

แบบฝึกหัด 6 LEAST-SQUARES REGRESSION

1. LINEAR REGRESSION

x	10	15	20	30	40	50	60	70	80
f(x)	5	9	15	18	22	30	35	38	43

1.1 จงใช้ LINEAR REGRESSION ในการคำนวณหา $f(x) = a_0 + a_1x_1$ และค่า $f(65)$

1.2 จงเขียน code

2. POLYNOMIAL REGRESSION

x	10	15	20	30	40	50	60	70	80
f(x)	5	9	15	18	22	30	35	38	43

2.1 จงใช้ POLYNOMIAL REGRESSION order $m = 2$ ในการสร้างสมการ $f(x) = a_0 + a_1x + a_2x^2$

2.2 จงเขียน code

3. MULTIPLE LINEAR REGRESSION

x_1	x_2	x_3	Y
1	0	1	4
0	1	3	-5
2	4	1	-6
3	2	2	0
4	1	5	-1
2	3	3	-7
1	6	4	-20

3.1 จงใช้ MULTIPLE LINEAR REGRESSION ในการสร้างสมการ $f(x) = a_0 + a_1x_1 + a_2x_2 + a_3x_3$

3.2 จงเขียน code

1. LINEAR REGRESSION

x	10	15	20	30	40	50	60	70	80
f(x)	5	9	15	18	22	30	35	38	43

1.1 จงใช้ LINEAR REGRESSION ในการคำนวณหา $f(x) = a_0 + a_1x_1$ และค่า $f(65)$

1.2 จงเขียน code

x_i	y_i	x_i^2	$x_i y_i$		
10	5	100	50	$a_0 = \frac{\left(\sum_{i=1}^n y_i\right)\left(\sum_{i=1}^n x_i^2\right) - \left(\sum_{i=1}^n x_i y_i\right)\left(\sum_{i=1}^n x_i\right)}{n\left(\sum_{i=1}^n x_i^2\right) - \left(\sum_{i=1}^n x_i\right)^2}$	$a_1 = \frac{n\left(\sum_{i=1}^n x_i y_i\right) - \left(\sum_{i=1}^n x_i\right)\left(\sum_{i=1}^n y_i\right)}{n\left(\sum_{i=1}^n x_i^2\right) - \left(\sum_{i=1}^n x_i\right)^2}$
15	9	225	135	$= \frac{(215)(20625) - (1165)(1375)}{9(20625) - (1165)^2}$	$= \frac{9(1165) - (1165)(215)}{9(20625) - (1165)^2}$
20	15	400	300		
30	18	900	540		
40	22	1600	880	$= 1.915555$	$= 0.519555$
50	30	2500	1500		
60	35	3600	2100	$f(65) = 1.915555 + 0.519555x$	
70	38	4900	2660	$= 36.180000$	
80	43	6400	3440		
Σ	375	215	20625	1165	

```
1 function sum(x){
2   let sum=0;
3   for(let i=0; i<x.length; i++){
4     sum += x[i];
5   }
6   return sum;
7 }
8
9 function pow_sum(x){
10  let sum=0;
11  for(let i=0; i<x.length; i++){
12    sum += Math.pow(x[i],2);
13  }
14  return sum;
15 }
16
17 function multiple_sum(x,y){
18  let sum=0;
19  for(let i=0; i<x.length; i++){
20    sum += x[i] * y[i];
21  }
22  return sum;
23 }
24
25 function linear_regression(points,x,y,fx){
26  let a0,a1;
27
28  a0 = (sum(y) * pow_sum(x) - multiple_sum(x,y) * sum(x)) / (points.length * pow_sum(x) - Math.pow(sum(x),2));
29  a1 = (points.length * multiple_sum(x,y) - sum(x) * sum(y)) / (points.length * pow_sum(x) - Math.pow(sum(x),2));
30
31  return "f(x) = " + a0.toFixed(6) + " + " + a1.toFixed(6) + "x" + "\n" +
32    "f'(x) = " + " + ((a0 + (a1 * fx)).toFixed(6));
33 }
34
35 let points = [
36   {x: 10, y: 5},
37   {x: 15, y: 9},
38   {x: 20, y: 15},
39   {x: 30, y: 18},
40   {x: 40, y: 22},
41   {x: 50, y: 30},
42   {x: 60, y: 35},
43   {x: 70, y: 38},
44   {x: 80, y: 43}
45 ]
46
47 let x = points.map(points => points.x);
48 let y = points.map(points => points.y);
49 let fx = 65;
50
51 console.log(linear_regression(points,x,y,fx));
```

2. POLYNOMIAL REGRESSION

x	10	15	20	30	40	50	60	70	80
f(x)	5	9	15	18	22	30	35	38	43

2.1 จงใช้ POLYNOMIAL REGRESSION order m = 2 ในการสร้างสมการ $f(x) = a_0 + a_1x + a_2x^2$

2.2 จงเขียน code

x_i	y_i	x_i^2	x_i^3	x_i^4	$x_i \cdot f(x_i)$	$x_i^2 \cdot f(x_i)$
10	5	100	1000	10000	50	500
15	9	225	3375	50625	135	2025
20	15	400	8000	160000	300	6000
30	18	900	27000	810000	540	16200
40	22	1600	64000	2560000	880	35200
50	30	2500	125000	6250000	1500	75000
60	35	3600	216000	12960000	2100	126000
70	38	4900	343000	24010000	2660	186200
80	43	6400	512000	40960000	3440	275200
Σ	375	215	20625	1165175	11605	783525

$$y_i = a_0 + a_1x_i + a_2x_i^2 \quad f(x) = a_0 + a_1x + a_2x^2$$
$$\begin{bmatrix} 11 & \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 \\ \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i^3 \\ \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i^3 & \sum_{i=1}^n x_i^4 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^n f(x_i) \\ \sum_{i=1}^n x_i \cdot f(x_i) \\ \sum_{i=1}^n x_i^2 \cdot f(x_i) \end{bmatrix}$$
$$\downarrow$$
$$\begin{bmatrix} 11 & 375 & 20625 \\ 375 & 20625 & 1165175 \\ 20625 & 1165175 & 783525 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 11605 \\ 783525 \\ 372955 \end{bmatrix}$$
$$A \cdot B$$
$$\begin{bmatrix} -0.706980 \\ 0.641524 \\ -0.001945 \end{bmatrix}$$

$$f(x) = -0.706980 + 0.641524x + (-0.001945)x^2$$
$$= 36.18730$$

```
1 const math = require('mathjs'); //import math.js
2
3 function polynomial_regression(x,y,fx){
4   let sumx2=0,sumx3=0,sumx4=0,sumy=0,sumxy=0,n=x.length;
5
6   for(let i=0; i<x.length; i++){
7     sumx2 += Math.pow(x[i],2);
8     sumx3 += Math.pow(x[i],3);
9     sumx4 += Math.pow(x[i],4);
10    sumy += Math.pow(x[i],4);
11    sumxy += x[i] * y[i];
12    sumxy2 += Math.pow(x[i],2) * y[i];
13  }
14
15  let A = [
16    [n, math.sum(x), sumx2],
17    [math.sum(x), sumx2, sumx3],
18    [sumx2, sumx3, sumx4],
19  ];
20  A = math.inv(A); //Matrix Inverse
21
22  let B = [math.sum(y), sumxy, sumxy2];
23
24  let a = math.multiply(A,B); //Multiply matrix A and B
25
26  return "f(x) = " + a[0].toFixed(6) + " + " + a[1].toFixed(6) + "x" + " + " + a[2].toFixed(6) + "x^2" + "\n" +
27    "f'(x) = " + a[0].toFixed(6) + a[1] * fx + a[2] * Math.pow(fx,2)).toFixed(6);
28 }
29
30 let points = [
31   {x: 10, y: 5},
32   {x: 15, y: 9},
33   {x: 20, y: 15},
34   {x: 30, y: 18},
35   {x: 40, y: 22},
36   {x: 50, y: 30},
37   {x: 60, y: 35},
38   {x: 70, y: 38},
39   {x: 80, y: 43}
40 ]
41
42 let x = points.map(points => points.x);
43 let y = points.map(points => points.y);
44 let fx = 65;
45
46 console.log(polynomial_regression(x,y,fx));
```

3. MULTIPLE LINEAR REGRESSION

X ₁	X ₂	X ₃	Y	x ₁ x ₁	x ₁ x ₂	x ₁ x ₃	x ₁ x ₂	x ₁ x ₃	x ₂ x ₃	x ₁ y	x ₂ y	x ₃ y
1	0	1	4	1	0	1	0	1	0	4	0	4
0	1	3	-5	0	1	3	0	0	3	0	-5	-15
2	4	1	-6	4	16	1	8	2	4	-12	-24	-6
3	2	2	0	9	4	4	6	6	4	0	0	0
4	1	5	-1	16	1	25	4	20	5	-4	-1	-5
2	3	3	-7	4	9	9	6	6	9	-14	-21	-21
1	6	4	-20	1	36	16	6	4	24	-10	-120	-80
Σ	13	17	19	-35	35	67	65	30	39	49	-46	-171

3.1 จงใช้ MULTIPLE LINEAR REGRESSION ในการสร้างสมการ $f(x) = a_0 + a_1x_1 + a_2x_2 + a_3x_3$

3.2 จงเขียน code

$$\begin{bmatrix} n & \sum_{i=1}^n x_{1i} & \sum_{i=1}^n x_{2i} & \sum_{i=1}^n x_{3i} \\ \sum_{i=1}^n x_{1i} & \sum_{i=1}^n x_{1i}^2 & \sum_{i=1}^n x_{1i}x_{2i} & \sum_{i=1}^n x_{1i}x_{3i} \\ \sum_{i=1}^n x_{2i} & \sum_{i=1}^n x_{1i}x_{2i} & \sum_{i=1}^n x_{2i}^2 & \sum_{i=1}^n x_{2i}x_{3i} \\ \sum_{i=1}^n x_{3i} & \sum_{i=1}^n x_{1i}x_{3i} & \sum_{i=1}^n x_{2i}x_{3i} & \sum_{i=1}^n x_{3i}^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^n y_i \\ \sum_{i=1}^n x_{1i}y_i \\ \sum_{i=1}^n x_{2i}y_i \\ \sum_{i=1}^n x_{3i}y_i \end{bmatrix}$$

$$\begin{bmatrix} 7 & 13 & 17 & 19 \\ 13 & 35 & 30 & 39 \\ 17 & 30 & 67 & 49 \\ 19 & 39 & 49 & 65 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} -35 \\ -46 \\ -171 \\ -123 \end{bmatrix}$$

$$A^{-1} \cdot B$$

$$\begin{bmatrix} 4.000000 \\ 2.000000 \\ -1.000000 \end{bmatrix}$$

$$f(x) = 4.000000 + 2.000000x_1 + (-1.000000)x_2 + (-1.000000)x_3$$

```

1  const math = require('mathjs');
2
3  function multiple_linear_regression(x1,x2,x3,y,fx){
4      let A = [
5          [x1.length,math.sum(x1),math.sum(x2),math.sum(x3)],
6          [math.sum(x1),math.multiply(x1,x1),math.multiply(x1,x2),math.multiply(x1,x3)],
7          [math.sum(x2),math.multiply(x1,x2),math.multiply(x2,x2),math.multiply(x2,x3)],
8          [math.sum(x3),math.multiply(x1,x3),math.multiply(x2,x3),math.multiply(x3,x3)]
9      ]
10
11      let B = [math.sum(y),math.multiply(x1,y),math.multiply(x2,y),math.multiply(x3,y)];
12
13      A = math.inv(A); //matrix inverse
14
15      let a = math.multiply(A,B); //multiply matrix A and B
16
17      return "f(x) = "+parseFloat(a[0]).toFixed(6)+" + "+parseFloat(a[1]).toFixed(6)+"x1"+" + "+parseFloat(a[2]).toFixed(6)+"x2"+" + "+parseFloat(a[3]).toFixed(6)+"x3";
18  }
19
20  let points = [
21      {x1: 1,x2: 0,x3: 1,y: 4},
22      {x1: 0,x2: 1,x3: 3,y: -5},
23      {x1: 2,x2: 4,x3: 1,y: -6},
24      {x1: 3,x2: 2,x3: 2,y: 0},
25      {x1: 4,x2: 1,x3: 5,y: -1},
26      {x1: 2,x2: 3,x3: 3,y: -7},
27      {x1: 1,x2: 6,x3: 4,y: -20},
28  ]
29
30  let x1 = points.map(points => points.x1);
31  let x2 = points.map(points => points.x2);
32  let x3 = points.map(points => points.x3);
33  let y = points.map(points => points.y);
34  let fx = 65;
35
36  console.log(multiple_linear_regression(x1,x2,x3,y,fx));

```