
Subject: Fitting several Gaussians
Posted by [gotwols](#) on Mon, 19 Sep 1994 23:07:02 GMT
[View Forum Message](#) <> [Reply to Message](#)

I am looking for a fitting routine that will do a nonlinear least squares fit of the sum of N Gaussian functions to measured data. I am aware of the routine in the IDL Users library that fits a single Gaussian plus a polynomial, but in my case I want to fit 3 Gaussian functions (or even better yet, N). If anyone knows of the whereabouts of such a function I would appreciate hearing about it, other wise I will have to write it myself.

Thanks in advance, bruce

--

Bruce L. Gotwols
Johns Hopkins University, Applied Physics Lab., Laurel MD 20723
Internet: gotwols@tesla.jhuapl.edu (128.244.147.15)
SPAN: APLSP::STR::GOTWOLS

Subject: Re: Fitting several Gaussians
Posted by [agrap](#) on Sat, 24 Sep 1994 00:00:58 GMT
[View Forum Message](#) <> [Reply to Message](#)

gotwols@strdev.jhuapl.edu (Bruce Gotwols) writes:

> I am looking for a fitting routine that will do a nonlinear least
> squares fit of the sum of N Gaussian functions to measured data. I am aware
> of the routine in the IDL Users library that fits a single Gaussian plus a
> polynomial, but in my case I want to fit 3 Gaussian functions (or even
> better yet, N). If anyone knows of the whereabouts of such a function I would
> appreciate hearing about it, other wise I will have to write it myself.
> Thanks in advance, bruce
> --
> Bruce L. Gotwols
> Johns Hopkins University, Applied Physics Lab., Laurel MD 20723
> Internet: gotwols@tesla.jhuapl.edu (128.244.147.15)
> SPAN: APLSP::STR::GOTWOLS

You can use the same routine CURVEFIT in the IDL library. What will be different will be your FUNCT.PRO function. It should be straight-forward, maybe just a few lines of code.

In the IDL library, FUNCT.PRO has the Gaussian function defined as
(I've left off the polynomial):

$$F = A(0)*EXP(-Z^2/2)$$

What you could do instead is pass in your "A" estimate array defining the 1st estimates for your number of Gaussians:

PGauss = N_ELEMENTS(A) -1

Then, initialize the argument, using the N from the length of your indep x array:

N=N_ELEMENTS(X)
F = DBLARR(N) + A(0)

Then create your F, term by term:

FOR K=1, PGauss DO F = F + A(K) * EXP(-Z^2/2)

Another way to do this problem is make it linear:
take the log of it, and do a linear least-squares fit. (but
if you have polynomial terms, then this approach won't work.

(If someone else has answered this, sorry my response is late.
I'm experiencing a delay in the newsfeed at my Internet
provider site.)

Amara

Amara Graps
Computational Physicist
Intergalactic Reality Boycott the WireTap Chip!
agraps@netcom.com
