Subject: Re: More Kriging Problems
Posted by David Fanning on Thu, 17 Oct 2013 13:01:24 GMT
View Forum Message <> Reply to Message

Chris Torrence writes:

> I think I see the problem. In my latest krig2d.pro code, I am
incorrectly calculating the X and Y arrays for the /REGULAR grid case.
Near the end of the code are the following lines:
> xx = (SIZE(x, /N_DIM) eq 2) ? x : REBIN(TRANSPOSE(x), nx, n)
> yy = (SIZE(y, /N_DIM) eq 2) ? y : REBIN(TRANSPOSE(y), nx, n)
>
> They should be:
> xx = REBIN(REFORM(x, 1, n), nx, n)
> yy = REBIN(REFORM(y, 1, n), nx, n)

Yes, this appears to solve the problem nicely. Thanks!

> Do you want to make the changes to the code, and see what kind of
> results you get?

OK, I made the above changes to my cgKrig2D code and to your new Krig2D
code, which I call Krig2D_Fast, to distinguish it from the normal Krig2D
code.

Here is my test code:

```
;*****************************************************
number = 21
values = Dist(number)
cgSurface, values, Title='Original Values'

values = Dist(number)
tic
sampled1 = Krig2d(values, Spherical=[5.0, 0.0])
toc
cgSurface, sampled1, Title='Original Krig2D Values'

values = Dist(number)
tic
sampled2 = cgKrig2d(values, Spherical=[5.0, 0.0])
toc
cgSurface, sampled2, Title='cgKrig2D Values'

values = Dist(number)
tic
sampled3 = Krig2d_Fast(values, Spherical=[5.0, 0.0])
toc
```

```
cgSurface, sampled3, Title='Fast Krig2D Values'
;*****************************************************
```

Here are the times I get on my machine:

```
 Elapsed Time:  0.952000   ; Original Krig2D code
 Elapsed Time:  0.187000   ; cgKrig2D
 Elapsed Time:  0.202000   ; Krig2D_Fast
```

I've run this several times and cgKrig2D is always *slightly* faster than Krig2D_Fast, but essentially the same, even if I set the "number" variable to different values.

There are a couple of other differences (which may explain the time difference). cgKrig2D preserves the grid size of the original data set, while the original Krig2D code and the Krig2D_Fast code default to their regular 26x26 size grid. (Humm, if this is true, then if I make the grid 31x31 then cgKrig2D should be slower than Krig2D_Fast, and this in fact happens.)

```
 Elapsed Time:  5.616000   ; Original Krig2D code
 Elapsed Time:  2.137000   ; cgKrig2D
 Elapsed Time:  2.106000   ; Krig2D_Fast
```

Also, now the original Krig2D routine produces a CRAZY surface! Not at all like the original surface. This wasn't happening last night, but you know how things get when it is late in the day and you have been working 12 hours straight trying to make sense of things. All kinds of strangeness sets in. Anyway, based on today's tests, I wouldn't use the original Krig2D routine.

In terms of fidelity to the original data, here are the results I get when I use the 21x21 data set. I still don't totally understand the difference between a semivariogram and a covariance matrix, but I guess cgKrig2D is using the former and Krig2D_Fast is using the latter.

```
MinMax:      0.000000      14.1421 ; Original data
MinMax:      -2.16058      15.0750 ; Original Krig2D code
MinMax: -2.38419e-006      14.1422 ; cgKrig2D
MinMax: -7.15256e-006      14.2410 ; Krig2D_Fast
```

I did add other kriging models to cgKrig2D, but I couldn't make sense of them yesterday. I'll look at this again today in light of the changes we have made to the 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 thue. ("Perhaps thou speakest truth.")
