General Linear Least Squares

James R. Graham
10/12/2009

Equations of condition

Suppose we consider a model to describe a data set (x_i, y_i) where $y = y(x)$ and the function can be written in the form

$$y_i = \alpha_1\beta_1(x_i) + \alpha_2\beta_2(x_i) + \cdots \alpha_n\beta_n(x_i)\,, \tag{1}$$

where β is some known function of the independent variable x, and α_i are constants. How do we find the constants α_i given the data?

If the problem can be expressed in this manner it is a linear one, because the dependent variable is a linear combination of known functions of the independent variable. If we write

$$B_{ij} = \beta_j(x_i)\,, \tag{2}$$

and

$$\mathbf{a} = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{pmatrix}, \tag{3}$$

then we can write the problem in matrix form as

$$\mathbf{y} = \mathbf{Ba}\,. \tag{4}$$

The equations represented by Eq. (4) are known as the ***equations of condition***.

Least squares

The equality in Eq. (4) holds only for data with no measurement errors. We are interested in finding **a** when there are errors in y and therefore Eq. (4) is not solved exactly: all we can hope for is a solution that in some sense is optimal. The method of least squares yields such a solution.

We can write a compact expression for the sum of the squares of the residuals,

$$\chi^2 = \|\mathbf{y} - \mathbf{Ba}\|_2^2 \,, \tag{5}$$

where the notation $\|\ldots\|_2$ is used to denote the Euclidian vector norm[1],

$$\|\mathbf{x}\|_2^2 = \mathbf{x}^T\mathbf{x} = \sum_i x_i^2 \;, \tag{6}$$

and the superscript T denotes the transpose. Expanding Eq. (5) we find

$$\begin{aligned} \chi^2 &= (\mathbf{y} - \mathbf{Ba})^T (\mathbf{y} - \mathbf{Ba}) \\ &= \mathbf{y}^T\mathbf{y} - 2\mathbf{y}^T\mathbf{Ba} + \mathbf{a}^T\mathbf{B}^T\mathbf{Ba}. \end{aligned} \tag{7}$$

We want to minimize this expression as a function of **a**, so that the first derivatives with respect to **a** are zero

$$\frac{\partial \chi^2}{\partial \mathbf{a}} = -2\mathbf{B}^T\mathbf{y} + 2\mathbf{B}^T\mathbf{Ba} = 0 \,, \tag{8}$$

or

$$\mathbf{B}^T\mathbf{Ba} = \mathbf{B}^T\mathbf{y} \,. \tag{9}$$

Thus, the unknown vector **a** is found by multiplying each side by the inverse matrix $(\mathbf{B}^T\mathbf{B})^{-1}$

$$\begin{aligned} (\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T\mathbf{Ba} &= (\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T\mathbf{y} \\ \mathbf{a} &= (\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T\mathbf{y}. \end{aligned} \tag{10}$$

The quantity $(\mathbf{B}^T\mathbf{B})^{-1}$ is known as the generalized or Moore-Penrose pseudo-inverse of **B**. Sophisticated versions of general least squares methods use ***singular value decomposition*** to compute the inverse of $\mathbf{B}^T\mathbf{B}$.

---

[1] In two or three dimensions the Euclidian vector norm is just the magnitude (length) of the vector (think Pythagoras' theorem).

An simple example: uniform acceleration from rest

Suppose we have a set of data described by a parabolic relation

$$x = \frac{1}{2} g t^2,$$

e.g., the distance traveled by a body dropped from rest. How do we find the value of g? Some data are shown in Figure 1.

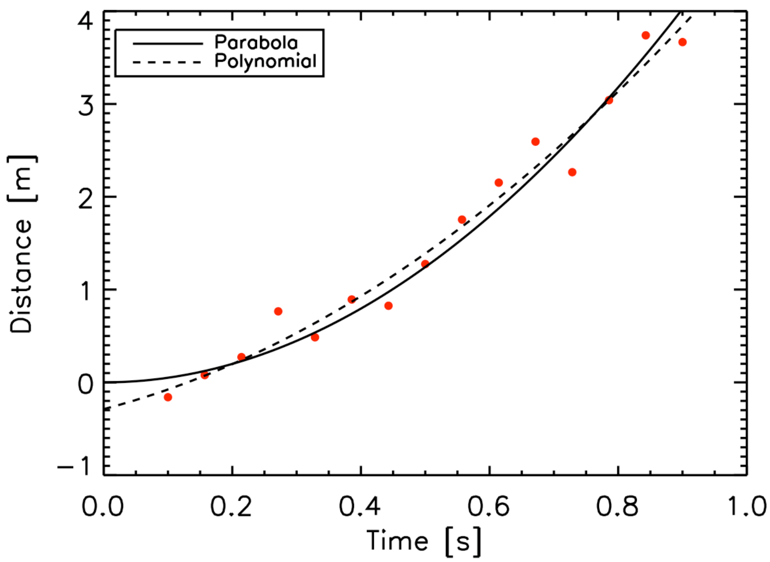

Figure 1: Measurement of the position of a body falling from rest under gravity with g = 9.81 m s^{-2}. The dotted line shows the fit that you get if you fit a general quadratic.

If the data vectors are `t` and `x`, in IDL the solution is implemented as follows:

```
b = transpose([0.5*t^2])
y = transpose(x)
psi = invert( transpose(b) ## b)
ans = psi ## transpose( b) ## y
```

Note that the `transpose` function in the first two steps is to convert the row vectors into column vectors. In IDL the matrix multiplication operator is `##`.

The conventional (but wrong) approach would be to fit a second order polynomial to the data:

```
res = poly_fit(t,x,2)
```

In the example in Figure 1, the parabolic fit gives $g = 9.93$ m s^{-2}, whereas the polynomial fit implies that the initial position is -0.29 m, the initial velocity is 1.81 m s^{-1} and the acceleration is 6.18 m s^{-2}. Polynomial fitting fails to take account of our knowledge that the initial position and velocity are zero, and as a consequence gives an inaccurate value for the acceleration.

Example: Circular motion

Now suppose our task is to determine the radius of a wheel by measuring the x-coordinate of a point on the circumference as the wheel rotates at a known frequency ω. The position of that point is given by

$$x = x_0 + R\sin(\omega t).$$

From measurements of (t, x) we want to find x_0 and R. For this example, the relevant fragment of IDL code is

```
b = transpose([[replicate(1.,npts)],[sin(omega*t)]])
y = transpose(x)
psi = invert( transpose(b) ## b)
ans = psi ## transpose( b) ## y
```

An example of such a fit is shown in Figure 2.

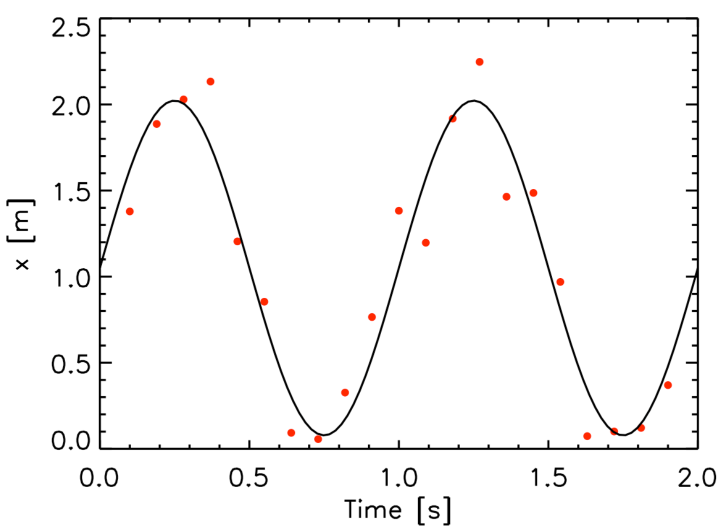

Figure 2: Time series of measurement of a point on the circumference of a wheel rotating at known angular frequency ω. A linear least squares fit to $x = x_0 + R\sin(\omega t)$ yields the radius and the x-coordinate of the point of rotation.

Note the limitation of this method—we cannot determine ω from the data; we have to know the rotation rate. Problems where unknowns enter other than in linear combinations fall into the category of ***non-linear least squares***. There are no closed-form solutions to non-linear problems: they are solved using iterative methods that require an initial guess for the model parameters.

At first sight some problems appear non-linear, e.g., the case of the rotating wheel when the phase, ϕ, is unknown

$$x = x_0 + R\sin(\omega t + \phi)\,.$$

However, by use of trigonometric identities we can write

$$x = x_0 + R\cos(\phi)\sin(\omega t) + R\sin(\phi)\cos(\omega t)\,,$$

which is a linear problem.