Subject: Re: MPFIT question
Posted by j.coenia@gmail.com on Wed, 14 Jan 2009 21:37:13 GMT
View Forum Message <> Reply to Message

The user function has help statements at its beginning and end now, and I tried the other suggestions, but I'm still stumped. I'm doing something foolish.

Below is the problem code. Most of it is just the 353-element data array. Without the data, it's only two short functions, 20 lines of code or so total. I hope it's not bad netiquette to post so many lines. I can't figure out why these data crash the program. I can fit most other data acquired in this manner -- I have fit hundreds of thousands of these curves with this method, but the error crops up once every 50 cases or so (usually for data that don't fit the model very well, but it still shouldn't crash?).

The data are being fit to a gamma variate function. Two things I'm not sure about: (1) I'm passing t0 as a fitting parameter, and I'm not sure if this is acceptable. (2) Worse, I don't have a good idea of how to estimate error, so I'm just using the standard deviation of the baseline (pre-arrival) curve, which is probably wrong. Suggestions?

The code crashes after just two MPFIT iterations. Just type 'MPFIT_problem' to run.

Thanks for any help.

```
function gamma_variate, x, p

print, 'p start...'
help, p

baseline = p[0]
t0 = p[1]
tmax = p[2]  ; time of peak signal
smax = p[3]  ; signal max in mVolts
alpha = p[4]

 ; shift time, apply gamma variate only to data after t0

wh = Where(x GE t0, ct)
t = (x - t0)[wh]
tmax = tmax - t0

 ; gamma variate function
```

```
s = baseline + (smax) * (tmax^(-alpha)) * exp(alpha) * (t^alpha) *  exp
((-alpha)  * t / tmax)

 ; add the pre-t0 data (baseline)

n = n_elements(x)
pre = make_array(n - ct, Value = baseline)
s = [pre, s]

print, 'p end...'
help, p

return, s

 end;---------------------------------------------------- --


pro mpfit_problem

!except = 2

data = [ $
 64.1682 , $
 66.3804 , $
 73.4243 , $
 64.1682 , $
 64.1682 , $
 55.9551 , $
 47.0046 , $
 52.2084 , $
 48.6855 , $
 52.2084 , $
 57.9162 , $
 75.9137 , $
 64.1682 , $
 75.9137 , $
 86.6244 , $
 89.4994 , $
 89.4994 , $
 101.845 , $
 105.153 , $
 89.4994 , $
 95.4993 , $
 92.457 , $
 105.153 , $
 89.4994 , $
 101.845 , $
```

92.457 , $
86.6244 , $
83.8301 , $
89.4994 , $
92.457 , $
92.457 , $
95.4993 , $
95.4993 , $
95.4993 , $
89.4994 , $
89.4994 , $
75.9137 , $
71.0068 , $
78.4766 , $
57.9162 , $
64.1682 , $
59.9377 , $
52.2084 , $
45.3754 , $
39.3496 , $
39.3496 , $
48.6855 , $
62.0212 , $
73.4243 , $
50.4197 , $
73.4243 , $
66.3804 , $
66.3804 , $
68.6594 , $
68.6594 , $
86.6244 , $
57.9162 , $
89.4994 , $
73.4243 , $
73.4243 , $
78.4766 , $
75.9137 , $
83.8301 , $
95.4993 , $
86.6244 , $
73.4243 , $
81.1148 , $
86.6244 , $
89.4994 , $
40.785 , $
54.053 , $
68.6594 , $
78.4766 , $

73.4243 , $
95.4993 , $
83.8301 , $
81.1148 , $
95.4993 , $
83.8301 , $
75.9137 , $
57.9162 , $
68.6594 , $
83.8301 , $
78.4766 , $
75.9137 , $
68.6594 , $
78.4766 , $
75.9137 , $
83.8301 , $
75.9137 , $
64.1682 , $
71.0068 , $
78.4766 , $
73.4243 , $
64.1682 , $
92.457 , $
68.6594 , $
73.4243 , $
75.9137 , $
62.0212 , $
73.4243 , $
55.9551 , $
57.9162 , $
62.0212 , $
73.4243 , $
75.9137 , $
75.9137 , $
92.457 , $
83.8301 , $
112.047 , $
98.6279 , $
123.118 , $
92.457 , $
105.153 , $
92.457 , $
75.9137 , $
86.6244 , $
92.457 , $
73.4243 , $
78.4766 , $
66.3804 , $

75.9137 , $
119.327 , $
127.011 , $
105.153 , $
73.4243 , $
54.053 , $
73.4243 , $
75.9137 , $
66.3804 , $
86.6244 , $
68.6594 , $
78.4766 , $
75.9137 , $
78.4766 , $
81.1148 , $
127.011 , $
152.651 , $
95.4993 , $
83.8301 , $
119.327 , $
50.4197 , $
68.6594 , $
95.4993 , $
75.9137 , $
131.009 , $
54.053 , $
108.553 , $
62.0212 , $
83.8301 , $
57.9162 , $
152.651 , $
119.327 , $
119.327 , $
54.053 , $
73.4243 , $
86.6244 , $
89.4994 , $
105.153 , $
73.4243 , $
78.4766 , $
112.047 , $
50.4197 , $
89.4994 , $
92.457 , $
30.5028 , $
55.9551 , $
119.327 , $
131.009 , $

68.6594 , $
71.0068 , $
81.1148 , $
112.047 , $
73.4243 , $
64.1682 , $
52.2084 , $
123.118 , $
127.011 , $
75.9137 , $
112.047 , $
101.845 , $
95.4993 , $
105.153 , $
55.9551 , $
135.114 , $
83.8301 , $
95.4993 , $
54.053 , $
36.6134 , $
42.2669 , $
71.0068 , $
68.6594 , $
95.4993 , $
115.638 , $
57.9162 , $
148.095 , $
48.6855 , $
75.9137 , $
95.4993 , $
127.011 , $
187.992 , $
112.047 , $
172.084 , $
71.0068 , $
92.457 , $
135.114 , $
148.095 , $
36.6134 , $
89.4994 , $
105.153 , $
68.6594 , $
119.327 , $
27.2923 , $
55.9551 , $
95.4993 , $
83.8301 , $
152.651 , $

43.7966 , $
62.0212 , $
112.047 , $
123.118 , $
135.114 , $
105.153 , $
32.8277 , $
62.0212 , $
39.3496 , $
47.0046 , $
92.457 , $
108.553 , $
86.6244 , $
152.651 , $
55.9551 , $
119.327 , $
52.2084 , $
75.9137 , $
71.0068 , $
35.3102 , $
43.7966 , $
119.327 , $
112.047 , $
127.011 , $
127.011 , $
62.0212 , $
54.053 , $
177.256 , $
311.315 , $
123.118 , $
101.845 , $
92.457 , $
152.651 , $
127.011 , $
101.845 , $
68.6594 , $
162.121 , $
172.084 , $
167.04 , $
152.651 , $
108.553 , $
172.084 , $
143.655 , $
152.651 , $
162.121 , $
279.383 , $
162.121 , $
123.118 , $

34.0487 , $
62.0212 , $
83.8301 , $
177.256 , $
172.084 , $
157.326 , $
73.4243 , $
135.114 , $
92.457 , $
59.9377 , $
131.009 , $
42.2669 , $
71.0068 , $
78.4766 , $
135.114 , $
78.4766 , $
143.655 , $
112.047 , $
78.4766 , $
105.153 , $
66.3804 , $
73.4243 , $
89.4994 , $
143.655 , $
75.9137 , $
152.651 , $
89.4994 , $
98.6279 , $
95.4993 , $
66.3804 , $
24.3893 , $
86.6244 , $
57.9162 , $
119.327 , $
135.114 , $
75.9137 , $
40.785 , $
112.047 , $
47.0046 , $
89.4994 , $
92.457 , $
95.4993 , $
89.4994 , $
50.4197 , $
81.1148 , $
57.9162 , $
45.3754 , $
66.3804 , $

```
 73.4243 , $
59.9377 , $
73.4243 , $
105.153 , $
78.4766 , $
64.1682 , $
54.053 , $
62.0212 , $
39.3496 , $
47.0046 , $
89.4994 , $
28.327 , $
81.1148 , $
43.7966 , $
68.6594 , $
48.6855 , $
54.053 , $
40.785 , $
35.3102 , $
42.2669 , $
75.9137 , $
64.1682 , $
64.1682 , $
39.3496 , $
95.4993 , $
78.4766 , $
101.845 , $
64.1682 , $
68.6594 , $
48.6855 , $
59.9377 , $
66.3804 , $
83.8301 , $
40.785 , $
39.3496 , $
78.4766 , $
127.011 , $
172.084 , $
52.2084 , $
119.327 , $
37.9595 , $
66.3804 , $
108.553 , $
86.6244 , $
83.8301  $
]

; calculate times @ 30Hz
```

```
n = n_elements(data)
t = findgen(n)/30.0

 ; define and contrain parameters

parinfo = replicate({value:0.D, fixed:0, limited:[0,0], $
       limits:[0.D,0], mpmaxstep: 0.0D}, 5)

 ; baseline
parinfo(0).fixed = 0b
parinfo(0).limited = [1,1]
parinfo(0).limits = [0, 3675.36]
parinfo(0).value = 84.551201

 ; t0
t0 = 4.0333333
parinfo(1).fixed = 0b
parinfo(1).limited(0) = 1
parinfo(1).limits(0)  = 0.D
parinfo(1).value = 4.0333333

 ; tmax
parinfo(2).fixed = 0b
parinfo(2).limited = [1,1]
parinfo(2).limits = [t0, n/30.0 - 0.5] ; 0.5 secs before end
parinfo(2).value =   8.5333338

 ; max
parinfo(3).limited  = [1,1]
parinfo(3).limits = [0, 3675.36]
parinfo(3).value = 127.96773

 ; alpha
parinfo(4).fixed = 0b
parinfo(4).limited = [1,1]
parinfo(4).limits = [0.5, 12]
parinfo(4).value = 1.0

 ; calculate std deviation of signal up to t0
 ; probably not the best way to estimate error...

err = stddev(data[0:t0*30]) > 1.0

 ; plot the signal

window, /free
plot, t, data, $
```

```
 Title = 'Signal', $
 XTitle = 'Time (s)', $
 YTitle = 'mVolts'

 ; fit

parms = mpfitfun('gamma_variate', t, data, replicate(err, n), $
  YFit = fit, $
  Bestnorm = chisq, $
  /NaN, $
  Parinfo = parinfo, $
  Quiet = 0 $
  )

 ; plot fit

oplot, t, gamma_variate(t, parms), Color = FSC_Color('red')

 end;---------------------------------------------------------- --
```