Subject: Re: IDL fitting of piecewise continuous function
Posted by Jeremy Bailin on Tue, 17 May 2011 19:21:49 GMT
View Forum Message <> Reply to Message

Craigh Markwardt's MPFIT (and its varients) is most people's preferred fitting routine:

http://www.physics.wisc.edu/~craigm/idl/fitting.html

You define your own function to feed in, so it can have whatever behaviour you want. To ensure continuity, I would recommend re-parametrizing the functional form so that it is continuous by definition. In this, case, A and B aren't actually independent, so calculate B in terms of A by setting them equal at x=x_m, which gives
B = A x_m^(3/2)

So you're actually fitting:

f(x) = A x_m^(1/2) x^(1/2) x < x_m
  A x_m^(3/2) x^(-1) x > x_m

-Jeremy.

Subject: Re: IDL fitting of piecewise continuous function
Posted by Jeremy Bailin on Tue, 17 May 2011 19:24:25 GMT
View Forum Message <> Reply to Message

sorry, that should be:

f(x) = A x^(1/2) x < x_m
  A x_m^(3/2) x^(-1) x > x_m

-Jeremy.

Subject: Re: IDL fitting of piecewise continuous function
Posted by Sasha Singh on Tue, 17 May 2011 19:26:41 GMT
View Forum Message <> Reply to Message

Thanks Jeremy,

I did look look at MPFIT. I am confused as to how do I define the constraints x<x_m and x > x_m in the function. Sorry I am quite new to IDL.

Sasha

On May 17, 3:21 pm, Jeremy Bailin <astroco...@gmail.com> wrote:
> Craigh Markwardt's MPFIT (and its varients) is most people's preferred fitting routine:
>
> http://www.physics.wisc.edu/~craigm/idl/fitting.html
>
> You define your own function to feed in, so it can have whatever behaviour you want. To ensure continuity, I would recommend re-parametrizing the functional form so that it is continuous by definition. In this, case, A and B aren't actually independent, so calculate B in terms of A by setting them equal at x=x_m, which gives
> B = A x_m^(3/2)
>
> So you're actually fitting:
>
> f(x) = A x_m^(1/2) x^(1/2) x < x_m
> A x_m^(3/2) x^(-1) x > x_m
>
> -Jeremy.

Subject: Re: IDL fitting of piecewise continuous function
Posted by Jeremy Bailin on Tue, 17 May 2011 19:53:19 GMT
View Forum Message <> Reply to Message

If you define your function analogously to the example in the MPFIT code, it might look like:

```
FUNCTION MYFUNCT, p, X=x, Y=y, ERR=err
  ; p[0] is A
  x_m = SOME_CONSTANT_VALUE

  ; x lt x_m is 1 if x<x_m and 0 otherwise
  ; x ge x_m is 1 if x >= x_m and 0 otherwise
  model = (x lt x_m) * p[0] * x^(0.5)  +  (x ge x_m) * p[0] * x_m^(1.5) / x

  return, (y-model)/err
END
```

Of course, that assumes that x_m is a constant, not a parameter - but if it's a parameter, you would just use p[1] instead of a constant value, for example.

-Jeremy.

Subject: Re: IDL fitting of piecewise continuous function
Posted by Craig Markwardt on Wed, 18 May 2011 02:35:13 GMT
View Forum Message <> Reply to Message

On May 17, 3:53 pm, Jeremy Bailin <astroco...@gmail.com> wrote:
> If you define your function analogously to the example in the MPFIT code, it might look like:
>
> FUNCTION MYFUNCT, p, X=x, Y=y, ERR=err
> ; p[0] is A
> x_m = SOME_CONSTANT_VALUE
>
> ; x lt x_m is 1 if x<x_m and 0 otherwise
> ; x ge x_m is 1 if x >= x_m and 0 otherwise
> model = (x lt x_m) * p[0] * x^(0.5) + (x ge x_m) * p[0] * x_m^(1.5) / x
>
> return, (y-model)/err
> END
>
> Of course, that assumes that x_m is a constant, not a parameter - but if it's a parameter, you would just use p[1] instead of a constant value, for example.

Yep, what he said.
Craig
