Subject: mpfit: what's different
Posted by Wox on Wed, 06 Feb 2008 16:35:52 GMT
View Forum Message <> Reply to Message

Hi everyone,

Is there any documentation on the theory behind Craig Markwardt's mpfit? What's different from the Levenberg-Marquardt algorithm as given in "Numerical Recipes in C" (used in IDL's curvefit)?

Is it a difference in choosing the damping factor in the modified curvature matrix or is there more to it?

Thanks,

Wout

Subject: Re: mpfit: what's different
Posted by Craig Markwardt on Mon, 11 Feb 2008 13:39:28 GMT
View Forum Message <> Reply to Message

Wox <nomail@hotmail.com> writes:

> Hi everyone,
>
> Is there any documentation on the theory behind Craig Markwardt's
> mpfit? What's different from the Levenberg-Marquardt algorithm as
> given in "Numerical Recipes in C" (used in IDL's curvefit)?
>

Hi, I'm back!

First off, I should say that MPFIT is based on MINPACK (by Jose More' et al), so I don't take any of the credit for the underlying algorithm.

Basically, Numerical Recipes is a quick and dirty implementation of Levenberg Marquardt, so what you get are quick and dirty results. Some of it boils down to how the normal equations are determined (MPFIT doesn't; it uses direct QR factorization of the Jacobian, which improves numerical precision); how the LM parameter and trust region are determined (Numerical Recipes doesn't really have a concept of "trust region"); and the convergence criteria.

Also, as David points out, MPFIT has more features than CURVEFIT, such as the abillity to fix or vary certain parameters, the ability to set simple parameter boundaries, to tie parameters together, and some

advanced options for keeping track of the fit status. The MPFIT* library also has a few different flavors of the main routine, to make things easy in certain cases (1d fitting, 2d fitting, using an expression insted of a function, and so on).

> Is it a difference in choosing the damping factor in the modified
> curvature matrix or is there more to it?

I'm not sure what this means. I encourage you to consult the primary references in the mpfit.pro documentation section.

Good luck, and happy fitting,
Craig Markwardt

Subject: Re: mpfit: what's different
Posted by Wox on Mon, 11 Feb 2008 17:13:26 GMT
View Forum Message <> Reply to Message

On 11 Feb 2008 08:39:28 -0500, Craig Markwardt
<craigmnet@REMOVEcow.physics.wisc.edu> wrote:

<snip>
> I'm not sure what this means. I encourage you to consult the primary
> references in the mpfit.pro documentation section.

Thanks, I will.

One question: Is there a way to make sure that the sum of certain parameters stays below 1? I was looking at the tied and limit possibilities, but I don't thing they can do this...

Subject: Re: mpfit: what's different
Posted by Craig Markwardt on Tue, 12 Feb 2008 10:19:49 GMT
View Forum Message <> Reply to Message

Wox <nomail@hotmail.com> writes:

> On 11 Feb 2008 08:39:28 -0500, Craig Markwardt
> <craigmnet@REMOVEcow.physics.wisc.edu> wrote:
>
> <snip>
>> I'm not sure what this means. I encourage you to consult the primary
>> references in the mpfit.pro documentation section.
>
> Thanks, I will.

>
> One question: Is there a way to make sure that the sum of certain
> parameters stays below 1? I was looking at the tied and limit
> possibilities, but I don't thing they can do this...

Not really. MPFIT only supports simple boundary constraints on individual parameters. If you can re-write your problem so that a new parameter is the sum of original parameters, then you can do it.

Craig

Subject: Re: mpfit: what's different
Posted by Wox on Tue, 12 Feb 2008 17:40:44 GMT
View Forum Message <> Reply to Message

Btw, can it be that mpcurvefit expects a different analytical derivative (dResid/dP) than curvefit (dFunc/dP)? I think this should be added in mpcurvefit_eval:

```
;; Compute the deviates, applying the weights
result = (y-f)*wts
if n_params() GT 1 then $
dp*=rebin(wts,n_elements(x),n_elements(p),/sample)
```

Actually, it should be
dp*=rebin(-wts,n_elements(x),n_elements(p),/sample)
but since mpfit_fdjac2 flips the sign (why?), we have to do it here to correct for that.
