Subject: Re: Interpolation
Posted by ben.bighair on Sat, 12 Apr 2008 15:57:08 GMT
View Forum Message <> Reply to Message

On Apr 11, 11:05 pm, tarequea...@gmail.com wrote:
> Hello All (IDL Gods),
>
> I am back with yet another problem.
> I know...I know...its friday night. I apologize for that. But I am
> really stuck here for a while.
>
> The problem:
>
> In a certain part of my code I need to do interpolation. The data that
> I am dealing with are from a XY grid. I need to convert them to polar
> coordinate. So, what I do is following:
>
> a. I generate a radius vector r_vec and a theta array containing
> values from zero to 2 pi.
> b. Now use the simple polar-to-rectangular coordinate transform i.e. x
> = r cos(theta) etc.
> c. Using these values I have a xy grid generated through a known r-
> theta values.
> d. Now think of superimposing the new xy grid(lets call it x'y') on
> to the old xy grid which contains the real data.
> e. This is the part where I need interpolation. I do interpolations to
> get the x'y' values from the xy points.
>
> So here's the question:
>
> ----- Is there any elegant way of doing this coordinate
> transformation ? (And in case you are thinking, "well you already have
> the xy data, so why not just convert to r-theta?", I have to say that
> the interpolation method actually gives me a way nicer dataset).
>
> My 2nd trouble is, and this is probably the biggest and dumbest
> problem for me.
>
> ----- i was playing around with several interpolation routines from
> IDl. My boss's suggestion was to use 'bilinear'.
> but I thought to give others' a shot too. Problem is, when I am
> done with interpolation, result is nothing like what I was
> expecting. A run down version of the code is shown below:
>
> ======================start of code==============================
>
> Nth= 10.
> dth= 1/Nth

```
>  r_vec= findgen(Nth)/Nth
>  theta_vec = findgen(Nth)/Nth  * 2.*!Pi
>
>  for i=0L,Nth - 1 do begin
>
>  x[i]= r_vec[i]* cos(theta_vec)
>
>  endfor
>
>  for j=0L,Nth - 1 do begin
>
>  y[j]= r_vec[j]* sin(theta_vec)
>
>  endfor
>
>  ;print,y
>
>  ;plot,x,y
>
>   ------------------------------------------------------------ -----------
>  Now I create the 'main' dataset on which I am going to use
>  interpolation scheme.
>
>  rr = findgen(20.)/30.
>  tht = findgen(20.)/30.  *2*!Pi
>
>  m = fltarr(20,20)
>
>  for j=0,19 do begin
>     for i=0,19 do begin
>
>     m[i,j] = rr[i]*cos(tht[j]) + 5.*rr[i]*sin(tht[j])
>
>     ;print,i
>     endfor
>  endfor
>
>  m_p=bilinear(m,x,y)
>
>  End
>  ======================End of Code==================================
>
>  The problem is, as I mentioned above, when I plot m and the
>  interpolated m_p, they do not look like similar at all.
>
>  Any help will be greatly appreciated.
>
>  Thanks in advance.
```

>
> ~tareque

Hi,

Coordinate transforms are very easy with CV_COORD().

I don't understand where you want to go with the interpolation. The input coordinates to BILINEAR are rectangular and formed as indexed based (as in subscript indices in X and Y.) As near as I can reconstruct, X ranges from -0.500000 to 0.728115 and Y ranges from -0.760845 to 0.285317. My reconstruction of your code is below.

Cheers,
Ben

```
PRO tareque

Nth= 10L
dth= 1.0/Nth
r_vec= findgen(Nth)/Nth
theta_vec = findgen(Nth)/Nth  * 2.*!Pi
x = fltarr(nth)
y = fltarr(nth)

for i=0L,Nth - 1 do begin

x[i]= r_vec[i]* cos(theta_vec[i])

endfor

for j=0L,Nth - 1 do begin

y[j]= r_vec[j]* sin(theta_vec[j])

endfor

;print,y

;plot,x,y

 ;------------------------------------------------------------ ------------
;Now I create the 'main' dataset on which I am going to use
;interpolation scheme.

rr = findgen(20.)/30.
tht = findgen(20.)/30.  *2*!Pi
```

```
m = fltarr(20,20)

for j=0,19 do begin
   for i=0,19 do begin

   m[i,j] = rr[i]*cos(tht[j]) + 5.*rr[i]*sin(tht[j])

   ;print,i
   endfor
endfor

m_p=bilinear(m,x,y)

STOP
End
```