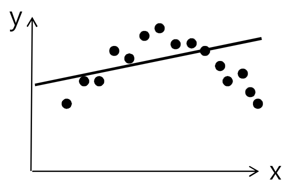
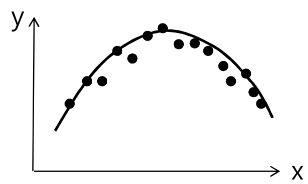
15. General Linear Least-Squares & Nonlinear Regression

HoHee Kim

Polynomial Regression



a)



b)

☞ 1차 다항식의 regression 보다 2차 다항식의 regression 이 바람직

- 2차 polynomial 으로 fitting 가정,

  n 개의 점 $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ 일 때,

$$y = a_0 + a_1 x + a_2 x^2 + e \,, \qquad S_r = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} \left(y_i - a_0 - a_1 x_i - a_2 x_i^2\right)^2$$

Residual 의 제곱의 합

☞ S_r 이 최소화 되도록 a_0, a_1, a_2 를 결정하여 유일한 2차 다항식을 유도

$$\frac{\partial S_r}{\partial a_0} = -2\sum(y_i - a_0 - a_1x_i - a_2x_i^2) = 0$$
$$\frac{\partial S_r}{\partial a_1} = -2\sum[(y_i - a_0 - a_1x_i - a_2x_i^2)x_i] = 0$$
$$\frac{\partial S_r}{\partial a_2} = -2\sum[(y_i - a_0 - a_1x_i - a_2x_i^2)x_i^2] = 0$$

최소가 되려면
미분 = 0

Normal equation

$$na_0 + (\sum x_i)a_1 + (\sum x_i^2)a_2 = \sum y_i$$
$$(\sum x_i)a_0 + (\sum x_i^2)a_1 + (\sum x_i^3)a_2 = \sum x_i y_i$$
$$(\sum x_i^2)a_0 + (\sum x_i^3)a_1 + (\sum x_i^4)a_2 = \sum x_i^2 y_i$$

☞ 수치데이터를 이용하여 a_0, a_1, a_2 를 구함
→ 유일한 2차 다항식

- m th-order polynomial 으로 fitting 가정,

$$y = a_0 + a_1x + a_2x^2 + \cdots + a_mx^m + e \ , \quad S_r = \sum_{i=1}^{n}(y_i - a_0 - a_1x_i - a_2x_i^2 - \cdots - a_mx_i^m)^2$$

- standard error & coefficient of determination :

$$s_{y/x} = \sqrt{\frac{S_r}{n-(m+1)}} \qquad r^2 = \frac{S_t - S_r}{S_t}$$

m차 다항식의 계수를 구하기 위해 (m+1)개 data가 있어야 하므로

Multiple Linear Regression: y 가 두 개 이상의 독립변수로 된 선형 함수 형태로 fitting , 여기서 두 개의 독립변수 가정,

$$y = a_0 + a_1x_1 + a_2x_2 + e \qquad S_r = \sum_{i=1}^{n}(y_i - a_0 - a_1x_{1,i} - a_2x_{2,i})^2$$

$$\frac{\partial S_r}{\partial a_0} = -2\sum(y_i - a_0 - a_1x_{1,i} - a_2x_{2,i}) = 0$$
$$\frac{\partial S_r}{\partial a_1} = -2\sum[(y_i - a_0 - a_1x_{1,i} - a_2x_{2,i})x_{1,i}] = 0$$
$$\frac{\partial S_r}{\partial a_2} = -2\sum[(y_i - a_0 - a_1x_{1,i} - a_2x_{2,i})x_{2,i}] = 0$$

최소가 되려면
미분 = 0

Normal equation

$$na_0 + (\sum x_{1,i})a_1 + (\sum x_{2,i})a_2 = \sum y_i$$
$$(\sum x_{1,i})a_0 + (\sum x_{1,i}^2)a_1 + (\sum x_{1,i}x_{2,i})a_2 = \sum x_{1,i}y_i$$
$$(\sum x_{2,i})a_0 + (\sum x_{1,i}x_{2,i})a_1 + (\sum x_{2,i}^2)a_2 = \sum x_{2,i}y_i$$

$$s_{y/x} = \sqrt{\frac{S_r}{n-(2+1)}}$$

$$r^2 = \frac{S_t - S_r}{S_t}$$

☞ 수치데이터를 이용하여 a_0, a_1, a_2 를 구함

General Linear Regression: linear 과 multiple 과 polynomial 을 포함

$$y = a_0 z_0 + a_1 z_1 + a_2 z_2 + \cdots a_m z_m + e$$

- z_0, z_1, …, z_m 는 (m+1)개의 basic 함수이며 a 들에 대해 선형이라야 함

multiple : $z_0=1$, $z_1= x_1$, $z_2= x_2$, … , $z_m= x_m$

polynomial : $z_0=1$, $z_1= x$, $z_2= x^2$, … , $z_m= x^m$

예) $y = a_0 + a_1 \cos(\omega x) + a_2 \sin(\omega x)$ (O) , $y = a_0(1 - e^{-a_1 x})$ (X)

$$\{y\} = [Z]\{a\} + \{e\}$$

독립변수의 측정값에서 basic 함수의 계산 값으로 이루어진 행렬

n X (m+1)

$$[Z] = \begin{bmatrix} z_{01} & z_{11} & \cdots & z_{m1} \\ z_{02} & z_{12} & \cdots & z_{m2} \\ \vdots & \vdots & \ddots & \vdots \\ z_{0n} & z_{1n} & \cdots & z_{mn} \end{bmatrix}$$

n ≥ (m+1)

$$\{y\}^T = \lfloor y_1 \quad y_2 \quad \cdots \quad y_n \rfloor$$

$$\{a\}^T = \lfloor a_0 \quad a_1 \quad \cdots \quad a_m \rfloor$$

$$\{e\}^T = \lfloor e_1 \quad e_2 \quad \cdots \quad e_n \rfloor$$

$$S_r = \sum_{i=1}^{n} \left(y_i - \sum_{j=0}^{m} a_j z_{ji} \right)^2$$

최소화 되도록 a_j 들에 대해 편미분 = 0 하면

$$\boxed{[Z]^T [Z]\{a\} = [Z]^T \{y\}}$$

Normal equation

$$\longrightarrow \{a\} = \left[[Z]^T [Z]\right]^{-1} [Z]^T \{y\}$$

- Normal equation 이 ill-conditioned 시스템 경우 반올림오차에 민감

→ QR factorization 을 사용하여 {a} 를 구함 (Z=QR , $\{a\} = R^{-1}Q^T\{y\}$)

→ MATLAB의 polyfit() 과 left division 에서 적용

- standard error & coefficient of determination :

$$s_{y/x} = \sqrt{\frac{S_r}{n-(m+1)}} \qquad r^2 = \frac{S_t - S_r}{S_t}$$

예) 4개의 Data 점 (x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4) 을 2차 다항식으로 fitting

$$y = a_0 + a_1 x + a_2 x^2 + e \longrightarrow z_0 = 1,\quad z_1 = x,\quad z_2 = x^2$$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ 1 & x_3 & x_3^2 \\ 1 & x_4 & x_4^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \end{bmatrix}$$

$$e_i = y_i - a_0 - a_1 x_i - a_2 x_i^2$$

$$S_r = \sum_{i=1}^{4} e_i^2 = \sum_{i=1}^{4} \left(y_i - a_0 - a_1 x_i - a_2 x_i^2\right)^2$$

$$[Z]^T [Z]\{a\} = [Z]^T \{y\}$$

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ x_1 & x_2 & x_3 & x_4 \\ x_1^2 & x_2^2 & x_3^2 & x_4^2 \end{bmatrix} \begin{bmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ 1 & x_3 & x_3^2 \\ 1 & x_4 & x_4^2 \end{bmatrix} = \begin{bmatrix} n & \sum x_i & \sum x_i^2 \\ \sum x_i & \sum x_i^2 & \sum x_i^3 \\ \sum x_i^2 & \sum x_i^3 & \sum x_i^4 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ x_1 & x_2 & x_3 & x_4 \\ x_1^2 & x_2^2 & x_3^2 & x_4^2 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} \sum y_i \\ \sum x_i y_i \\ \sum x_i^2 y_i \end{bmatrix}$$


```
>> x = [0 1 2 3 4 5]';
>> y = [2.1  7.7  13.6  27.2  40.9  61.1]';
>> z = [ones(size(x)) x x.^2]
 z =     1      0      0
         1      1      1
         1      2      4
         1      3      9
         1      4     16
         1      5     25
>> z'*z
 ans =    6     15     55
         15     55    225
         55    225    979
>> a = (z'*z)\(z'*y)
 a =     2.4786
         2.3593
         1.8607
```

polynomial regression

Normal equation 을 적용

$$\Rightarrow y = 1.8607x^2 + 2.3593x + 2.4786$$

```
>> Sr = sum((y-z*a).^2)
 Sr =
        3.7466
```

$$S_r = \sum_{i=1}^{n}\left(y_i - \sum_{j=0}^{m} a_j z_{ji}\right)^2$$

```
>> r2 = 1-Sr/sum((y-mean(y)).^2)
 r2 =
        0.9985
```

$$r^2 = 1 - \frac{S_r}{S_t}$$

```
>> syx = sqrt(Sr/(length(x)-length(a)))
 syx =
        1.1175
```

$$s_{y/x} = \sqrt{\frac{S_r}{n-(m+1)}}$$

```
>> a = polyfit(x,y,2)
 a =
      1.8607    2.3593    2.4786
```

QR factorization 을 적용

$\Rightarrow y = 1.8607x^2 + 2.3593x + 2.4786$

```
>> a= z\y
 a =    2.4786
        2.3593
        1.8607
```

z 가 square 가 아닌 경우
Left division은
QR factorization 을 적용

Nonlinear Regression : 비선형 모델로 fitting

예) $y = a_0(1-e^{-a_1 x}) + e$

① Gauss-Newton method : Taylor series 사용하여 비선형을 선형으로 근사

② optimization method를 이용: $S_r = f(a_0, a_1) = \sum_{i=1}^{n}\left(y_i - a_0(1-e^{-a_1 x_i})\right)^2$

☞ S_r 을 최소화 하는 a_0, a_1 를 구함

예)

fSSR.m

```
function f = fSSR( a, xm, ym )
  yp = a(1)*xm.^a(2);
  f = sum((ym-yp).^2);
end
```

```
>> x = [10 20 30 40 50 60 70 80];
>> y = [25 70 380 550 610 1220 830 1450];
>> fminsearch( @fSSR, [1,1], [], x, y )
 ans =   2.5384    1.4359
```

초기값벡터 option 이 없을 때

$\Rightarrow y = 2.5384x^{1.4359}$