

Лабораторная работа №5

Ишанова А.И. группа НФИ-02-19

Содержание

1	Цель работы	5
2	Задание работы	6
3	Выполнение лабораторной работы	7
3.1	Подгонка полиномиальной кривой	7
3.2	Матричные преобразования	18
3.2.1	Вращение	20
3.2.2	Отражение	23
3.2.3	Дилатация	25
4	Вывод	29

List of Figures

3.1	подготовка к лабораторной работе	7
3.2	начало журналирования	7
3.3	набор точек	8
3.4	выделение координат по x и y	9
3.5	команда построения графика	10
3.6	полученный график	10
3.7	инициализация матрицы A	11
3.8	замена первого столбца матрицы A	11
3.9	замена второго столбца матрицы A	12
3.10	вычисление матриц $A^T A$ и $A^T y$	13
3.11	расширенная матрица B	14
3.12	решение методом Гаусса	15
3.13	команда построения графика	15
3.14	полученный график	16
3.15	решение методом Гаусса	16
3.16	значения полинома в точках	17
3.17	команда построения графика	17
3.18	полученный график	18
3.19	работа с данными	19
3.20	полученное изображение	20
3.21	угол θ_1 и матрица вращения R_1	20
3.22	перемножение матриц R_1 и D	21
3.23	извлечение координат	21
3.24	угол θ_2 и матрица вращения R_2	21
3.25	перемножение матриц R_2 и D	22
3.26	извлечение координат	22
3.27	команда построения графика	22
3.28	полученный график	23
3.29	матрица отражения	23
3.30	перемножение матриц R и D	24
3.31	извлечение координат	24
3.32	команда построения графика	25
3.33	полученный график	25
3.34	матрица дилатации	26
3.35	матрица дилатации	26
3.36	матрица дилатации	27
3.37	команда построения графика	27

3.38	полученный график	28
3.39	завершение сессии журналирования	28

1 Цель работы

Научиться решать проблему подгонки полиномиальной кривой и научиться реализовывать некоторые матричные преобразования.

2 Задание работы

Выполнить лабораторную работу и сделать отчет по лабораторной работе в форматах md, docx и pdf.

3 Выполнение лабораторной работы

1. Создаем каталог для работы в папке laboratory. (mkdir) (fig. 3.1)

```
iMac-Alina:~ alinaishanova$ cd work/2020-2021/"Введение в научное программирование"/laboratory
iMac-Alina:laboratory alinaishanova$ mkdir lab05
iMac-Alina:laboratory alinaishanova$ █
```

Figure 3.1: подготовка к лабораторной работе

2. Начинаем сессию журналирования. (fig. 3.2)



```
|>> diary on
```

Figure 3.2: начало журналирования

3.1 Подгонка полиномиальной кривой

1. Задаем набор точек через матрицу D. (fig. 3.3)

```
>> D = [1 1 ; 2 2 ; 3 5 ; 4 4 ; 5 2 ; 6 -3]
D =
```

1	1
2	2
3	5
4	4
5	2
6	-3

Figure 3.3: набор точек

2. Извлекаем вектора x и y . (fig. 3.4)


```
>> xdata = D(:, 1)
xdata =
```

```
1
2
3
4
5
6
```

```
>> ydata = D(:, 2)
ydata =
```

```
1
2
5
4
2
-3
```

Figure 3.4: выделение координат по x и y

3. Рисуем точки на графике. (plot) (fig. 3.5 и fig. 3.6)

```
>> plot(xdata, ydata, 'o-')
```

Figure 3.5: команда построения графика

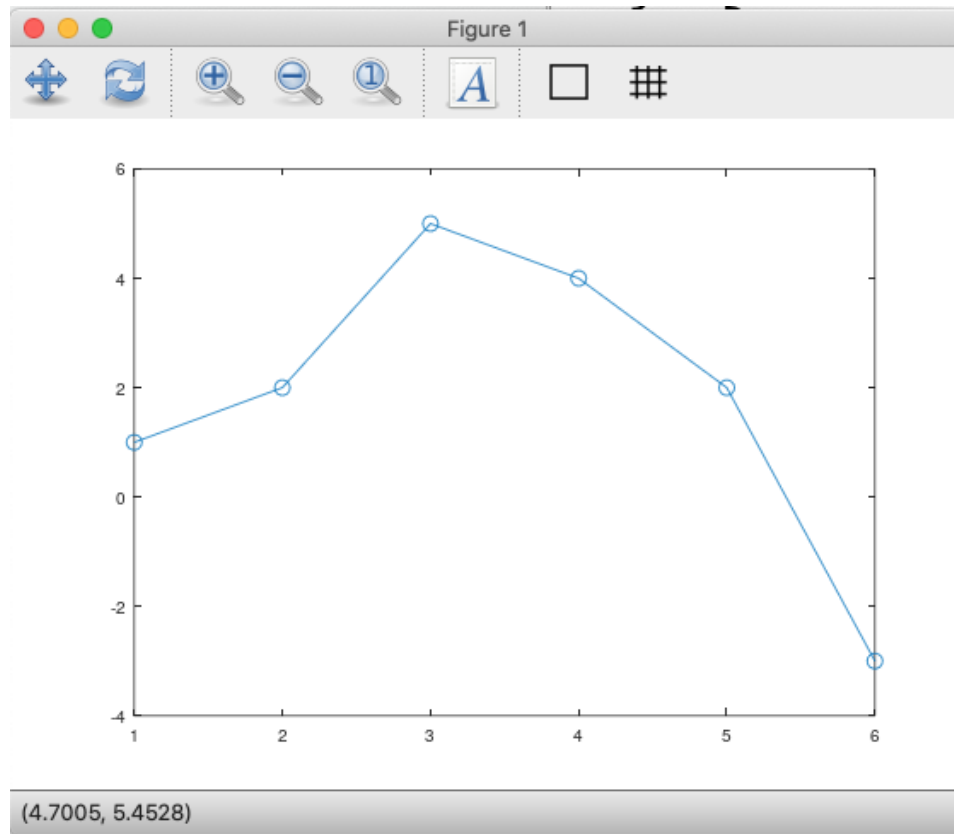


Figure 3.6: полученный график

4. Задаем матрицу коэффициентов A: сначала делаем ее единичной матрицей, потом меняем первый столбец на квадрат координат точек по x, а второй — просто на координаты. (fig. 3.7, fig. 3.8 и fig. 3.9)

```
>> A = ones(6,3)
A =
```

1	1	1
1	1	1
1	1	1
1	1	1
1	1	1
1	1	1

Figure 3.7: инициализация матрицы A

```
>> A(:,1) = xdata .^ 2
A =
```

1	1	1
4	1	1
9	1	1
16	1	1
25	1	1
36	1	1

Figure 3.8: замена первого столбца матрицы A

```
>> A(:,2) = xdata  
A =
```

1	1	1
4	2	1
9	3	1
16	4	1
25	5	1
36	6	1

Figure 3.9: замена второго столбца матрицы A

5. Находим матрицы $A^T A$ и $A^T y$, необходимые нам для решения. (fig. 3.10)

```

>> A' * A
ans =

    2275    441    91
     441     91    21
      91     21     6

>> A' * ydata
ans =

    60
    28
    11

```

Figure 3.10: вычисление матриц $A^T A$ и $A^T y$

6. Из них задаем расширенную матрицу B, для решения методом Гаусса.
(fig. 3.11)

```
>> B = A' * A  
B =
```

```
2275    441    91  
441     91    21  
91     21     6
```

```
>> B(:, 4) = A' * ydata  
B =
```

```
2275    441    91    60  
441     91    21    28  
91     21     6    11
```

Figure 3.11: расширенная матрица B

7. Решаем методом Гаусса (rref) и записываем ответ в отдельные переменные.
(fig. 3.12)

```

>> B_res = rref(B)
B_res =

    1.0000    0    0   -0.8929
         0    1.0000    0    5.6500
         0    0    1.0000   -4.4000

>> a1 = B_res(1,4)
a1 = -0.8929
>> a2 = B_res(2,4)
a2 = 5.6500
>> a3 = B_res(3,4)
a3 = -4.4000

```

Figure 3.12: решение методом Гаусса

8. Задаем значения по x для графика и функцию y как квадратное уравнение с полученными коэффициентами. Строим график исходных данных и нашей подогранной параболы. (fig. 3.13 и fig. 3.14)

```

>> x = linspace(0,7, 50);
>> y = a1 * x.^2 + a2 * x + a3;
>> plot(xdata, ydata, 'o', x,y, 'linewidth',2)
>> grid on
>> legend('data values', 'least-squares parabola')
>> title('y = -0.89286 x^2 + 5.65 x - 4.4')

```

Figure 3.13: команда построения графика

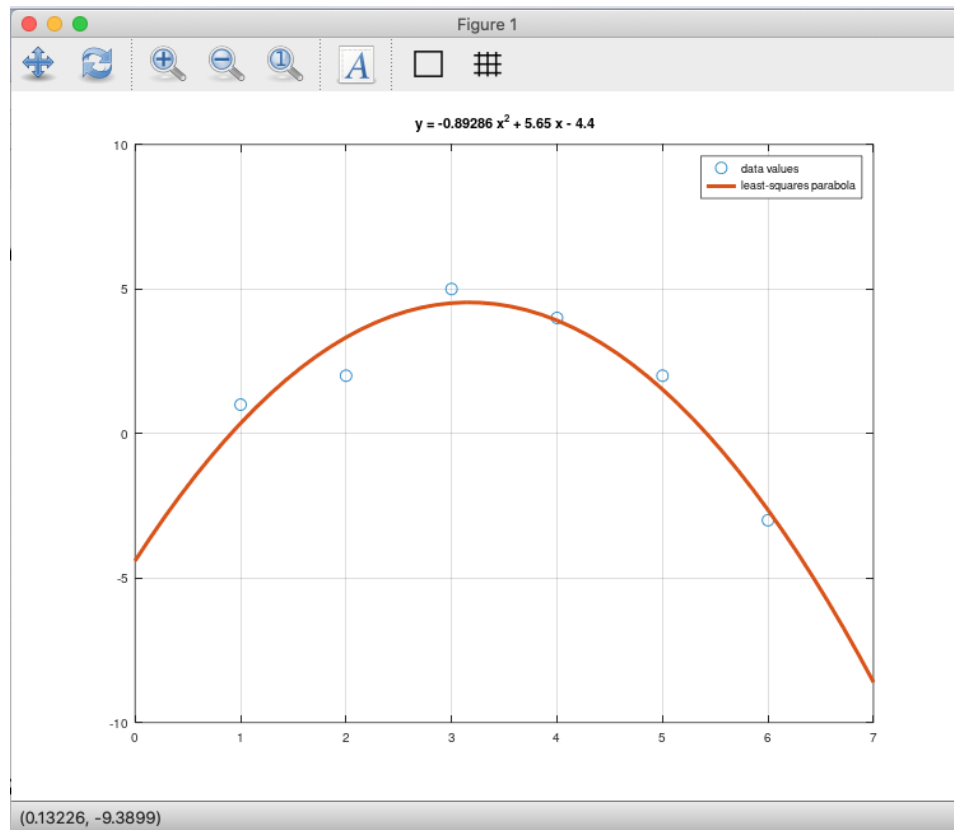


Figure 3.14: полученный график

9. Получим коэффициенты с помощью встроенной программы polyfit.
(fig. 3.15)

```
>> P = polyfit (xdata, ydata, 2)
P =
-0.8929    5.6500   -4.4000
```

Figure 3.15: решение методом Гаусса

10. Найдём значения полинома в точках. (polyval) (fig. 3.16)


```
>> y = polyval (P, xdata)
y =
    0.3571
    3.3286
    4.5143
    3.9143
    1.5286
   -2.6429
```

Figure 3.16: значения полинома в точках

1. Построим график по полученным точкам. (fig. 3.17 и fig. 3.18)

```
>> plot(xdata, ydata, 'o-', xdata, y, '+-')
>> grid on
>> legend ('original data', 'polyfit data')
```

Figure 3.17: команда построения графика

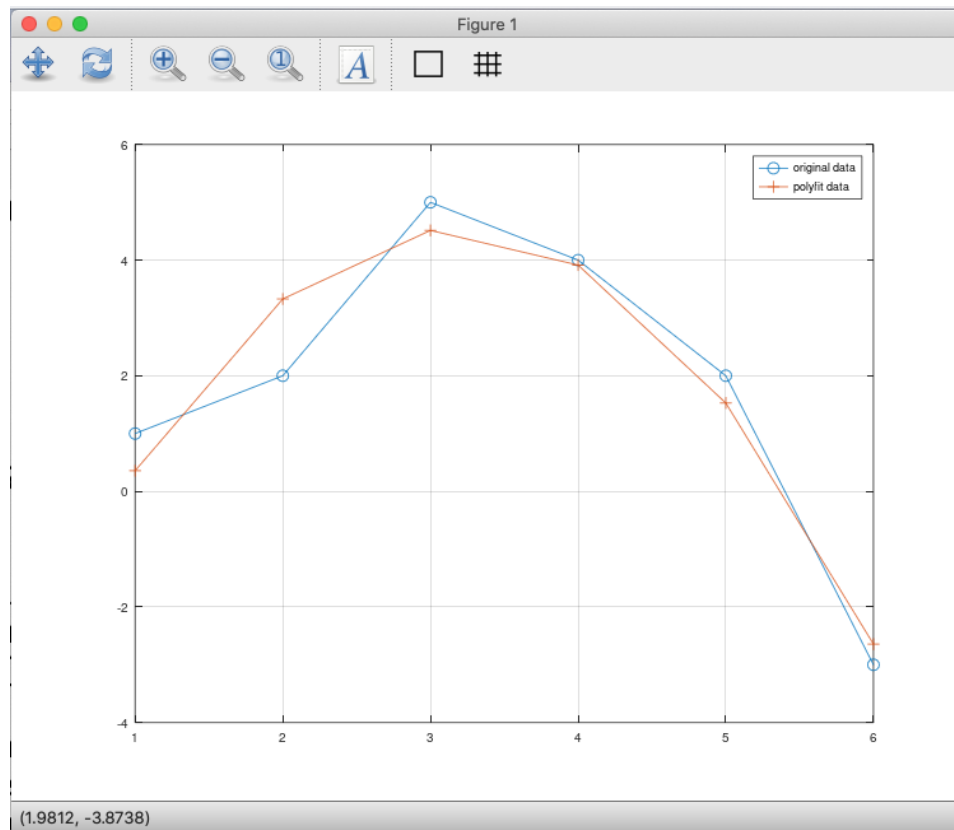


Figure 3.18: полученный график

3.2 Матричные преобразования

1. Строим изображение по циклу Эйлера: задаем матрицу точек, выделяем координаты по x и y , строим график. (fig. 3.19 и fig. 3.20)

```

>> D = [ 1 1 3 3 2 1 3 ; 2 0 0 2 3 2 2]
D =

    1    1    3    3    2    1    3
    2    0    0    2    3    2    2

>> x = D(1, :)
x =

    1    1    3    3    2    1    3

>> y = D(2, :)
y =

    2    0    0    2    3    2    2

>> plot (x,y)

```

Figure 3.19: работа с данными

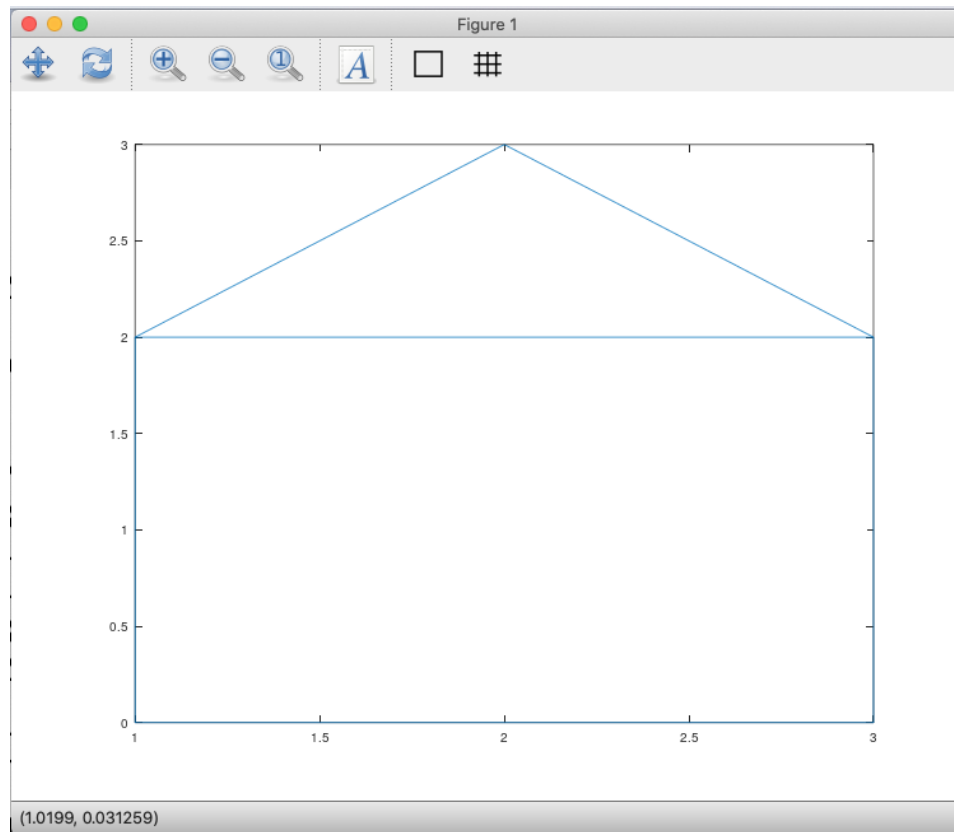


Figure 3.20: полученное изображение

3.2.1 Вращение

1. Задаем угол поворота θ_1 и соответствующую ему матрицу поворота R_1 . (fig. 3.21)

```
>> theta1 = 90*pi/180
theta1 = 1.5708
>> R1 = [cos(theta1) -sin(theta1); sin(theta1) cos(theta1)]
R1 =

    6.1232e-17   -1.0000e+00
    1.0000e+00    6.1232e-17
```

Figure 3.21: угол θ_1 и матрица вращения R_1

1. Для того чтобы воспроизвести поворот матрицы умножаем R_1 на D . (fig. 3.22)

```
>> RD1 = R1*D
RD1 =

Columns 1 through 6:

-2.0000e+00    6.1232e-17    1.8370e-16   -2.0000e+00   -3.0000e+00   -2.0000e+00
 1.0000e+00    1.0000e+00    3.0000e+00    3.0000e+00    2.0000e+00    1.0000e+00

Column 7:

-2.0000e+00
 3.0000e+00
```

Figure 3.22: перемножение матриц R_1 и D

3. Выделяем координаты точек. (fig. 3.23)

```
>> x1 = RD1(1, :)
x1 =

Columns 1 through 5:

-2.0000e+00    6.1232e-17    1.8370e-16   -2.0000e+00   -3.0000e+00

Columns 6 and 7:

-2.0000e+00   -2.0000e+00

>> y1 = RD1(2, :)
y1 =

 1.0000    1.0000    3.0000    3.0000    2.0000    1.0000    3.0000
```

Figure 3.23: извлечение координат

4. Задаем угол поворота θ_2 и проделываем то же самое для него. (fig. 3.24, fig. 3.25 и fig. 3.26)

```
>> theta2 = 225*pi/180
theta2 = 3.9270
>> R2 = [cos(theta2) -sin(theta2); sin(theta2) cos(theta2)]
R2 =

-0.7071    0.7071
-0.7071   -0.7071
```

Figure 3.24: угол θ_2 и матрица вращения R_2

```
>> RD2 = R2*D
RD2 =
    0.7071   -0.7071   -2.1213   -0.7071    0.7071    0.7071   -0.7071
   -2.1213   -0.7071   -2.1213   -3.5355   -3.5355   -2.1213   -3.5355
```

Figure 3.25: перемножение матриц R_2 и D

```
>> x2 = RD2(1, :)
x2 =
    0.7071   -0.7071   -2.1213   -0.7071    0.7071    0.7071   -0.7071

>> y2 = RD2(2, :)
y2 =
   -2.1213   -0.7071   -2.1213   -3.5355   -3.5355   -2.1213   -3.5355
```

Figure 3.26: извлечение координат

5. Строим график с исходным изображением и двумя перевернутыми. (fig. 3.27 и fig. 3.28)

```
>> plot(x,y, 'bo-', x1, y1, 'ro-', x2, y2, 'go-')
>> axis([-4 4 -4 4], 'equal')
>> grid on
>> legend ('original', 'rotated 90 deg', 'rotated 225')
```

Figure 3.27: команда построения графика

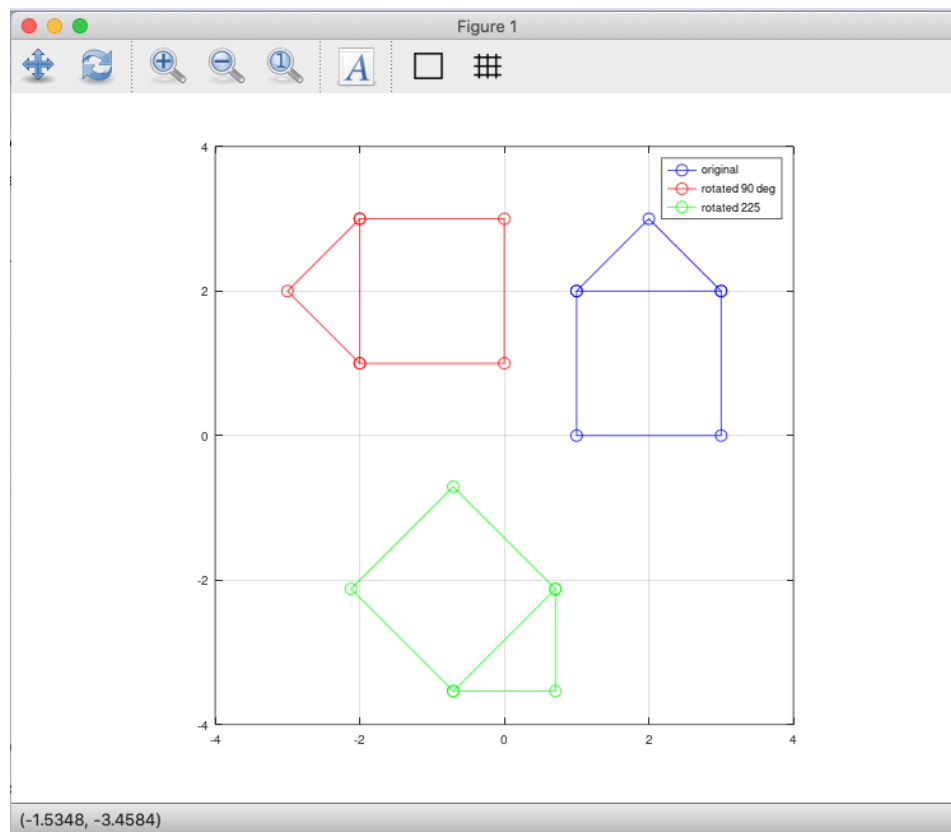


Figure 3.28: полученный график

3.2.2 Отражение

1. Задаем матрицу отражения относительно прямой l ($y = x$). (fig. 3.29)

```
>> R = [ 0 1; 1 0]
R =

     0     1
     1     0
```

Figure 3.29: матрица отражения

2. Отражаем, перемножая матрицу R на D . (fig. 3.30)

```
>> RD = R * D
RD =

     2     0     0     2     3     2     2
     1     1     3     3     2     1     3
```

Figure 3.30: перемножение матриц R и D

3. Выделяем координаты по x и по y . (fig. 3.31)

```
>> x1 = RD(1, :)
x1 =

     2     0     0     2     3     2     2

>> y1 = RD(2, :)
y1 =

     1     1     3     3     2     1     3
```

Figure 3.31: извлечение координат

4. Строим изображения. (fig. 3.32 и fig. 3.33)


```

>> plot (x,y, 'o-', x1, y1, 'o-')
>> axis([-1 4 -1 4], 'equal')
>> axis([-1 5 -1 5], 'equal')
>> grid on
>> legend ('original', 'reflected')

```

Figure 3.32: команда построения графика

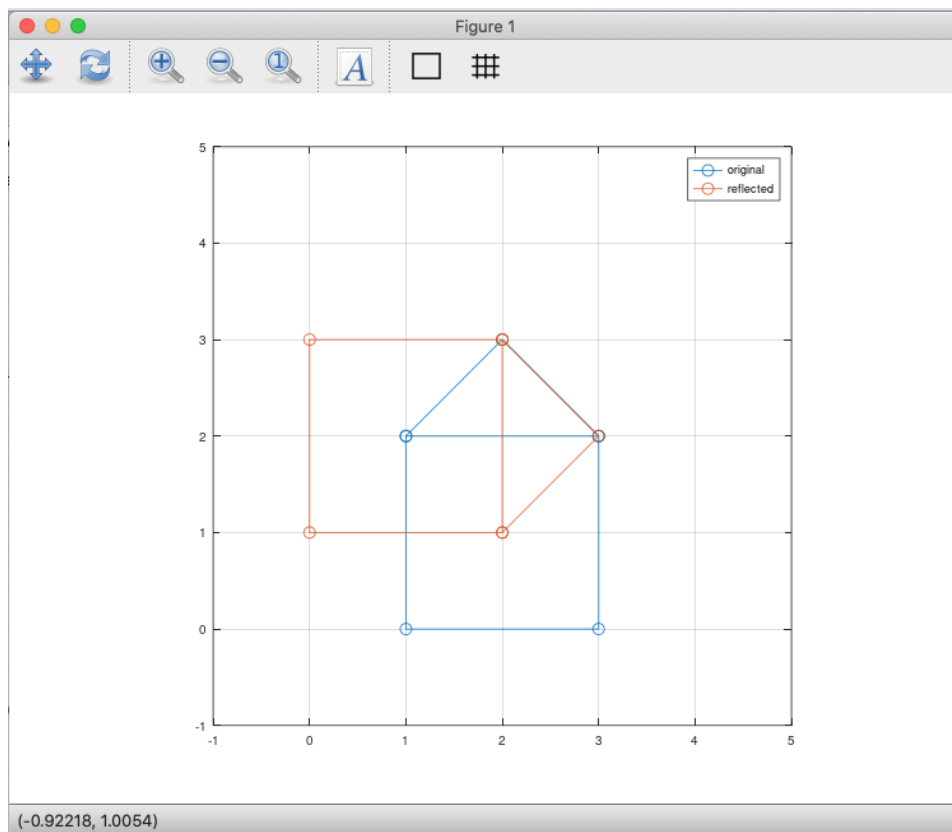


Figure 3.33: полученный график

3.2.3 Дилатация

1. Задаем матрицу дилатации T . (fig. 3.34)

```
>> T = [2 0; 0 2]
T =
```

2	0
0	2

Figure 3.34: матрица дилатации

2. Выполняем матричное преобразование. (fig. 3.35)

```
>> TD = T*D
TD =
```

2	2	6	6	4	2	6
4	0	0	4	6	4	4

Figure 3.35: матрица дилатации

3. Выделяем координаты точек. (fig. 3.36)

```

>> x1 = TD(1, :)
x1 =
    2    2    6    6    4    2    6

>> y1 = TD(2, :)
y1 =
    4    0    0    4    6    4    4

```

Figure 3.36: матрица дилатации

4. Строим изображения. (fig. 3.37 и fig. 3.38)

```

>> y1 = TD(2, :)
y1 =
    4    0    0    4    6    4    4

>> plot (x,y, 'o-', x1, y1, 'o-')
>> axis([-1 7 -1 7], 'equal')
>> grid on
>> legend ('original', 'expanded')

```

Figure 3.37: команда построения графика

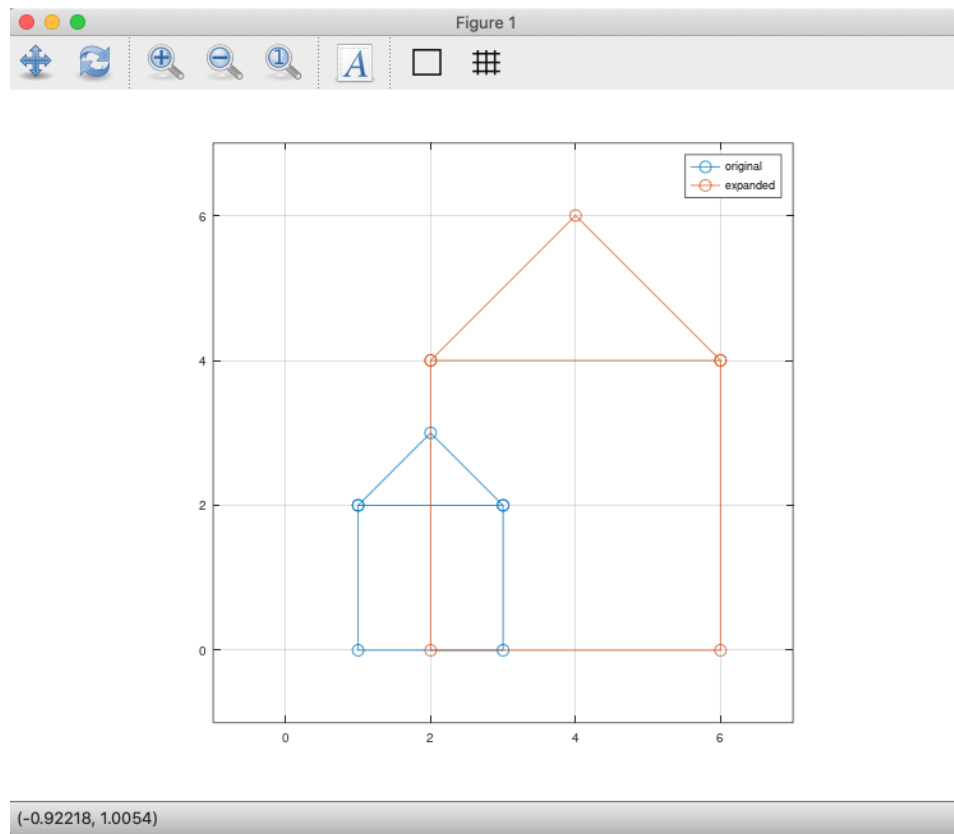


Figure 3.38: полученный график

5. Завершаем сессию журналирования. (fig. 3.39)

```
>> diary off
```

Figure 3.39: завершение сессии журналирования

4 Вывод

В ходе выполнения работы мы научились подгонке данных к общим уравнениям (в ручную и через встроенную команду `polyfit`), а так же матричным преобразованиям, таким как вращение, отражение и дилатация.