Subject: Interpolation
Posted by jun on Thu, 02 Jun 1994 17:44:56 GMT
View Forum Message <> Reply to Message

I am using IDL to process and display 3D data. At some stage, I have to
interpolate a volume p=p(x,y,z) or a surface z=z(x,y) defined on an irregular
grid onto a regular grid. Does any one know any IDL program that has this
capability?

Thanks in advance.

jun@eureka.columbia.edu

Subject: Re: Interpolation
Posted by stl on Fri, 03 Jun 1994 08:14:33 GMT
View Forum Message <> Reply to Message

In article <2sl5qo$h1q@apakabar.cc.columbia.edu> jun@eureka.phys.columbia.edu (Jun Xu)
writes:
> I am using IDL to process and display 3D data. At some stage, I have to
> interpolate a volume p=p(x,y,z) or a surface z=z(x,y) defined on an irregular
> grid onto a regular grid. Does any one know any IDL program that has this
> capability?

hi,

Well, it depends on how mathimaticly correct you need this stuff. I
believe you could do some very very complicated stuff here, but there is
a relatively simple solution.
Use the trigrid and triangulate procedures in IDL. In the Reference
manual, there is a good example in the Trigrid section of how this
works.

-hope this help (sorry, but I am no mathimatician. Cannot spell
fro squate either!)

-stephen

--
Stephen C Strebel / SKI TO DIE
stl@maz.sma.ch / and
Swiss Meteorological Institute, Zuerich / LIVE TO TELL ABOUT IT
01 256 93 85 / (and pray for snow)

Subject: Re: Interpolation

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
```

Subject: Re: Interpolation
Posted by tarequeaziz on Sat, 12 Apr 2008 20:24:26 GMT
View Forum Message <> Reply to Message

On Apr 12, 11:57 am, "ben.bighair" <ben.bigh...@gmail.com> wrote:
> 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
```

hi Ben,

Thanks for getting back to me on this.

I guess I could not clearly state my problem before. So, I am going to give it another try:

The data I deal with comes from a ccd camera and this goes some tinkering and tweaking (to account for background noise and stuff). Then this 'processed' data is in need for interpolation. why? well the reason being this, that these processed data are actually in XY

frame.
We want them to be on a polar coordinate. The reason I cannot use
'cv_coord' is because in that case I will get the data in a r-theta
plane which is
directly correlated to the experimental XY grid. If we use
interpolation, then we can be on a XY grid (yes, it is XY grid) which
is carefully 'designed' according to
our 'own' r-theta values. So in this way we have more leverage over
data manipulation.

Was it any clearer than before?

And one more thing, I could not find much changes in your version
except for an inclusion of 'STOP'. Did i miss something here??

Thanks again for your time.
Much appreciated !

~ Tareque

Subject: Re: Interpolation
Posted by ben.bighair on Sat, 12 Apr 2008 22:26:48 GMT
View Forum Message <> Reply to Message

On Apr 12, 4:24 pm, tarequea...@gmail.com wrote:
> On Apr 12, 11:57 am, "ben.bighair" <ben.bigh...@gmail.com> wrote:
>
>
>
>> 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
>
>  hi Ben,
>
>  Thanks for getting back to me on this.
>
```

> I guess I could not clearly state my problem before. So, I am going to
> give it another try:
>
> The data I deal with comes from a ccd camera and this goes some
> tinkering and tweaking (to account for background noise and stuff).
> Then this 'processed' data is in need for interpolation. why? well the
> reason being this, that these processed data are actually in XY
> frame.
> We want them to be on a polar coordinate. The reason I cannot use
> 'cv_coord' is because in that case I will get the data in a r-theta
> plane which is
> directly correlated to the experimental XY grid. If we use
> interpolation, then we can be on a XY grid (yes, it is XY grid) which
> is carefully 'designed' according to
> our 'own' r-theta values. So in this way we have more leverage over
> data manipulation.
>
> Was it any clearer than before?
>
> And one more thing, I could not find much changes in your version
> except for an inclusion of 'STOP'. Did i miss something here??
>
> Thanks again for your time.
> Much appreciated !
>
> ~ Tareque

Hi Again,

The only thing I did other than STOP (which I forgot to remove) was redefine X and Y, which in your example were missing. I thought it would be helpful for others if the whole thing worked. I should have been clearer.

Speaking of clearer... I think I am more confused now. You might need the intervention from the IDL gods that you originally petitioned. Regardless of the parts I don't understand, the thing I do get is that BILINEAR is looking for you to provide interpolation coords that fit within the dimensions of the input array to be sampled.

Take a peek at the second example in the online help for BILINEAR...
(slightly modified here...)

```
P = FINDGEN(3,3)
IX = [[0.5, 1.9], [1.1, 2.2]] ;Define the X subscripts.
JY = [[0.1, 0.9], [1.2, 1.8]] ;Define the Y subscripts.
Z = BILINEAR(P, IX, JY, MISSING = !VALUES.F_NAN) ;Interpolate.
PRINT, P
```

PRINT,Z
P prints as...
0.00000 1.00000 2.00000
3.00000 4.00000 5.00000
6.00000 7.00000 8.00000
Z prints as ...
0.800000 4.60000
4.70000 NaN

Note that IX and JY are mostly falling within the 0-2 range of indices for a 3x3 array. BILINEAR can extrapolate - just remove the MISSING keyword - but it can only extrapolate a little ways. In your example IX are sampled at

0.00000 0.0809017 0.0618034 -0.0927051 -0.323607
-0.500000 -0.485410 -0.216312 0.247214 0.728115

and JY are sampled at ...

0.00000
0.0587785
0.190211
0.285317
0.235114
-4.37114e-08
-0.352671
-0.665740
-0.760845
-0.529007

Which to my way of thinking your are interpolating all around either side of the [0,0] location. Is that what you intended? It is possible that in my redo of your example I corrupted the X and Y values you intended. In which case I would be not helping very much at all!

Cheers,
Ben

Subject: Re: Interpolation
Posted by tarequeaziz on Sun, 13 Apr 2008 00:56:57 GMT
View Forum Message <> Reply to Message

On Apr 12, 6:26 pm, "ben.bighair" <ben.bigh...@gmail.com> wrote:
> On Apr 12, 4:24 pm, tarequea...@gmail.com wrote:
>
>

>
>> On Apr 12, 11:57 am, "ben.bighair" <ben.bigh...@gmail.com> wrote:
>
>>> 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
>
>> hi Ben,
>
>> Thanks for getting back to me on this.
>
>> I guess I could not clearly state my problem before. So, I am going to
>> give it another try:
>
>> The data I deal with comes from a ccd camera and this goes some
>> tinkering and tweaking (to account for background noise and stuff).
>> Then this 'processed' data is in need for interpolation. why? well the
>> reason being this, that these processed data are actually in XY
>> frame.
>> We want them to be on a polar coordinate. The reason I cannot use
>> 'cv_coord' is because in that case I will get the data in a r-theta
>> plane which is
>> directly correlated to the experimental XY grid. If we use
>> interpolation, then we can be on a XY grid (yes, it is XY grid) which
>> is carefully 'designed' according to
>> our 'own' r-theta values. So in this way we have more leverage over
>> data manipulation.
>
>> Was it any clearer than before?
>
>> And one more thing, I could not find much changes in your version
>> except for an inclusion of 'STOP'. Did i miss something here??
>
>> Thanks again for your time.
>> Much appreciated !
>
>> ~ Tareque
>

> Hi Again,
>
> The only thing I did other than STOP (which I forgot to remove) was
> redefine X and Y, which in your example were missing. I thought it
> would be helpful for others if the whole thing worked. I should have
> been clearer.
>
> Speaking of clearer... I think I am more confused now. You might
> need the intervention from the IDL gods that you originally
> petitioned. Regardless of the parts I don't understand, the thing I
> do get is that BILINEAR is looking for you to provide interpolation
> coords that fit within the dimensions of the input array to be
> sampled.
>
> Take a peek at the second example in the online help for BILINEAR...
> (slightly modified here...)
> P = FINDGEN(3,3)
> IX = [[0.5, 1.9], [1.1, 2.2]] ;Define the X subscripts.
> JY = [[0.1, 0.9], [1.2, 1.8]] ;Define the Y subscripts.
> Z = BILINEAR(P, IX, JY, MISSING = !VALUES.F_NAN) ;Interpolate.
> PRINT, P
> PRINT,Z
> P prints as...
> 0.00000 1.00000 2.00000
> 3.00000 4.00000 5.00000
> 6.00000 7.00000 8.00000
> Z prints as ...
> 0.800000 4.60000
> 4.70000 NaN
>
> Note that IX and JY are mostly falling within the 0-2 range of indices
> for a 3x3 array. BILINEAR can extrapolate - just remove the MISSING
> keyword - but it can only extrapolate a little ways. In your example
> IX are sampled at
>
> 0.00000 0.0809017 0.0618034 -0.0927051 -0.323607
> -0.500000 -0.485410 -0.216312 0.247214 0.728115
>
> and JY are sampled at ...
>
> 0.00000
> 0.0587785
> 0.190211
> 0.285317
> 0.235114
> -4.37114e-08
> -0.352671
> -0.665740

> -0.760845
> -0.529007
>
> Which to my way of thinking your are interpolating all around either
> side of the [0,0] location. Is that what you intended? It is
> possible that in my redo of your example I corrupted the X and Y
> values you intended. In which case I would be not helping very much
> at all!
>
> Cheers,
> Ben

Hi Ben,

Thanks once again for your reply and also taking time to look into
this problem.
I am trying to attach a picture of my problem. well, looks like I cant
attach a picture here.

Anyways, I am going to give another shot at this. And,once again Boss,
thanks so much for your help.
Alright, lets start:

Think of a big circle. Close to the periphery, there is another little
circle inside of that big circle. Now, this little guy's positions are
the
bones of contentions. The coordinate system used are just rectangular
ones. We want them to be in polar.
At the same time I want the these positions in such a way that they
corresponds to a r-theta grid 'designed' by me.

That's why we are trying to interpolate.We want to get the
interpolated points based on our desired r-theta values.

In other words

Step one: Real data in a XY frame
Step two: 'Design a new xy frame, say X'Y' frame, whose values are
generated from a chosen r_vec and theta_vec.
Step 3: Now interpolate from XY to X'Y'.

Now how does it sound?

Once again, Thank you so much for your help.

Best,

Tareque

Subject: Re: Interpolation
Posted by Spon on Mon, 14 Apr 2008 15:34:15 GMT
View Forum Message <> Reply to Message

On Apr 13, 1:56 am, tarequea...@gmail.com wrote:
> Step one: Real data in a XY frame
> Step two: 'Design a new xy frame, say X'Y' frame, whose values are
> generated from a chosen r_vec and theta_vec.
> Step 3: Now interpolate from XY to X'Y'.
>
> Tareque

Hi Tareque,

I'm guessing you know whereabouts your small frame is, within your big frame, right?
So, if you take your big normal (x'y') frame, your small frame can be defined by two points,
bottom-left and top-right - let's call them (b,l) & (t,r) - in terms of x'y' grid co-oords.

; Once you've worked out where these two points are, you can use CONGRID on your xy dataset:
tempx = r - l ; How many data points of the x'y' grid does the xy grid span
tempy = t - b ; in each dimension?

; interpolate to new sub-grid
newdata = congrid(data, tempx, tempy)

; Your x'y' frame co-ordinates for this data are
newx = l + lindgen(tempx)
newy = b + lindgen(tempy)

; (this bit is just array juggling to avoid for loops)
newx = rebin(newx,tempx,tempy)
newy = rebin(reform(newy,1,tempy),tempx,tempy)
newx = reform(newx,n_elements(newx))
newy = reform(newy,n_elements(newy))

; x'y' co-ordinates for ever datapoint in 'newdata'
xycoords = transpose([[newx],[newy]])

; so your new data should be at r/theta co-ordinates defined by:
polarcoords = cv_coord(from_rect = xycoords, /to_polar)

I've assumed that your big circle is centered on the origin.
I've also assumed your small circle is in the upper-right quadrant of
your large circle here,
so I don't have to wrap my mind around minus-signs and the like...

I hope this helps and that I've understood your question
correctly. :-)

Regards,
Chris

Subject: Re: Interpolation
Posted by Spon on Mon, 14 Apr 2008 18:44:48 GMT
View Forum Message <> Reply to Message

Weird. I've just noticed that while on my work computer (XP Pro /
Firefox) the font I read google groups in can distinguish between
1(one) and l(L), at home, (XP Home/Firefox) they both look the same.

In the above code, all 'l' are the letter L except in this line:
  newy = rebin(reform(newy,1,tempy),tempx,tempy)
where the 1 is a 'one'.

Must see if I can set up my default browser fonts better, or
something.
Chris

Subject: Re: Interpolation
Posted by tarequeaziz on Mon, 14 Apr 2008 20:13:12 GMT
View Forum Message <> Reply to Message

On Apr 14, 11:34 am, Spon <christoph.b...@gmail.com> wrote:
> On Apr 13, 1:56 am, tarequea...@gmail.com wrote:
>
>> Step one: Real data in a XY frame
>> Step two: 'Design a new xy frame, say X'Y' frame, whose values are
>> generated from a chosen r_vec and theta_vec.
>> Step 3: Now interpolate from XY to X'Y'.
>
>> Tareque

>
> Hi Tareque,
>
> I'm guessing you know whereabouts your small frame is, within your big
> frame, right?
> So, if you take your big normal (x'y') frame, your small frame can be
> defined by two points,
> bottom-left and top-right - let's call them (b,l) & (t,r) - in terms
> of x'y' grid co-oords.
>
> -----
> ; Once you've worked out where these two points are, you can use
> CONGRID on your xy dataset:
> tempx = r - l ; How many data points of the x'y' grid does the xy grid
> span
> tempy = t - b ; in each dimension?
>
> ; interpolate to new sub-grid
> newdata = congrid(data, tempx, tempy)
>
> ; Your x'y' frame co-ordinates for this data are
> newx = l + lindgen(tempx)
> newy = b + lindgen(tempy)
>
> ; (this bit is just array juggling to avoid for loops)
> newx = rebin(newx,tempx,tempy)
> newy = rebin(reform(newy,1,tempy),tempx,tempy)
> newx = reform(newx,n_elements(newx))
> newy = reform(newy,n_elements(newy))
>
> ; x'y' co-ordinates for ever datapoint in 'newdata'
> xycoords = transpose([[newx],[newy]])
>
> ; so your new data should be at r/theta co-ordinates defined by:
> polarcoords = cv_coord(from_rect = xycoords, /to_polar)
> -----
>
> I've assumed that your big circle is centered on the origin.
> I've also assumed your small circle is in the upper-right quadrant of
> your large circle here,
> so I don't have to wrap my mind around minus-signs and the like...
>
> I hope this helps and that I've understood your question
> correctly. :-)
>
> Regards,
> Chris

Hi Chris,

Thank you so much for getting back at this.
Without your permission I sent a picture of my set up. Hope that will be able to shed some light on it.

Once again, much appreciated!

Best,
Tareque

Subject: Re: interpolation
Posted by anil on Wed, 07 Dec 2011 13:48:26 GMT
View Forum Message <> Reply to Message

On Dec 7, 3:46 pm, anil <akpinar.a...@gmail.com> wrote:
> Hi,
> I have many ascii files which contain a header , 5 columns of data .
> Number of rows for each data is different. Here is an example of the
> data:
>
> 7.5 14.552 17.698 14.551 12.7623
> 9.6 14.547 17.700 14.546 12.7648
> 14.6 14.462 17.713 14.460 12.7903
> 19.4 14.341 17.737 14.339 12.8307
> 24.4 14.161 17.780 14.158 12.8962
> 29.3 13.704 17.921 13.701 13.0851
> 34.6 12.313 17.986 12.309 13.3670
> 39.2 9.622 18.062 9.619 13.8134
> 44.5 8.659 18.165 8.655 14.0121
> 49.7 8.451 18.194 8.447 14.0590
> 54.3 8.221 18.257 8.217 14.1343
> 59.4 7.938 18.324 7.934 14.2180
> 64.8 7.808 18.441 7.803 14.3237
> 69.5 7.880 18.617 7.875 14.4536
> 74.5 7.923 18.924 7.917 14.6890
> 79.4 8.040 19.190 8.034 14.8840
> 84.3 8.100 19.372 8.093 15.0196
> 89.8 8.050 19.613 8.043 15.2137
> 94.6 8.069 19.781 8.062 15.3429
> 99.3 8.076 19.922 8.068 15.4524
> 109.7 8.235 20.314 8.226 15.7404
> 169.3 8.564 21.193 8.550 16.3880
> 179.3 8.602 21.269 8.587 16.4428
> 189.3 8.632 21.327 8.616 16.4844
> 199.1 8.659 21.378 8.642 16.5210
> 209.6 8.680 21.424 8.662 16.5544

> 219.3 8.708 21.482 8.689 16.5963
> 229.2 8.733 21.527 8.713 16.6283
> 239.4 8.755 21.570 8.734 16.6592
> 249.4 8.768 21.617 8.746 16.6944
> 259.3 8.777 21.649 8.754 16.7183
> 268.9 8.788 21.678 8.764 16.7397
> 278.9 8.796 21.700 8.771 16.7559
> 289.1 8.806 21.733 8.781 16.7805
> 298.9 8.834 21.764 8.808 16.8013
> 324.2 8.834 21.813 8.805 16.8398
> 349.1 8.854 21.864 8.823 16.8773
> 374.4 8.861 21.896 8.828 16.9017
> 399.5 8.861 21.916 8.825 16.9176
> 424.1 8.866 21.935 8.828 16.9321
> 449.1 8.871 21.958 8.831 16.9497
> 474.3 8.878 21.982 8.835 16.9678
> 499.5 8.881 22.005 8.836 16.9857
> 524.2 8.885 22.028 8.837 17.0034
> 549.1 8.889 22.048 8.839 17.0188
> 574.0 8.892 22.072 8.840 17.0375
> 599.0 8.895 22.100 8.840 17.0592
> 649.4 8.900 22.137 8.840 17.0881
> 699.5 8.907 22.169 8.842 17.1128
> 749.4 8.913 22.195 8.843 17.1329
> 799.3 8.921 22.215 8.847 17.1481
> 849.3 8.929 22.230 8.850 17.1595
> 899.5 8.938 22.246 8.853 17.1714
> 949.5 8.947 22.258 8.857 17.1803
> 999.3 8.954 22.266 8.859 17.1863
> 1049.6 8.965 22.276 8.865 17.1933
> 1099.1 8.971 22.282 8.866 17.1979
> 1149.0 8.980 22.288 8.870 17.2021
> 1198.9 8.989 22.294 8.873 17.2063
> 1249.3 8.996 22.297 8.875 17.2084
> 1299.4 9.003 22.301 8.877 17.2113
> 1349.4 9.011 22.304 8.879 17.2133
> 1399.0 9.018 22.305 8.881 17.2139
> 1449.1 9.025 22.308 8.882 17.2160
> 1499.2 9.033 22.309 8.885 17.2165
> 1549.2 9.040 22.311 8.886 17.2179
> 1551.8 9.041 22.311 8.887 17.2178
>
> The columns are depth,temperature,salinity,potential temperature and
> density respectively. The depth values are changing in each data set
> (7.5 is 6.7 and 9.6 is 10 in the next data and so on...). In some data
> files, depth values end at 300 or 400 etc... . What I want to do is to
> interpolate these depth values and their corresponding
> temperature,salinity, etc.. to certain levels i assign

> [5,10,15,20,25.....200,210,220.... 300,325,350,.....] or to 1m
> depth levels starting from 5m, such as (5,6,7,.....50,....1550). The
> problem here is with the 2nd and 4th column where the data first
> descends to a certain level and then starts ascending again. Therefore
> i can not use function interpol. I have done such an interpolation
> earlier:
>
> pval=100; depth value
> ind=0
> if (p[0] le pval) then begin
> while (ind le (nrows-2) and p[ind] le 100) do ind=ind+1
> tempx(i)=t[ind]-((p[ind]-100)/(p[ind]-p[ind-1])*(t[ind]-t[in d-1]))
> salx(i)=s[ind]-((p[ind]-100)/(p[ind]-p[ind-1])*(s[ind]-s[ind -1]))
> potempx(i)=q[ind]-((p[ind]-100)/(p[ind]-p[ind-1])*(q[ind]-q[ind-1]))
> denx(i)=d[ind]-((p[ind]-100)/(p[ind]-p[ind-1])*(d[ind]-d[ind -1]))
> endif
>
> ; i is the data index(1st file,2nd file etc..) and nrows is the
> number of rows, which are assigned earlier in the code.
>
> with this, I interpolate the values and obtain temperature,salinity
> etc.., for this case at 100m.
>
> I think i can find all the values i want with this way, but there
> should be a much easier way I guess. I need help on this.
> Regards

My aim here is to obtain a regular data set and then use it to contour
such a figure:
http://argo.jcommops.org/FTPRoot/Argo/Doc/Floats/2008-03/pod ens200803.jpg

---

Subject: Re: interpolation
Posted by anil on Wed, 07 Dec 2011 13:51:00 GMT
View Forum Message <> Reply to Message

On Dec 7, 3:46 pm, anil <akpinar.a...@gmail.com> wrote:
> Hi,
> I have many ascii files which contain a header , 5 columns of data .
> Number of rows for each data is different. Here is an example of the
> data:
>
> 7.5 14.552 17.698 14.551 12.7623
> 9.6 14.547 17.700 14.546 12.7648
> 14.6 14.462 17.713 14.460 12.7903
> 19.4 14.341 17.737 14.339 12.8307
> 24.4 14.161 17.780 14.158 12.8962
> 29.3 13.704 17.921 13.701 13.0851

```
>    34.6 12.313 17.986 12.309 13.3670
>    39.2  9.622 18.062  9.619 13.8134
>    44.5  8.659 18.165  8.655 14.0121
>    49.7  8.451 18.194  8.447 14.0590
>    54.3  8.221 18.257  8.217 14.1343
>    59.4  7.938 18.324  7.934 14.2180
>    64.8  7.808 18.441  7.803 14.3237
>    69.5  7.880 18.617  7.875 14.4536
>    74.5  7.923 18.924  7.917 14.6890
>    79.4  8.040 19.190  8.034 14.8840
>    84.3  8.100 19.372  8.093 15.0196
>    89.8  8.050 19.613  8.043 15.2137
>    94.6  8.069 19.781  8.062 15.3429
>    99.3  8.076 19.922  8.068 15.4524
>   109.7  8.235 20.314  8.226 15.7404
>   169.3  8.564 21.193  8.550 16.3880
>   179.3  8.602 21.269  8.587 16.4428
>   189.3  8.632 21.327  8.616 16.4844
>   199.1  8.659 21.378  8.642 16.5210
>   209.6  8.680 21.424  8.662 16.5544
>   219.3  8.708 21.482  8.689 16.5963
>   229.2  8.733 21.527  8.713 16.6283
>   239.4  8.755 21.570  8.734 16.6592
>   249.4  8.768 21.617  8.746 16.6944
>   259.3  8.777 21.649  8.754 16.7183
>   268.9  8.788 21.678  8.764 16.7397
>   278.9  8.796 21.700  8.771 16.7559
>   289.1  8.806 21.733  8.781 16.7805
>   298.9  8.834 21.764  8.808 16.8013
>   324.2  8.834 21.813  8.805 16.8398
>   349.1  8.854 21.864  8.823 16.8773
>   374.4  8.861 21.896  8.828 16.9017
>   399.5  8.861 21.916  8.825 16.9176
>   424.1  8.866 21.935  8.828 16.9321
>   449.1  8.871 21.958  8.831 16.9497
>   474.3  8.878 21.982  8.835 16.9678
>   499.5  8.881 22.005  8.836 16.9857
>   524.2  8.885 22.028  8.837 17.0034
>   549.1  8.889 22.048  8.839 17.0188
>   574.0  8.892 22.072  8.840 17.0375
>   599.0  8.895 22.100  8.840 17.0592
>   649.4  8.900 22.137  8.840 17.0881
>   699.5  8.907 22.169  8.842 17.1128
>   749.4  8.913 22.195  8.843 17.1329
>   799.3  8.921 22.215  8.847 17.1481
>   849.3  8.929 22.230  8.850 17.1595
>   899.5  8.938 22.246  8.853 17.1714
>   949.5  8.947 22.258  8.857 17.1803
```

> 999.3 8.954 22.266 8.859 17.1863
> 1049.6 8.965 22.276 8.865 17.1933
> 1099.1 8.971 22.282 8.866 17.1979
> 1149.0 8.980 22.288 8.870 17.2021
> 1198.9 8.989 22.294 8.873 17.2063
> 1249.3 8.996 22.297 8.875 17.2084
> 1299.4 9.003 22.301 8.877 17.2113
> 1349.4 9.011 22.304 8.879 17.2133
> 1399.0 9.018 22.305 8.881 17.2139
> 1449.1 9.025 22.308 8.882 17.2160
> 1499.2 9.033 22.309 8.885 17.2165
> 1549.2 9.040 22.311 8.886 17.2179
> 1551.8 9.041 22.311 8.887 17.2178
>
> The columns are depth,temperature,salinity,potential temperature and
> density respectively. The depth values are changing in each data set
> (7.5 is 6.7 and 9.6 is 10 in the next data and so on...). In some data
> files, depth values end at 300 or 400 etc... . What I want to do is to
> interpolate these depth values and their corresponding
> temperature,salinity, etc.. to certain levels i assign
> [5,10,15,20,25.....200,210,220.... 300,325,350,.....] or to 1m
> depth levels starting from 5m, such as (5,6,7,.....50,....1550). The
> problem here is with the 2nd and 4th column where the data first
> descends to a certain level and then starts ascending again. Therefore
> i can not use function interpol. I have done such an interpolation
> earlier:
>
> pval=100; depth value
> ind=0
> if (p[0] le pval) then begin
> while (ind le (nrows-2) and p[ind] le 100) do ind=ind+1
> tempx(i)=t[ind]-((p[ind]-100)/(p[ind]-p[ind-1])*(t[ind]-t[in d-1]))
> salx(i)=s[ind]-((p[ind]-100)/(p[ind]-p[ind-1])*(s[ind]-s[ind -1]))
> potempx(i)=q[ind]-((p[ind]-100)/(p[ind]-p[ind-1])*(q[ind]-q[ind-1]))
> denx(i)=d[ind]-((p[ind]-100)/(p[ind]-p[ind-1])*(d[ind]-d[ind -1]))
> endif
>
> ; i is the data index(1st file,2nd file etc..) and nrows is the
> number of rows, which are assigned earlier in the code.
>
> with this, I interpolate the values and obtain temperature,salinity
> etc.., for this case at 100m.
>
> I think i can find all the values i want with this way, but there
> should be a much easier way I guess. I need help on this.
> Regards

My aim here is to obtain a regular data set. And use it to contour

such a figure:
http://argo.jcommops.org/FTPRoot/Argo/Doc/Floats/2008-03/pod ens200803.jpg

---

Subject: Re: interpolation
Posted by Russell[1] on Wed, 07 Dec 2011 16:42:22 GMT
View Forum Message <> Reply to Message

I'm not exactly sure what the issue is, but interpol seems to work
fine?

readcol,'data.dat',depth,temp,salinity,pot_temp,density
thisdepth=findgen(300)*5 ;the values where you want the temperature
-- whose derivative changes sign
thistemp=interpol(temp,depth,thisdepth)

You might add /quad or /lsq for a better fit, or even use a better
spline routine...

Russell

On Dec 7, 8:51 am, anil <akpinar.a...@gmail.com> wrote:
> On Dec 7, 3:46 pm, anil <akpinar.a...@gmail.com> wrote:
>
>
>
>
>
>
>
>
>
>
>> Hi,
>> I have many ascii files which contain a header , 5 columns of data .
>> Number of rows for each data is different. Here is an example of the
>> data:
>
>>   7.5 14.552 17.698 14.551 12.7623
>>   9.6 14.547 17.700 14.546 12.7648
>>   14.6 14.462 17.713 14.460 12.7903
>>   19.4 14.341 17.737 14.339 12.8307
>>   24.4 14.161 17.780 14.158 12.8962
>>   29.3 13.704 17.921 13.701 13.0851
>>   34.6 12.313 17.986 12.309 13.3670
>>   39.2   9.622 18.062   9.619 13.8134

```
>>    44.5  8.659  18.165  8.655  14.0121
>>    49.7  8.451  18.194  8.447  14.0590
>>    54.3  8.221  18.257  8.217  14.1343
>>    59.4  7.938  18.324  7.934  14.2180
>>    64.8  7.808  18.441  7.803  14.3237
>>    69.5  7.880  18.617  7.875  14.4536
>>    74.5  7.923  18.924  7.917  14.6890
>>    79.4  8.040  19.190  8.034  14.8840
>>    84.3  8.100  19.372  8.093  15.0196
>>    89.8  8.050  19.613  8.043  15.2137
>>    94.6  8.069  19.781  8.062  15.3429
>>    99.3  8.076  19.922  8.068  15.4524
>>   109.7  8.235  20.314  8.226  15.7404
>>   169.3  8.564  21.193  8.550  16.3880
>>   179.3  8.602  21.269  8.587  16.4428
>>   189.3  8.632  21.327  8.616  16.4844
>>   199.1  8.659  21.378  8.642  16.5210
>>   209.6  8.680  21.424  8.662  16.5544
>>   219.3  8.708  21.482  8.689  16.5963
>>   229.2  8.733  21.527  8.713  16.6283
>>   239.4  8.755  21.570  8.734  16.6592
>>   249.4  8.768  21.617  8.746  16.6944
>>   259.3  8.777  21.649  8.754  16.7183
>>   268.9  8.788  21.678  8.764  16.7397
>>   278.9  8.796  21.700  8.771  16.7559
>>   289.1  8.806  21.733  8.781  16.7805
>>   298.9  8.834  21.764  8.808  16.8013
>>   324.2  8.834  21.813  8.805  16.8398
>>   349.1  8.854  21.864  8.823  16.8773
>>   374.4  8.861  21.896  8.828  16.9017
>>   399.5  8.861  21.916  8.825  16.9176
>>   424.1  8.866  21.935  8.828  16.9321
>>   449.1  8.871  21.958  8.831  16.9497
>>   474.3  8.878  21.982  8.835  16.9678
>>   499.5  8.881  22.005  8.836  16.9857
>>   524.2  8.885  22.028  8.837  17.0034
>>   549.1  8.889  22.048  8.839  17.0188
>>   574.0  8.892  22.072  8.840  17.0375
>>   599.0  8.895  22.100  8.840  17.0592
>>   649.4  8.900  22.137  8.840  17.0881
>>   699.5  8.907  22.169  8.842  17.1128
>>   749.4  8.913  22.195  8.843  17.1329
>>   799.3  8.921  22.215  8.847  17.1481
>>   849.3  8.929  22.230  8.850  17.1595
>>   899.5  8.938  22.246  8.853  17.1714
>>   949.5  8.947  22.258  8.857  17.1803
>>   999.3  8.954  22.266  8.859  17.1863
>>  1049.6  8.965  22.276  8.865  17.1933
```

>> 1099.1 8.971 22.282 8.866 17.1979
>> 1149.0 8.980 22.288 8.870 17.2021
>> 1198.9 8.989 22.294 8.873 17.2063
>> 1249.3 8.996 22.297 8.875 17.2084
>> 1299.4 9.003 22.301 8.877 17.2113
>> 1349.4 9.011 22.304 8.879 17.2133
>> 1399.0 9.018 22.305 8.881 17.2139
>> 1449.1 9.025 22.308 8.882 17.2160
>> 1499.2 9.033 22.309 8.885 17.2165
>> 1549.2 9.040 22.311 8.886 17.2179
>> 1551.8 9.041 22.311 8.887 17.2178
>
>> The columns are depth,temperature,salinity,potential temperature and
>> density respectively. The depth values are changing in each data set
>> (7.5 is 6.7 and 9.6 is 10 in the next data and so on...). In some data
>> files, depth values end at 300 or 400 etc... . What I want to do is to
>> interpolate these depth values and their corresponding
>> temperature,salinity, etc.. to certain levels i assign
>> [5,10,15,20,25.....200,210,220.... 300,325,350,.....] or to 1m
>> depth levels starting from 5m, such as (5,6,7,.....50,....1550). The
>> problem here is with the 2nd and 4th column where the data first
>> descends to a certain level and then starts ascending again. Therefore
>> i can not use function interpol. I have done such an interpolation
>> earlier:
>
>> pval=100; depth value
>> ind=0
>> if (p[0] le pval) then begin
>> while (ind le (nrows-2) and p[ind] le 100) do ind=ind+1
>> tempx(i)=t[ind]-((p[ind]-100)/(p[ind]-p[ind-1])*(t[ind]-t[in d-1]))
>> salx(i)=s[ind]-((p[ind]-100)/(p[ind]-p[ind-1])*(s[ind]-s[ind -1]))
>> potempx(i)=q[ind]-((p[ind]-100)/(p[ind]-p[ind-1])*(q[ind]-q[ind-1]))
>> denx(i)=d[ind]-((p[ind]-100)/(p[ind]-p[ind-1])*(d[ind]-d[ind -1]))
>> endif
>
>> ; i is the data index(1st file,2nd file etc..) and nrows is the
>> number of rows, which are assigned earlier in the code.
>
>> with this, I interpolate the values and obtain temperature,salinity
>> etc.., for this case at 100m.
>
>> I think i can find all the values i want with this way, but there
>> should be a much easier way I guess. I need help on this.
>> Regards
>
> My aim here is to obtain a regular data set. And use it to contour
> such a figure:http://argo.jcommops.org/FTPRoot/Argo/Doc/Floats/2008 -03/podens200803...

Subject: Re: interpolation
Posted by anil on Wed, 07 Dec 2011 17:50:31 GMT
View Forum Message <> Reply to Message

On Dec 7, 6:42 pm, Russell <rryan....@gmail.com> wrote:
> I'm not exactly sure what the issue is, but interpol seems to work
> fine?
>
> readcol,'data.dat',depth,temp,salinity,pot_temp,density
> thisdepth=findgen(300)*5 ;the values where you want the temperature
> -- whose derivative changes sign
> thistemp=interpol(temp,depth,thisdepth)
>
> You might add /quad or /lsq for a better fit, or even use a better
> spline routine...
>
> Russell
>
>
>

This is not working for sure! When I do this I get negative values and numbers as -3000,-4165 etc.. whereas my temperature range is between 4 and 26.

Subject: Re: interpolation
Posted by Jeremy Bailin on Wed, 07 Dec 2011 20:07:58 GMT
View Forum Message <> Reply to Message

On 12/7/11 5:50 PM, anil wrote:
> On Dec 7, 6:42 pm, Russell<rryan....@gmail.com> wrote:
>> I'm not exactly sure what the issue is, but interpol seems to work
>> fine?
>>
>> readcol,'data.dat',depth,temp,salinity,pot_temp,density
>> thisdepth=findgen(300)*5 ;the values where you want the temperature
>> -- whose derivative changes sign
>> thistemp=interpol(temp,depth,thisdepth)
>>
>> You might add /quad or /lsq for a better fit, or even use a better
>> spline routine...
>>
>> Russell
>>
>>
>>
>

> This is not working for sure! When I do this I get negative values and
> numbers as -3000,-4165 etc.. whereas my temperature range is between 4
> and 26.

Russell's code certainly works perfectly well on the data you provided.

-Jeremy.

Subject: Re: interpolation
Posted by David Fanning on Wed, 07 Dec 2011 20:13:04 GMT
View Forum Message <> Reply to Message

Jeremy Bailin writes:

> Russell's code certainly works perfectly well on the data you provided.

You may have to consider the unthinkable: a problem with
your own code! ;-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: interpolation
Posted by Brian Wolven on Wed, 07 Dec 2011 20:27:52 GMT
View Forum Message <> Reply to Message

It does seem unlikely that one could *interpolate* and get values that were outside the bounds of the original data. ;)

Subject: Re: interpolation
Posted by Kenneth P. Bowman on Wed, 07 Dec 2011 22:25:39 GMT
View Forum Message <> Reply to Message

In article < 12296280.447.1323289672656.JavaMail.geo-discussion-forums@yq q16 >,
Brian Wolven <brian.wolven@gmail.com> wrote:

> It does seem unlikely that one could *interpolate* and get values that were
> outside the bounds of the original data. ;)

With linear interpolation that is certainly true, but it can occur
with higher order schemes. Interpolation usually guarantees only
that the interpolating function passes through the known data points.

If you take the spline interpolation example in my book

http://csrp.tamu.edu/pdf/idl/sample_chapter.pdf

and compute the result at high resolution, you will see that
the spline function overshoots the tabulated points.

Cheers, Ken Bowman

Subject: Re: interpolation
Posted by anil on Thu, 08 Dec 2011 12:15:10 GMT
View Forum Message <> Reply to Message

On Dec 7, 10:13 pm, David Fanning <n...@dfanning.com> wrote:
> Jeremy Bailin writes:
>> Russell's code certainly works perfectly well on the data you provided.
>
> You may have to consider the unthinkable: a problem with
> your own code! ;-)
>
> Cheers,
>
> David
>
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming:http://www.idlcoyote.com/
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Ok. I've found the problem.It is because of the loops etc.. i have
used previously. I've fixed it and now Russell's code works fine for
my case. The only problem I have now is that:
I have certain number of desired depth values which I assign as :
depth2=indgen(310)*5+5
temp2=interpol(t,p,depth2)
sal2=interpol(s,p,depth2)
potemp2=interpol(q,p,depth2)
den2=interpol(d,p,depth2)

and temp2,sal2,potemp2,den2 are the values corresponding to these depths. I should have a depth range of 5 to 1550(or 1540)m but in some data ,I only have depths up to 100m or 300 in some other. So i have done this instead:

step=(fix(max(p))-fix(min(p)))/5
depth2=indgen(step)*5+5
temp2=interpol(t,p,depth2)
sal2=interpol(s,p,depth2)
potemp2=interpol(q,p,depth2)
den2=interpol(d,p,depth2)
because of fix I lose 1 or 2 data points. But it is not so important for my case. Thank you all for helping out.
Regards,
ANIL

Subject: Re: interpolation
Posted by anil on Thu, 08 Dec 2011 12:15:20 GMT
View Forum Message <> Reply to Message

On Dec 7, 10:13 pm, David Fanning <n...@dfanning.com> wrote:
> Jeremy Bailin writes:
>> Russell's code certainly works perfectly well on the data you provided.
>
> You may have to consider the unthinkable: a problem with
> your own code! ;-)
>
> Cheers,
>
> David
>
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming:http://www.idlcoyote.com/
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Ok. I've found the problem.It is because of the loops etc.. i have used previously. I've fixed it and now Russell's code works fine for my case. The only problem I have now is that:
I have certain number of desired depth values which I assign as :
depth2=indgen(310)*5+5
temp2=interpol(t,p,depth2)
sal2=interpol(s,p,depth2)
potemp2=interpol(q,p,depth2)
den2=interpol(d,p,depth2)
and temp2,sal2,potemp2,den2 are the values corresponding to these

depths. I should have a depth range of 5 to 1550(or 1540)m but in some data ,I only have depths up to 100m or 300 in some other. So i have done this instead:

step=(fix(max(p))-fix(min(p)))/5
depth2=indgen(step)*5+5
temp2=interpol(t,p,depth2)
sal2=interpol(s,p,depth2)
potemp2=interpol(q,p,depth2)
den2=interpol(d,p,depth2)
because of fix I lose 1 or 2 data points. But it is not so important for my case. Thank you all for helping out.
Regards,
ANIL

Subject: Re: interpolation
Posted by anil on Thu, 08 Dec 2011 12:15:29 GMT
View Forum Message <> Reply to Message

On Dec 7, 10:13 pm, David Fanning <n...@dfanning.com> wrote:
> Jeremy Bailin writes:
>> Russell's code certainly works perfectly well on the data you provided.
>
> You may have to consider the unthinkable: a problem with
> your own code! ;-)
>
> Cheers,
>
> David
>
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming:http://www.idlcoyote.com/
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Ok. I've found the problem.It is because of the loops etc.. i have used previously. I've fixed it and now Russell's code works fine for my case. The only problem I have now is that:
I have certain number of desired depth values which I assign as :
depth2=indgen(310)*5+5
temp2=interpol(t,p,depth2)
sal2=interpol(s,p,depth2)
potemp2=interpol(q,p,depth2)
den2=interpol(d,p,depth2)
and temp2,sal2,potemp2,den2 are the values corresponding to these depths. I should have a depth range of 5 to 1550(or 1540)m but in some

data ,I only have depths up to 100m or 300 in some other. So i have
done this instead:

step=(fix(max(p))-fix(min(p)))/5
depth2=indgen(step)*5+5
temp2=interpol(t,p,depth2)
sal2=interpol(s,p,depth2)
potemp2=interpol(q,p,depth2)
den2=interpol(d,p,depth2)
because of fix I lose 1 or 2 data points. But it is not so important
for my case. Thank you all for helping out.
Regards,
ANIL

Subject: Re: interpolation
Posted by Brian Wolven on Thu, 08 Dec 2011 20:57:41 GMT
View Forum Message <> Reply to Message

I would add only that it seems difficult to justify using any type of higher order scheme for data of this type, but that's more of a philosophical and aesthetic issue to debate. Getting values like -4000 when the data range from 4 to 26 is way more than just an overshoot problem!

Subject: Re: interpolation
Posted by Yngvar Larsen on Thu, 08 Dec 2011 21:24:14 GMT
View Forum Message <> Reply to Message

On Dec 7, 6:50 pm, anil <akpinar.a...@gmail.com> wrote:
> On Dec 7, 6:42 pm, Russell <rryan....@gmail.com> wrote:
>
>> I'm not exactly sure what the issue is, but interpol seems to work
>> fine?
>
>> readcol,'data.dat',depth,temp,salinity,pot_temp,density
>> thisdepth=findgen(300)*5 ;the values where you want the temperature
>> -- whose derivative changes sign
>> thistemp=interpol(temp,depth,thisdepth)
>
>> You might add /quad or /lsq for a better fit, or even use a better
>> spline routine...
>
>> Russell
>
> This is not working for sure! When I do this I get negative values and
> numbers as -3000,-4165 etc.. whereas my temperature range is between 4
> and 26.

Maybe missing numbers represented as -9999, or something similar, in some of the files?

--
Yngvar