Subject: Re: regress question
Posted by Wout De Nolf on Mon, 01 Dec 2008 08:38:36 GMT
View Forum Message <> Reply to Message

On Sun, 30 Nov 2008 23:42:38 -0800 (PST), James McCreight
<mccreigh@gmail.com> wrote:

> I have some vague recollection of doing this once within an IDL
> function. A quick look turned up this, looks promising and like
> something i've seen before:
>
> Curvefit(X, Y, Weights, A [, Sigma] [, CHISQ=variable] [, /DOUBLE] [,
> FITA=vector] [, FUNCTION_NAME=string] [, ITER=variable] [,
> ITMAX=value] [, /NODERIVATIVE] [, STATUS={0 | 1 | 2}] [, TOL=value] [,
> YERROR=variable])
> A
> A vector with as many elements as the number of terms in the user-
> supplied function, containing the initial estimate for each parameter.
> On return, the vector A contains the fitted model parameters.
>
> FITA
> Set this keyword to a vector, with as many elements as A, which
> contains a zero for each fixed parameter, and a non-zero value for
> elements of A to fit. If not supplied, all parameters are taken to be
> non-fixed.

Why using a non-linear least squares fitting algorithm for a linear
problem? Fixing parameters is not all that difficult using the linear
algorithms (i.e. orthogonal decomposition methods like SVD), although
you have to do it yourself.

Suppose y=a.x1+b.x2+c then you find the least squares solution by (X1
and X2 column vectors)
SVDC, [X1,X2,replicate(1,1,n_elements(X1))], W, U, V
result=SVSOL(U, W, V, Y) ; gives LSSol. [a,b,c]

Suppose I want to fix b=3 then you would do this
SVDC, [X1,replicate(1,1,n_elements(X1))], W, U, V
result=SVSOL(U, W, V, Y-3*X2); gives LSSol. [a,c]