Subject: Re: Matching Lats and Lons from two arrays
Posted by Conor on Tue, 26 Aug 2008 18:41:27 GMT
View Forum Message <> Reply to Message

On Aug 26, 11:47 am, wilsona <awils...@bigred.unl.edu> wrote:
> I have 2 seperate arrays of Latittudes and Longitudes.
> CS_LATLON(0,4607) is one latitude array and dlat(192,139) is the
> other
> CS_LATLON(1,4607) is one longitude array and dlon(192,139) is the
> other.
> I want to index through each element in both CS_LATLON arrays and
> find
> which point(s) in the dlat and dlong arrays are closest.
> I tried using nested loops but that gave me 12 million+ loops which
> is
> too many for my liking. I now am trying search2d
> NUM_PNTS = N_ELEMENTS(CS_LATLON(0, *)) - 1
>
> FOR J = 0, NUM_PNTS DO BEGIN
> CLOSE_LATS = SEARCH2D(dlat, 0, 0, CS_LATLON(0,J),
> CS_LATLON(0,J), INCREASE=0.5, $
> DECREASE=0.5)
> lat1 = CS_LATLON(0,J) * PI / 180.0
> FOR K = 0, CLOSE_LATS DO BEGIN
> lat2 = dlat(K) * PI / 180.0
> d_long = CS_LATLON(1,J) - dlon(K)) * PI / 180.0
> DISTANCE = 10800.0 / PI * acos(sin(lat1) * sin(lat2)
> +
> cos(lat1) * $
> cos(lat2) * cos(d_long))
> ENDFOR ; K
> ENDFOR ; J
> This is not working they way I would like. Any suggestions on this
> would be greatly appreciated.

I often have to match up two star fields, which is pretty much the
same thing. You can download the routine I use here:

astro.ufl.edu/~cmancone/pros/qfind.pro

Here's the source. It basically just uses histogram to bin everyhing
and then searches one bin left and right:

function qfind,x1,y1,x2,y2,posshift=posshift

if n_elements(posshift) eq 0 then posshift = 1

n1 = n_elements(x1)

```
n2 = n_elements(x2)

; this is where I'll store the result
res = lonarr(2,n1)

; this mask list will tell us if an entry is from list one or list two
allinds = lindgen(n2)

; the histogram will tell us how many stars left and right we have to
check
hist = histogram(x2,binsize=posshift,omin=minx,reverse_indices=ri)

; calculate which bin each x went into
bins = floor((x1-minx)/posshift)>0
nbins=n_elements(hist)

f = 0L
for i=0L,n1-1 do begin
 ; figure out which bin this star is in
 bin = bins[i]

 ; adjust the indexes accordingly
 inds = ri[ri[(bin-1)>0]:ri[((bin+2)<nbins)]-1]

 ; calculate the distance from this star to its neighbors
 dist = sqrt( (x2[inds]-x1[i])^2 + (y2[inds]-y1[i])^2 )

 ; get the closest star within posshift that is from the second list
 mindist = min( dist, wm )

 ; no stars found
 if mindist gt posshift then continue

 ; record result
 res[*,f] = [i,inds[wm]]
 ++f
endfor

if f eq 0 then return,-1

; get rid of any blank entries
res = res[*,0:f-1]


return,res

end
```