Subject: Re: Simple? problem
Posted by James Kuyper on Tue, 09 Apr 2002 17:23:47 GMT
View Forum Message <> Reply to Message

Ivan Valtchanov wrote:

```
>  Hi,
>
>  I have a small problem concerning a good programming techniques, so here it is:
>  ----------------------
>  pro test,nx,ny,image
>
>  ; Make some random X and Y arrays
>  x = randomu(s,1000)
>  y = randomu(s,1000)
>
>  ; now give arbitrary weights for each point
>  w = randomu(s,1000)/100.0
>  ; take the square
>  w2=2.0*w*w
>
>  ; I want to construct a 2-D image with a specified dimension
>  image = fltarr(nx,ny)
>
>  ; I want to sum up the contribution of each point as Gaussian
>  ; with width=w to the image pixels
>
>  for i=0, nx-1 do begin
>   xgi = i/float(nx)
>   dx = x-xgi
>   for j=0, ny-1 do begin
>    ygi = j/float(ny)
>    dy = y-ygi
>    dr2 = dx*dx+dy*dy
>    arg = -dr2/w2 > (-20.0) ; to avoid overflows
```

The only thing that statement prevents is underflows. It prevents arg from ever being smaller than -20.0. exp(-20.0) is an extremely small number, not an extremely large one. Ordinarily, underflows aren't much of a problem.

```
>    image[i,j] =  total(exp(arg))
```

'arg' is a scalar, so exp(arg) is also a scalar. Therefore, the total() doesn't do anything useful. You could just as well say exp(arg).

```
>   endfor
>  endfor
```

>
> return
> end
> ---------------------------
>
> This is obviously quite unoptimised - two cycles etc. Do you have any ideas, references or do you know if it is already solved in IDL?
>
> I have looked for something similar in David Fanning pages and IDL astronomical libraries but I couldn't find something to adapt, maybe I have missed it?

```
image =  exp((x-(indgen(nx)#replicate(1.0/float(nx),ny))^2 - $
 (y-replicate(1.0/float(ny),nx)#indgen(ny))^2))/w2)
```
