
Subject: Re: Vectorize procedure

Posted by [Kenneth P. Bowman](#) on Thu, 10 Dec 2009 15:25:13 GMT

[View Forum Message](#) <> [Reply to Message](#)

In article

<041dea05-b2af-435a-b9e9-7ca360287c69@p35g2000yqh.googlegroups.com>,
Romolo Politi <romolo.politi@gmail.com> wrote:

> Hi,
> I'm vectorizing a procedure in order to delete three for loops and
> increase the speed.
> I have to calculating the interpolation on a regular grid.
> newy=interpol(y,x,newx,/spline)
> in the loop manner for each index of the loop there are three
> different arrays .
> in the vectorize manner I have three 3D matrices, But I do not found a
> way to calculate the interpolation matrix.
> Any one have suggestions?
>
> Thanks
>
> Romolo

I'm afraid that you have not clearly explained your problem. Are you trying to do 3-D cubic spline interpolation? As far as I know, standard IDL does not contain a multi-dimensional spline interpolation procedure, only the 1-D functionality available in INTERPOL, SPLINE, and SPL_INTERP. You can do vectorized multi-dimensional *linear* interpolation by using INTERPOLATE.

The Advanced Math and Statistics package (extra cost) includes the IMSL spline interpolation functions. You can view the help files and see if that is what you need.

<http://127.0.0.1:60523/help/index.jsp>

Ken Bowman

Subject: Re: Vectorize procedure

Posted by [Romolo Politi](#) on Thu, 10 Dec 2009 18:03:02 GMT

[View Forum Message](#) <> [Reply to Message](#)

On 10 Dic, 16:25, "Kenneth P. Bowman" <k-bow...@null.edu> wrote:

> In article
> < 041dea05-b2af-435a-b9e9-7ca360287...@p35g2000yqh.googlegroup s.com >,
> Romolo Politi <romolo.pol...@gmail.com> wrote:
>

```

>> Hi,
>> I'm vectorizing a procedure in order to delete three for loops and
>> increase the speed.
>> I have to calculating the interpolation on a regular grid.
>> newy=interpol(y,x,newx,/spline)
>> in the loop manner for each index of the loop there are three
>> different arrays .
>> in the vectorize manner I have three 3D matrices, But I do not found a
>> way to calculate the interpolation matrix.
>> Any one have suggestions?
>
>> Thanks
>
>> Romolo
>
> I'm afraid that you have not clearly explained your problem. Are you
> trying to do 3-D cubic spline interpolation? As far as I know,
> standard IDL does not contain a multi-dimensional spline interpolation
> procedure, only the 1-D functionality available in INTERPOL, SPLINE, and
> SPL_INTERP. You can do vectorized multi-dimensional *linear* interpolation
> by using INTERPOLATE.
>
> The Advanced Math and Statistics package (extra cost) includes the IMSL
> spline interpolation functions. You can view the help files and
> see if that is what you need.
>
> http://127.0.0.1:60523/help/index.jsp
>
> Ken Bowman

```

Sorry for the not clarity.

my situation is

```

x=fltarr(3200,201,201)
y=fltarr(3200,201,201)
newx=fltarr(799,201,201)
newy=fltarr(799,201,201)
for i=0,200 do begin
  for j=0,200 do begin
    newy[* ,i,j]=interpol(y[* ,i,j],x[* ,i,j],newx[* ,i,j],/SPLine)
  endfor
endfor

```

I like vectorize this loop.

Thanks, and sorry again

Romolo Politi

Subject: Re: Vectorize procedure

Posted by [Kenneth P. Bowman](#) on Thu, 10 Dec 2009 20:00:41 GMT

[View Forum Message](#) <> [Reply to Message](#)

In article

<b599bcd-fd42e-45d8-b643-7ea4620fd7ed@k19g2000yqc.googlegroups.com>,

Romolo Politi <romolo.politi@gmail.com> wrote:

```
> Sorry for the not clarity.  
> my situation is  
> x=fltarr(3200,201,201)  
> y=fltarr(3200,201,201)  
> newx=fltarr(799,201,201)  
> newy=fltarr(799,201,201)  
> for i=0,200 do begin  
>   for j=0,200 do begin  
>     newy[* ,i,j]=interpol(y[* ,i,j],x[* ,i,j],newx[* ,i,j],/SPLine)  
>   endfor  
> endfor
```

You are doing ~40,000 separate spline fits. I don't think there is any way to vectorize this, as each fit is an independent problem. Spline fitting is a coupled problem, so you cannot, for example, simply string your segments together into one long fit.

If the newx[* ,i,j] are the same for all i and j, you can probably get a significant speed-up by calling SPL_INIT once and then calling SPL_INTERP for each i and j. Since you are storing newx separately for each i and j, I gather this is not the case.

Your program is accessing memory efficiently, but you will get a small speed up by saying

```
newy[0,i,j]=interpol(y[* ,i,j],x[* ,i,j],newx[* ,i,j],/SPLine)
```

You might also try the other built in spline functions: SPLINE, SPLINE_P, SPL_INIT and SPL_INTERP to see whether one is faster.

Finally, I notice that you are downsampling your data substantially. Are you sure you really need to use a spline fit to do that?

Ken Bowman
