Subject: Gridding options
Posted by Ben Tupper on Tue, 29 Aug 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Hello,

I'm staring (again) at largish set of CTD casts from a recent cruise. The cast data is comprised of sample information from every 0.5 meters from the surface to the seafloor. The 20 or so casts are separated from each other by about 10-20km and are nearly colinear. I need to interpolate a 2d grid from these values. In the past I have used the techniques described to grid the data. I list them here in hopes that someone familiar with this kind of data can suggest alternatives.

1. Triangulation and TRIGRID, this method works quickly and preserves the different 'clines' (pycno-, halo- thermo-, ...) very well. Since the seafloor is very irregular, I spend a good deal of effort fiddling with masks and blanking the boundaries. This method also accentuates the noisiness of the data (median filtering of each cast prior to gridding helps.) Despite having to play twister with the boundary/masking stuff, this is the method we use right now.

2. MIN_CURVE_SURF, this method is fairly fast but the details are lost.

3. A home grown Inverse-Distance-To-A-Power method, this method is slow for large datasets and tends to produce a bull's eye pattern around isolated features (especially common in the biological data set.) (Sorry, JD, it does have loopity-loops. You can flick a worm into the air... but that doesn't mean it knows how to fly! I'm still trying to figure out what .....

```
****************************************
tt=total(a[(((dy=((di=lindgen(((n=nx<ny)),nx+ny-1)))/n))*(nx gt
ny?1:nx)+ $
            (nx gt ny?1:-1)*((dx=di mod n))*(nx-1))>0<(nx*ny-1)]* $
         (dy ge dx AND (dy-dx) lt nx>ny),1)
****************************************
```

means. Maybe I need to drink less/more coffee.)

4. KRIG_2D, this method is so slow (for the size of the data set) that I haven't had the patience to wait for the result. I know that recent version of SURFER (Golden Software) has introduced an improved KRIGING routine that will interpolate large grids quickly (I have seen it produce a grid in a matter of minutes from similar data... I have waited hours for a similar result from IDL.)

So, are there alternatives for gridding in IDL?

Thanks,

Ben
--
Ben Tupper
Bigelow Laboratory for Ocean Science
West Boothbay Harbor, Maine
btupper@bigelow.org
    note: email address new as of 25JULY2000

Subject: Re: Gridding options
Posted by Craig Markwardt on Wed, 30 Aug 2000 07:00:00 GMT

View Forum Message <> Reply to Message

tclement@ucsd.edu (Todd Clements) writes:

```
>  arrays larger than 128x128, and starts giving incorrect values. It seems
>  like just a "short" integer problem, but heck if I can figure out where it
>  might be in there!!
```

Actually, I think this is a problem that you made NX and NY 16-bit
integers. If you promote them to long then it should work again.

```
>  The times are as follows (I'm not sure they mean anything for incorrect
>  results, but here they are anyway):
>
>  Array size     New (1 line)      Craig (2 line)
>  512x512         0.493             0.615
>  1024x1024       2.531             3.039
>  2048x2048       10.523            12.89
```

Hmm, surprisingly I found that my version was about 4 times faster on
two different architectures.

	Craig (otest)	JD (test)	1024x1024
Linux	0.35	1.94 s	{ x86 linux unix 5.2.1 Jun 4 1999}
Alpha	1.47	6.78 s	{ alpha OSF unix 5.2 Oct 30 1998}

The codes I used are below, in all their ugly, wake-up in the morning,
hair of the dog glory. Did I do something wrong? Note that I tried
both INT and FLOAT, and also did a comparison test. As long as you
pass 1024L instead of 1024 you shouldn't get incorrect answers.

Craig
[cc to Clements]

```
.comp
function otest, nx, iter=iter, integer=doint
ny = nx
if keyword_set(doint) then a = lindgen(nx,ny) else a = findgen(nx,ny)
if n_elements(iter) EQ 0 then iter = 10

tt = fltarr(nx+ny-1)

;; Do the work
t0 = systime(1)
ll = lindgen(nx>ny)
for j = 0, iter-1 do begin
 for i = 0, ny-1 do tt(i)      = total((a(0+ll,i-ll))(0:i<(nx-1)))
 for i = 1, nx-1 do tt(i+ny-1) = total((a(i+ll,ny-1-ll))(0:(nx-1-i)<(ny-1)))
end
if iter GT 1 then $
  print, (systime(1)-t0)/10.
return, tt
end

.comp
pro test, nx, integer=doint
ny = nx
if keyword_set(doint) then a = lindgen(nx,ny) else a = findgen(nx,ny)
t0 = systime(1)
for i = 0, 9 do $
tt=total(a[(((dy=((di=lindgen(((n=nx<ny)),nx+ny-1)))/n))*(nx gt ny?1:nx)+ $
               (nx gt ny?1:-1)*((dx=di mod n))*(nx-1))>0<(nx*ny-1)]* $
            (dy ge dx AND (dy-dx) lt nx>ny),1)
print, (systime(1)-t0)/10.
print, max(abs(tt-otest(nx,iter=1)))
end
```

--

 -- --------------

Craig B. Markwardt, Ph.D.   EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response

 -- --------------

Subject: Re: Gridding options
Posted by davidf on Wed, 30 Aug 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Todd Clements (tclement@ucsd.edu) writes:

> my news server seems to only get about 30% of the articles

> posted here

Turn the "joke" filter off and you will get the rest.

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Gridding options
Posted by tclement on Wed, 30 Aug 2000 07:00:00 GMT

View Forum Message <> Reply to Message

In article <onhf82y1wb.fsf@cow.physics.wisc.edu>,
craigmnet@cow.physics.wisc.edu wrote:

```
> "J.D. Smith" <jdsmith@astro.cornell.edu> writes:
>
>>  tt=total(a[(((dy=((di=lindgen(((n=nx<ny)),nx+ny-1)))/n))*(nx gt ny?1:nx)+ $
>>              (nx gt ny?1:-1)*((dx=di mod n))*(nx-1))>0<(nx*ny-1)]* $
>>           (dy ge dx AND (dy-dx) lt nx>ny),1)

>
> Todd, did you compare the run time?
>
```

I did, (only after I read your post responding to someones post responding to that...my news server seems to only get about 30% of the articles posted here, and so I often don't see things until someone else quotes it and I see it then) and the new one is faster, except that it dies at arrays larger than 128x128, and starts giving incorrect values. It seems like just a "short" integer problem, but heck if I can figure out where it might be in there!!

The times are as follows (I'm not sure they mean anything for incorrect results, but here they are anyway):

Array size	New (1 line)	Craig (2 line)
512x512	0.493	0.615
1024x1024	2.531	3.039
2048x2048	10.523	12.89

Todd

Subject: Re: Gridding options
Posted by Craig Markwardt on Wed, 30 Aug 2000 07:00:00 GMT
View Forum Message <> Reply to Message

"J.D. Smith" <jdsmith@astro.cornell.edu> writes:

```
> Ben Tupper wrote:
>> (Sorry, JD, it does have loopity-loops.  You can flick a worm into the
>> air... but that doesn't mean it knows how to fly!  I'm still trying to
>> figure out what .....
>>
> ****************************************
> tt=total(a[(((dy=((di=lindgen(((n=nx<ny)),nx+ny-1)))/n))*(nx gt ny?1:nx)+ $
>              (nx gt ny?1:-1)*((dx=di mod n))*(nx-1))>0<(nx*ny-1)]* $
>           (dy ge dx AND (dy-dx) lt nx>ny),1)
> ****************************************
>>
>> means.   Maybe I need to drink less/more coffee. )
>
> It's written in an ancient IDL runic language lost in the dark ages (1993).
> Simply run it on an array "a" after defining the dimensions "nx" and "ny" and
> all will be clear.
```

Another disadvantage to vectorizing with very large arrays is that IDL
has to generate the indices for the array dynamically. This can
consume a lot of memory (sometimes more than the array itself, if it's
a byte or int array!). The expression, lindgen((nx<ny), nx+ny-1),
[some deleted] is the giveaway that JD had to essentially create a
whole new array even bigger than the original. Still, it's a pretty
impressive computation.

Todd, did you compare the run time?

Craig

--
--- --------------
Craig B. Markwardt, Ph.D.   EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
--- --------------

Subject: Re: Gridding options
Posted by John-David T. Smith on Wed, 30 Aug 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Ben Tupper wrote:

```
>  (Sorry, JD, it does have loopity-loops.  You can flick a worm into the
>  air... but that doesn't mean it knows how to fly!  I'm still trying to
>  figure out what .....
>
***************************************
tt=total(a[(((dy=((di=lindgen(((n=nx<ny)),nx+ny-1)))/n))*(nx gt ny?1:nx)+ $
              (nx gt ny?1:-1)*((dx=di mod n))*(nx-1))>0<(nx*ny-1)]* $
           (dy ge dx AND (dy-dx) lt nx>ny),1)
***************************************
>
>  means.   Maybe I need to drink less/more coffee. )
```

It's written in an ancient IDL runic language lost in the dark ages (1993).
Simply run it on an array "a" after defining the dimensions "nx" and "ny" and
all will be clear.

JD

```
--
 J.D. Smith                             /*\    WORK: (607) 255-6263
 Cornell University Dept. of Astronomy  \*/         (607) 255-5842
 304 Space Sciences Bldg.               /*\    FAX: (607) 255-5875
 Ithaca, NY 14853                       \*/
```

Subject: Re: Gridding options
Posted by Ben Tupper on Wed, 30 Aug 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Craig Markwardt wrote:

```
>
>  The more appropriate question is probably, how broad should the
>  gaussian be in X and Y?  This depends on how much smoothing you want
>  to acomplish, and the new sampling.  For example, if your original
>  sampling was 10-20 km, then the interpolated image might have ~2 km
>  resolution.  With minimal smoothing, the gaussian sigma would be
>  around 15 km (ie, comparable to your sampling).  The response function
>  should have around +/- 2 sigmas = +/- 30 km, which is about 30 pixels.
>
```

Ah! Got it!

THANKS!

Ben

--
Ben Tupper
Bigelow Laboratory for Ocean Science
West Boothbay Harbor, Maine
btupper@bigelow.org
note: email address new as of 25JULY2000

Subject: Re: Gridding options
Posted by tclement on Thu, 31 Aug 2000 15:42:40 GMT

View Forum Message <> Reply to Message

craigmnet@cow.physics.wisc.edu wrote:

> Hmm, surprisingly I found that my version was about 4 times faster on
> two different architectures.

And eventually, so did I once I revised my testing code. My mistake entirely... I got running times of 2.537 vs. 0.5115 for a 1024x1024 array.

{ alpha OSF unix 5.3 Nov 11 1999}

Todd

Subject: Re: Gridding options
Posted by Craig Markwardt on Thu, 31 Aug 2000 15:56:21 GMT

View Forum Message <> Reply to Message

tclement@ucsd.edu (Todd Clements) writes:
> craigmnet@cow.physics.wisc.edu wrote:
>
>> Hmm, surprisingly I found that my version was about 4 times faster on
>> two different architectures.
>
> And eventually, so did I once I revised my testing code. My mistake
> entirely... I got running times of 2.537 vs. 0.5115 for a 1024x1024 array.
>
> { alpha OSF unix 5.3 Nov 11 1999}

The triumph of the FOR loops, then! Sometimes even ancient scripture can be misinterpretted.

Craig

--
 -- --------------
Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
 -- --------------

Subject: Re: Gridding options
Posted by John-David T. Smith on Thu, 31 Aug 2000 18:33:04 GMT

View Forum Message <> Reply to Message

Craig Markwardt wrote:
>
> tclement@ucsd.edu (Todd Clements) writes:
>> craigmnet@cow.physics.wisc.edu wrote:
>>
>>> Hmm, surprisingly I found that my version was about 4 times faster on
>>> two different architectures.
>>
>> And eventually, so did I once I revised my testing code. My mistake
>> entirely... I got running times of 2.537 vs. 0.5115 for a 1024x1024 array.
>>
>> { alpha OSF unix 5.3 Nov 11 1999}
>
> The triumph of the FOR loops, then! Sometimes even ancient scripture
> can be misinterpretted.
>
> Craig

The rampant literalism in this newsgroup is repulsing me. I mean really, people, look at it!

```
tt=total(a[(((dy=((di=lindgen(((n=nx<ny)),nx+ny-1)))/n))*(nx gt ny?1:nx)+ $
              (nx gt ny?1:-1)*((dx=di mod n))*(nx-1))>0<(nx*ny-1)]* $
           (dy ge dx AND (dy-dx) lt nx>ny),1)
```

Does it really look like something anyone would ever actually write?!? I can see my sarcasm has been utterly wasted on this crowd.

JD

--
 J.D. Smith   /*\   WORK: (607) 255-6263
 Cornell University Dept. of Astronomy */   (607) 255-5842

```
304 Space Sciences Bldg.          /*\     FAX: (607) 255-5875
Ithaca, NY 14853                 \*/
```

Subject: Re: Gridding options
Posted by tclement on Thu, 31 Aug 2000 19:08:34 GMT

View Forum Message <> Reply to Message

```
"J.D. Smith" <jdsmith@astro.cornell.edu> wrote:
> Craig Markwardt wrote:
>>  The triumph of the FOR loops, then!  Sometimes even ancient scripture
>>  can be misinterpretted.
>
> The rampant literalism in this newsgroup is repulsing me.  I mean really,
> people, look at it!
>
> tt=total(a[(((dy=((di=lindgen(((n=nx<ny)),nx+ny-1)))/n))*(nx gt ny?1:nx)+ $
>             (nx gt ny?1:-1)*((dx=di mod n))*(nx-1))>0<(nx*ny-1)]* $
>          (dy ge dx AND (dy-dx) lt nx>ny),1)
>
> Does it really look like something anyone would ever actually write?!?  I can
> see my sarcasm has been utterly wasted on this crowd.
```

Well, I hate to be too literal about it, but you _did_ write it, did you not? =)

I do believe that it does fall under the realm of obfuscated. Perhaps, as in the world of C where there is an actual contest to create such things, we could add entries such as the one above to our own IDL obfuscated code contest. Perhaps the "evil side" of being an "expert programmer" is, like JD, you get the urge to write code like that. ;>

Todd

Subject: Re: Gridding options
Posted by davidf on Thu, 31 Aug 2000 19:18:42 GMT

View Forum Message <> Reply to Message

J.D. Smith (jdsmith@astro.cornell.edu) writes:

```
>  The rampant literalism in this newsgroup is repulsing me.  I mean really,
>  people, look at it!
>
>  tt=total(a[(((dy=((di=lindgen(((n=nx<ny)),nx+ny-1)))/n))*(nx gt ny?1:nx)+ $
>              (nx gt ny?1:-1)*((dx=di mod n))*(nx-1))>0<(nx*ny-1)]* $
>           (dy ge dx AND (dy-dx) lt nx>ny),1)
>
```

> Does it really look like something anyone would ever actually write?!? I can
> see my sarcasm has been utterly wasted on this crowd.

The problem, JD, is that you are not putting in the proper thingy (I know there is a proper word for this, but I can never remember it) that indicates sarcasm. I consulted the canonical list of smiley faces:

http://www.indiadirect.com/kkjg/CanonicalLists.html

and the only thing I could find is ;-), which is purported to be "flirtatious or sarcastic". I wouldn't go with flirtatious. I tried that yesterday with, well, bad results.

And, geez, for all I know you may have already have put the proper similey in there. I'm still working through the first inside pair of parentheses. :-(

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155