Subject: How to grid pixel level data where latitude and longitude are 2D arrays
Posted by masterjedirobyn on Wed, 19 Jun 2013 19:31:06 GMT
View Forum Message <> Reply to Message

Hi,

I have been having a problem gridding a very large dataset which contains pixel level data into a gridded average. My data looks like this:

Lat and Lon are float [409,13248], the variables I wish to interpolate are also [409,13248]. I wish to grid these into arrays of [360,180] (1 degree spacing). Lon and Lat are irregular.

I have tried several methods of doing this. First, I looked at http://www.idlcoyote.com/code_tips/griddata.html and followed the process there, using qhull. Then, using the griddata command gave the following error: GRIDDATA: Value of Triangle index is out of allowed range. I then used triangulate instead of qhull and did not run into an error, but the result I got does not seem to be correct. Even if it was correct, the amount of time this calculation takes is huge; it runs overnight, and that's only on one file. I have many.

Does anyone know a faster, more memory efficient way of gridding data when your latitude and longitude are irregular and in 2D? This calculation typically freezes my machine with IDL using over 100% of the CPU. Is it possible that there could be some trick using Value_Locate? (Although from what I read, value_locate only works when your lon/lats are monotonically increasing/decreasing)

Thanks