

Re: integration + Levenberg–Marquardt (Numerical Recipes)

Source: <http://sci.tech–archive.net/Archive/sci.math.num–analysis/2008–04/msg00039.html>

- *From:* spellucci@xx (Peter Spellucci)
 - *Date:* Wed, 2 Apr 2008 19:52:26 +0200 (CEST)
-

In article <d256ff43–4f70–49b8–8059–991633bf0353@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx>, Jamie <Jamie.McQuay@xxxxxxxxxxxxxxxx> writes:

Hello,

I am having problems getting my fitting function derivatives correct for the Levenburg–Marquardt fitting routine.

I have a differential in the following format:
$$dAB/dt = ka \cdot A(Bo - AB) - kd \cdot AB$$

additional:
A is constant
I'm fitting ka,Bo and kd

I am getting the integrated format of the formula via numerical integration (this is a test equation, they will get more difficult).

The integration is correct when I test the resulting data but I don't know how to properly get parameter derivatives for the fitting function.

How do I generate the parameter derivatives (dyda in the LM fitting function) without the integrated form of the formula (i only have it numerically).

Thank you for any hints.

Jamie

in your example, you have the time dependent function AB, which also depends on the real parameters ka, Bo and kd which you want to compute (from measured instances of AB, I guess).
one can prove, that AB depends differentiable on these parameters, and that the derivatives, which of course also depend on t, obey

Re: integration + Levenberg–Marquardt (Numerical Recipes)

differential equations. for easier notation I name these

$$g1(t) = d(AB)/d(ka) (t)$$

$$g2(t) = d(AB)/d(B0) (t)$$

$$g3(t) = d(AB)/d(kb) (t)$$

the differential equations for the $g1, g2, g3$ are obtained by differentiating the given differential equation with respect to the parameters using the chain rule and keeping in mind that AB also is a function of them:

$$"d(AB)/dt = ka*A(B0-AB) - kd*AB" \Rightarrow$$

$$dg1/dt = A*(B0-AB) - kd*g1 + ka*A*(-g1)$$

$$dg2/dt = ka*A(1-g2) - kd*g2$$

$$dg3/dt = ka*A*(-g3) - AB - kd*g3$$

(hope I got it right)

you add these differential equations to your original one (observe, that also AB itself enters) and integrate then as one system with your integrator using the output at the grid points (t_i given, $AB_{measured_i}$) then you can set up the residual and the Jacobian which is required by Levenberg Marquardt. This is much better than forcing the Levenberg Marquardt code to do the derivative using its implemented finite difference formula: remember that in the integrator, say a Runge Kutta code, you will specify a "tolerance". this tolerance will yield an internally used grid, which will, as a rule, be much finer than the grid of your measured points. then, in finite differences, you might have a too large change of the parameters which might change the grid. then the output will be not a smooth function of the parameters and the finite difference will yield nonsense. or, contrary, the discretization errors in the integrator will be much larger than the change in the solution due to a small change in the parameters, and again the finite difference formula will yield nonsense. with the above approach, you will get a useful jacobian

hth
peter

.