Subject: Re: Nearest Neighbor ... again!
Posted by penteado on Thu, 08 Jul 2010 16:27:31 GMT
View Forum Message <> Reply to Message

On Jul 8, 10:40 am, Fabzi <fabien.mauss...@gmail.com> wrote:

```
> I am sorry, I did not make clear that grid1 and grid2 are two
> different irregular grids (lat1,lon1),(lat2,lon2). It is indeed
> classical, I know, and I feel quite embarrassed!
>
> So the code here:
>
>   n1 = n_elements(ilon)
>   triangulate, x1, y1, c ; Compute Delaunay triangulation
>   out = GRIDDATA(x1,y1, LINDGEN(n1), XOUT=x2, YOUT=y2, /NEAREST_N,
> TRIANGLES = c)
>
> Is giving me, for each point in Grid2, the index of the closest point
> in grid1. My question is, how to get the four closest points? The "bad
> trick" that I used with GRIDDATA won't work here...
```

If you need to interpolate from one irregular rectangular grid to the other, you can use bilinear() or interpolate(). There are several other options if the output grid is regular. If you really need the nearest points, voronoi() might serve.