Subject: Correlate function (bug?)
Posted by Theo Brauers on Fri, 13 Mar 1998 08:00:00 GMT
View Forum Message <> Reply to Message

Hi

I was facing a problem with the correlate function
which is demonstrated by the following IDL procedure:

```
pro testc
x=[1.D-5, 1.D-6, 0.0]
y=[1.D-5, 1.1D-6, 0.0]
f=[1.D0, 1.D1, 1.D2, 1.D3, 1.D4, 1.D5, 1.D6]
for i=0,6 do begin
print, i, correlate(x,y, /double) $
    , correlate(x,y*f[i], /double) $
    , correlate(x*f[i],y, /double) $
    , correlate(x*f[i],y*f[i], /double)
endfor
END
```

giving the following result:

```
 0      1.#QNAN00      1.#QNAN00      1.#QNAN00      1.#QNAN00
 1      1.#QNAN00      1.#QNAN00      1.#QNAN00      1.#QNAN00
 2      1.#QNAN00      1.#QNAN00      1.#QNAN00      1.#QNAN00
 3      1.#QNAN00      1.#QNAN00      1.#QNAN00     0.99995432
 4      1.#QNAN00      1.#QNAN00      1.#QNAN00     0.99995432
 5      1.#QNAN00     0.99995432     0.99995432     0.99995432
 6      1.#QNAN00     0.99995432     0.99995432     0.99995432
```

In my opinion there is no reason for an underflow or whatever causes
correlate to return NAN.

Best Theo

Dr. Theo Brauers
Institut fuer Atmosphaerische Chemie (ICG-3)
Forschungszentrum Juelich
52425 JUELICH, Germany
Tel. +49-2461-61-6646 Fax. +49-2461-61-5346

Subject: Re: Correlate function (bug?)
Posted by thompson on Mon, 16 Mar 1998 08:00:00 GMT

Theo Brauers <Th.Brauers@fz-juelich.de> writes:

> Hi

> I was facing a problem with the correlate function
> which is demonstrated by the following IDL procedure:

```
> pro testc
> x=[1.D-5, 1.D-6, 0.0]
> y=[1.D-5, 1.1D-6, 0.0]
> f=[1.D0, 1.D1, 1.D2, 1.D3, 1.D4, 1.D5, 1.D6]
> for i=0,6 do begin
> print, i, correlate(x,y, /double) $
>     , correlate(x,y*f[i], /double) $
>     , correlate(x*f[i],y, /double) $
>     , correlate(x*f[i],y*f[i], /double)
> endfor
> END
```

> giving the following result:

```
>  0      1.#QNAN00      1.#QNAN00      1.#QNAN00      1.#QNAN00
>  1      1.#QNAN00      1.#QNAN00      1.#QNAN00      1.#QNAN00
>  2      1.#QNAN00      1.#QNAN00      1.#QNAN00      1.#QNAN00
>  3      1.#QNAN00      1.#QNAN00      1.#QNAN00     0.99995432
>  4      1.#QNAN00      1.#QNAN00      1.#QNAN00     0.99995432
>  5      1.#QNAN00     0.99995432     0.99995432     0.99995432
>  6      1.#QNAN00     0.99995432     0.99995432     0.99995432
```

> In my opinion there is no reason for an underflow or whatever causes
> correlate to return NAN.

> Best Theo

This behavior seems to have been introduced in IDL version 5. IDL/v4 doesn't have any problem, and returns the result

```
      0      0.99995432     0.99995432     0.99995432     0.99995432
      1      0.99995432     0.99995432     0.99995432     0.99995432
      2      0.99995432     0.99995432     0.99995432     0.99995432
      3      0.99995432     0.99995432     0.99995432     0.99995432
      4      0.99995432     0.99995432     0.99995432     0.99995432
      5      0.99995432     0.99995432     0.99995432     0.99995432
      6      0.99995432     0.99995432     0.99995432     0.99995432
```

William Thompson

Subject: Re: Correlate function (bug?)
Posted by John Smith on Tue, 17 Mar 1998 08:00:00 GMT
View Forum Message <> Reply to Message

William Connolley wrote:

> In article 9AA03CE1@fz-juelich.de, Theo Brauers <Th.Brauers@fz-juelich.de> writes:
>> I was facing a problem with the correlate function
>> which is demonstrated by the following IDL procedure:
>
>> pro testc
>> x=[1.D-5, 1.D-6, 0.0]
>> y=[1.D-5, 1.1D-6, 0.0]
>> print, correlate(x,y, /double)
>
>> giving the following result:
>>
>> NAN
>
> The problem is due to stupid programming in the correlate function.
> You can look at the source in $IDL_DIR/lib/correlate.pro.
> There is a tolerance (set to 1e-12 at /double, or 1e-6 otherwise) and if
> ||x'||^2*||y'||^2 is less than this, NAN is returned. Double precision should be
> good for a lot better than 1e-12, and anyway its more the ratio between x and y
> that would matter not the absolute value. Ho hum.
>
> Just take the code and insert tol=0 in the appropriate place...
>
> ps - the documentation doesn't mention this tolerance factor, presumably to avoid
> worrying our pretty little heads...
>
> - William

This bug was reported long time ago to RSI and they said they will fix it in one of
their
next releases. Meanwhile I use the old version.

Samuel Haimov
haimov@uwyo.edu