by T.B.Co

MAR 2 1 1996

Least Squares Method.

1. **Linear Regression:** Suppose you are given a set of x and y data, and you wish to obtain a linear relationship between y and x given by

$$y = mx + b$$

where m is the slope and b is the intercept. Then a formula for obtaining these parameters that would yield a "least squares" fit (as given in Felder and Rousseau, p. 590) is:

$$\begin{aligned} m &= \frac{n\sum_{i=1}^{n} x_i y_i - (\sum_{i=1}^{n} x_i)(\sum_{i=1}^{n} y_i)}{n\sum_{i=1}^{n} x_i^2 - (\sum_{i=1}^{n} x_i)^2} \\ b &= \frac{(\sum_{i=1}^{n} x_i^2)(\sum_{i=1}^{n} y_i) - (\sum_{i=1}^{n} x_i y_i)(\sum_{i=1}^{n} x_i)}{n\sum_{i=1}^{n} x_i^2 - (\sum_{i=1}^{n} x_i)^2} \end{aligned}$$

where n is the number of data points.

2. **Multivariable Linear Regression:** This time, suppose you are given a set of y, x and w data and you wish to obtain a linear relationship given by

$$y = a_2 x + a_1 w + a_0$$

where a_2, a_1 and a_0 are the parameters which you need to find. A set of formulas similar to the single independent variable case above can be obtained but is very complicated. Instead, a short-hand method is given by the following procedure:

(a) Construct the following matrices:

$$A = \begin{pmatrix} x_1 & w_1 & 1 \\ x_2 & w_2 & 1 \\ x_3 & w_3 & 1 \\ \vdots & \vdots & \vdots \\ x_n & w_n & 1 \end{pmatrix} \qquad B = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{pmatrix}$$

(b) Solve for the parameters as:

$$\begin{pmatrix} a_2 \\ a_1 \\ a_0 \end{pmatrix} = (A^t A)^{-1} A^t B$$

where A^t is the transpose of A. (Note: because matrices are not commutative, you have to be careful of the order in which you do matrix multiplication.) This equation is also known as the "normal equation".

Example: Suppose the following data is given

$y(i)$	$x(i)$	$w(i)$
1.00	0.00	2.00
24.00	1.00	10.00
-3.00	2.00	-2.00
30.00	3.00	10.00
15.50	4.00	3.00
14.75	5.00	1.50
24.00	6.00	4.00
67.00	7.00	20.00
20.75	8.00	0.30
23.00	9.00	0.00
33.50	10.00	3.00

Then we get the following matrices:

$$A = \begin{pmatrix} 0.00 & 2.00 & 1 \\ 1.00 & 10.00 & 1 \\ 2.00 & -2.00 & 1 \\ 3.00 & 10.00 & 1 \\ 4.00 & 3.00 & 1 \\ 5.00 & 1.50 & 1 \\ 6.00 & 4.00 & 1 \\ 7.00 & 20.00 & 1 \\ 8.00 & 0.30 & 1 \\ 9.00 & 0.00 & 1 \\ 10.00 & 3.00 & 1 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 1.00 \\ 24.00 \\ -3.00 \\ 30.00 \\ 15.50 \\ 14.75 \\ 24.00 \\ 67.00 \\ 20.75 \\ 23.00 \\ 33.50 \end{pmatrix}$$

from which we get the following

$$A^tA = \begin{pmatrix} 385.0 & 251.90 & 55.0 \\ 251.9 & 644.34 & 51.8 \\ 55.0 & 51.80 & 11.0 \end{pmatrix} \longrightarrow (A^tA)^{-1} = \begin{pmatrix} 0.0091 & 0.0002 & -0.0463 \\ 0.0002 & 0.0025 & -0.0126 \\ -0.0463 & -0.0126 & 0.3815 \end{pmatrix}$$

$$A^tB = \begin{pmatrix} 1564.8 \\ 2159.4 \\ 250.5 \end{pmatrix}$$

Finally we get

$$(A^tA)^{-1}A^tB = \begin{pmatrix} 3.0 \\ 2.5 \\ -4.0 \end{pmatrix}$$

Thus $a_2 = 3.0$, $a_1 = 2.5$ and $a_0 = -4.0$ and the least squares fit is given by

$$y = 3.0 * x + 2.5 * w - 4.0$$

3. **Least Squares Polynomial Fit** The scheme for the multivariable case can also be used to find the parameters of a polynomial curve fit. For instance, for a second order polynomial case,

$$y = a_2 z^2 + a_1 z + a_0$$

If we let $x = z^2$ and $w = z$, then we get back the previous example, i.e.

$$y = a_2 x + a_1 w + a_0$$

Let us then generalize this to the pth order polynomial. Suppose we are given y-z data (with z as the independent variable). We want to get the parameters a_i, $i = 0$ to p, such that the following polynomial

$$y = a_p z^p + a_{p-1} z^{p-1} + \ \ldots \ + a_2 z^2 + a_1 z + a_0$$

yields the best curve fit to the raw data. All we need to do is to construct the matrices A and B from raw data as before:

$$A = \begin{pmatrix} z_1^p & z_1^{p-1} & \cdots & z_1^2 & z_1 & 1 \\ z_2^p & z_2^{p-1} & \cdots & z_2^2 & z_2 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ z_n^p & z_n^{p-1} & \cdots & z_n^2 & z_n & 1 \end{pmatrix}$$

$$B = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{pmatrix}$$

Then the required parameters are given by the normal equation

$$\begin{pmatrix} a_p \\ a_{p-1} \\ \vdots \\ a_2 \\ a_1 \\ a_0 \end{pmatrix} = (A^t A)^{-1} A^t B$$

Example: Given the following data:

y_i	z_i
13.88	8.728×10^{-3}
15.23	8.237×10^{-2}
17.37	0.2113
18.66	0.2749
20.18	0.4524
19.96	0.4832
20.00	0.6135
19.11	0.7599
18.76	0.8075
18.87	0.8096
18.29	0.8474

Then the matrices A and B are

$$A = \begin{pmatrix} z_1^3 & z_1^2 & z_1 & 1 \\ z_2^3 & z_2^2 & z_2 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ z_n^3 & z_n^2 & z_n & 1 \end{pmatrix} = \begin{pmatrix} 6.649 \times 10^{-7} & 7.618 \times 10^{-5} & 8.728 \times 10^{-3} & 1 \\ 5.588 \times 10^{-4} & 6.784 \times 10^{-3} & 8.237 \times 10^{-2} & 1 \\ 9.437 \times 10^{-3} & 4.466 \times 10^{-2} & 0.2113 & 1 \\ 2.077 \times 10^{-2} & 7.556 \times 10^{-2} & 0.2749 & 1 \\ 9.261 \times 10^{-2} & 0.2047 & 0.4524 & 1 \\ 0.1128 & 0.2334 & 0.4832 & 1 \\ 0.2309 & 0.3764 & 0.6135 & 1 \\ 0.4388 & 0.5774 & 0.7599 & 1 \\ 0.5265 & 0.6520 & 0.8075 & 1 \\ 0.5308 & 0.6555 & 0.8096 & 1 \\ 0.6086 & 0.7182 & 0.8474 & 1 \end{pmatrix}$$

$$B = \begin{pmatrix} 13.88 \\ 15.23 \\ 17.37 \\ 18.66 \\ 20.18 \\ 19.96 \\ 20.00 \\ 19.11 \\ 18.76 \\ 18.87 \\ 18.29 \end{pmatrix}$$

Then the method should yield

$$\begin{pmatrix} a_3 \\ a_2 \\ a_1 \\ a_0 \end{pmatrix} = (A^tA)^{-1}A^tB = \begin{pmatrix} -1.151 \\ -20.10 \\ 23.43 \\ 13.56 \end{pmatrix}$$

or

$$y = -1.151z^3 - 20.10z^2 + 23.43z + 13.56$$

It is necessary to plot the function derived to see how good the fit is. For this example Figure 0.1 shows that the cubic polynomial does give a good fit. Also note that the curve does not actually pass through the data points. Thus we call this process "curve fitting". This is acceptable in most cases because the data could actually contain small level of error.

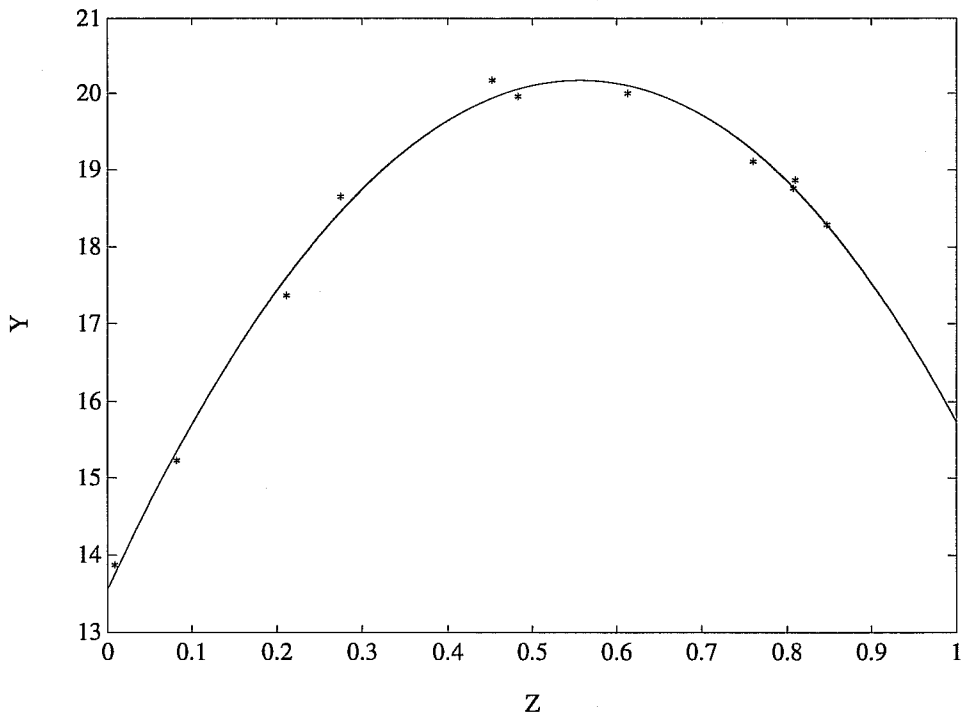

Figure 0.1: A cubic polynomal fit shown together with raw data.

4. Finally, there are some other cases for which the method could be applied to. For instance, in obtaining the parameters of the Antoine Equation, the following manipulations could be done:

$$\begin{aligned} \log_{10}(P) &= A - \frac{B}{T+C} \\ T\log_{10}(P) + C\log_{10}(P) &= AT + AC - B \\ T\log_{10}(P) &= -C\log_{10}(P) + AT + AC - B \end{aligned}$$

Now use the following substitution: $y = T\log_{10}(P)$, $x = \log_{10}(P)$, $w = T$, $a_2 = -C$, $a_1 = A$ and $a_0 = AC - B$, we should get back the following

$$y = a_2 x + a_1 w + a_0$$

and procede as we did before. After the coefficients a_2, a_1 and a_0 are obtained, the actual parameters can be recovered, i.e.

$$\begin{aligned} C &= -a_2 \\ A &= a_1 \\ B &= AC - a_0 = -a_2 a_1 - a_0 \end{aligned}$$

5. **Notes on Lotus 123.** Certain matrix operations can be done on Lotus 123. Matrix multiplication and inversion can be found in the menu item Data and submenu item Matrix. Matrix transposes can be done via the menu item Range and submenu item Transpose. Since matrix transposition exchanges the rows and columns which could contain formulas with cell addresses, it is best to build the matrices by using Value under the Range submenu (instead of copy) before any other matrix operation is done.