



Subject: Re: avoiding for loop when calculating median  
Posted by [a2652099](#) on Fri, 30 Jan 1998 08:00:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

george.mccabe@gssc.nasa.gov (George McCabe) wrote:

```
> calculating the median of each i set of values M(i,*), I would like to
> 'vectorize' the operation to avoid a FOR loop like the following -
>
> for cnt=0, idim-1 do begin
>   calcmedian(cnt)=median(mat(cnt,*))
> end
>
> but, you can't do -
>
> cnt=indgen(idim)
> calcmedian(cnt)=median(M(cnt,*))
>
> and get the desired result, where each element of calcmedian() will be
> assigned the same value.
>
> can it be done without a FOR loop?
```

I don't think so. But maybe the routine above can be made a bit faster?

I assume that mat is an idim x jdim array. mat(cnt,\*) gives the elements  
no. cnt, cnt+idim, cnt+2\*idim etc., I guess it's easier to use  
mat2=reform(mat), so all the elements are in line.

Another idea:

```
index = lindgen( jdim )
for cnt = 0, idim-1 do begin
  calcmedian(cnt) = median( mat2(cnt*jdim+index) )
end
```

Whether there will be a measurable improvement in speed will depend on  
the sizes of idim and jdim.

Alex

--

Alex Schuster    Wonko@weird.cologne.de    PGP Key available  
alex@pet.mpin-koeln.mpg.de

---

Subject: Re: avoiding for loop when calculating median  
Posted by [Michael Werger](#) on Fri, 30 Jan 1998 08:00:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

George McCabe wrote:

```
> for cnt=0,idim-1 do begin
>   calcmedian(cnt)=median(mat(cnt,*))
> end
> ...
>
> can it be done without a FOR loop?
>
> Thank you, George
>
> ....
```

If it is only due to speed that you do not want to use the FOR loop,  
just write it in a single line:  
for cnt=0, idim-1 do calcmedian(cnt) = median(mat(cnt,\*))

--

Michael Werger                    ESA ESTEC & Praesepe B.V.  
Astrophysics Division   mwerger@estec.esa.nl  
Postbus 299   http://astro.estec.esa.nl  
2200 AG Noordwijk   +31 71 565 3783 (Voice)  
The Netherlands   +31 71 565 4690 (FAX)

---

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

---

Subject: Re: avoiding for loop when calculating median  
Posted by [George McCabe](#) on Mon, 02 Feb 1998 08:00:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

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 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 didn't REFORM the data because I figured it wasn't worth the extra step as the cube can be addressed directly as 1D. But if yours ran much faster including the step to REFORMAT then I'll try it.

Thanks again, George

