Subject: Re: Oh what is wrong with WHERE() ?
Posted by David Fanning on Sat, 29 Apr 2006 15:14:47 GMT
View Forum Message <> Reply to Message

Sheldon writes:

> Can someone find the disappearing pixel here? I have been trying and
> can't seem to find it.

You should have a read about the razor's edge:

http://www.dfanning.com/math_tips/razoredge.html

And think about how that might apply to a WHERE function
that is trying to find a floating point value!

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/

Subject: Re: Oh what is wrong with WHERE() ?
Posted by Liberum on Sat, 29 Apr 2006 15:35:13 GMT
View Forum Message <> Reply to Message

Dam David,

What the hell did I come across? This is way over my head. I have been
using double precision as it solved my initial rounding error problem
but when pixels disappear like that, logic becomes magic. I will use
integers instead and convert to float only when I am taking the mean.
Will study your write-up on the issue later today.

You know the funny thing is that I sat many days with this error and
thought that I must be stupid or blind not to see the simple mistake.
Thanks for restoring my faith in me.

Thanks,
Sheldon

Subject: Re: Oh what is wrong with WHERE() ?
Posted by news.qwest.net on Sat, 29 Apr 2006 15:49:40 GMT

"Sheldon" <shejo284@gmail.com> wrote in message
news:1146322284.261031.55760@e56g2000cwe.googlegroups.com...
...
> m[nodata_tot] = 255.0
> p[nodata_tot] = 255.0
> ; finding the area with real data check
> va_data = WHERE(p EQ 255.0, COMPLEMENT=vdatap, NCOMPLEMENT=chp)
> va_data = WHERE(m EQ 255.0, COMPLEMENT=vdatam, NCOMPLEMENT=chm)
> print, 'chp: ', chp, ' chm: ', chm
...
> chp: 32271 chm: 32272
> ////////////////////////////
>
> What happened to one pixel??

Hi Sheldon,
could you reduce this problem to a very short example?
(and with m and p small enough to actually print out)?

This looks very strange in that you apparently set both m and p
to have the same number of 255s, then get different results from
the where call.. David makes a good point about the folly of
using where for equality to floats, however for this example I don't
think that causes your problem.

Offhand I would say that p already has a single value of 255 that was
not in the indexes described by nodata_tot. Thus you already had a value
for 255 before your line "p[nodata_tot] = 255.0". This line had set
1443953 points to 255, leaving 32272 points that should be other,
however on of those 32272 points in fact had a point equal to 255,
thus leaving the chp value one short at 32721.

Cheers,
bob

PS you should think hard about converting your arrays to integers
before doing any equality test.

Subject: Re: Oh what is wrong with WHERE() ?
Posted by Liberum on Sat, 29 Apr 2006 16:15:52 GMT

Hi Bob,

This is good point that you make and I have thought about this at great length before. I have to set the arrays m and p to the equal amount of 255, i.e., no data values so as do a good pixel to pixel comparison over a period of one month. This means that I will do this m and p comparison about 355 times for one month. Now m and p varies when it comes to which pixels are valid and which are not. Some times p has all 255 values and the comparison cannot be done.
Now in the checks that I have placed in the program checks of the the count for 255 in p is greater than the 255 count in m and sets nodata_tot to the greater number:
IF count_m GE count_p THEN nodata_tot = nodata_m ELSE nodata_tot = nodata_p
The results showed that these two amounts were identical:
1443953
1443953
This is what I did to solve the problem your pointing out. Yet is does not work.
The numbers add up until the last where statement where the one pixel disappears. This is why I am trying now David's approach and using LONG instead of DOUBLE. Still a pixel disappears :(I think that it is something about this array as I tried another and it works. The thing is what? I can shrink the arrays but I will have to exclude some data. I did just that with the following check (each arrays should only have the values 255, 1 or 2):

```
r = where((p EQ 255.0 OR p EQ 1.0 OR p EQ 2.0), check_p,
COMPLEMENT=compp, NCOMPLEMENT=ncompp)
print, 'check_p:  ', check_p
IF ncompp GT 0 THEN stop, p[compp]
```

Yet this test is passed and ncompp is always zero. If I reduce the array, then what am I looking for in that case? I am lost.

/Sheldon

Subject: Re: Oh what is wrong with WHERE() ?
Posted by news.qwest.net on Mon, 01 May 2006 17:02:14 GMT
View Forum Message <> Reply to Message

"Sheldon" <shejo284@gmail.com> wrote in message
news:1146327351.938523.112920@e56g2000cwe.googlegroups.com.. .
> Hi Bob,

> IF count_m GE count_p THEN nodata_tot = nodata_m ELSE nodata_tot =
> nodata_p
> The results showed that these two amounts were identical:

> 1443953
> 1443953

Just a thought:

Even though the total is equal, it may be that notdata_m is not
exactly equal to nodata_p.
A short example of what I mean.
p = [0,1,0]
m = [1,0,0]

nodata_p = (where p eq 1, count_p)
nodata_m = (where m eq 1, count_m)
IF count_m GE count_p THEN nodata_tot = nodata_m ELSE nodata_tot =
nodata_p

p[nodata_tot] = 1
m[nodata_tot] = 1

so your results are now:
p = [1,1,0]
m = [1,0,0]

(Hence the missing pixel in p is due to the fact that your 'p' had
a '1' in a place outside of 'nodata_tot'.)

Cheers,
bob

Subject: Re: Oh what is wrong with WHERE() ?
Posted by Liberum on Thu, 04 May 2006 08:18:55 GMT
View Forum Message <> Reply to Message

Thanks Bob for pointing out that bug to me.
I will fix it right away.

/Sheldon
