Aim

To understand the polynomial curve fitting using least square regression technique

Theory

A typical experiment collects data related to one parameter (say x) independent variable. These data (or the observations can be stored in to vectors (list of row or column of numbers) namely x and y. Using least square regression, it is possible to compute the coefficients of a polynomial function, of some selected degree, for y in terms of x.

Exercises

Let us consider the sample data given below:

x	-1	0	1	2	3	5	7	9
y	-1	3	2.5	5	4	2	5	4

1. Fit a 4^{th} order polynomial equation to the above data using least square regression in SCILAB.

   The general form of 4^{th} order polynomial is written as

   $$y = a_1 + a_2x + a_3x^2 + a_4x^3 + a_5x^4$$

   The simultaneous equation obtained from the regression analysis and written in the matrix form as follows:

   $$\begin{bmatrix} 8 & 26 & 170 & 1232 & 9686 \\ 26 & 170 & 1232 & 9686 & 79256 \\ 170 & 1232 & 9686 & 79256 & 665510 \\ 1232 & 9686 & 79256 & 665510 & 5686952 \\ 9686 & 79256 & 665510 & 5686952 & 49208966 \end{bmatrix} \times \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} = \begin{bmatrix} 24.5 \\ 106.5 \\ 676.5 \\ 5032.5 \\ 39902.5 \end{bmatrix}$$

   The above matrix representation can be represented in the form of

   $[A] = [X] \times [B]$ from this relation we can estimate the unknowns as

   $[X] = [A]^{-1} B$ using matrix computations.

   Solution for the above simultaneous equation can be obtained using SCILAB.

```
-->A = [8 26 170 1232 9686; 26 170 1232 9686
79256; 170 1232 9686 79256 665510; 1232 9686
79256 665510 5686952; 9686 79256 665510 5686952
49208966];

-->B=[24.5 106.5 676.5 5032.5 39902.5]'
B  =

24.5
106.5
676.5
5032.5
39902.5

-->x=A\B
 x  =

  2.6904606
  2.3044649
- 1.2374996
  0.2180404
- 0.0118925
```

Now the 4th order polynomial takes the form

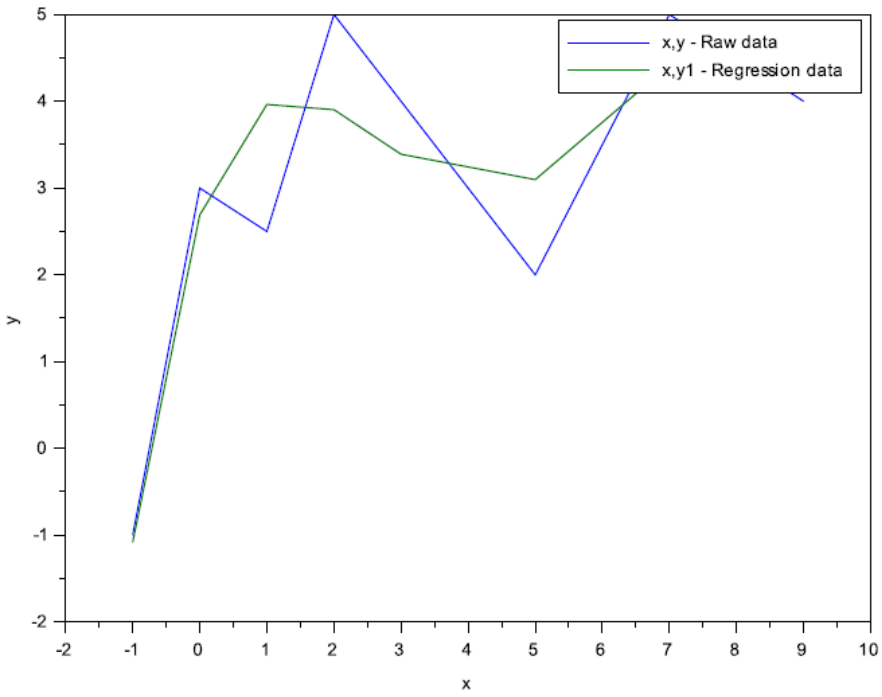
$y_1 = 2.6904606 + 2.3044649x - 1.2374996x^2 + 0.2180404\,x^3 - 0.0118925x^4$

Now plot the data (raw *x-y* data before the polynomial fit and *x-y* data in the above regression equation).

Plot for the raw data and the regression data

```
-->x=[-1 0 1 2 3 5 7 9]';
 x  =
   - 1.    0.    1.    2.    3.    5.    7.    9.
 -->y=[-1 3 2.5 5 4 2 5 4]';
 y  =
   - 1.    3.    2.5    5.    4.    2.    5.    4.
-->y1=[2.6904606+2.3044649*x-
1.2374996*x^2+0.2180404*x^3-
      0.0118925*x^4]
y1  =
   - 1.0814368
    2.6904606
    3.9635738
    3.9034352
```

```
    3.3901572
    3.0975326
    4.4181992
    4.1179362
-->plot(x,y,x,y1)
-->xlabel('x')
-->ylabel('y')
-->legend('x,y - Raw data','x,y1 - Regression
data')
```



2. Fit 3rd order polynomial for the data given in problem 1. using linear regression analysis technique. Plot the data (raw *x-y* data before the polynomial fit and the *x-y* data from the regression equation) and show the graph in SCILAB.

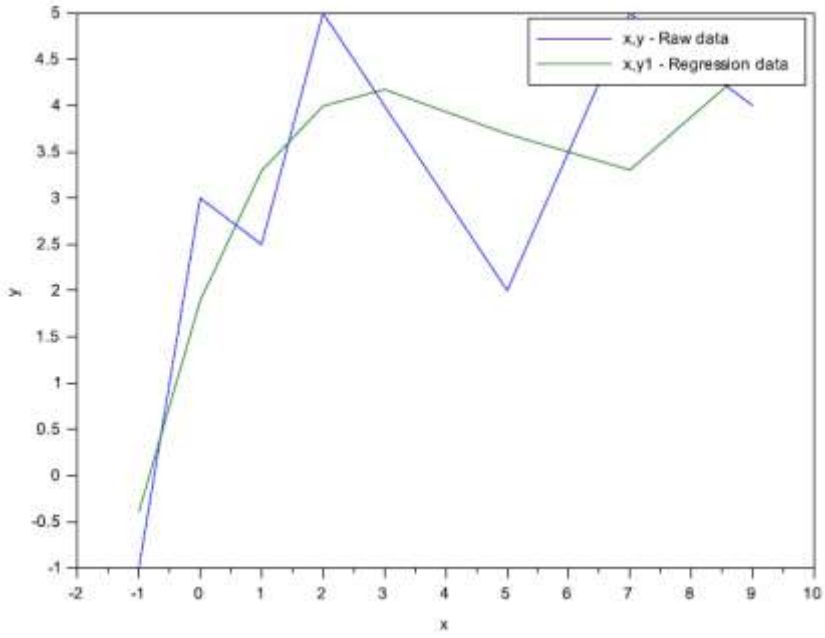
```
-->A=[8 26 170 1232; 26 170 1232 9686; 170 1232 9686
  79256; 1232 9686 79256 665510]
 A  =

    8.       26.      170.      1232.
    26.      170.     1232.     9686.
    170.     1232.    9686.     79256.
    1232.    9686.    79256.    665510.

-->B=[24.5 106.5 676.5 5032.5]'
 B  =

    24.5
```

```
    106.5
    676.5
    5032.5

-->x=A\B
 x   =

    1.8942743
    1.8132651
  - 0.4416466
    0.0302604
```

Now the 3rd order polynomial takes the form

$y_1 = 1.8942743 + 1.8132651\,x - 0.4416466\,x^2 + 0.0302\,x^3$

Plot for the raw data and the regression data



3. Fit 2nd order polynomial for the data given in problem 1. using linear regression analysis technique. Plot the data (raw *x-y* data before the polynomial fit and the *x-y* data from the regression equation) and show the graph in SCILAB.

```
-->A=[8 26 170; 26 170 1232; 170 1232 9686]
 A   =

    8.      26.     170.
    26.     170.    1232.
    170.    1232.   9686.

-->B=[24.5 106.5 676.5]'
 B   =
```

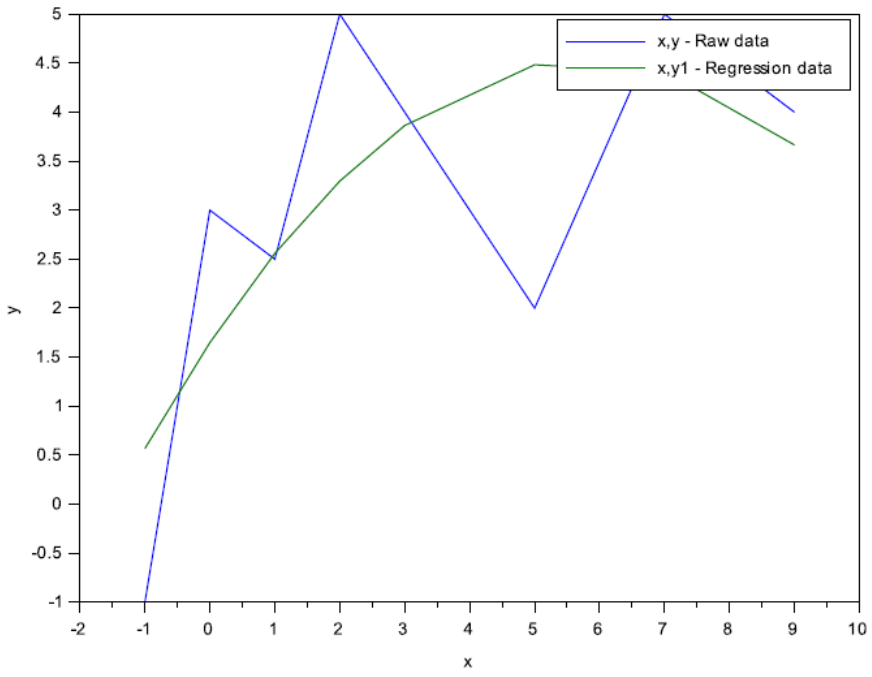
```
    24.5
    106.5
    676.5

-->x=A\B
 x   =

    1.6484391
    0.9955571
  - 0.0857176
```

Now the 2nd order polynomial takes the form

$y_1 = 1.8942743 + 1.8132651\,x - 0.4416466\,x^2$

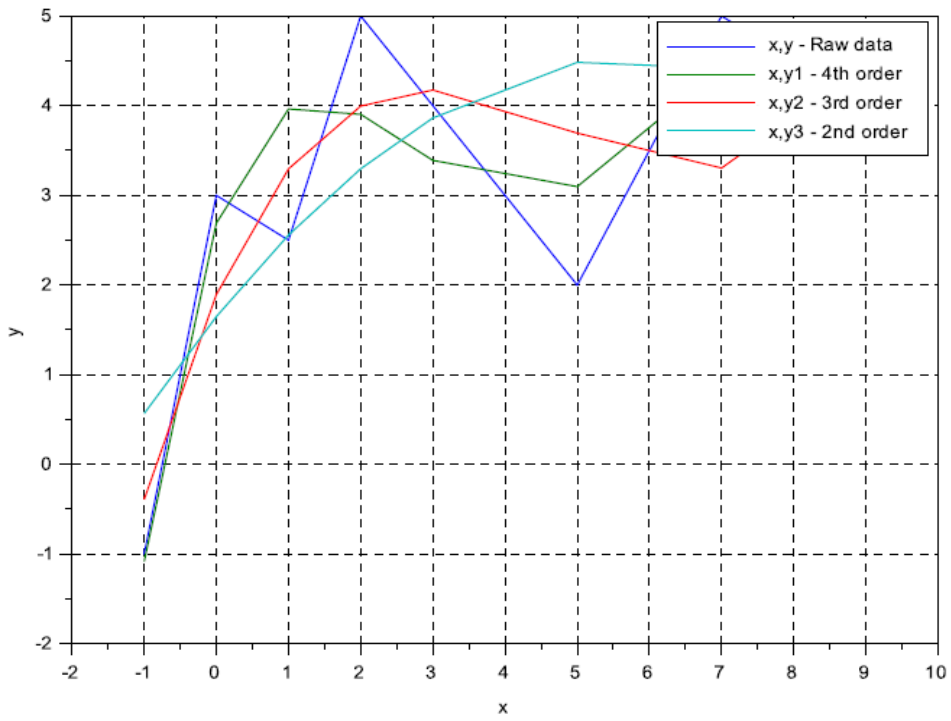
Plot for the raw data and the regression data

```
-->x=[-1 0 1 2 3 5 7 9]';
-->y=[-1 3 2.5 5 4 2 5 4]'
-->y1=[1.6484391+0.9955571*x-0.0857176*x^2]
-->plot(x,y,x,y1)
-->legend('x,y - Raw data','x,y1 - Regression
data')
-->xlabel('x')
-->ylabel('y')
```



Plot for all the three regression fit data

```
-->x=[-1 0 1 2 3 5 7 9]';
-->y=[-1 3 2.5 5 4 2 5 4]';
-->y1=[2.6904606+2.3044649*x-1.2374996*x^2+0.2180404*x^3-
       0.0118925*x^4]
-->y2=[1.8942743+1.8132651*x-0.4416466*x^2+0.0302*x^3]
-->y3=[1.6484391+0.9955571*x-0.0857176*x^2]
-->plot(x,y,x,y1,x,y2,x,y3)
-->legend('x,y - Raw data','x,y1 - 4th order','x,y2 - 3rd
order','x,y3 - 2nd order')
-->xlabel('x')
-->ylabel('y')
-->xgrid(1)
```



Result

Thus we learned the tool SCILAB for solving the polynomials and matrices.