
Subject: Re: Counting the number of irregular value in regular grid

Posted by [jkj](#) on Sun, 27 May 2007 11:25:46 GMT

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

On May 26, 1:04 am, Nick <junbin...@hotmail.com> wrote:

```
> I was trying to count the number of irregular data in regular grid.
> I made code but it's too slow to calculate my lots of data.
> Could you give me some advice to make this code more faster?
> Thank you, Jungbin
>
> ;1 X 1 Grid
> glon = -179.5 + findgen(360)
> glat = 89.5 - findgen(180)
>
>     for j = 0, 359 do begin
>         if [ (lon(x)-0.5 ge glon(j)) and (lon(x)+0.5 lt glon(j)) ]
> then begin
>         for k = 0, 179 do begin
>             if [ (lat(x)+0.5 le glat(k)) and (lat(x)-0.5 gt
> glat(k)) ] then begin
>                 if ( finite( value(j,k) ) eq 1 ) then nn(j,k) = nn(j,k)
> + 1
>             endif
>         endfor ;k
>     endif
```

I think your comparison logic is backwards (won't you always end up with a null set?), so I changed the comparisons, so my suggestion:

```
lowlon  = lon - 0.5
highlon = lon + 0.5
lowlat  = lat - 0.5
highlat = lat + 0.5
j_loop = where(lowlon[x] le glon and highlon[x] gt glon)
if(j_loop[0] ge 0)then begin
    k_loop = where(lowlat[x] le glat and highlat[x] gt glat)
    if(k_loop[0] ge 0)then begin
        ; ...now you have a list of j and k indices which
        ; meet your criteria
    endif
endif
```

...I'm actually not certain how to gracefully use the resulting j_loop and k_loop arrays to index the value and nn arrays... there must be some slick array method to do that, however

-Kevin
