Subject: Re: Yet another user with poly_fit problems
Posted by Heinz Stege on Wed, 02 Oct 2013 19:57:38 GMT
View Forum Message <> Reply to Message

On Wed, 2 Oct 2013 12:31:58 -0700 (PDT), suicidaleggroll@gmail.com
wrote:

> On Wednesday, October 2, 2013 11:35:35 AM UTC-6, Ken G wrote:
>> I feel compelled to jump in here and say that the interpolation method is mathematically dicey
and I wouldn't do it. What you're essentially doing is weighting your data points in a very
non-linear way. We see that the first few points are clustered closely together in x; the last few
points are widely separated. The uniform x spacing in the linear interpolation therefore devotes
many more points to the large-x-value region relative to the number of y points out there to
support the data. So in your fit result, you're biasing or weighting the data as if the large y were
the most significant point by far. You can see this in the results if you plot newX, D . There's a
downward bulge in D between the last two points as the weighting pulls the curve down toward
the linear interpolation. Plus, D fails to come close to the first 2 points.
>
> That is true. I mentioned in my post that this approach was suitable for the application in which I
used it (my points were more or less evenly spaced), but you're right that it might not be ideal for
Gus. The same kind of approach could probably still be used though with different abscissa
values to prevent the heavy weighting at higher X values.

I also fully agree with Ken. And I tried to get the newX values
without different weighting for the X values (except for the first and
the last point). I ran:

newX=X
for i=0,4 do begin &$
   newX=[newX,(newX+newX[1:*])/2.] &$
   newX=newX[sort(newX)] &end
newY = interpol(Y, X, newX)

The result is still as Ken describes: Very bad matching for the first
four points and a downward bulge between the last two points.

For me it is a fact, that the given data points can not be fairly
fitted by a polynomial.

Heinz