Subject: How to represent the spatial distribution of a parameter
Posted by duxiyu@gmail.com on Wed, 10 Dec 2008 09:16:29 GMT
View Forum Message <> Reply to Message

Dear all,

I have three array X, Y and V.
X and Y represent the position, and they are not distributed uniformly
on the X-Y plane.
V represent the parameter values corresponding to their position.

I want to take some 2D figures which can represent the spatial
distribution of the parameter values.
Is there any recommended representation?

I find some figures in a paper which is shown in the web link below.
http://lh5.ggpht.com/_NSjLmdFf0ac/ST9_aR0A8zI/AAAAAAAAAFA/8H 3m5pkbLO0/1.png
http://lh6.ggpht.com/_NSjLmdFf0ac/ST9_a51bQII/AAAAAAAAAFI/2Z oP59xdyYw/2.png
He divided the X-Y plane into small grids, and used the mean of the
parameter at the points, which are located in each grid, to represent
the parameter value in that grid.
I wander whether there is a simple way to do this in IDL.
For example, I want to use the 0.1*0.1 grid to divide the X-Y plane.
How do I plot the figures like that from X, Y and V?

Best regards,
Du

Subject: Re: How to represent the spatial distribution of a parameter
Posted by Jean H. on Wed, 10 Dec 2008 13:05:22 GMT
View Forum Message <> Reply to Message

Hi,

I did something similar a while ago... here is part of it:

```
newX = Xdata / CellSizeX    ;Agregate the data
newY = Ydata / CellSizeY

nbCol = ceil((maxX+1.0) / cellSizeX)
nbRow = ceil((maxY+1.0) / cellSizeY)
nPoint = n_elements(newX)

image = lonarr(nbCol, nbRow)
nbPointsXY = lonarr(nbCol, nbRow)

for i = 0L, nPoint-1 do begin
```

```
    image[newX[i], newY[i]] += v[i]
    nbPointsXY[newX[i], newY[i]] += 1
endfor

image /= nbPointsXY   ;do the average

tvscl, image
```

Jean

Subject: Re: How to represent the spatial distribution of a parameter
Posted by duxiyu@gmail.com on Wed, 10 Dec 2008 17:37:26 GMT
View Forum Message <> Reply to Message

Excuse me!
I do not understand your routine.
Could you show me an example? or give me some detailed explanations?

Du

On Dec 10, 9:05 pm, "Jean H." <jghas...@DELTHIS.ucalgary.ANDTHIS.ca>
wrote:
> Hi,
>
> I did something similar a while ago... here is part of it:
>
> newX = Xdata / CellSizeX   ;Agregate the data
> newY = Ydata / CellSizeY
>
> nbCol = ceil((maxX+1.0) / cellSizeX)
> nbRow = ceil((maxY+1.0) / cellSizeY)
> nPoint = n_elements(newX)
>
> image = lonarr(nbCol, nbRow)
> nbPointsXY = lonarr(nbCol, nbRow)
>
> for i = 0L, nPoint-1 do begin
>   image[newX[i], newY[i]] += v[i]
>   nbPointsXY[newX[i], newY[i]] += 1
> endfor
>
> image /= nbPointsXY   ;do the average
>
> tvscl, image
>

> Jean

Subject: Re: How to represent the spatial distribution of a parameter
Posted by Giorgio on Wed, 10 Dec 2008 18:59:27 GMT
View Forum Message <> Reply to Message

One way is to define a grid from the x and y arrays and then
interpolate the values V to match the grid.
Please see David's website: http://www.dfanning.com/code_tips/griddata.html

Giorgio

Subject: Re: How to represent the spatial distribution of a parameter
Posted by Jean H. on Wed, 10 Dec 2008 21:20:57 GMT
View Forum Message <> Reply to Message

Hi Du,

ok, so you have a set of X and Y coordinates

You want to plot them in a regular grid, so to know in which cell to
plot the corresponding data, you divide the coords by the cell size
ex: X = [1,2,2.5]
cellSizeX = 0.5

NewX = X/cellSizeX
==> 2.00000 4.00000 5.00000
==> on the grid, your point will be in cell 2,4 and 5
grid: 0-0.5-1-1.5-2-2.5-3-3.5-4-4.5-5 etc

Now, you want to know the size of your grid, so you do
nbCol = ceil((maxX+1.0) / cellSizeX)

this assume that the 1st cell of the grid is coord 0;0, do maxX-minX if not.
So basically, you take the highest X value and divide by the cell size.
Up-round this, so that you are sure that the new coord is within your grid.

Create a grid (an array) with the number of columns and rows. Create a
"count" layer, that you will use for doing the average.

For each point that you have, using the new X and Y coordinate, you
1)add the value V to the image,
2) add 1 to the count layer

then, simply divide the image by the count, and you get the average

value in each pixel.

Jean

> Excuse me!
> I do not understand your routine.
> Could you show me an example? or give me some detailed explanations?
>
> Du
>
>
> On Dec 10, 9:05 pm, "Jean H." <jghas...@DELTHIS.ucalgary.ANDTHIS.ca>
> wrote:
>> Hi,
>>
>> I did something similar a while ago... here is part of it:
>>
>> newX = Xdata / CellSizeX ;Agregate the data
>> newY = Ydata / CellSizeY
>>
>> nbCol = ceil((maxX+1.0) / cellSizeX)
>> nbRow = ceil((maxY+1.0) / cellSizeY)
>> nPoint = n_elements(newX)
>>
>> image = lonarr(nbCol, nbRow)
>> nbPointsXY = lonarr(nbCol, nbRow)
>>
>> for i = 0L, nPoint-1 do begin
>> image[newX[i], newY[i]] += v[i]
>> nbPointsXY[newX[i], newY[i]] += 1
>> endfor
>>
>> image /= nbPointsXY ;do the average
>>
>> tvscl, image
>>
>> Jean
>

Subject: Re: How to represent the spatial distribution of a parameter
Posted by Jeremy Bailin on Thu, 11 Dec 2008 15:34:42 GMT
View Forum Message <> Reply to Message

On Dec 10, 8:05 am, "Jean H." <jghas...@DELTHIS.ucalgary.ANDTHIS.ca>
wrote:
> Hi,

>
> I did something similar a while ago... here is part of it:
>
> newX = Xdata / CellSizeX ;Agregate the data
> newY = Ydata / CellSizeY
>
> nbCol = ceil((maxX+1.0) / cellSizeX)
> nbRow = ceil((maxY+1.0) / cellSizeY)
> nPoint = n_elements(newX)
>
> image = lonarr(nbCol, nbRow)
> nbPointsXY = lonarr(nbCol, nbRow)
>
> for i = 0L, nPoint-1 do begin
> image[newX[i], newY[i]] += v[i]
> nbPointsXY[newX[i], newY[i]] += 1
> endfor
>
> image /= nbPointsXY ;do the average
>
> tvscl, image
>
> Jean

If there are on average lots of points per grid cell, it would be more efficient to use the reverse indices in David's HIST_ND:

```
histimage = hist_nd(transpose([[X],[Y]], [binX,binY], $
 min=[minX,minY], max=[maxX,maxY], reverse_indices=hiri)
meanimage = fltarr(size(histimage,/dimen))
for i=0l,n_elements(histimage)-1 do if histimage[i] gt 0 then $
  meanimage[i] = mean(V[hiri[hiri[i]:hiri[i+1]-1]])
```

If performance is still a serious issue, you should go read http://www.dfanning.com/code_tips/drizzling.html (it's effectively the same thing - you just need to divide by histimage at the end). In fact, go read it anyway - it's very illuminating. :-)=

-Jeremy.

Subject: Re: How to represent the spatial distribution of a parameter
Posted by Jeremy Bailin on Thu, 11 Dec 2008 15:36:17 GMT
View Forum Message <> Reply to Message

On Dec 11, 10:34 am, Jeremy Bailin <astroco...@gmail.com> wrote:
> On Dec 10, 8:05 am, "Jean H." <jghas...@DELTHIS.ucalgary.ANDTHIS.ca>
> wrote:

>
>
>
>> Hi,
>
>> I did something similar a while ago... here is part of it:
>
>> newX = Xdata / CellSizeX ;Agregate the data
>> newY = Ydata / CellSizeY
>
>> nbCol = ceil((maxX+1.0) / cellSizeX)
>> nbRow = ceil((maxY+1.0) / cellSizeY)
>> nPoint = n_elements(newX)
>
>> image = lonarr(nbCol, nbRow)
>> nbPointsXY = lonarr(nbCol, nbRow)
>
>> for i = 0L, nPoint-1 do begin
>> image[newX[i], newY[i]] += v[i]
>> nbPointsXY[newX[i], newY[i]] += 1
>> endfor
>
>> image /= nbPointsXY ;do the average
>
>> tvscl, image
>
>> Jean
>
> If there are on average lots of points per grid cell, it would be more
> efficient to use the reverse indices in David's HIST_ND:
>
> histimage = hist_nd(transpose([[X],[Y]], [binX,binY], $
> min=[minX,minY], max=[maxX,maxY], reverse_indices=hiri)
> meanimage = fltarr(size(histimage,/dimen))
> for i=0l,n_elements(histimage)-1 do if histimage[i] gt 0 then $
> meanimage[i] = mean(V[hiri[hiri[i]:hiri[i+1]-1]])
>
> If performance is still a serious issue, you should go
readhttp://www.dfanning.com/code_tips/drizzling.html(it's effectively the
> same thing - you just need to divide by histimage at the end). In
> fact, go read it anyway - it's very illuminating. :-)=
>
> -Jeremy.

...although, to answer the original poster's question, I'd agree with
Giorgio - your easiest option is probably to use GRIDDATA.

-Jeremy.