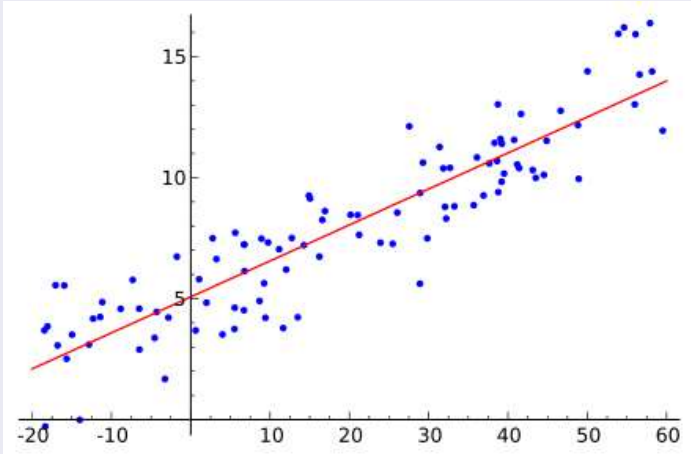
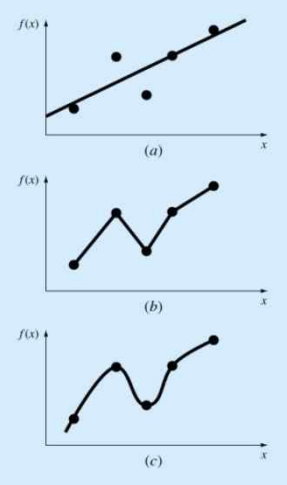
Part 4
Chapter 14

Linear Regression



CURVE FITTING

- There are two general approaches two curve fitting
  - Data exhibit a significant degree of scatter
    - The strategy is to derive a single curve that represents the general trend of the data.
  - Data is very precise
    - The strategy is to pass a curve or a series of curves through each of the points.
- In engineering two types of applications are encountered
  - Trend analysis
    - Predicting values of dependent variable
  - Hypothesis testing
    - Comparing existing mathematical model with measured data



Statistics Review : Measure of Location

- Arithmetic mean
  - the sum of the individual data points (y_i) divided by the number of points n:
- Median
  - the midpoint of a group of data.
- Mode
  - the value that occurs most frequently in a group of data.

$$\bar{y} = \frac{\sum y_i}{n}$$

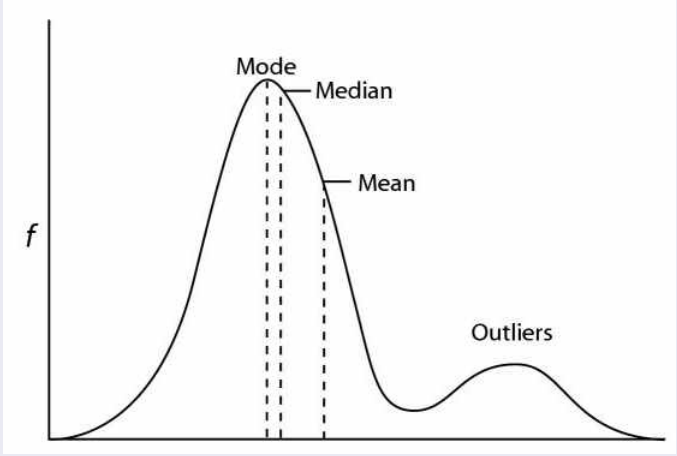
13, 18, 13, 14, 13, 16, 14, 21, 13

mean $(13 + 18 + 13 + 14 + 13 + 16 + 14 + 21 + 13) \div 9 = 15$

median $(9 + 1) \div 2 = 5\text{th}$ 13, 13, 13, 13, 14, 14, 16, 18, 21

mode 13

range $21 - 13 = 8$



Statistics Review: Measures of Spread

- (Sample) Variance : $$s_y^2 = \frac{\sum (y_i - \bar{y})^2}{n-1} = \frac{\sum y_i^2 - \left(\sum y_i\right)^2 / n}{n-1}$$

$$s_y^2 = \frac{S_t}{n-1}$$

  - S_t : sum of squares (of residuals) $S_t = \sum (y_i - \bar{y})^2$
  - (n-1) : degrees of freedom
  - (cf.) Population Variance : $\sigma^2 = \frac{S_t}{n}$

- Standard deviation : $s_y = \sqrt{\frac{S_t}{n-1}}$

- Coefficient of variation: $\text{c.v.} = \frac{s_y}{\bar{y}} \times 100\%$

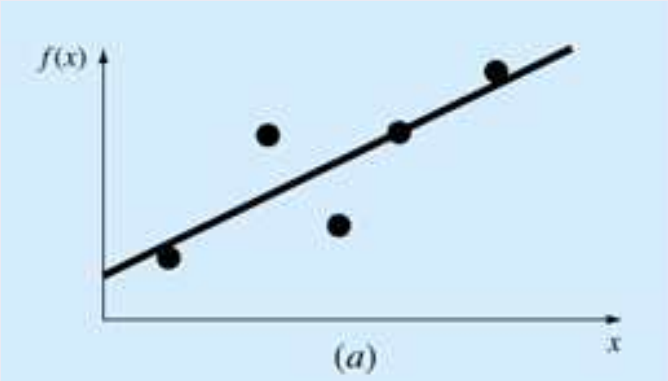
Descriptive Statistics in MATLAB

- MATLAB : assuming some column vector *s*
  - mean(*s*), median(*s*), mode(*s*)
    - Calculate the mean, median, and mode of *s*.
  - min(*s*), max(*s*)
    - Calculate the minimum and maximum value in *s*.
  - var(*s*), std(*s*)
    - Calculate the variance and standard deviation of *s*
- Note : if a matrix is given, the statistics will be returned for each column.

```
A = [1 2 3; 3 3 6; 4 6 8; 4 7 7];
mean(A)
ans =
    3.0000    4.5000    6.0000
```

Least Squares Regression

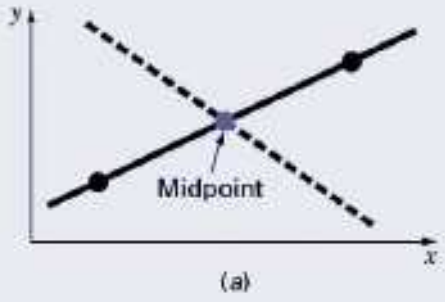
- Linear Regression
  - Fitting a straight line to a set of paired observations: $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$.
  - $y=a_0+a_1x+e$
    - a_1 : slope
    - a_0 : intercept
    - e : error/residual between model and observations



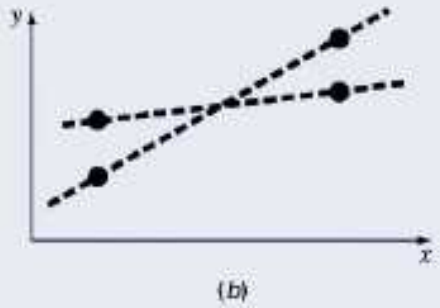
Linear Regression

- Determine a_0 *and* a_1 *by* minimizing:
  - Sum of error
  - Sum of absolute errors : +/- errors trade-off
  - minimax : 각점의 최대오차를 최소화

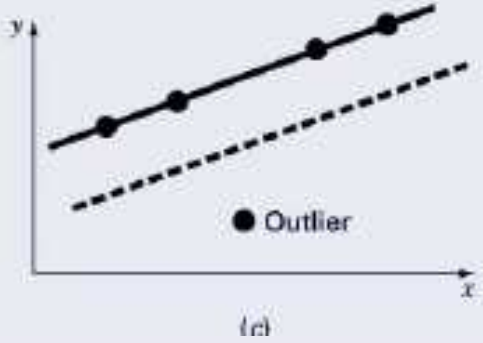
기준 적용시 문제



잔차합 : 연결선의 중앙을 지나는 모든 직선은 잔차합 0



잔차 절대값 합: 두점선 사이의 모든 직선 은 잔차의 절대값 합을 최소화



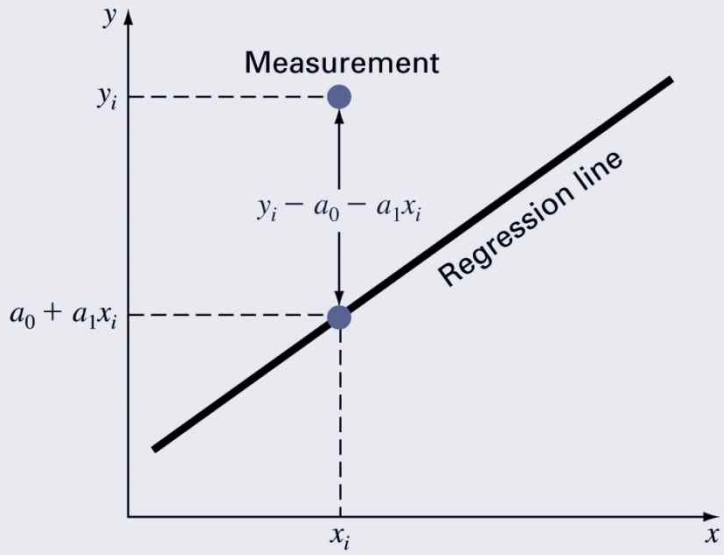
Minimax: 각 데이터 점이 직선으로 부터 떨어진 최대거리를 최소화

Linear Least-Squares Regression

- Determine the ‘best’ coefficients in a linear model for given data set.
  - ‘Best’ means minimizing the sum of the squares of the *estimate* residuals.
  - For a straight line model, this gives:

$$y = a_0 + a_1 x$$

$$S_r = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (y_i - a_0 - a_1 x_i)^2$$



Least-Squares Fit

$$S_r = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (y_i - a_0 - a_1 x_i)^2$$

$$\frac{\partial S_r}{\partial a_o} = -2\sum (y_i - a_o - a_1 x_i) = 0$$

$$\frac{\partial S_r}{\partial a_1} = -2\sum [(y_i - a_o - a_1 x_i) x_i] = 0$$

$$0 = \sum y_i - \sum a_0 - \sum a_1 x_i$$

$$0 = \sum y_i x_i - \sum a_0 x_i - \sum a_1 x_i^2$$

$$a_1 = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{n\sum x_i^2 - \left(\sum x_i\right)^2}$$

$$a_0 = \bar{y} - a_1 \bar{x}$$

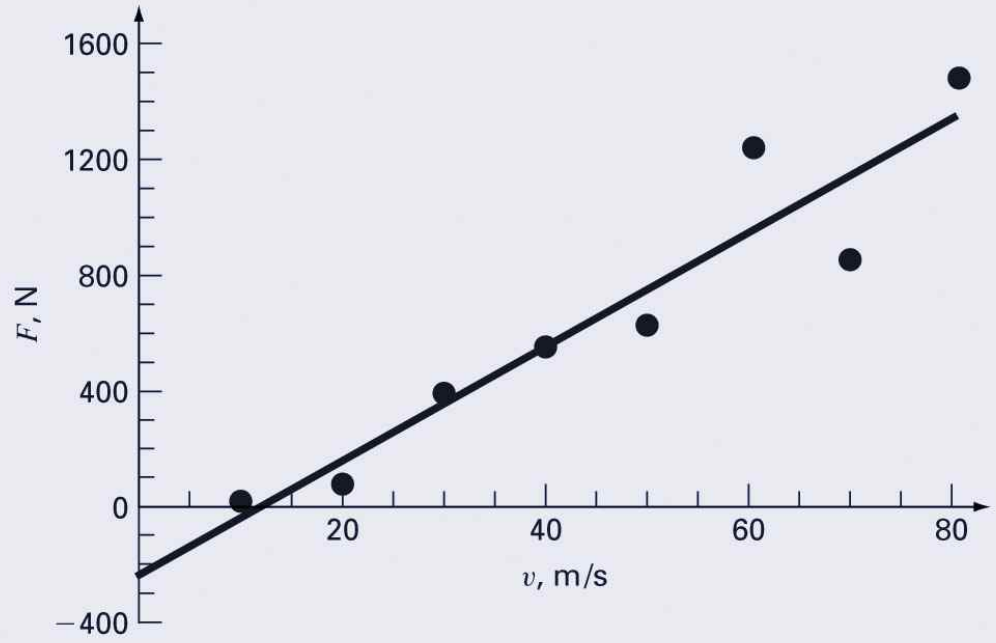
Example

	V (m/s)	F (N)		
i	x_i	y_i	$(x_i)^2$	$x_i y_i$
1	10	25	100	250
2	20	70	400	1400
3	30	380	900	11400
4	40	550	1600	22000
5	50	610	2500	30500
6	60	1220	3600	73200
7	70	830	4900	58100
8	80	1450	6400	116000
Σ	360	5135	20400	312850

$$a_1 = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{n\sum x_i^2 - \left(\sum x_i\right)^2} = \frac{8(312850) - (360)(5135)}{8(20400) - (360)^2} = 19.47024$$

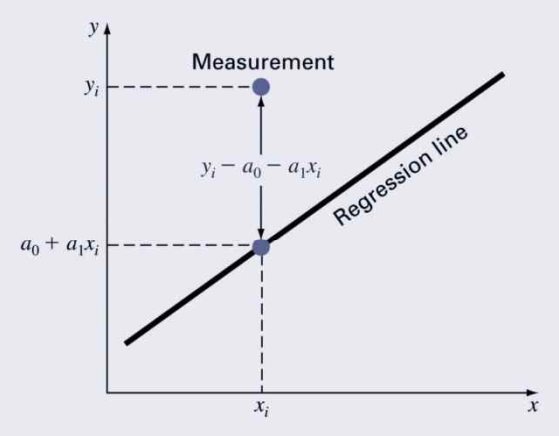
$$a_0 = \bar{y} - a_1 \bar{x} = 641.875 - 19.47024(45) = -234.2857$$

$$F_{est} = -234.2857 + 19.47024v$$



Quantification of Error

- Recall for a straight line, the sum of the squares of the estimate residuals:

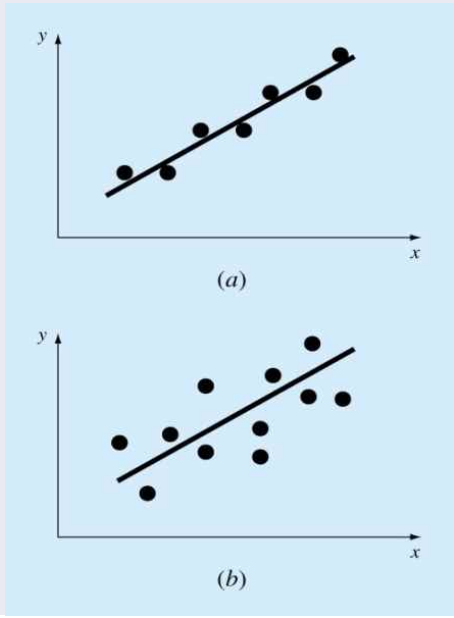


$$S_r = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (y_i - a_0 - a_1 x_i)^2$$

- *Standard error of the estimate:*

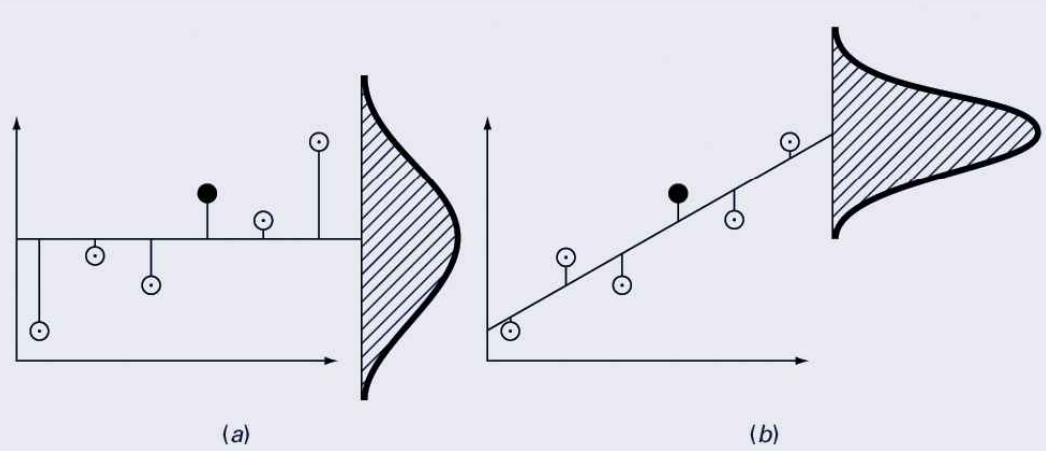
$$s_{y/x} = \sqrt{\frac{S_r}{n-2}}$$

n-2 : (모수 2개),
(점이 2개이면 직선,
데이터 분포 의미
없음).



Standard Error of the Estimate

- Regression data showing (a) the spread of data around the mean of the dependent data and (b) the spread of the data around the best fit line:

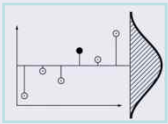


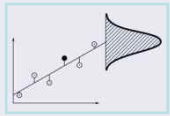
- The reduction in spread represents the improvement due to linear regression.

Coefficient of Determination

- The *coefficient of determination* (R^2)
  - the difference between the sum of the squares of the data residuals and the sum of the squares of the estimate residuals, normalized by the sum of the squares of the data residuals :

$$S_t = \sum (y_i - \bar{y})^2 \qquad S_r = \sum (y_i - a_0 - a_1 x_i)^2 \qquad R^2 = \frac{S_t - S_r}{S_t}$$





- R^2 represents the percentage of the original uncertainty explained by the model.
  - For a perfect fit, S_r=0 and R^2=1
  - If R^2=0, there is no improvement over simply picking the mean
  - If R^2<0, the model is *worse* than simply picking the mean
    - can yield negative values, depending on the definition used
  - Sqrt(R^2) : correlation coefficient (r)

Example

	V (m/s)	F (N)			
i	x_i	y_i	$a_0+a_1x_i$	$(y_i-\bar{y})^2$	$(y_i-a_0-a_1x_i)^2$
1	10	25	-39.58	380535	4171
2	20	70	155.12	327041	7245
3	30	380	349.82	68579	911
4	40	550	544.52	8441	30
5	50	610	739.23	1016	16699
6	60	1220	933.93	334229	81837
7	70	830	1128.63	35391	89180
8	80	1450	1323.33	653066	16044
Σ	360	5135		1808297	216118

$$F_{est} = -234.2857 + 19.47024v$$

$$S_t = \sum (y_i - \bar{y})^2 = 1808297$$

$$S_r = \sum (y_i - a_0 - a_1 x_i)^2 = 216118$$

$$s_y = \sqrt{\frac{1808297}{8-1}} = 508.26$$

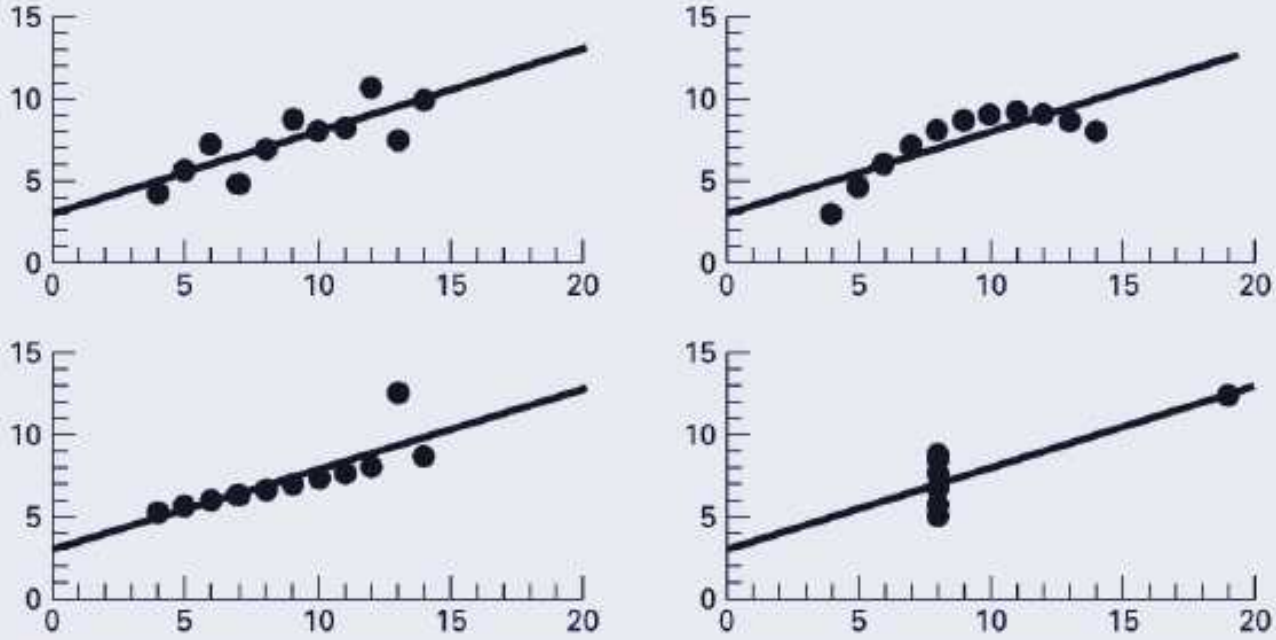
$$s_{y/x} = \sqrt{\frac{216118}{8-2}} = 189.79$$

$$R^2 = \frac{1808297 - 216118}{1808297} = 0.8805$$

88.05% of the original uncertainty has been explained by the linear model

Coefficient of Determination

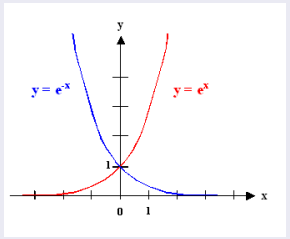
- Not always correct.
  - The following all have the same best-fit line and R^2.
    - $y = 3+0.5x$, $R^2 = 0.67$
  - Needs evaluation of the scatter diagram



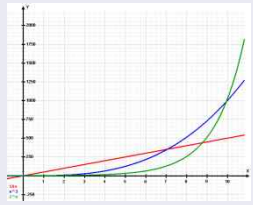
Nonlinear Relationships

- Relationship between the dependent and independent variables is linear
  - This is not always the case.
  - Three common examples are:

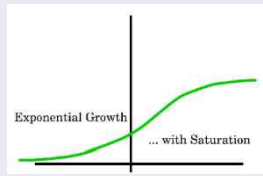
$$\text{exponential}: \quad y = \alpha_1 e^{\beta_1 x}$$



$$\text{power}: \quad y = \alpha_2 x^{\beta_2}$$



$$\text{saturation - growth - rate}: \quad y = \alpha_3 \frac{x}{\beta_3 + x}$$

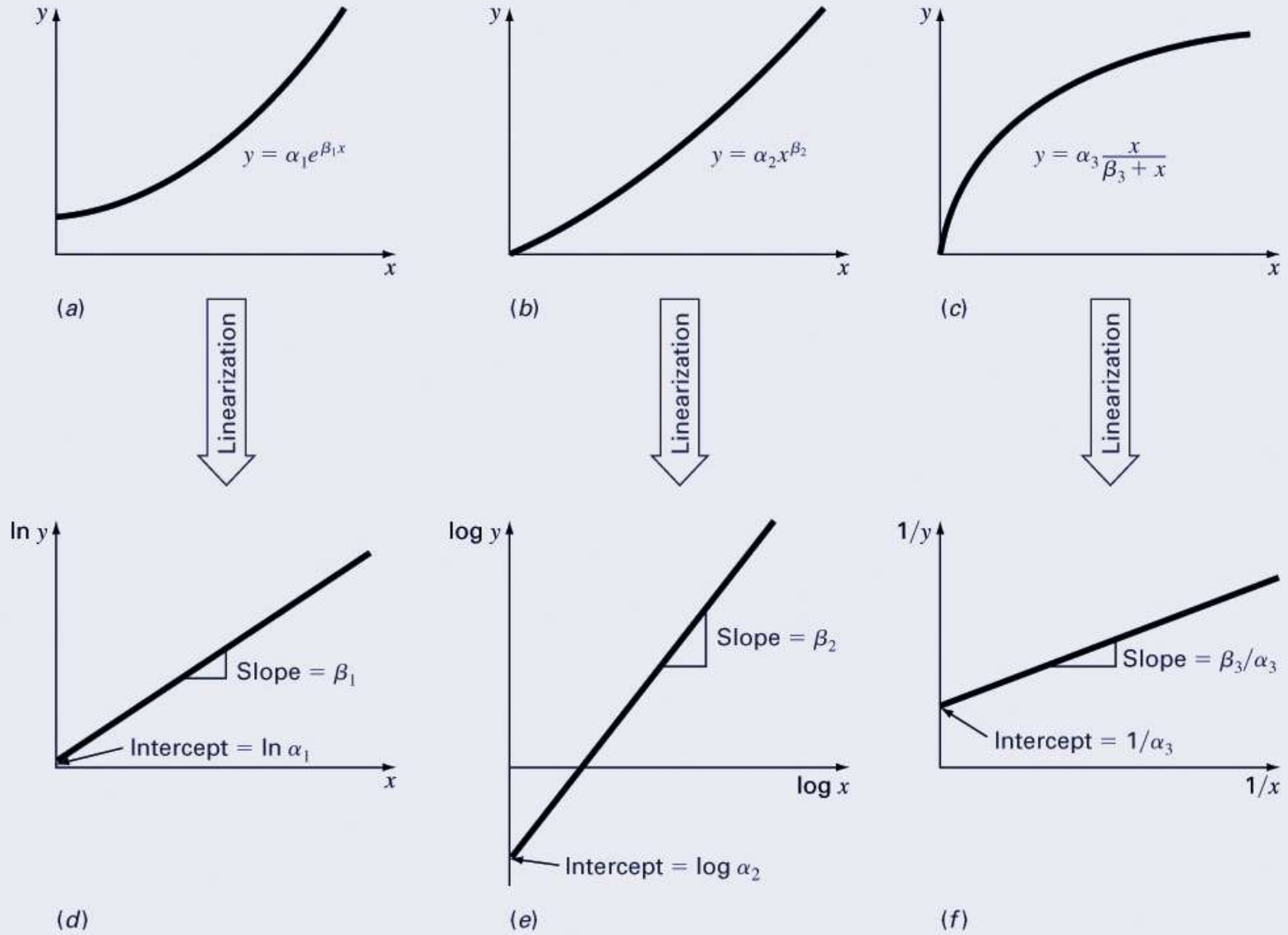


Linearization of Nonlinear Relationships

- For the three common models, this may involve taking logarithms or inversion
  - Issue: unequal error weighting (1, 10, 100,1000,…), $log_{10}\ x$ $(x>0)$

Model	Nonlinear	Linearized
exponential :	$y = \alpha_1 e^{\beta_1 x}$	$\ln y = \ln \alpha_1 + \beta_1 x$
power :	$y = \alpha_2 x^{\beta_2}$	$\log y = \log \alpha_2 + \beta_2 \log x$
saturation - growth - rate :	$y = \alpha_3 \frac{x}{\beta_3 + x}$	$\frac{1}{y} = \frac{1}{\alpha_3} + \frac{\beta_3}{\alpha_3}\frac{1}{x}$

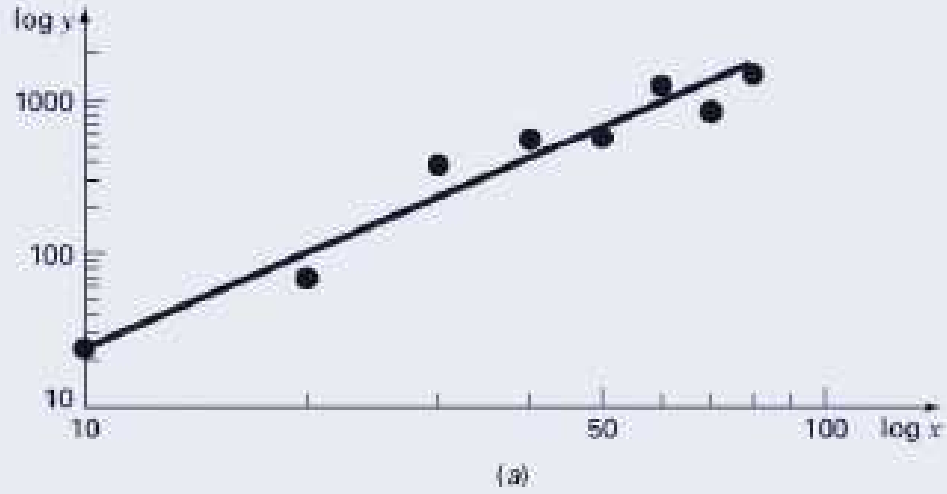
Transformation Examples



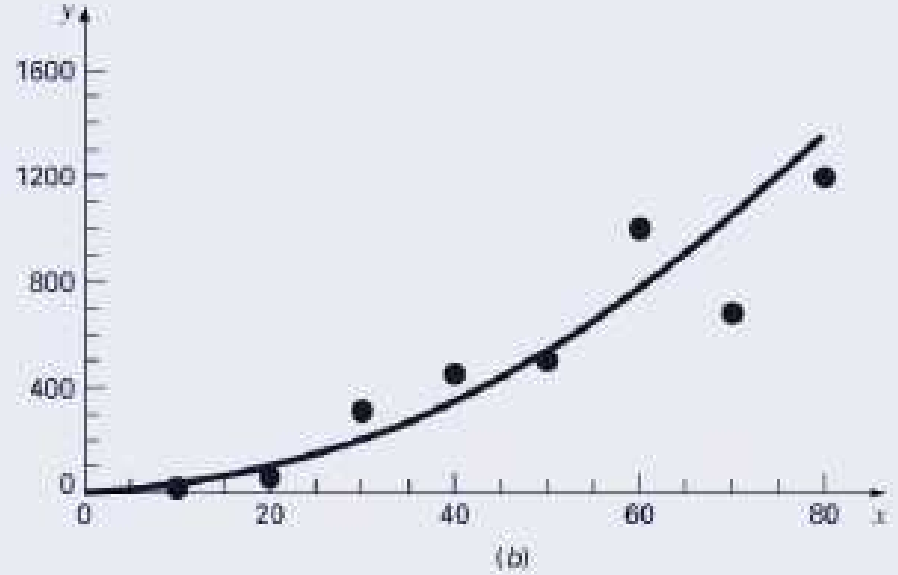
Example

- $y = \alpha x^{\beta}$

α=0.274
β=1.9842



(a)



(b)

MATLAB Functions

- polyfit : fits a least-squares nth order polynomial to data
  - *p* = polyfit(*x*, *y*, *n*)
    - *x*: independent data
    - *y*: dependent data
    - *n*: order of polynomial to fit
    - *p*: coefficients of polynomial : $f(x)=p_1x^n+p_2x^{n-1}+\ldots+p_nx+p_{n+1}$
      - *(cf.)* $f(x)=a_0+a_1x^1+a_2x^2+\ldots+a_nx^n$ (다항식 차수 순서 주의)

- polyval : compute a value using the coefficients
  - *y* = polyval(*p*, *x*)

```
x = (0: 0.1: 2.5)';
y = erf(x);
```

The coefficients in the approximating polynomial of degree 6 are

```
p = polyfit(x,y,6)

p =

 0.0084   -0.0983    0.4217   -0.7435   0.1471    1.1064   0.0004
```

$$.0084x^6 - 0.0983x^5 + 0.4217x^3 + 0.1471x^2 + 1.106x + 0.0004 .$$

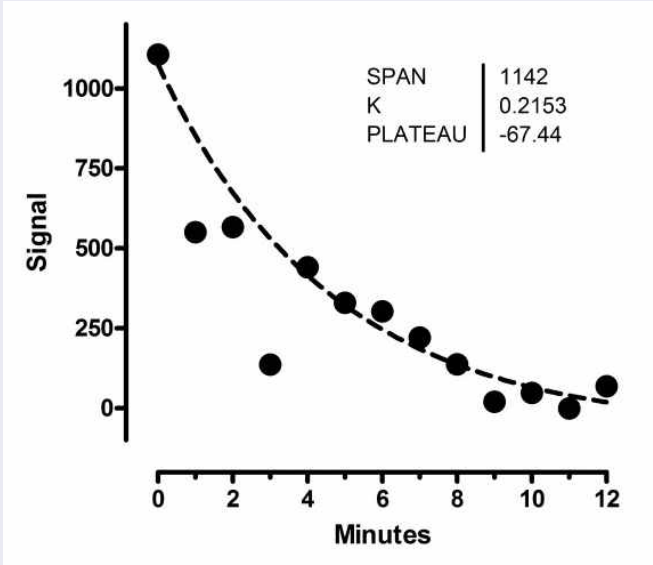
```
f = polyval(p,x);

table = [x y f y-f]

table =

    0          0          0.0004    -0.0004
    0.1000     0.1125     0.1119     0.0006
    0.2000     0.2227     0.2223     0.0004
    0.3000     0.3286     0.3287    -0.0001
    0.4000     0.4284     0.4288    -0.0004
    ...
    2.1000     0.9970     0.9969     0.0001
    2.2000     0.9981     0.9982    -0.0001
```

THE END

Homework : MATLAB (14.5.1, 14.5.2)
Report : 14.5, 14.24

Part 4
Chapter 15

General Linear Least-Squares and Nonlinear Regression



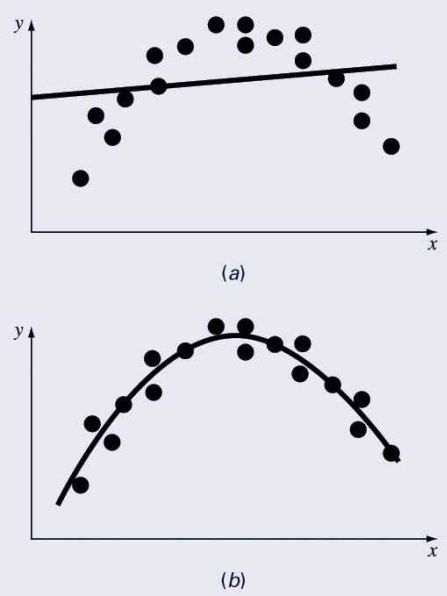
Chapter Objectives

- Polynomial regression.
- Multiple linear regression.
- General linear least-squares model.
- Nonlinear regression with optimization techniques.

Polynomial Regression

- The least-squares procedure can be readily extended to fit data to a higher-order polynomial.
  - the idea is again to minimize the sum of the squares of the estimate residuals

- The figure shows the same data fit with
  a) first order polynomial
  b) second order polynomial



Process and Measures of Fit

- For a second order polynomial, the best fit would mean minimizing:

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

- In general, this would mean minimizing:

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

- The standard error for fitting an m^{th} order polynomial to n data points is:

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

because the m^{th} order polynomial has (m+1) coefficients.

- The coefficient of determination R^2 is still found using:

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

Process of Polynomial Regression

- Minimizing the least square error

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

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

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

$$\begin{aligned} na_0 + (\sum x_i)a_1 + (\sum x_i^2)a_2 &= \sum y_i \\ (\sum x_i)a_0 + (\sum x_i^2)a_1 + (\sum x_i^3)a_2 &= \sum x_i y_i \\ (\sum x_i^2)a_0 + (\sum x_i^3)a_1 + (\sum x_i^4)a_2 &= \sum x_i^2 y_i \end{aligned}$$

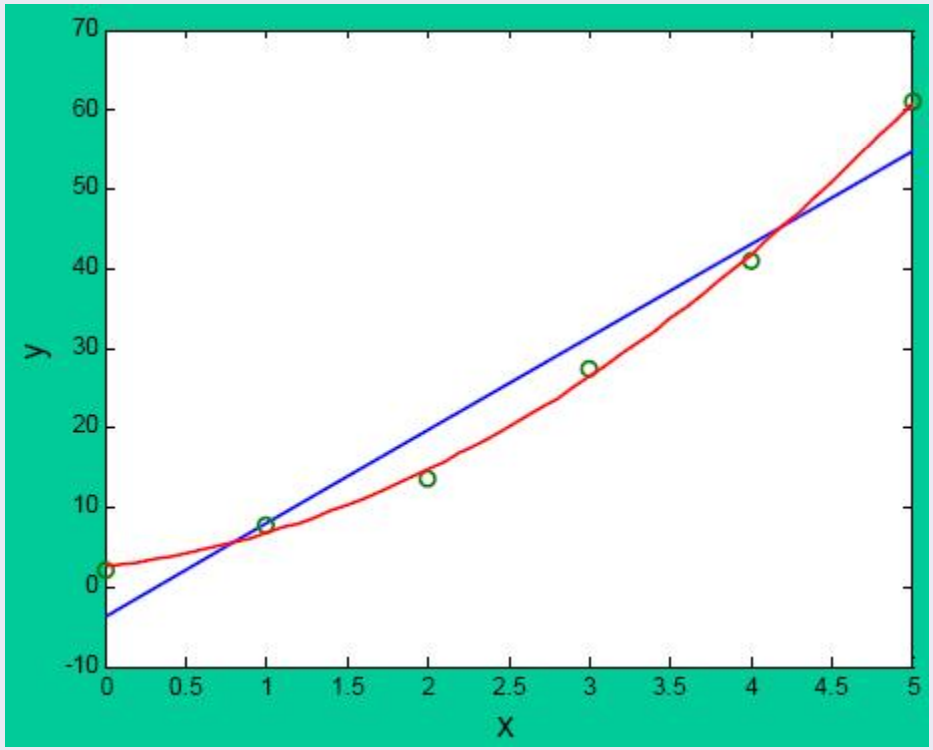
A linear equation of a_0, a_1 and a_2. Solvable.

Example 15.1

x_i	y_i
0	2.1
1	7.7
2	13.6
3	27.2
4	40.9
5	61.1
Σ	152.6

$y = -3.7238 + 11.663x$
$s_{y/x} = 5.1576$
$r^2 = 0.94708$

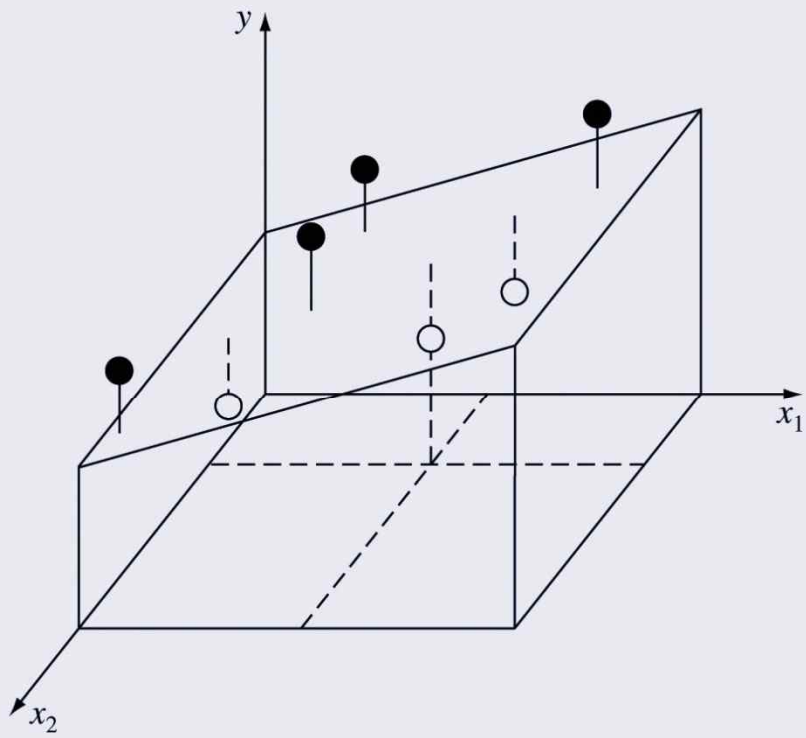
$y = 2.4786 + 2.3593x + 1.8607x^2$
$s_{y/x} = 1.1175$
$r^2 = 0.99851$



Multiple Linear Regression

- Another useful extension of linear regression is the case where *y* is a linear function of two or more independent variables:

$$y = a_0 + a_1 x_1 + a_2 x_2 + \cdots a_m x_m$$



- The best fit is obtained by minimizing the sum of the squares of the estimate residuals:

$$S_r = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} \left(y_i - a_0 - a_1 x_{1,i} - a_2 x_{2,i} - \cdots a_m x_{m,i} \right)^2$$

Multiple Linear Regression

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

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

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

$$\begin{aligned}
n a_0 &+ (\sum x_{1,i}) a_1 &+ (\sum x_{2,i}) a_2 &= \sum y_i \\
(\sum x_{1,i}) a_0 &+ (\sum x_{1,i}^2) a_1 &+ (\sum x_{1,i} x_{2,i}) a_2 &= \sum x_{1,i} y_i \\
(\sum x_{2,i}) a_0 &+ (\sum x_{1,i} x_{2,i}) a_1 &+ (\sum x_{2,i}^2) a_2 &= \sum x_{2,i} y_i
\end{aligned}$$

General Linear Least Squares

- Linear, polynomial, and multiple linear regression all belong to the general linear least-squares model:

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

where $z_0, z_1, \ldots, z_m$ are a set of $(m+1)$ *basis functions* and e is the error of the fit.

- (ex) Polynomial bases
  - The collection of quadratic polynomials with real coefficients has $\{1, t, t^2\}$ as a basis.
  - Every quadratic can be written as $(a \cdot 1 + b \cdot t + c \cdot t^2)$, that is, as a linear combination of the basis functions 1, t, and t^2

Solving General Linear Least Squares Coefficients

- $y = a_0 + a_1 z_1 + a_2 z_2 + \ldots + a_m z_m + e$

- Recast into matrix form

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

Where

$$\{y\} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad \{a\} = \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_m \end{bmatrix}, \quad \{e\} = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix}, \quad [Z] = \begin{bmatrix} z_{10} & z_{11} & \cdots & z_{1m} \\ z_{20} & z_{21} & \cdots & z_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ z_{n0} & z_{n1} & \cdots & z_{nm} \end{bmatrix}$$

Solving General Linear Least Squares Coefficients

- Generally, [*Z*] is not a square matrix, so simple inversion cannot be used to solve for {*a*}

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

Where

$$\{y\} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad \{a\} = \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_m \end{bmatrix}, \quad \{e\} = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{bmatrix}, \quad [Z] = \begin{bmatrix} z_{10} & z_{11} & \cdots & z_{1m} \\ z_{20} & z_{21} & \cdots & z_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ z_{n0} & z_{n1} & \cdots & z_{nm} \end{bmatrix}$$

- The outcome of this minimization yields:

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

(m x n)*(n x m) = m x m

MATLAB Example

- Given *x* and *y* data in columns, solve for the coefficients of the best fit line for $y=a_0+a_1x+a_2x^2$

```
Z = [ones(size(x)) x x.^2]
a = (Z'*Z)\(Z'*y)
```

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

- To calculate measures of fit:

```
St = sum((y-mean(y)).^2)
Sr = sum((y-Z*a).^2)
r2 = 1-Sr/St
syx = sqrt(Sr/(length(x)-length(a)))
```

$$S_t = \sum (y_i - \bar{y})^2 \qquad S_r = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n}\left(y_i - \sum_{j=0}^{m} a_j z_{ji}\right)^2 \qquad r^2 = \frac{S_t - S_r}{S_t} \qquad s_{y/x} = \sqrt{\frac{S_r}{n-(m+1)}}$$

Nonlinear Regression

- As seen in the previous chapter, not all fits are linear equations of coefficients and basis functions.

- One method to handle this is to transform the variables and solve for the best fit of the transformed variables. There are two problems with this method:
  - Not all equations can be transformed easily or at all
  - The best fit line represents the best fit for the transformed variables, not the original variables.

- Another method is to perform nonlinear regression to directly determine the least-squares fit.

Nonlinear Regression in MATLAB

- To perform nonlinear regression in MATLAB,
  - Write a function that returns the sum of the squares of the estimate residuals for a fit and then
  - Use MATLAB's fminsearch function to find the values of the coefficients where a minimum occurs (unconstrained optimization).

Nonlinear Regression in MATLAB Example 15.5

- Given dependent force data *F* for independent velocity data *v*, determine the coefficients for the fit:

$$F = a_0 v^{a_1}$$

- First , write a function (fSSR.m) containing the following:

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

- Then, use fminsearch in the command window to obtain the values of a that minimize fSSR:

```
a = fminsearch(@fSSR, [1, 1], [ ], x, y)
```

where [1, 1] is an initial guess for the [a0, a1] vector, and [ ] is a placeholder for the options

Display : 'off ' displays no output ; 'iter' displays output at each iteration; 'final' displays just the final output ; 'notify' (default) displays output only if the function does not converge.
MaxIter : Maximum number of iterations allowed.
TolX : Termination tolerance on x.

Example 15.5

- Linear regression (Log transform, ref. chap 14)

$F = (0.274)v^{1.9842}$

- Nonlinear regression
  - m file : fSSR.m

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

	V (m/s)	F (N)
i	x_i	y_i
1	10	25
2	20	70
3	30	380
4	40	550
5	50	610
6	60	1220
7	70	830
8	80	1450

```
>> fminsearch (@fSSR, [1,1], [ ], x, y)
```

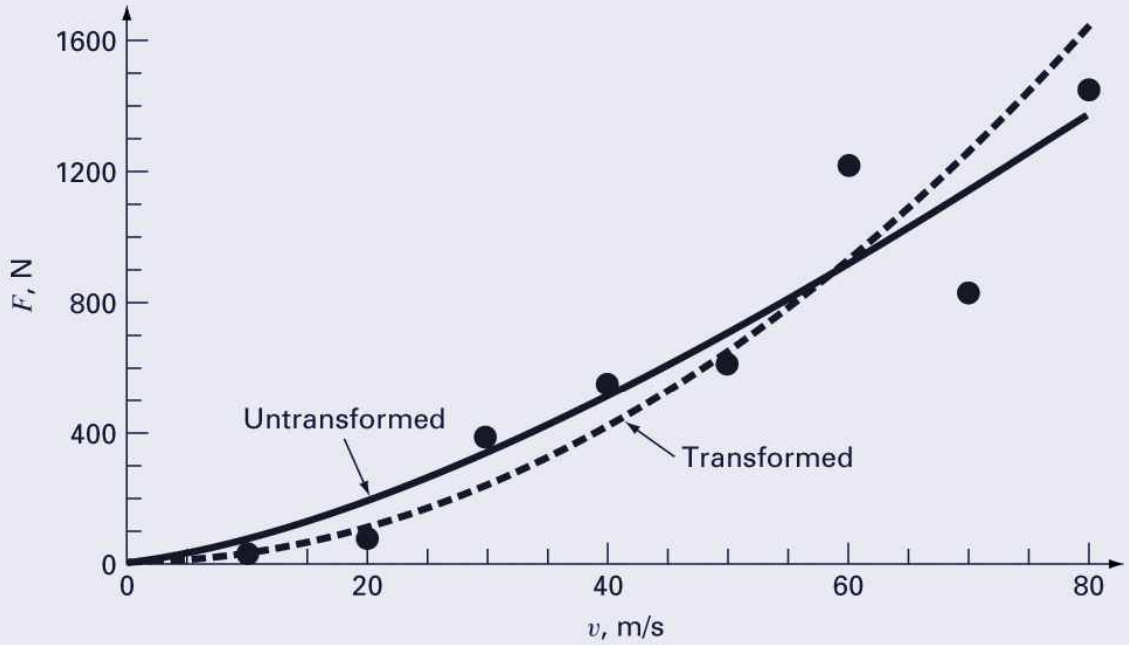
```
ans =  2.5384   1.4359
```

$F = (2.5384)v^{1.4359}$

```
1.0   1.1
1  function f=fssr(a, xm, ym)
2 -    yp=a(1)*xm.^a(2);
3 -    f=sum((ym-yp).^2);
4 -  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
```

Nonlinear Regression Results



- Hard to tell which is better
  - Nonlinear regression : minimize the error of original data
  - Transformed linear regression : minimize the error of transformed data

THE END

Homework : 예제 15.4, 15.5
Report : 15.3, 15.20