Subject: Gridding to the Surface of a Sphere
Posted by David Fanning on Sun, 14 Apr 2013 16:20:09 GMT
View Forum Message <> Reply to Message

Folks,

Quite frequently you find yourself with randomly positioned data values that are associated with a latitude and longitude value. You often want to display this kind of data as a contour plot on a map projection. Traditionally, the Triangulate/Trigrid method is used to grid random data values into a 2D grid that can be contoured. And, there is provision in this method for gridding to the "surface of a sphere," which seems like a good thing to do for latitude/longitude data.

But, you would be gravely mistaken. :-)

Personally, I think the Triangulate/Trigrid gridding method for creating a grid on the surface of a sphere is tragically flawed. (Although I would be happy to discover otherwise.) I have outlined in some detail my reasons for thinking this in the following article:

http://www.idlcoyote.com/code_tips/sphericalgrid.php

I also illustrate how this can be done correctly by using GridData to do the gridding to the sphere, rather than the Triangulate/Trigrid method.

There is one strange thing about the GridData method that I don't understand and don't mention in the article. Maybe someone can help me with this. The GridData methods I illustrate (NaturalNeighbor and InverseDistance) require that I supply Delaunay triangles to the GridData program. If I create the triangles with Triangulate, all is well. If I create the triangles with QHull, the GridData program chokes. Does anyone have any insight into why that would be?

You can find code and data in the article if you care to fool around with this.

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thue. ("Perhaps thou speakest truth.")