Subject: the NaN effect :-|
Posted by rkombiyil on Tue, 12 Jun 2007 15:33:59 GMT
View Forum Message <> Reply to Message

This stumps me.. We had some discussions on NaN's earlier, but mostly
wrt 'TOTAL'
Lets say,

```
IDL>a=[6.2,12.5,14.1,0.,22,!values.f_nan]
IDL> print,max(a)
      22.0000
IDL> print,min(a)
      0.00000
```

; Now I increase number of NaN's in the array :D

```
IDL> a=[6.2,12.5,14.1,0.,!values.f_nan,22,!values.f_nan]
IDL> print,max(a)
      22.0000
IDL> print,min(a)
      0.00000
```

; go on, repeat this (it is 00:23 where I am @, so CARPE NOCTEM!) :-P

```
IDL>  a=[!values.f_nan,6.2,12.5,14.1,0.,!values.f_nan,22,!values.f _nan]
IDL> print,min(a)
          NaN
IDL> print,max(a)
          NaN
```

==
Same goes for MIN also. ??!!

IDL's docu says:
<snip from IDL ref guide: Page 1269/4090>
If the MAX function is run on an array containing NaN values and the
NAN keyword is not set, an invalid result will occur.
</snip>
The same is said for MIN also.

So the result (OPS with MIN/MAX) is directly proportional to the
number of NaN's we eat? er, add to the array? :P

So what is right and what is wrong? Enlighten, please.
/metachronist

Subject: Re: the NaN effect :-|
Posted by Jim Pendleton, ITT Vi on Wed, 13 Jun 2007 16:43:21 GMT

View Forum Message <> Reply to Message

```
<eddedmondson@hotmail.com> wrote in message
news:5d7uh7F33rq9aU1@mid.individual.net...
>  metachronist <rkombiyil@gmail.com> wrote:
>>  ; Now I increase number of NaN's in the array :D
>
>  Note what you say here...
>
>>
>>  IDL> a=[6.2,12.5,14.1,0.,!values.f_nan,22,!values.f_nan]
>>  IDL> print,max(a)
>>       22.0000
>>  IDL> print,min(a)
>>       0.00000
>>
>>  ; go on, repeat this (it is 00:23 where I am @, so CARPE NOCTEM! ) :-P
>>
>
>  and here
>
>>  IDL>  a=[!values.f_nan,6.2,12.5,14.1,0.,!values.f_nan,22,!values.f _nan]
>
>  but note that now the first value is an NaN. I think the change has
>  more to do with this - the positioning of the NaN - than the actual
>  number of NaNs in your array.
>
>>  If the MAX function is run on an array containing NaN values and the
>>  NAN keyword is not set, an invalid result will occur.
>
>  But yes, your getting invalid results, which isn't surprising if you
>  put in invalid data, so best to check for it before you do it.
>
>  --
>  Edd
```

Another item of note; you may well get differing results across different
platforms if you execute MIN, MAX, TOTAL, MOMENT, etc., on
the same array containing NANs if you fail to use the NAN keyword.
My rule of thumb is to always use the NAN keyword when performing statistics
like these on derived floating point or double precision values when I
know there's not a guarantee that my floats will be bounded.

Jim P.