Subject: Re: Griddata and Contour questions
Posted by Conor on Wed, 11 Jul 2007 16:17:40 GMT

View Forum Message <> Reply to Message

On Jul 11, 11:09 am, julia.kamenet...@gmail.com wrote:
> I'm somewhat new to IDL, and attempting to create contour plots of
> sets of astronomical data. We'd like to interpolate data points based
> on the known Gaussian beam size. It seems to me like the only routine
> in IDL to accomplish this would be the Kriging method of GRIDDATA
> using Gaussian covariance? I have a few questions about this...
>
> 1) In what units are parameters such as d and R (as specified in the
> help section on the variogram)? I am assuming that they are in units
> of spacing between adjacent grid points (ie, the distance from one to
> the next is equal to one), but please let me know if this is
> incorrect.
>
> 2) When I use this method, I receive an array that contains values way
> outside the range of the data. (My data is from about 20 to 200, and
> the returned array has values from about negative to positive
> 12,000!) I cannot get correct results, even with changing around my
> range value in the variogram, dimensions, or if I triangulate the data
> before.
>
> Some examples of ones I've tried:
>
> KRIGGRID2=GRIDDATA(158_ASCII.L,158_ASCII.B,158_ASCII.INTENSI TY, /
> KRIGING, DIMENSION=[500,386], VARIOGRAM=[3,24,0,1])
>
> KRIGGRID=GRIDDATA(158_ASCII.L,ASCII.B,158_ASCII.INTENSITY, /KRIGING,
> TRIANGLES=TR, ANISOTROPY=[1,1], DIMENSION=[500,386],
> VARIOGRAM=[3,24,0,1], SEARCH_ELLIPSE=20)
>
> Any idea what I'm doing wrong, and why I'm getting values that are
> clearly not interpolations of the defined points? I am able to grid
> using other methods and get reasonable contours with this data, but I
> cannot seem to get this method to work.
>
> 3) Once I have this data gridded, how can I then display it on the
> original coordinate axes, rather than axes that are just numbered by
> indices of the gridded array?
>
> It would be preferable to do this all using the iContour gridding
> wizard (as you can tell, I have little IDL command line experience),
> but as of yet I am not sure that any of the options in the wizard will
> interpolate the right way. If there is a suitable option, or one that
> is "close enough", I'd love to hear about it!
>

> Thank you for your time.

I have no idea if this is what you need, or what you want. Maybe it is though, so I can show you what I do. I often use interpolation for all sorts of things. I use a combination of triangulate and trigrid, as recommended by online_help. I have a (command line) program I wrote for this along time ago. The problem is that it was long enough ago that I don't remember the details of how it actually works. So I have to give you the program bereft of any useful comments. There should be some helpful hints in the online_help for interpolate though.

```
PRO inter,x,y,z,xval,yval,res, noextrapolate=noextrapolate

xx = x
yy = y
zz = z
triangulate, xx, yy, tr, b

res = fltarr(n_elements(xval))

FOR j = 0L, n_elements(xval)-1 DO BEGIN
   xi = xval[j]
   yi = yval[j]
   p = 0.1
   if keyword_set( noextrapolate ) then array = trigrid(xx, yy, zz,
tr, [0, 0], [xi-p, yi-p, xi+p, yi+p], NX=101, NY=101) else $
     array = trigrid(xx, yy, zz, tr, EXTRAPOLATE=b, [0, 0], [xi-p, yi-
p, xi+p, yi+p], NX=101, NY=101)
   res[j] = array[50, 50]

ENDFOR

END
```

You would call this by putting all the x data you want to interpolate between in x, all the corresponding y data in y, and all their corresponding values in z. Then, the values you want to interpolate at go in xval and yval, and the result comes out in res. So you could do something like this:

```
x = [findgen(10),findgen(10)]
y = [sin(findgen(10)),sin(findgen(10))+1]
z = [fltarr(10),fltarr(10)+1]
xval = [2.0,4.0,6.0]
yval = [1.5,0.0,0.0]
```

```
inter,x,y,z,xval,yval,res
```

In this case, the first sin curve has a value of 0, the second sine curve has a value of 1, and the values in res are the interpolated values between the two curves at the points [xval,yval]