
Subject: POLY_FIT gives wrong answer !
Posted by [Henk Schets](#) on Tue, 09 May 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

Maybe all of you already know this, but I didn't and I could not find any documentation about it.

The problem : I have two long arrays (x- and y-values) and I made subarrays out of them by using an array as subscript, like

```
x = fltarray(l)
```

```
y = fltarray(l)
```

```
indexarr = longarray(l)
```

(you know, the way to sort different arrays together)

However, when I do something like out =

poly_fit(x[indexarr],y[indexarr]), the outcome is simply wrong ! I know this because other programs gave another solution (all the same).

The only way to do it right is by making other arrays like x2 and y2 and doing a poly_fit on it.

Is this a known issue ?

Greetings,

Henk

Henk Schets
Royal Meteorological Institute of Belgium (R.M.I.B.)
Ringlaan 3
B-1180 Brussels
Belgium

Tel: (+32) 2 3730597

Fax: (+32) 2 3751259

E-mail: Henk.Schets@oma.be

WWW server: <http://www.meteo.oma.be/IRM-KMI>

Subject: Re: POLY_FIT gives wrong answer !
Posted by [davidf](#) on Wed, 10 May 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Henk Schets (Henk.Schets@oma.be) writes:

> David Fanning wrote:

```

>
>> Henk Schets (Henk.Schets@oma.be) writes:
>>
>>> Maybe all of you already know this, but I didn't and I could not find
>>> any documentation about it.
>>> The problem : I have two long arrays (x- and y-values) and I made
>>> subarrays out of them by using an array as subscript, like
>>>   x = fltarray(l)
>>>   y = fltarray(l)
>>>   indexarr = longarray(l)
>>> (you know, the way to sort different arrays together)
>>> However, when I do something like out =
>>> poly_fit(x[indexarr],y[indexarr]), the outcome is simply wrong ! I know
>>> this because other programs gave another solution (all the same).
>>> The only way to do it right is by making other arrays like x2 and y2 and
>>> doing a poly_fit on it.
>>>
>>> Is this a known issue ?
>>
>> I don't know. What is known is that things are not always
>> the way they seem to be. I'm guessing this is one of those
>> times. :-)
>>
>> Cheers,
>>
>> David
>>
>> --
>> David Fanning, Ph.D.
>
> Could you be a littlebit more cryptic, please ?

```

I'm just saying that about 5 times out of 10 when users describe problems to you, they are offering red herrings. I think this is one of those times. There is no explanation that I can think of why a subscripted array is not identical to a variable made from that subscripted array. Subscripted arrays are, of course, passed into the program by value, while variables are passed in by reference. But since the input variables don't change when they are passed into POLY_FIT, I think we can rule this out as a suspect.

And from what I learned from my Sherlock Holmes, if you eliminate everyone who can't possibly have committed the crime, who you are left with--no matter how improbable--is the murderer. My list of suspects is pretty much narrowed down to programmer error in the absence of a working (or mis-working, in this case) example. :-)

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting

Phone: 970-221-0438 E-Mail: davidf@dfanning.com

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: POLY_FIT gives wrong answer !

Posted by [Henk Schets](#) on Wed, 10 May 2000 07:00:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

David Fanning wrote:

> Henk Schets (Henk.Schets@oma.be) writes:

>

>> Maybe all of you already know this, but I didn't and I could not find

>> any documentation about it.

>> The problem : I have two long arrays (x- and y-values) and I made

>> subarrays out of them by using an array as subscript, like

>> x = fltarray(l)

>> y = fltarray(l)

>> indexarr = longarray(l)

>> (you know, the way to sort different arrays together)

>> However, when I do something like out =

>> poly_fit(x[indexarr],y[indexarr]), the outcome is simply wrong ! I know

>> this because other programs gave another solution (all the same).

>> The only way to do it right is by making other arrays like x2 and y2 and

>> doing a poly_fit on it.

>>

>> Is this a known issue ?

>

> I don't know. What is known is that things are not always

> the way they seem to be. I'm guessing this is one of those

> times. :-)

>

> Cheers,

>

> David

>

> --

> David Fanning, Ph.D.

Could you be a littlebit more cryptic, please ?

Henk

Subject: Re: POLY_FIT gives wrong answer !
Posted by [steinhh](#) on Thu, 11 May 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

In article <MPG.13848f1b4d7167dd989b0f@news.frii.com>
davidf@dfanning.com (David Fanning) writes:

> [Actually, Bill Thompson wrote:
>
> ..about getting different results with a sorted vs unsorted data
> set...]

Excuse my brevity in quoting - this post is also smuggled out from
Goddard, so I'm keeping it short to avoid detection :-)

Although I think Bill's going in the right direction with regard to
the POLY_FIT question (he's sitting in the cubicle next to me, so I
should know where he's going :-), it doesn't actually explain the
original problem as it was **stated** by Henk Schets:

> The only way to do it right is by making other arrays like x2 and y2 and
> doing a poly_fit on it.

Taken at face value, this means he's doing this (referring to Bill's
example code):

```
xs = x(s)
ys = y(s)
param3 = poly_fit(xs,ys,2,yfit2)
```

Doing so, I get **identical** results compared to using x(s) and y(s)
(whereas the sorted/unsorted versions differ). So, the problem lies
somewhere else.

My guess is that Henk is converting either x(s) or y(s) (or both) to
double precision, which will make the whole computation be performed
with double precision..

..or Something Else (tm). Unless we get an actual example (including
data) that will show the discrepancy, I'm leaning towards David's
general hypothesis that the problem is not in POLY_FIT, but in the
application of it..

Stein Vidar

Subject: Re: POLY_FIT gives wrong answer !
Posted by [davidf](#) on Thu, 11 May 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

The Powers That Be at Goddard Space Flight Center prefer their employees have no opinions about anything. Hence, they are refusing to publish any news articles sent from the place. (I'm ready to file a Freedom of Information Act request to get Stein Vidar back on-line.) Bill Thompson smuggled this article out in his lunch box and asked if I would post it for him. Guess we will have to re-open discussion. :-(

Newsgroups: comp.lang.idl-pvwave
Subject: Re: POLY_FIT gives wrong answer !
References: <3917F8C2.A6DB4535@oma.be>

Henk Schets <Henk.Schets@oma.be> writes:

> Hi,

> Maybe all of you already know this, but I didn't and I could not find
> any documentation about it.

> The problem : I have two long arrays (x- and y-values) and I made
> subarrays out of them by using an array as subscript, like

> x = fltarray(l)
> y = fltarray(l)
> indexarr = longarray(l)
> (you know, the way to sort different arrays together)

> However, when I do something like out =
> poly_fit(x[indexarr],y[indexarr]), the outcome is simply wrong ! I know
> this because other programs gave another solution (all the same).

> The only way to do it right is by making other arrays like x2 and y2 and
> doing a poly_fit on it.

> Is this a known issue ?

At first I thought that David was right, and that there was a subtle error in the calling program. However, some experimentation leads me to believe that the answer has to do with round-off error.

To test what was going on, I generated a random series of X points between 0 and 1, and a corresponding Y array based on a polynomial, plus some noise.

```
IDL> x = randomu(seed,1000)
IDL> y = 3+5*x-7*x^2+randomn(seed,1000)
```

Applying POLY_FIT to these data gave reasonable results.

```
IDL> param1 = poly_fit(x,y,2,yfit1)
```

I then sorted the input arrays into increasing X, and reapplied POLY_FIT, expecting to get the same answer.

```
IDL> s=sort(x)
IDL> param2 = poly_fit(x(s),y(s),2,yfit2)
```

To my surprise, the parameters from the two runs of POLY_FIT are very close, but not exactly the same, even though the input arrays are exactly the same, just ordered differently. My only conclusion is that the difference must be due to round-off error, which affects the data differently depending on the order.

Perhaps your case is a more extreme example of round-off error affecting the calibration. I've definitely had problems sometimes with completely bogus results from POLY_FIT caused by round-off error. In my case this has always occurred when the range in X from minimum to maximum is small compared to the values of X. In such cases, I've always found it useful to redefine the problem to fit a polynomial to $X - X_0$ instead of X, where X_0 is comparable to the average value of X. For example, if my values of X ranged from 13479.25 to 13480.16, subtracting a typical value of $X_0 = 13480$ gives me a new X' which ranges from -0.75 to +0.16. This reduces the effect of round-off errors dramatically.

William Thompson

Subject: Re: POLY_FIT gives wrong answer !
Posted by [davidf](#) on Thu, 11 May 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Henk Schets (Henk.Schets@oma.be) writes:

> OK, so if you think I am wrong, why is it that when I just plot the data with the
> array as a subscript, the outcome is the same as if I made a totally new array ?
> Isn't this an indication of some bug in poly_fit ?

Uh, oh. Now *I'm* confused. I thought the results were *different*.
(And I still do after reading the original post again.)

Oh, well, I'll tell you what. As long as you have the thing working, I'm willing to put this into the category of "One of Life's Mysteries" and drop the subject until next time. :-)

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting

Phone: 970-221-0438 E-Mail: davidf@dfanning.com

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: POLY_FIT gives wrong answer !

Posted by [Henk Schets](#) on Thu, 11 May 2000 07:00:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

David Fanning wrote:

> Henk Schets (Henk.Schets@oma.be) writes:

>

>> David Fanning wrote:

>>

>>> Henk Schets (Henk.Schets@oma.be) writes:

>>>

>>>> Maybe all of you already know this, but I didn't and I could not find
>>>> any documentation about it.

>>>> The problem : I have two long arrays (x- and y-values) and I made
>>>> subarrays out of them by using an array as subscript, like

>>>> x = fltarray(l)

>>>> y = fltarray(l)

>>>> indexarr = longarray(l)

>>>> (you know, the way to sort different arrays together)

>>>> However, when I do something like out =

>>>> poly_fit(x[indexarr],y[indexarr]), the outcome is simply wrong ! I know

>>>> this because other programs gave another solution (all the same).

>>>> The only way to do it right is by making other arrays like x2 and y2 and
>>>> doing a poly_fit on it.

>>>>

>>>> Is this a known issue ?

>>>

>>> I don't know. What is known is that things are not always

>>> the way they seem to be. I'm guessing this is one of those

>>> times. :-)

>>>

>>> Cheers,

>>>

>>> David

>>>

>>> --

>>> David Fanning, Ph.D.

>>
>> Could you be a littlebit more cryptic, please ?
>
> I'm just saying that about 5 times out of 10 when users
> describe problems to you, they are offering red herrings.
> I think this is one of those times. There is no explanation
> that I can think of why a subscripted array is not identical
> to a variable made from that subscripted array. Subscripted
> arrays are, of course, passed into the program by value, while
> variables are passed in by reference. But since the input
> variables don't change when they are passed into POLY_FIT,
> I think we can rule this out as a suspect.
>
> And from what I learned from my Sherlock Holmes, if you
> eliminate everyone who can't possibly have committed the
> crime, who you are left with--no matter how improbable--
> is the murderer. My list of suspects is pretty much narrowed
> down to programmer error in the absence of a working
> (or mis-working, in this case) example. :-)
>
> Cheers,
>
> David

OK, so if you think I am wrong, why is it that when I just plot the data with the array as a subscript, the outcome is the same as if I made a totally new array ? Isn't this an indication of some bug in poly_fit ?

Henk

Subject: Re: POLY_FIT gives wrong answer !
Posted by [Henk Schets](#) on Fri, 12 May 2000 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Stein Vidar Hagfors Haugan wrote:

> In article <MPG.13848f1b4d7167dd989b0f@news.frii.com>
> davidf@dfanning.com (David Fanning) writes:
>
>> [Actually, Bill Thompson wrote:
>>
>> ..about getting different results with a sorted vs unsorted data
>> set...]
>
> Excuse my brevity in quoting - this post is also smuggled out from
> Goddard, so I'm keeping it short to avoid detection :-)
>

> Although I think Bill's going in the right direction with regard to
> the POLY_FIT question (he's sitting in the cubicle next to me, so I
> should know where he's going :-), it doesn't actually explain the
> original problem as it was **stated** by Henk Schets:
>
>> The only way to do it right is by making other arrays like x2 and y2 and
>> doing a poly_fit on it.
>
> Taken at face value, this means he's doing this (referring to Bill's
> example code):
>
> xs = x(s)
> ys = y(s)
> param3 = poly_fit(xs,ys,2,yfit2)
>
> Doing so, I get **identical** results compared to using x(s) and y(s)
> (whereas the sorted/unordered versions differ). So, the problem lies
> somewhere else.
>
> My guess is that Henk is converting either x(s) or y(s) (or both) to
> **double** precision, which will make the whole computation be performed
> with double precision..
>
> ..or Something Else (tm). Unless we get an actual example (including
> data) that will show the discrepancy, I'm leaning towards David's
> general hypothesis that the problem is not in POLY_FIT, but in the
> application of it..
>
> Stein Vidar

Yep, found the error. It was hard to find, so I'm sorry I questioned the
perfectness of poly_fit. At least I learned about the rounding error ;-)

Thanks,

Henk
