Subject: Re: speher keyword of qhull Posted by David Fanning on Wed, 08 Nov 2006 14:47:52 GMT

View Forum Message <> Reply to Message

sebinjapan writes:

- > Does anyone knows was is returned in the sphere variable when using:
- > qhull, v0, v1, tr, /delaunay, sphere = s

As I read the documentation, it seems to me SPHERE is an input variable, with a value of 0 or 1.:-)

Cheers.

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: http://www.dfanning.com/

Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: speher keyword of qhull

Posted by Andy Heaps on Wed, 08 Nov 2006 16:12:53 GMT

View Forum Message <> Reply to Message

Hello Sebinjapan,

an interesting question for me as I was puzzling over the same point last week. I don't have any real answers but the following should maybe provide some illumination on the subject:

I have a strange grid of points in longitude and latitude on a globe:

LATS FLOAT = Array[722, 511]

LONGS FLOAT = Array[722, 511]

TEMPERATURE FLOAT = Array[722,511]

The longitudes (LONGS) go from -179.997 to 179.999 and the latitudes (LATS) go from -77.0105 to 89.8854. TEMPERATURE is the ocean surface temperature on this grid.

The location of the longitudes and latitudes are very irregular so triangulation data is needed to pass to the CONTOUR command. The TRIANGULATE procedure works and CONTOUR produces a pretty decent plot.

There is a wrapping problem, however, at the dateline seen as a line from the north to south poles. Adding the /SPHERE to the TRANGULATE call produces:

% TRIANGULATE: Points are co-linear, no solution.

No hope here then so I then used QHULL on the suggestion of a colleague:

```
If I use QHULL, longs, lats, tri, /DELAUNAY, SPHERE =1
I get:
% QHULL: Expression must be named variable in this context: <INT
( 1)>.

Okay, let's used a named variable:
QHULL, longs, lats, tri, SPHERE = s
HELP, s, tri
S DOUBLE = Array[3, 368943]
TRI LONG = Array[3, 733642]
```

Using the triangulation data TRI with CONTOUR produces a perfect plot with no dateline missing data.

So we're back to your original question - what is returned by SPHERE=s? THE IDL reference guide for version 6.0 QHULL says: DELAUNAY Performs a Delaunay triangulation and returns the vertex indicies of the resulting polyhedra; otherwise, the convex hull of the data are returned.

SPHERE Computes the Delaunay triangulation of the points which lie on the surface of a sphere. The V0 argument contains the longitude, in degrees, and V1 contains the latitude, in degrees, of each point.

It looks like the IDL documentation could be better just here. Does SPHERE=s imply a Delaunay triangulation and that s is the convex hull of the data? Printing off some values of s I get:

```
PRINT, s(*, 0:3)
```

```
PRINT, 'min, max of s are', MIN(s), MAX(s)
0.066654513 0.21466261 -0.97441118
0.064778714 0.21523610 -0.97441118
0.062897982 0.21579320 -0.97441118
```

0.061012460 0.21633386 -0.97441118 min, max of s are -0.99999048 0.99999800

Is this the sort of thing you're trying to work out as well?

Cheers Andy

David Fanning wrote:

> sebinjapan writes:

>

>> Does anyone knows was is returned in the sphere variable when using:

```
>> qhull, v0, v1, tr, /delaunav, sphere = s
>
> As I read the documentation, it seems to me SPHERE is
> an input variable, with a value of 0 or 1.:-)
>
> Cheers,
>
> David
```

Subject: Re: speher keyword of ghull Posted by David Fanning on Wed, 08 Nov 2006 16:34:03 GMT View Forum Message <> Reply to Message

Andy Heaps writes:

- an interesting question for me as I was puzzling over the
- > same point last week. I don't have any real answers but the following
- > should maybe provide some illumination on the subject:

Yikes! Here is a project for someone. (Not me. I'm still recovering from serving as a radio reporter doing election coverage last night. You have NO idea how hard it is to write a coherent story at 2:30 AM when you have a deadline in 5 minutes. Most nerve-wracking experience of my life!)

Cheers,

David

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: http://www.dfanning.com/

Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: speher keyword of ghull Posted by Andy Heaps on Thu, 09 Nov 2006 15:23:22 GMT View Forum Message <> Reply to Message

ITT-Vis help desk have resolved the guery:

The SPHERE keyword to QHULL will return the Delaunay triangulation of the points which lie on the surface of a sphere. Consider the following example:

```
> pro test
    seed = 1
>
    n = 50
>
    lon = randomu(seed, n) * 360. - 180.
>
    lat = randomu(seed, n) * 180. - 90.
>
    f = \sin(lon * !dtor)^2 * \cos(lat * !dtor)
>
>
    ghull, Ion, lat, gtr, $
>
       SPHERE = q_sphere, /DELAUNAY
>
>
    qpolys = lonarr(4,(size(qtr, /DIM))[1])
>
    qpolys[0,*] = 3
>
    qpolys[1:3,*] = qtr
>
>
    g poly = obj new('idlgrpolygon', $
>
       g sphere, COLOR = [0,255,0], $
>
       POLYGONS = apolys, STYLE = 2)
>
>
    xobjview, q_poly, XSIZE = 400, YSIZE = 400, $
>
       XOFF = 410, YOFF = 450, $
>
       TITLE = 'QHULL triangulation'
>
>
> end
```

Subject: Re: speher keyword of qhull Posted by Haje Korth on Fri, 10 Nov 2006 17:55:18 GMT View Forum Message <> Reply to Message

Andy,

yes I hope one of these days, ITTVIS provide a detailed tutorial on qhull/griddata, describing the right combination of keywords to get a particular job done. This should benefit anyone involved in Earth and/or space sciences, which is a pretty broad audience.

I got bit for the colinear-points problem with triangulate a few years ago because I wanted to interpolate data on a sphere which came on a regular grid. On regular grids the probability that three point lie along a great circle is pretty large, thus resulting in the colinear-point error message in triangulate.

Haje

Andy Heaps wrote:

```
> Hello Sebinjapan,
          an interesting question for me as I was puzzling over the
>
> same point last week. I don't have any real answers but the following
 should maybe provide some illumination on the subject:
>
> I have a strange grid of points in longitude and latitude on a globe:
                        = Array[722, 511]
> LATS
               FLOAT
> LONGS
                FLOAT
                          = Array[722, 511]
> TEMPERATURE
                       FLOAT = Array[722,511]
> The longitudes (LONGS) go from -179.997 to 179.999 and the latitudes
> (LATS) go from -77.0105 to 89.8854. TEMPERATURE is the ocean surface
> temperature on this grid.
>
 The location of the longitudes and latitudes are very irregular so
> triangulation data is needed to pass to the CONTOUR command. The
> TRIANGULATE procedure works and CONTOUR produces a pretty decent plot.
  There is a wrapping problem, however, at the dateline seen as a
> line from the north to south poles. Adding the /SPHERE to the
> TRANGULATE call produces:
> % TRIANGULATE: Points are co-linear, no solution.
>
> No hope here then so I then used QHULL on the suggestion of a colleague:
> If I use QHULL, longs, lats, tri, /DELAUNAY, SPHERE =1
> % QHULL: Expression must be named variable in this context: <INT
> (
        1)>.
>
> Okay, let's used a named variable:
> QHULL, longs, lats, tri, SPHERE = s
> HELP, s, tri
 S
            DOUBLE = Array[3, 368943]
> TRI
             LONG
                       = Array[3, 733642]
>
 Using the triangulation data TRI with CONTOUR produces a perfect plot
> with no dateline missing data.
>
> So we're back to your original question - what is returned by
> SPHERE=s? THE IDL reference guide for version 6.0 QHULL says:
> DELAUNAY Performs a Delaunay triangulation and returns the vertex
> indicies of the resulting polyhedra; otherwise, the convex hull of the
> data are returned.
> SPHERE Computes the Delaunay triangulation of the points which lie on
> the surface of a sphere. The V0 argument contains the longitude, in
> degrees, and V1 contains the latitude, in degrees, of each point.
>
> It looks like the IDL documentation could be better just here. Does
> SPHERE=s imply a Delaunay triangulation and that s is the convex hull
> of the data? Printing off some values of s I get:
```

```
> PRINT, s(*, 0:3)
  PRINT, 'min, max of s are', MIN(s), MAX(s)
      0.066654513
                      0.21466261
                                    -0.97441118
>
      0.064778714
                      0.21523610
                                    -0.97441118
>
      0.062897982
                      0.21579320
                                    -0.97441118
>
>
      0.061012460
                      0.21633386
                                    -0.97441118
> min, max of s are
                      -0.99999048
                                     0.99999800
>
>
  Is this the sort of thing you're trying to work out as well?
>
>
       Cheers
>
       Andy
>
>
>
>
>
    David Fanning wrote:
>
>> sebinjapan writes:
>>
>>> Does anyone knows was is returned in the sphere variable when using:
>>> qhull, v0, v1, tr, /delaunay, sphere = s
>>
>> As I read the documentation, it seems to me SPHERE is
>> an input variable, with a value of 0 or 1. :-)
>>
>> Cheers,
>>
>> David
```