Subject: Re: Problems getting CURVEFIT to work
Posted by Craig Markwardt on Wed, 13 Nov 2002 01:14:45 GMT
View Forum Message <> Reply to Message

Jonathan Greenberg <greenberg@ucdavis.edu> writes:
> Hi there, I'm trying to use CURVEFIT to fit data to a decay function of the
> form:
> f(x) = a(1-e^(bx))+c

Problem 1. Your parameters A and C are very highly (anti) correlated with each other. It would be better to recast as A*EXP(B*X) + C.

```
> My code is as follows:
>
> pro decayfunc, X,A,F,pder
> bx=EXP(A[1]*X)
> F=A[0]*(1-bx)+A[2]
> if N_PARAMS() GE 4 THEN $
> pder=[[1-bx],[-A[0]*X*bx],[replicate(1.0,N_ELEMENTS(X))]]
> end
> X=[30185.0,33897.0,35089.0,35377.0,35665.0]
> Y=[0.3002,1.3849,1.3004,1.226,1.3118]
> A=[1.25,-1.0,-0.1]
```

Problem 2. Your initial value of "B" of -1 is not a good choice. When the fitter tries to evaluate EXP(-1.0*30185.) the result is zero. A better choice would be about -1./30000.

Problem 3. Your data don't look very exponential to me! There is just one low point. You are going to have to live with some very large confidence intervals...

```
> weight=[1.0,1.0,1.0,1.0,1.0]
>  yfit=CURVEFIT[X,Y,weights,A,SIGMA,FUNCTION_NAME='decayfunc', /DOUBLE]
```

Suggestion. It might be worth trying MPCURVEFIT or MPFITFUN from my web page. The fitting routines appear to be much more robust than the stock CURVEFIT.

Good luck,
Craig

http://cow.physics.wisc.edu/~craigm/idl/idl.html (under curve fitting)

--

-- --------------
Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu

Subject: Re: Problems getting CURVEFIT to work
Posted by Jonathan Greenberg on Wed, 13 Nov 2002 07:43:10 GMT
View Forum Message <> Reply to Message

Thanks!

On 11/12/02 5:14 PM, in article onk7jiuv7e.fsf@cow.physics.wisc.edu, "Craig Markwardt" <craigmnet@cow.physics.wisc.edu> wrote:

>
> Jonathan Greenberg <greenberg@ucdavis.edu> writes:
>> Hi there, I'm trying to use CURVEFIT to fit data to a decay function of the
>> form:
>> f(x) = a(1-e^(bx))+c
>
> Problem 1. Your parameters A and C are very highly (anti) correlated
> with each other. It would be better to recast as A*EXP(B*X) + C.

The problem is I need a curve that starts low and asymptotes higher -- I have a reason to believe it will asymptote at some maximum, hence the a(1-e^(bx))+c --> given this, how can I get this to work?

>
>
>> My code is as follows:
>>
>> pro decayfunc, X,A,F,pder
>> bx=EXP(A[1]*X)
>> F=A[0]*(1-bx)+A[2]
>> if N_PARAMS() GE 4 THEN $
>> pder=[[1-bx],[-A[0]*X*bx],[replicate(1.0,N_ELEMENTS(X))]]
>> end
>> X=[30185.0,33897.0,35089.0,35377.0,35665.0]
>> Y=[0.3002,1.3849,1.3004,1.226,1.3118]
>> A=[1.25,-1.0,-0.1]
>
> Problem 2. Your initial value of "B" of -1 is not a good choice.
> When the fitter tries to evaluate EXP(-1.0*30185.) the result is zero.
> A better choice would be about -1./30000.

I'll try that out!

>
> Problem 3. Your data don't look very exponential to me! There is

> just one low point. You are going to have to live with some very
> large confidence intervals...
>
>> weight=[1.0,1.0,1.0,1.0,1.0]
>> yfit=CURVEFIT[X,Y,weights,A,SIGMA,FUNCTION_NAME='decayfunc', /DOUBLE]

Yeah, I have more data now, and will try this!

>
> Suggestion. It might be worth trying MPCURVEFIT or MPFITFUN from my
> web page. The fitting routines appear to be much more robust than the
> stock CURVEFIT.

I'll try them. Thanks!

>
> Good luck,
> Craig
>
> http://cow.physics.wisc.edu/~craigm/idl/idl.html (under curve fitting)

Subject: Re: Problems getting CURVEFIT to work
Posted by Craig Markwardt on Wed, 13 Nov 2002 15:18:05 GMT
View Forum Message <> Reply to Message

Jonathan Greenberg <greenberg@ucdavis.edu> writes:
> On 11/12/02 5:14 PM, in article onk7jiuv7e.fsf@cow.physics.wisc.edu, "Craig
> Markwardt" <craigmnet@cow.physics.wisc.edu> wrote:
>> Jonathan Greenberg <greenberg@ucdavis.edu> writes:
>>> Hi there, I'm trying to use CURVEFIT to fit data to a decay function of the
>>> form:
>>> f(x) = a(1-e^(bx))+c
>>
>> Problem 1. Your parameters A and C are very highly (anti) correlated
>> with each other. It would be better to recast as A*EXP(B*X) + C.
>
> The problem is I need a curve that starts low and asymptotes higher -- I
> have a reason to believe it will asymptote at some maximum, hence the
> a(1-e^(bx))+c --> given this, how can I get this to work?

A*EXP(B*X) + D will suit you just fine. It's the same function after
all, with (D=C+A). The fit you were doing had C and A almost
perfectly anti-correlated, so combining the two will relieve some of
that problem.

Craig

--
 -- --------------
Craig B. Markwardt, Ph.D.   EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
 -- --------------

Subject: Re: Problems getting CURVEFIT to work
Posted by James Kuyper on Wed, 13 Nov 2002 15:41:10 GMT
View Forum Message <> Reply to Message

Jonathan Greenberg wrote:
>
> Thanks!
>
> On 11/12/02 5:14 PM, in article onk7jiuv7e.fsf@cow.physics.wisc.edu, "Craig
> Markwardt" <craigmnet@cow.physics.wisc.edu> wrote:
>
>>
>> Jonathan Greenberg <greenberg@ucdavis.edu> writes:
>>> Hi there, I'm trying to use CURVEFIT to fit data to a decay function of the
>>> form:
>>> f(x) = a(1-e^(bx))+c
>>
>> Problem 1. Your parameters A and C are very highly (anti) correlated
>> with each other. It would be better to recast as A*EXP(B*X) + C.
>
> The problem is I need a curve that starts low and asymptotes higher -- I

Which is precisely what a*exp(b*x)+c will do, for the appropriate values
of a, b, and c. For example, y=-80690323.0D*exp(-0.0006D*x)+1.340D is a
curve with those features, which makes a rough match to your data. It
can be converted to the form of your equation by re-writing it as:

y = 80690323.0D*(1.0D - exp(-0.0006D*x))-80690321.66D

But as you can see, that form obscures the assymptotic value, which is
1.34 for this curve. The assymptote in your form is a-c, which is a
small difference of two very large numbers. That causes problems for any
numerical approach to
fitting this equation to data.

Subject: Re: Problems getting CURVEFIT to work
Posted by Jonathan Greenberg on Fri, 15 Nov 2002 18:35:23 GMT

Hello again Craig:

You MPCURVEFIT function is terrific! I've started using it instead of CURVEFIT. One quick question, how does it deal with NaN values? I started including NaN values in my regressions, and I noticed some strange behavior from the output -- ideally it should just ignore the X,Y that has a NaN for the X value, but is this actually the case?

--j

On 11/12/02 5:14 PM, in article onk7jiuv7e.fsf@cow.physics.wisc.edu, "Craig Markwardt" <craigmnet@cow.physics.wisc.edu> wrote:

>
> Jonathan Greenberg <greenberg@ucdavis.edu> writes:
>> Hi there, I'm trying to use CURVEFIT to fit data to a decay function of the
>> form:
>> f(x) = a(1-e^(bx))+c
>
> Problem 1. Your parameters A and C are very highly (anti) correlated
> with each other. It would be better to recast as A*EXP(B*X) + C.
>
>
>> My code is as follows:
>>
>> pro decayfunc, X,A,F,pder
>> bx=EXP(A[1]*X)
>> F=A[0]*(1-bx)+A[2]
>> if N_PARAMS() GE 4 THEN $
>> pder=[[1-bx],[-A[0]*X*bx],[replicate(1.0,N_ELEMENTS(X))]]
>> end
>> X=[30185.0,33897.0,35089.0,35377.0,35665.0]
>> Y=[0.3002,1.3849,1.3004,1.226,1.3118]
>> A=[1.25,-1.0,-0.1]
>
> Problem 2. Your initial value of "B" of -1 is not a good choice.
> When the fitter tries to evaluate EXP(-1.0*30185.) the result is zero.
> A better choice would be about -1./30000.
>
> Problem 3. Your data don't look very exponential to me! There is
> just one low point. You are going to have to live with some very
> large confidence intervals...
>
>> weight=[1.0,1.0,1.0,1.0,1.0]
>> yfit=CURVEFIT[X,Y,weights,A,SIGMA,FUNCTION_NAME='decayfunc', /DOUBLE]
>
> Suggestion. It might be worth trying MPCURVEFIT or MPFITFUN from my

> web page. The fitting routines appear to be much more robust than the
> stock CURVEFIT.
>
> Good luck,
> Craig
>
> http://cow.physics.wisc.edu/~craigm/idl/idl.html   (under curve fitting)