

Лабораторная работа №5

Дисциплина: Научное программирование

Аветисян Давид Артурович

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	17

List of Tables

List of Figures

3.1	Матрица D и вектор x и y	8
3.2	Точки на графике	9
3.3	Матрица коэффициентов A	10
3.4	Решение по методу наименьших квадратов	11
3.5	График параболы	12
3.6	Встроенная функция <code>polyfit</code>	13
3.7	Задание и построение матрицы D	13
3.8	Перевод в радианы и расчёт для угла 90°	14
3.9	Перевод в радианы и расчёт для угла 225°	14
3.10	Построение оригинальной и повернутых матриц	15
3.11	Отражение матрицы D	15
3.12	Дилатация матрицы D	16

1 Цель работы

Познакомиться с подгонкой полиномиальной кривой и с матричными преобразованиями.

2 Задание

1. Познакомиться с подгонкой полиномиальной кривой.
2. Познакомиться с вращением матрицы.
3. Познакомиться с отражением матрицы.
4. Познакомиться с дилатацией матрицы.

3 Выполнение лабораторной работы

- 1) Первым делом я познакомился с подгонкой полиномиальной кривой. Сначала я задал матрицу D с двумя столбцами, и извлёк из неё вектора x и y .

```
octave:1> diary on
octave:2> 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

octave:3> xdata = D(:,1)
xdata =

     1
     2
     3
     4
     5
     6

octave:4> ydata = D(:,2)
ydata =

     1
     2
     5
     4
     2
    -3
```

Figure 3.1: Матрица D и вектор x и y

Далее я нарисовал получившиеся точки на графике.

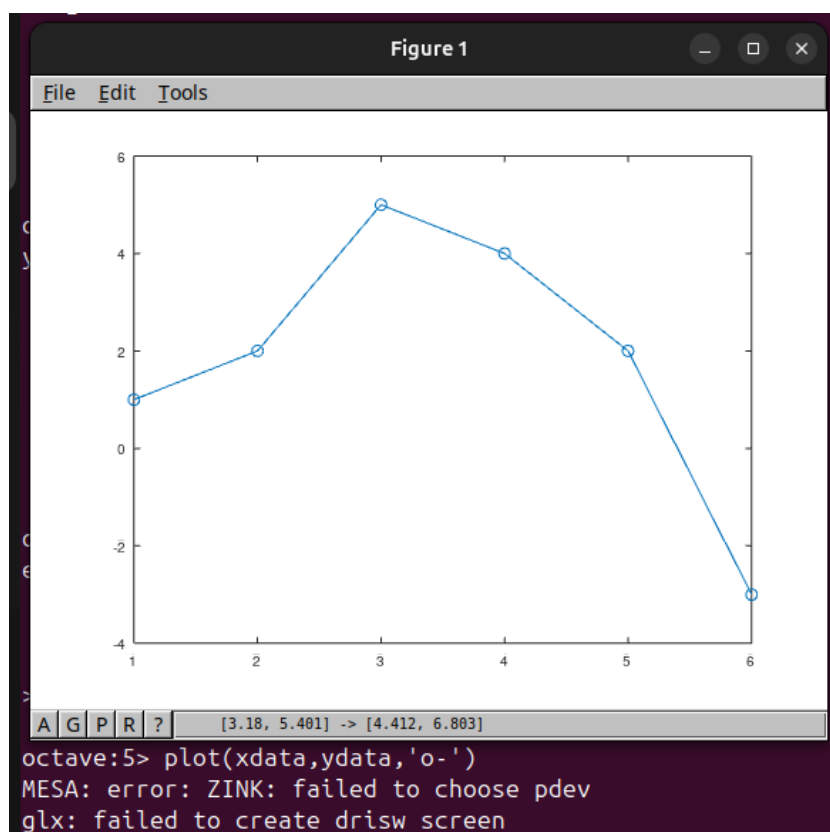


Figure 3.2: Точки на графике

Затем необходимо было построить уравнение вида $y = a * x^2 + b * x + c$. Я задал матрицу коэффициентов A .

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

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

octave:7> A(:,1) = xdata .^ 2
A =

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

octave:8> A(:,2) = xdata
A =

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

Figure 3.3: Матрица коэффициентов A

После я нашёл решение по методу наименьших квадратов из решения уравнения $A' * A * b = A' * y$. Получилось квадратное уравнение вида $y = -0.8929 * x^2 + 5.65 * x - 4.4$.

```
octave:9> A'*A
ans =

    2275    441    91
    441     91    21
     91     21     6

octave:10> A' * ydata
ans =

    60
    28
    11

octave:11> B = A' * A;
octave:12> B(:,4) = A' * ydata;
octave:13> B_res = rref (B)
B_res =

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

octave:14> a1=B_res(1,4)
a1 = -0.8929
octave:15> a2=B_res(2,4)
a2 = 5.6500
octave:16> a3=B_res(3,4)
a3 = -4.4000
```

Figure 3.4: Решение по методу наименьших квадратов

Потом я построил соответствующий график параболы.

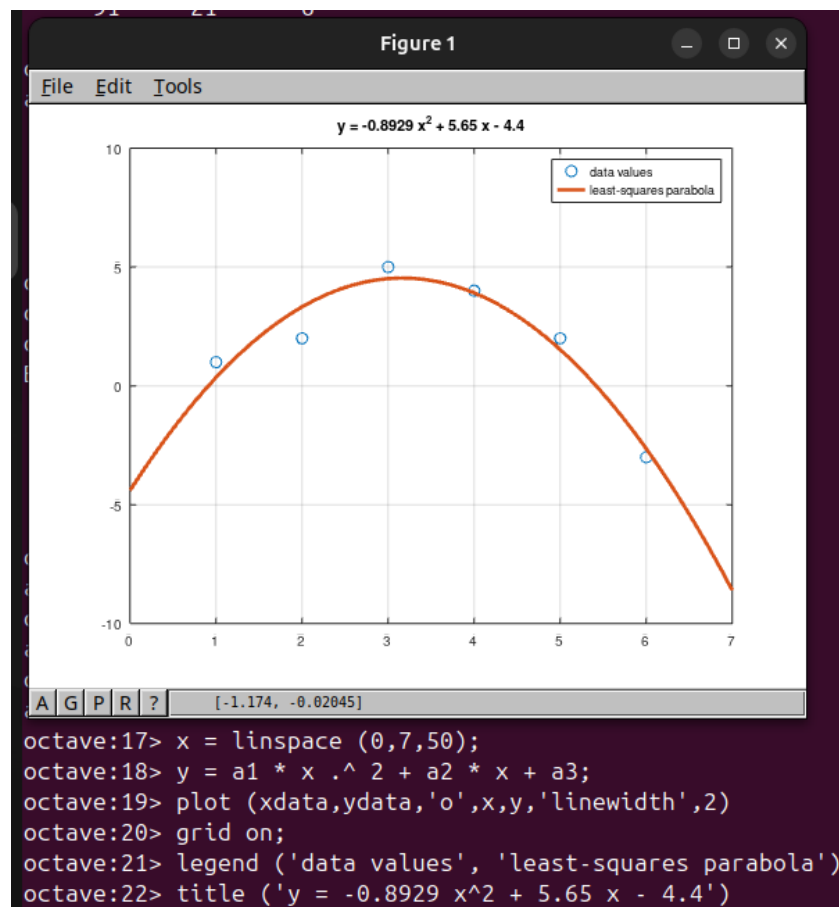


Figure 3.5: График параболы

Далее я познакомился с автоматизированной встроенной в Octave функцией подгонки - $\text{polyfit}(x, y, \text{order})$. Я получил подгоночный полином, рассчитал значения полинома в точках и построил исходные и подгоночные данные.

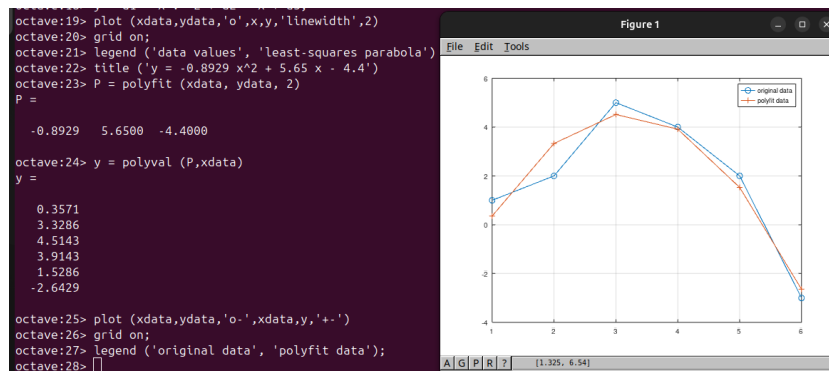


Figure 3.6: Встроенная функция polyfit

- 2) Затем я познакомился с матричными преобразованиями. Для начала я задал матрицу D и построил её граф. Получился домик.

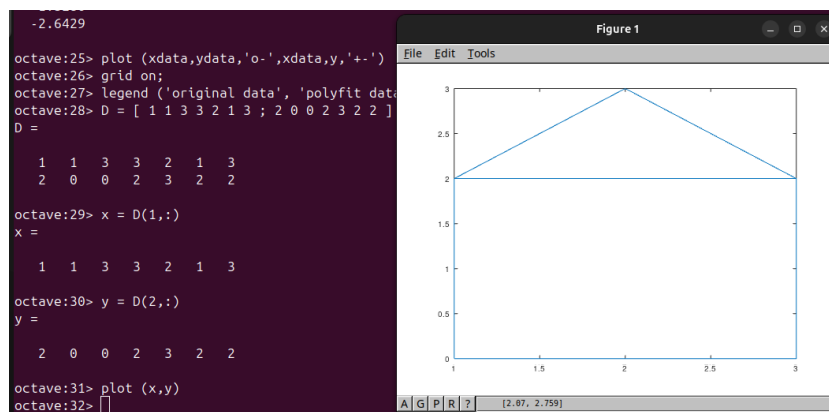


Figure 3.7: Задание и построение матрицы D

После я познакомился с вращением матрицы на 90 и 225 градусов. Для этого я перевёл углы в радианы и использовал специальную матрицу, умножение на которую даёт эффект поворота матрицы D .

```

octave:32> theta1 = 90*pi/180
theta1 = 1.5708
octave:33> R1 = [cos(theta1) -sin(theta1); sin(theta1) cos(theta1)]
R1 =

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

octave:34> RD1 = R1*D
RD1 =

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

octave:35> x1 = RD1(1,:)
x1 =

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

octave:36> y1 = RD1(2,:)
y1 =

    1    1    3    3    2    1    3

```

Figure 3.8: Перевод в радианы и расчёт для угла 90

```

octave:37> theta2 = 225*pi/180
theta2 = 3.9270
octave:38> R2 = [cos(theta2) -sin(theta2); sin(theta2) cos(theta2)]
R2 =

   -0.7071    0.7071
   -0.7071   -0.7071

octave:39> 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

octave:40> x2 = RD2(1,:)
x2 =

    0.7071   -0.7071   -2.1213   -0.7071    0.7071    0.7071   -0.7071

octave:41> y2 = RD2(2,:)
y2 =

   -2.1213   -0.7071   -2.1213   -3.5355   -3.5355   -2.1213   -3.5355

```

Figure 3.9: Перевод в радианы и расчёт для угла 225

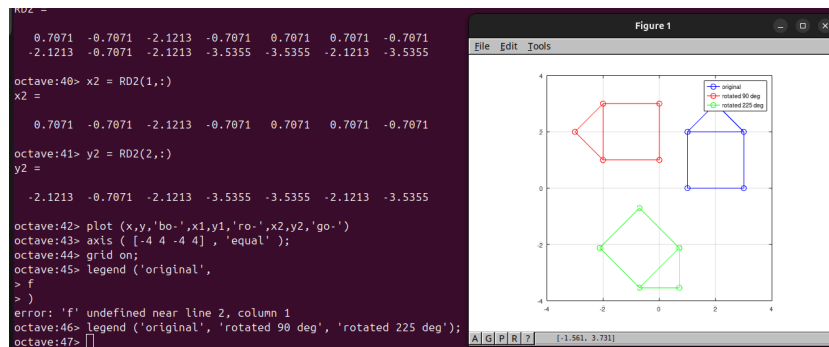


Figure 3.10: Построение оригинальной и повернутых матриц

Потом я прознакомился с отражением матрицы. Для этого необходимо основную матрицу D умножить на специальную матрицу для отражения. В данном случае отражение происходило относительно прямо $y = x$.

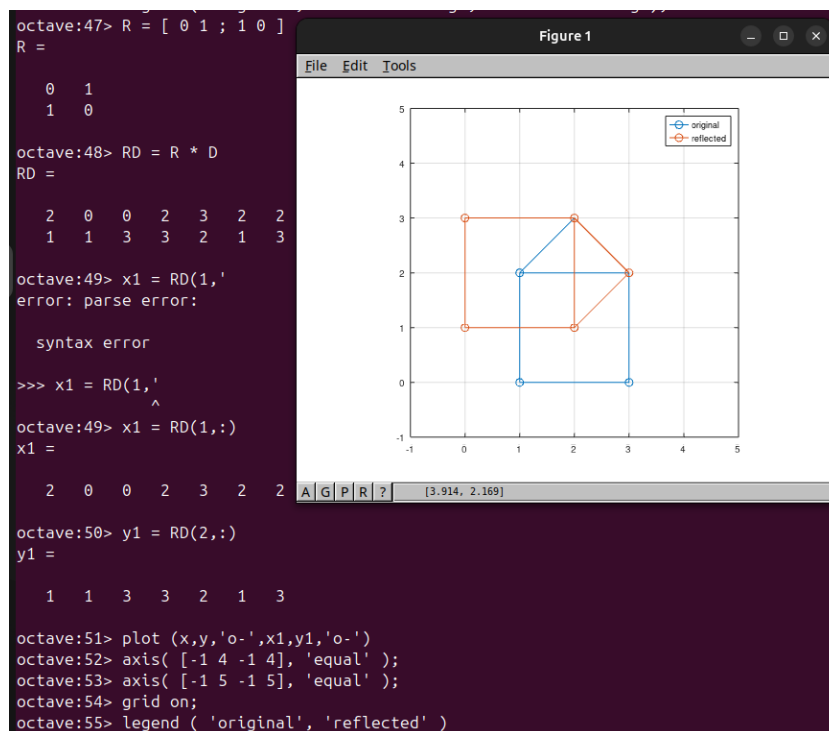


Figure 3.11: Отражение матрицы D

И наконец я познакомился с дилатацией (расширением или сжатием) матрицы. Оно может быть выполнено путём умножения матрицы на единичную матрицу, где вместо единиц задан коэффициент k , который является множителем дилата-

ции. В данном случае $k = 2$ и матрица D увеличилась вдвое.

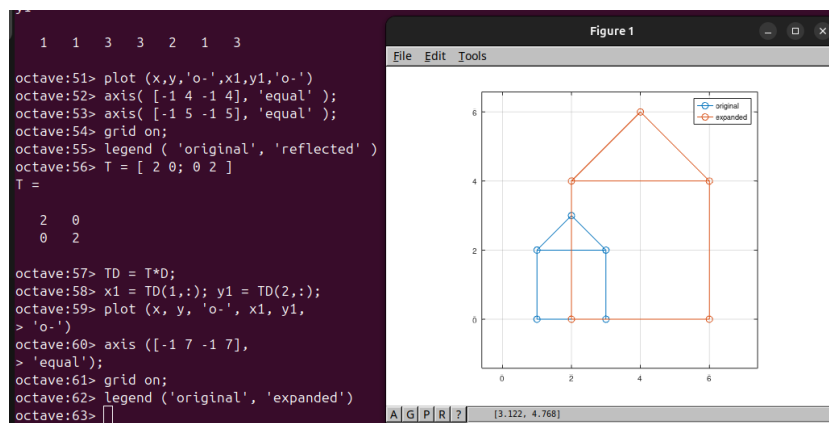


Figure 3.12: Дилатация матрицы D

4 Выводы

Я познакомился с подгонкой полиномиальной кривой и с матричными преобразованиями.