Subject: Re: Matching Lats and Lons from two arrays
Posted by JD Smith on Fri, 05 Sep 2008 22:53:53 GMT
View Forum Message <> Reply to Message

I should point out that the MATCH_2D routine mentioned in the above thread and at:

http://www.dfanning.com/code_tips/matchlists.html

assumes euclidean distance measures apply, which of course is *not* strictly correct on a sphere. For points of latitude and longitude (or ra/dec), this will work for small patches of earth (or sky), but if you are near either pole, or cover appreciable areas, the 2D distance measure will falter.

You can "fake" a "pretty good" euclidean distance measure by pre-multiplying all longitudes by the cos of latitude. This works if the latitude range is not overly large, and if you don't cross any poles. Do do this correctly would require replacing the d=dx^2 + dy^2 with:

d = 2 asin(sqrt(sin(ddec/2)^2 + cos(dec1)cos(dec2)sin(dra/2)^2))

which is obviously much costlier to evaluate. More importantly, you'll need to think a bit about whether the stock "platte carre" projection (i.e. (ra, dec) -> (x,y)) is wasteful or not for your distribution of points on the sphere, or adversely affects your matching radius criterion. Adopting another projection prior to binning and running the histogram might offer gains in efficiency and more "uniform" distance performance (the issue being that square bins in a projected sphere do not have constant area). It should be straightforward to "wrap-around" the poles.

JD