Subject: Re: "Correct" Data Philosophy
Posted by Laura on Mon, 21 Dec 2009 17:19:14 GMT
View Forum Message <> Reply to Message

On Dec 18, 3:22 pm, David Fanning <n...@dfanning.com> wrote:
> Laura writes:
>> GRID_TPS use "thin plate spline" as the interpolating function, which
>> I used a lot in 3D modeling before moving to IDL. They can estimate
>> the values using data samples on irregular grid (which means as long
>> as you know the sample data locations and values, you are fine, they
>> don't need to be on regular grids).
>
> OK, I'm thinking of this problem sort of like that time I missed
> an easy overhead and lost to that smart-aleck young kid and
> came home and maybe pushed the door a little too hard with my
> tennis bag and there was a bit of a hole in the dry wall.
> "Thin Plate Spline" sounds like the wire gauze I had to
> use to repair the darn thing. Is it like that?
>
> If so, how could I use it to "repair" some dropped
> data points in the center of my image, for example?
>

Here's how I use the GRID_TPS in IDL:

In my example, I have the original data on very sparse grids with some missing values, but I want to interpolate the data at a higher resolution:

```
FUNCTION TPSInterpolation, org_data, missValue, newDimx, newDimy,
minX, minY, maxX, maxY
; org_data is the original data on a regular grid located at (or
bounded by) [minX, minY, maxX, maxY]
; missValue is the filled-in value in org_data indicating the real
value is missing there
; newDimx and newDimy are the dimensions of the resulting data,
; if you just want to fill in the value on orginal grids, I think you
can use the dimensions of the org_data

  data =fltarr(newDimx, newDimy)
  orgInd = where(org_data NE missValue, count)
  if (count EQ 0) then begin
    data = congrid(org_data, newDimx, newDimy)
    return, data
  endif

  sz=size(org_data)
  dimx = sz[1]
```

```
  dimy = sz[2]

  xSpan = maxX-minX
  ySpan = maxY-minY

  dx0=xSpan/(dimx-1)
  dy0=ySpan/(dimy-1)

  xVector=findgen(dimx)*dx0 + minX ;xlocation
  yVector=findgen(dimy)*dy0 + minY ;ylocation
  indices = array_indices(org_data, orgInd)

  xPos = xVector[indices[0,*]] ;Xp
  yPos = yVector[indices[1,*]] ;Yp
  values=org_data(orgInd) ; Values

  dx=xSpan/(newDimy-1)
  dy=ySpan/(newDimy-1)

  data = grid_tps(xPos, yPos, values, COEFFICIENTS = coef, NGRID=
[newDimx, newDimy], START=[minX, minY], DELTA=[dx,dy])
  return, data
END
```

Note: If you want to use MIN_CURVE_SURF then the call function can be set as:

```
data = min_curve_surf(values, xPos, yPos, GS=[dx, dy], BOUNDS = [minX,
minY, maxX, maxY], NX=newDimx, NY=newDimy)
```

If you want to fill in the missing value in a large array, I think dividing them into blocks and working on each block separately will be a good idea.

Hope this helps.

Laura