i=1}^n \varepsilon_i^2 = 0,6994$

Среднеквадратическое отклонение: $\delta = \sqrt{\frac{S}{n}} = 0,2522$

Коэффициент корреляции: $r = 0,5890$

График:



Квадратичная аппроксимация:

$$\sum_{i=1}^n x_i = 11; \quad \sum_{i=1}^n x_i^2 = 15,4; \quad \sum_{i=1}^n x_i^3 = 24,2; \quad \sum_{i=1}^n x_i^4 = 40,5328;$$

$$\sum_{i=1}^n y_i = 7,3738; \quad \sum_{i=1}^n x_i y_i = 8,7916; \quad \sum_{i=1}^n x_i^2 y_i = 11,9051$$

Решим систему:

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

$$\begin{cases} 11a_0 + 11a_1 + 15,4a_2 = 7,3738 \\ 11a_0 + 15,4a_1 + 24,2a_2 = 8,7916 \\ 15,4a_0 + 24,2a_1 + 40,5328a_2 = 11,9051 \end{cases} \Rightarrow \begin{cases} a_0 = -0,1999 \\ a_1 = 2,1489 \\ a_2 = -0,9133 \end{cases}$$

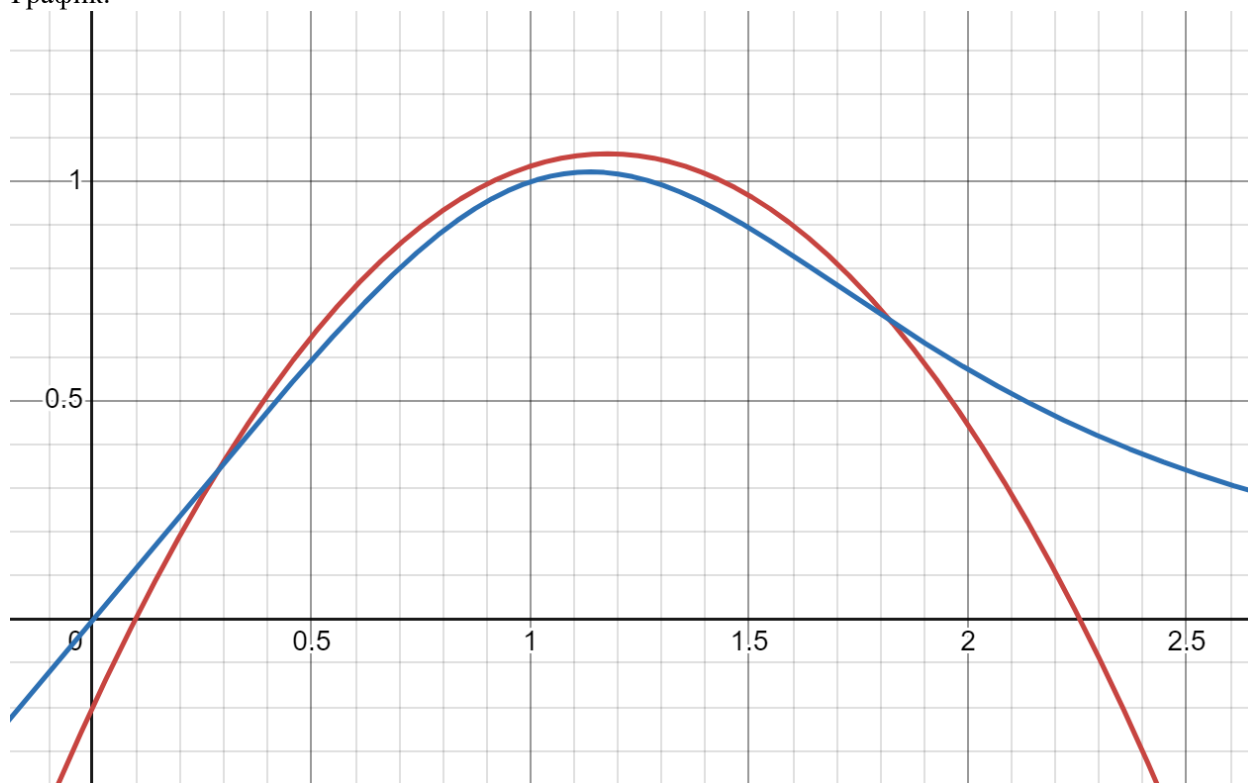
Полученная квадратичная аппроксимация: $P(x) = -0,9133x^2 + 2,1489x - 0,1999$

№	X	Y	P(x)	ε_i
1	0,0	0,0000	-0,1999	-0,1999
2	0,2	0,2399	0,1934	-0,0465
3	0,4	0,4776	0,5135	0,0359
4	0,6	0,7018	0,7607	0,0589
5	0,8	0,8873	0,9347	0,0474
6	1,0	1,0000	1,0357	0,0357
7	1,2	1,0179	1,0636	0,0457
8	1,4	0,9501	1,0185	0,0684
9	1,6	0,8309	0,9003	0,0694
10	1,8	0,6969	0,7090	0,0121
11	2,0	0,5714	0,4447	-0,1267

Мера отклонения: $S = \sum_{i=1}^n \varepsilon_i^2 = 0,0782$

Среднеквадратическое отклонение: $\delta = \sqrt{\frac{S}{n}} = 0,0843$

График:



Наилучшее приближение: квадратичная аппроксимация.

Листинг программы

```
def linear_approximation(x, y):
    SX: float = sum([p for p in x])
    SXX: float = sum([p ** 2 for p in x])
    SY: float = sum([p for p in y])
    SXY: float = sum([x[i] * y[i] for i in range(len(x))])
    delta = SXX * len(x) - SX * SX
    delta1 = SXY * len(x) - SX * SY
    delta2 = SXX * SY - SX * SXY

    a = delta1 / delta
    b = delta2 / delta
    func = lambda x: a * x + b
    return func, a, b
```

```
def polynomial_approximation_2(x, y):
    SX = sum([p for p in x])
    SX2 = sum([p ** 2 for p in x])
    SX3 = sum([p ** 3 for p in x])
    SX4 = sum([p ** 4 for p in x])
    SY = sum([p for p in y])
    SXY = sum([x[i] * y[i] for i in range(len(x))])
    SX2Y = sum([x[i] * x[i] * y[i] for i in range(len(x))])

    x1 = np.array([[len(x), SX, SX2], [SX, SX2, SX3], [SX2, SX3, SX4]])
    y1 = np.array([SY, SXY, SX2Y])
    a = np.linalg.solve(x1, y1)

    func = lambda x: a[2] * x ** 2 + a[1] * x + a[0]
    return func, a[2], a[1], a[0]
```

```
def polynomial_approximation_3(x, y):
    SX = sum([p for p in x])
    SX2 = sum([p ** 2 for p in x])
    SX3 = sum([p ** 3 for p in x])
    SX4 = sum([p ** 4 for p in x])
    SX5 = sum([p ** 5 for p in x])
    SX6 = sum([p ** 6 for p in x])
    SY = sum([p for p in y])
    SXY = sum([x[i] * y[i] for i in range(len(x))])
    SX2Y = sum([x[i] * x[i] * y[i] for i in range(len(x))])
    SX3Y = sum([x[i] * x[i] * x[i] * y[i] for i in range(len(x))])

    x = np.array([[len(x), SX, SX2, SX3], [SX, SX2, SX3, SX4], [SX2, SX3,
    SX4, SX5], [SX3, SX4, SX5, SX6]])
    y = np.array([SY, SXY, SX2Y, SX3Y])
    a = np.linalg.solve(x, y)

    func = lambda x: a[3] * x ** 3 + a[2] * x ** 2 + a[1] * x + a[0]
    return func, a[3], a[2], a[1], a[0]
```

```
def exponential_approximation(x, y):
    if not all([p > 0 for p in x]):
        print("Экспотенциальная аппроксимация невозможна")
        return None, None, None
    y_ln = [np.log(p) for p in y]
    _, b1, a1 = linear_approximation(x, y_ln)
    a = np.exp(a1)
    b = b1
    func = lambda x: a * np.exp(b * x)
    return func, a, b
```

```
def logarithmic_approximation(x, y):
    if not all([p > 0 for p in x]):
        print("Логарифмическая аппроксимация невозможна")
        return None, None, None
    x_ln = [np.log(p) for p in x]
    _, a1, b1 = linear_approximation(x_ln, y)
    func = lambda x: a1 * np.log(x) + b1
    return func, a1, b1
```

```
def power_approximation(x, y):
    if not (all([p > 0 for p in x]) and all([p > 0 for p in y])):
        print("Степенная аппроксимация невозможна")
        return None, None, None
    x_ln = [np.log(p) for p in x]
    y_ln = [np.log(p) for p in y]
    _, b1, a1 = linear_approximation(x_ln, y_ln)
    a = np.exp(a1)
    b = b1
    func = lambda x: a * x ** b
    return func, a, b
```

Полный код: https://github.com/aulouu/comp_math_labs/blob/main/lab4/lab4

Результат выполнения программы

Набор данных №1:

1 2
2 5
3 7
4 8
5 10
6 12
7 13

Выберите способ ввода данных (file/console): *file*
Введите имя файла: *inp.txt*

```
+-----+
|          Линейная аппроксимация          |
+---+-----+-----+-----+-----+
| № | X | Y | P | eps |
+---+-----+-----+-----+-----+
| 1 | 1.000 | 2.000 | 2.786 | 0.786 |
| 2 | 2.000 | 5.000 | 4.571 | -0.429 |
| 3 | 3.000 | 7.000 | 6.357 | -0.643 |
| 4 | 4.000 | 8.000 | 8.143 | 0.143 |
| 5 | 5.000 | 10.000 | 9.929 | -0.071 |
| 6 | 6.000 | 12.000 | 11.714 | -0.286 |
| 7 | 7.000 | 13.000 | 13.500 | 0.500 |
+---+-----+-----+-----+-----+
```

Коэффициент детерминации: 0.99255

Высокая точность аппроксимации

Коэффициент Пирсона: 0.99131

Связь весьма высокая

```
+-----+
|          Квадратичная аппроксимация          |
+---+-----+-----+-----+-----+
| № | X | Y | P | eps |
+---+-----+-----+-----+-----+
| 1 | 1.000 | 2.000 | 2.310 | 0.310 |
| 2 | 2.000 | 5.000 | 4.571 | -0.429 |
| 3 | 3.000 | 7.000 | 6.643 | -0.357 |
| 4 | 4.000 | 8.000 | 8.524 | 0.524 |
| 5 | 5.000 | 10.000 | 10.214 | 0.214 |
| 6 | 6.000 | 12.000 | 11.714 | -0.286 |
| 7 | 7.000 | 13.000 | 13.024 | 0.024 |
+---+-----+-----+-----+-----+
```

Коэффициент детерминации: 0.99616

Высокая точность аппроксимации

Кубическая аппроксимация				
№	X	Y	P	eps
1	1.000	2.000	2.143	0.143
2	2.000	5.000	4.738	-0.262
3	3.000	7.000	6.810	-0.190
4	4.000	8.000	8.524	0.524
5	5.000	10.000	10.048	0.048
6	6.000	12.000	11.548	-0.452
7	7.000	13.000	13.190	0.190
Коэффициент детерминации: 0.99695				
Высокая точность аппроксимации				

Экспоненциальная аппроксимация				
№	X	Y	P	eps
1	1.000	2.000	3.086	1.086
2	2.000	5.000	4.066	-0.934
3	3.000	7.000	5.358	-1.642
4	4.000	8.000	7.059	-0.941
5	5.000	10.000	9.301	-0.699
6	6.000	12.000	12.256	0.256
7	7.000	13.000	16.148	3.148
Коэффициент детерминации: 0.92371				
Удовлетворительная аппроксимация				

Логарифмическая аппроксимация				
№	X	Y	P	eps
1	1.000	2.000	1.339	-0.661
2	2.000	5.000	5.211	0.211
3	3.000	7.000	7.477	0.477
4	4.000	8.000	9.084	1.084
5	5.000	10.000	10.330	0.330
6	6.000	12.000	11.349	-0.651
7	7.000	13.000	12.210	-0.790
Коэффициент детерминации: 0.98559				
Высокая точность аппроксимации				

Степенная аппроксимация				
№	X	Y	P	eps
1	1.000	2.000	2.267	0.267
2	2.000	5.000	4.327	-0.673
3	3.000	7.000	6.316	-0.684
4	4.000	8.000	8.260	0.260
5	5.000	10.000	10.171	0.171
6	6.000	12.000	12.056	0.056
7	7.000	13.000	13.920	0.920

Коэффициент детерминации: 0.99081
Высокая точность аппроксимации

Выбор аппроксимирующей функции							
Вид функции	a	b	c	d	S	del	R^2
$ax + b$	1.78571	1.00000	-	-	1.57143	0.47380	0.99255
$ax^2 + bx + c$	-0.09524	2.54762	-0.14286	-	0.80952	0.34007	0.99616
$ax^3 + bx^2 + cx + d$	0.02778	-0.42857	3.68651	-1.14286	0.64286	0.30305	0.99695
$ae^{(bx)}$	2.34211	0.27582	-	-	16.09824	1.51649	0.92371
$a \ln x + b$	5.58649	1.33918	-	-	3.04015	0.65902	0.98559
ax^b	2.26698	0.93268	-	-	1.93878	0.52628	0.99081

Наилучшая аппроксимирующая функция: Кубическая

Набор данных №2:

-1 -1

2 5

3 9

4 17

5 25

Выберите способ ввода данных (file/console): *file*

Введите имя файла: *inp2.txt*

Экспотенциальная аппроксимация невозможна

Логарифмическая аппроксимация невозможна

Степенная аппроксимация невозможна

```
+-----+
|          Линейная аппроксимация          |
+---+-----+-----+-----+-----+
| № | X   | Y   | P   | eps |
+---+-----+-----+-----+-----+
| 1 | -1.000 | -1.000 | -3.943 | -2.943 |
| 2 | 2.000 | 5.000 | 8.509 | 3.509 |
| 3 | 3.000 | 9.000 | 12.660 | 3.660 |
| 4 | 4.000 | 17.000 | 16.811 | -0.189 |
| 5 | 5.000 | 25.000 | 20.962 | -4.038 |
+---+-----+-----+-----+-----+
Коэффициент детерминации: 0.93403
Удовлетворительная аппроксимация
Коэффициент Пирсона: 0.93706
Связь весьма высокая
```

```
+-----+
|          Квадратичная аппроксимация          |
+---+-----+-----+-----+-----+
| № | X   | Y   | P   | eps |
+---+-----+-----+-----+-----+
| 1 | -1.000 | -1.000 | -0.964 | 0.036 |
| 2 | 2.000 | 5.000 | 4.574 | -0.426 |
| 3 | 3.000 | 9.000 | 9.744 | 0.744 |
| 4 | 4.000 | 17.000 | 16.576 | -0.424 |
| 5 | 5.000 | 25.000 | 25.070 | 0.070 |
+---+-----+-----+-----+-----+
Коэффициент детерминации: 0.99880
Высокая точность аппроксимации
```

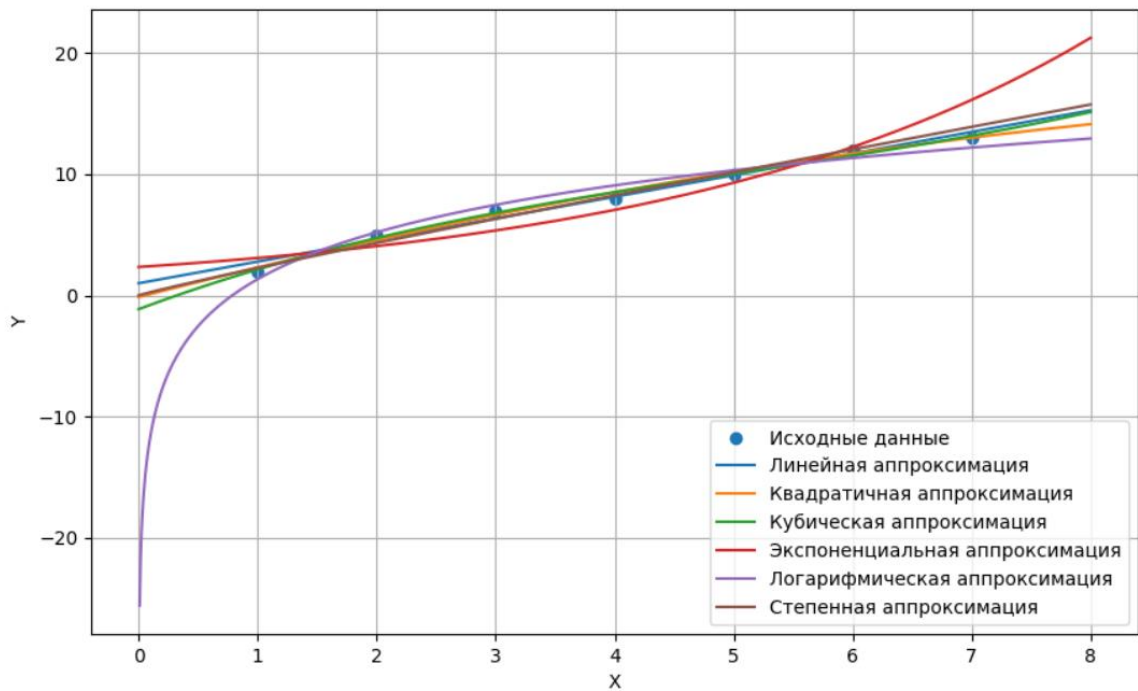
```
+-----+
|          Кубическая аппроксимация          |
+---+-----+-----+-----+-----+
| № | X   | Y   | P   | eps |
+---+-----+-----+-----+-----+
| 1 | -1.000 | -1.000 | -0.985 | 0.015 |
| 2 | 2.000 | 5.000 | 4.696 | -0.304 |
| 3 | 3.000 | 9.000 | 9.683 | 0.683 |
| 4 | 4.000 | 17.000 | 16.454 | -0.546 |
| 5 | 5.000 | 25.000 | 25.152 | 0.152 |
+---+-----+-----+-----+-----+
Коэффициент детерминации: 0.99886
Высокая точность аппроксимации
```


Выбор аппроксимирующей функции								
Вид функции	a	b	c	d	S	del	R^2	
$ax + b$	4.15094	0.20755	-	-	50.71698	3.18487	0.93403	
$ax^2 + bx + c$	0.83098	1.01518	-0.78022	-	0.92098	0.42918	0.99880	
$ax^3 + bx^2 + cx + d$	0.02381	0.67792	1.14443	-0.49451	0.88017	0.41956	0.99886	

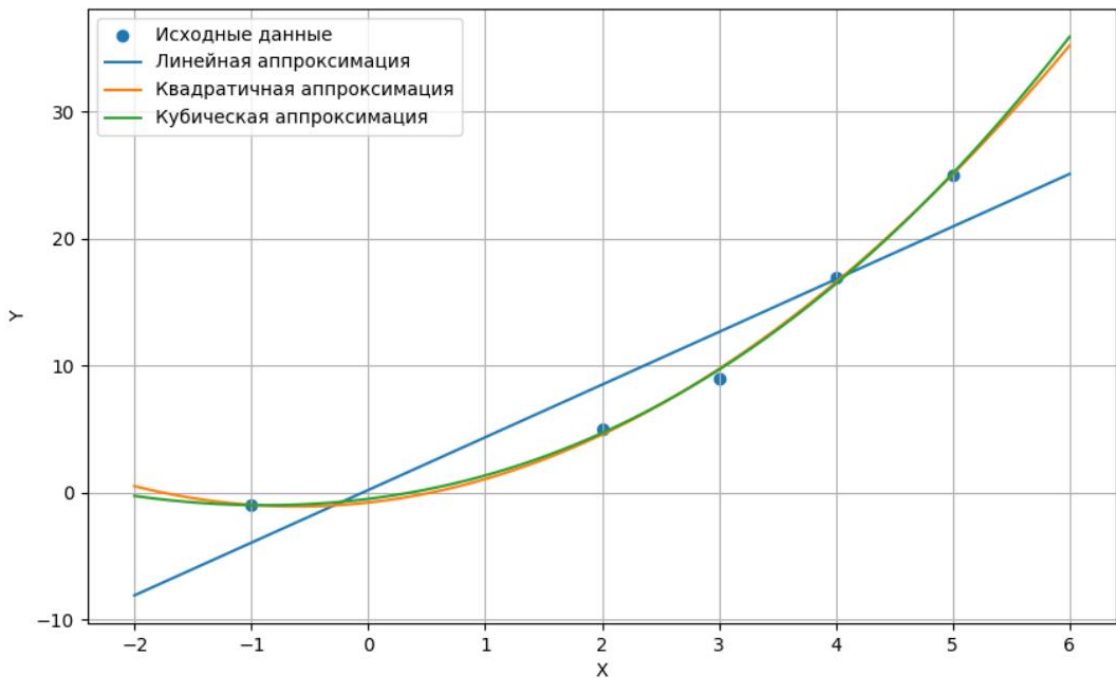
Наилучшая аппроксимирующая функция: Кубическая

Графики аппроксимирующих функций

Для набора данных №1:



Для набора данных №2:



Вывод

В ходе данной лабораторной работы были изучены различные виды аппроксимации функции через метод наименьших квадратов. Данный метод довольно просто программируем.