Computational Mathematics/Information Technology

Dr Oliver Kerr

2009–10

Goodness of fit

Last week we saw that if the **statistics** option is set to be true LINEST returns statistical data about how well the data fits the estimated straight line. In particular we started to consider

- the coefficient of determination
- standard error of the y-estimate

We looked last week at the standard error and saw its problems: If we are told the standard error is 3.542, unless we know more about the problem this number tells us nothing!

We will now look at the **the Coefficient of Determination**.

Coefficient of Determination — r^2

When fitting $Y = mx + c$ to the data set $\{(x_0, y_0),\ (x_1, y_1) \ldots (x_n, y_n)\}$ the **coefficient of determination** is defined as

$$r^2 = \frac{\displaystyle\sum_{i=0}^{n} (\overline{y} - Y_i)^2}{\displaystyle\sum_{i=0}^{n} (\overline{y} - y_i)^2}$$

The coefficient compares the spread of the fitted Y_i values about $\overline{y}$, the mean of the data, with the spread of the y-data values y_i about the data mean $\overline{y}$.

- If the data lies on the best fit line then $y_i = Y_i$ and so $r^2 = 1$. This is the largest possible value of r^2 and corresponds to perfect agreement between the fit and the data.

- If the data lies on the best fit line then $y_i = Y_i$ and so $r^2 = 1$. This is the largest possible value of r^2 and corresponds to perfect agreement between the fit and the data.
- As the fit gets worse then r^2 decreases, with the worst case being indicated by $r^2 = 0$ (?? must have $Y_i = \overline{y}$ for all i).

- Recalling we can write our estimates from our line of best fit as
$$Y_i = \frac{\overline{xy} - \overline{x}\,\overline{y}}{\overline{x^2} - \overline{x}^2}\,(x_i - \overline{x}) + \overline{y}$$
it is possible to write r^2 as
$$r^2 = \frac{\left(\displaystyle\sum_{i=0}^{n}(x_i - \overline{x})(y_i - \overline{y})\right)^2}{\displaystyle\sum_{i=0}^{n}(x_i - \overline{x})^2 \sum_{i=0}^{n}(y_i - \overline{y})^2}$$
This takes a bit of working (which is not in Dr Bowtell's notes!)

- Statisticians also use the use the **Pearson product-moment correlation coefficient** (or the **coefficient of correlation**). This is defined to be

$$r = \frac{\displaystyle\sum_{i=0}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\displaystyle\sum_{i=0}^{n}(x_i - \overline{x})^2 \sum_{i=0}^{n}(y_i - \overline{y})^2}}$$

This is the square root of the coefficient of determination, but can take either a positive or negative sign, and lies in the range $-1 \leq r \leq 1$.

- A value of r close to $+1$ represents a good linear relationship between the x–y data values such that as x increases the y data also tends to increase.
- A value of r close to -1 represents a good linear relationship between the x–y data values such that as x increases the y values tend to decrease.

Excel's Statistics

When using LINEST in order to display the c and m values together with all the fitting statistics one needs to enter

=LINEST(*y-values*, *x-values*, TRUE,**TRUE**).

When fitting $y = m_1x_1 + m_2x_2 + \cdots + m_kx_k + c$ to $(n+1)$ data points, each of the form $(x_1,\ x_2, \ldots x_k,\ y)$, LINEST will return an array with 5 rows and $k+1$ columns (which you have to highlight before you do Ctrl-Alt-Return):

m_k	m_{k-1}	$\ldots$	m_1	c
se_k	se_{k-1}	$\ldots$	se_1	se_c
r^2	SE_y			
F	df			
S_{reg}	S			

m_k	m_{k-1}	$\ldots$	m_1	c
se_k	se_{k-1}	$\ldots$	se_1	se_c
r^2	SE_y			
F	df			
S_{reg}	S			

The first row contains the c and m fitting parameters which we have been seen before.

m_k	m_{k-1}	$\dots$	m_1	c
se_k	se_{k-1}	$\dots$	se_1	se_c
r^2	SE_y			
F	df			
S_{reg}	S			

The second row consists of standard errors for each of the parameters — these can be ignored for now.

m_k	m_{k-1}	$\dots$	m_1	c
se_k	se_{k-1}	$\dots$	se_1	se_c
r^2	SE_y			
F	df			
S_{reg}	S			

The third row contains the two important parameters — the coefficient of determination r^2 and the standard error of the y estimate SE_y.

m_k	m_{k-1}	$\dots$	m_1	c
se_k	se_{k-1}	$\dots$	se_1	se_c
r^2	SE_y			
F	df			
S_{reg}	S			

The fourth row contains the F statistic for the purpose of hypothesis testing, and df, the number of degrees of freedom for the system.

m_k	m_{k-1}	$\dots$	m_1	c
se_k	se_{k-1}	$\dots$	se_1	se_c
r^2	SE_y			
F	df			
S_{reg}	S			

Row five contains S_{reg} — the squares of the differences between the fitted y-values and the mean of the y-data, and S — the sum of the squares of the differences between the fitted y-values and the data y-values.

Non-Linear Least Squares Fitting

So far we have used least squares fitting to find a linear function of the independent variables — we found the values of the parameters $\{m_1, m_2, \ldots, m_k, c\}$ so that $y = m_1 x_1 + \cdots + m_k x_k + c$ best fits the data according to the criteria of least squares.

Not all data sets lie on straight lines:

- We may know something about the underlying process that gives rise to the data.
- It may be clear from the data that there isn't a linear relationship.

Polynomial fitting

Suppose we have the data set $\{(x_0, y_0),\ (x_1, y_1),\ \ldots (x_n, y_n)\}$ and we want to fit the polynomial

$$y = c + m_1 x + m_2 x^2 + \cdots + m_k x^k$$

What do we do?

Polynomial fitting

Suppose we have the data set $\{(x_0, y_0),\ (x_1, y_1),\ \ldots (x_n, y_n)\}$ and we want to fit the polynomial

$$y = c + m_1 x + m_2 x^2 + \cdots + m_k x^k$$

What do we do?

As before we can seek to find c, m_1, ..., m_k to minimise

$$S = \sum_{i=0}^{n} \left(y_i - (c + m_1 x + m_2 x^2 + \cdots + m_k x^k) \right)^2$$

This is equivalent to the problem we had before where instead of x, x^2, ..., x^k we had x_1, x_2, ..., x_k. So we can solve it in the same way using LINEST.

For example, suppose it is decided to fit a cubic to 10 values (so $k = 3$ and $n = 9$). We would create four columns of values, one for $x_1 = x$, one for $x_2 = x^2$, one for $x_3 = x^3$ and one for y.

- In A1:A10 enter the given x values, $\{x_0,\ x_1,\ \ldots, x_9\}$
- In B1 enter $=$A1$\wedge$2, and copy down to B10. (*column B now contains the values* x^2)
- In C1 enter $=$A1$\wedge$3, and copy down to C10 (*column C now contains the values* x^3)
- In D1:D10 enter the given y values, $\{y_0,\ y_1, \ldots,\ y_9\}$.
- Highlight E1:H1, Enter **=LINEST(D1:D10,A1:C10)**, use **Ctrl-Shift-Enter** to give the values of m_3, m_2, m_1 and c in E1 to H1 respectively.

Over to Excel...

Fitting power, exponential and logarithmic functions

This approach can be used for other more complex examples where we can transform our approximating function into one where the coefficients, or a function of them, appear in a linear form.

Consider

Given the data set $\{(x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n)\}$

Fit one of:

power curve:	$y = bx^m$
exponential curve:	$y = be^{mx}$
logarithmic curve:	$y = m \ln x + c$

Power curve

If we want to fit a curve of the form

$$y = bx^m$$

to the data, we take logarithms:

$$\ln y = m \ln x + \ln b$$

This is now of the form

$$Y = mX + c$$

where $Y = \ln y$, $X = \ln x$ and $c = \ln b$.

So we can use LINEST by setting up a column with $\ln x_i$ and one with $\ln y_i$ and fit the data.

Warning: You are not finding b and m to minimize

$$S = \sum_{i=0}^{n} (y_i - b x_i^m)$$

Instead you are finding b and m to minimize

$$S = \sum_{i=0}^{n} ((\ln y_i) - (m \ln x_i + \ln b))$$

These are not the same. This applies also to the following example.

Exponential curve

If we want to fit a curve of the form

$$y = be^{mx}$$

to our data, we take logarithms:

$$\ln y = mx + \ln b$$

This is now of the form

$$Y = mX + c$$

where $Y = \ln y$, $X = x$ and $c = \ln b$.

So we can use LINEST by setting up a column with x_i and one with $\ln y_i$ and fit the data.

Logarithmic curve

If we want to fit a curve of the form

$$y = m \ln x + c$$

This is already in the form

$$Y = mX + c$$

where $Y = y$, $X = \ln x$.

So we can use LINEST by setting up a column with $\ln x_i$ and one with y_i and fit the data.

General fitting

This process can readily be applied to fitting any function of the form

$$y = m_1 f_1(x_1, \ldots, x_k) + m_2 f_2(x_1, \ldots, x_k) + \cdots + m_n f_n(x_1, \ldots, x_k) + c$$

Over to Excel...

General fitting

This process can readily be applied to fitting any function of the form

$$y = m_1 f_1(x_1, \ldots, x_k) + m_2 f_2(x_1, \ldots, x_k) + \cdots + m_n f_n(x_1, \ldots, x_k) + c$$

Over to Excel...

However, other functions cannot be transformed into this form, for example

$$y = axe^{bx}\cos(cx + d)$$

What can we do here?

Suppose we want to fit the curve

$$y = axe^{bx}\cos(cx + d)$$

to our data points. We may still want to minimize

$$S(a, b, c, d) = \sum_{i=0}^{n} \left(y_i - ax_i e^{bx_i}\cos(cx_i + d)\right)^2$$

There may be no straightforward analytical approach, but it can be solved numerically. Not a trivial task to do yourself, but Excel can help!