
Subject: Re: Help converting different data sets into one data set and correcting for mismatch

Posted by [Craig Markwardt](#) on Sat, 03 Jan 2009 03:36:29 GMT

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

On Jan 2, 4:56 pm, mbwel...@gmail.com wrote:

> Hello,
>
> I have a bit of a dilemma that I was hoping someone could advise me
> on. I have two separate data sets, that are set up slightly
> differently, but show similar data and need to be displayed together.
>
> The first data set is of the form array1[6,32400], where positions 0-3
> represent the top, bottom, left and right values of a box with a range
> for top and bottom of -90 to 90 and left to right of 180 to 0. It's in
> latitude and longitude for a hemisphere in 1x1 deg cell sizes. i.e.
> 90,89,180,179...
>
> The second data set is the same range for latitude and longitude but
> it is instead a simple ascii grid with each cell position representing
> the latitude and longitude position. i.e. column 0, row 0 =
> 90,89,180,179 from the above example, I think.
>
> I need a way to map the second data set to the first data set so that
> I can compare them. The only way I know how to do it right now would be
> very time intensive and this time I don't have right now. The other
> issue (which I'm checking right now) is that they may not be exactly
> one to one and if that's the case, how might one stretch or shrink the
> second cell size in order to make them one to one?

Seems to me you need a reverse-lookup index to your ARRAY1 table. If you really have 1 degree spatial bins, then why not make a lookup table that is 180x180?

INDEX = LONARR(180,180)-1

Now fill them with the index values,

II = LINDGEN(npts) ;; npts is number of rows in ARRAY1
INDEX[ARRAY1[1,*],ARRAY1[3,*]] = II

Each cell of INDEX points back to the row number in ARRAY1 corresponding the desired geographic position. If a cell of INDEX contains -1, then there was no matching row in ARRAY1 to begin with.

Now do as many lookups as you want. For example,

JJ = INDEX[ARRAY2[0,*], ARRAY2[1,*]]

So now `ARRAY2[* ,i]` matches `ARRAY1[* ,JJ[i]]` (except of course when `JJ[i]` is -1)

Another way: If `ARRAY1` has all 180x180 entries -- no gaps -- then you could get away with something similar by pre-sorting `ARRAY1` so that they are in geographic order.

Craig
