Subject: POLY_FIT 5.0: inaccurate results. use 4.0 if possible.
Posted by Russ Welti (on mapper on Wed, 25 Mar 1998 08:00:00 GMT
View Forum Message <> Reply to Message

The 5.0 POLY_FIT has accuracy problems.

My app uses POLY_FIT extensively to solve a problem
which can be expressed as "choosing the very best of
many possible curve fittings".

I noticed the app was producing some wrong results, after
upgrading to IDL 5.0.

(I can presently run both 4.0 and 5.0)

I traced the problem to a key loop which performs
say 1000 curve fits and then chooses the X/Y set which had
the lowest error (sigma).

I inserted statements into the loop to print out the
X and Y values being fit, and then the resulting sigma values.

The X and Y values are all identical, and the sigma values
are mostly similar except on the important fit, the best
one, where 4.0's POLY_FIT gives a value like .37 but
5.0's gives ~ 1.9, a sixfold difference!

Unfortunately the app behaves poorly because any fit
over 1.0 error in the Y domain is considered bad, so the
program continues to search for a better one when it
has already found what is in fact the best one.

The fix for me was to copy the old POLY_FIT into
my app's directory and continue to use it.

I'd rather use the new one, if it's faster or somehow better,
but unless anyone can tell me where I went wrong, the better
accuracy of the old version is what I need.

 ---
here's a nutshell showing the problem:

```
IDL5> .run
- pks=[692,760,879,1030,1159,1177,1312,1436,1667,1790,1908]
- szs=[105,120,145,175,200,204,230,255,300,325,350]
-
- ; perform fits
```

```
-
- fit4=POLY_FIT4(pks,szs,3,yfit4,yband,sigma4,a)   ; 4.0 version
- fit5=POLY_FIT (pks,szs,3,yfit5,yband,sigma5,a)   ; 5.0 version
-
- ; print results
-
- print,'POLY_FIT4 gives sigma of:',sigma4
- print,'POLY_FIT5 gives sigma of:',sigma5
- print
- print,format='("POLY_FIT4 best fit:",20(F7.2))',yfit4
- print,format='("POLY_FIT5 best fit:",20(F7.2))',yfit5
- end
% Compiled module: $MAIN$.
% Compiled module: POLY_FIT4.
% Compiled module: POLY_FIT.
POLY_FIT4 gives sigma of:      0.37212247
POLY_FIT5 gives sigma of:      1.98881

POLY_FIT4 best fit: 105.28 119.82 144.59 175.05 200.47 203.99 230.27 254.39 300.05 325.13
349.95
POLY_FIT5 best fit: 105.65 120.27 145.19 175.86 201.50 205.05 231.58 255.97 302.20 327.63
352.81
```

Subject: Re: POLY_FIT 5.0: inaccurate results. use 4.0 if possible.
Posted by davidf on Mon, 30 Mar 1998 08:00:00 GMT

View Forum Message <> Reply to Message

Hi Folks,

Just to keep you up to date on this inaccurate POLY_FIT 5 saga,
I now hear from RSI technical support people that there is
a bug in the DOUBLE implementation. So, just to summarize.

Russ Welti reported that IDL 5 POLY_FIT gave different (and
apparently inaccurate) results compared to the IDL 5 POLY_FIT.
RSI reports that in the interest of execution speed, they
now make the calculation in single precision and to get the
same results as before we should set the DOUBLE keyword. (I
agree with Peter, by the way, that a SINGLE keyword would
have made more sense here.) But there is apparently still
a bug in the DOUBLE implementation, so the result is perhaps
even worse than doing it in single precision.

The work-around it to make the *data* a double precision
value before passing it to POLY_FIT. I'm told this will
be fixed before the final release of 5.1.

This reminds me of one of those times when I tried to do
something nice for my wife and it gets all fouled up and
she ends up not speaking to me for a week. :-(

Cheers,

David

David Fanning, Ph.D.
Fanning Software Consulting
E-Mail: davidf@dfanning.com
Phone: 970-221-0438
Coyote's Guide to IDL Programming: http://www.dfanning.com/

Subject: Re: POLY_FIT 5.0: inaccurate results. use 4.0 if possible.
Posted by pit on Mon, 30 Mar 1998 08:00:00 GMT
View Forum Message <> Reply to Message

In article <MPG.f859b08896d2aad989765@news.frii.com>,
davidf@dfanning.com (David Fanning) writes:

> I submitted this problem to the RSI technical support staff
> to see what they knew about it. Here is their response.
> This is one of those "nice things to do" that cause confusion
> because the change is not communicated carefully enough. I'm
> not sure how to solve this "upgrade" problem, but I do know
> it causes grief from time to time.
>
> **********
> To obtain similar results for the POLY_FIT procedure from IDL 4.0.1 to
> IDL 5.0, please set (/DOUBLE) the DOUBLE keyword when using POLY_FIT in
> IDL 5.0.ï¿½ For most of IDL 4.0.1 numerical analysis routines, the
> internal computations are performed on double precision variables.
> However, because of timing feature requests, these routines were changed
> to perform single precision computations.ï¿½ Fortunately, the DOUBLE
> keyword was added to these routines to force double precision
> computations.ï¿½ Thus, to force IDL 5.0's POLY_FIT to perform like IDL
> 4.0.1's POLY_FIT, please use this DOUBLE keyword.ï¿½ Sorry for any
> inconvenience this change may have caused.
> **********

So, why don't they use a keyword /SINGLE and stay backward compatible?
This is bad style, IHMO.

Peter

--
~~~~~~

Peter "Pit" Suetterlin http://www.uni-sw.gwdg.de/~pit
Universitaets-Sternwarte Goettingen
Tel.: +49 551 39-5048 pit@uni-sw.gwdg.de
-- * -- * ...-- * -- * ...-- * -- * ...-- * -- * ...-- * -- * ...-- * --
Come and see the stars! http://www.kis.uni-freiburg.de/~ps/SFB
Sternfreunde Breisgau e.V. Tel.: +49 7641 3492
~~~~~~