Subject: Re: avoiding for loop when calculating median
Posted by Alex Schuster on Mon, 02 Feb 1998 08:00:00 GMT
View Forum Message <> Reply to Message

George McCabe wrote:

> Thanks for your inputs, Alex.
>
> following an earlier hint on the group I wrote the loop like you
> describe, but without REFORM'ing the matrix. To be honest my matrix is

Actually, it's TRANSPOSE, not REFORM!

> a large data cube, but I chose a 2D example to make the description less
> opaque. The reduction in execution time was measured - 20%, which on 45
> seconds is significant. When you say HUGE is that the scale of the
> increase you experienced.

I did something like that:

```
IDL> n = 2000L
IDL> m = randomu( seed, n, n )
IDL> m2 = transpose( m )
IDL> t=systime(1) & for i = 0, n-1 do c(i) = median(m(i,*)) & print,
systime(1)-t, format='(F4.1)'
 3.0
IDL> t=systime(1) & for i = 0, n-1 do c(i) = median(m2(*,i)) & print,
systime(1)-t, format='(F4.1)'
 0.6
```

That's a factor of five, and this is HUGE. Of course, the TRANSFORMing
has to be done, too, this also takes a second or so.
With 3d data it's not that easy. Does the data need to be in this form?
Changing the x, y, and z direction could speed it up.

Using 1d only, I get this:

```
IDL> index = lindgen( n )
IDL> t=systime(1) & for i = 0, n-1 do c(i) = median(m(index+i*n)) &
print, systime(1)-t, format='(F4.1)'
 2.2
```

Whoops, this time it's faster than the original routine.

Alex
--
Alex Schuster   Wonko@weird.cologne.de   PGP Key available

alex@pet.mpin-koeln.mpg.de