
Subject: Re: newbie sees iterative solution, wants to know the IDL Way

Posted by [Tom Roche](#) on Mon, 21 Jul 2008 19:48:58 GMT

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

Chris Mon, 21 Jul 2008 02:08:49 -0700 (PDT) (rearranged)

> A couple of things:

> First, you have your diagram mislabeled. In IDL, the first index is
> the column.

Does "column" always means dimension=2 in multi-dimensional arrays?
I'll be working mostly in 3- and 4-space.

> I will assume you still want to sum over time,

Correct, and it's still the first dimension in the netCDFs I'm using.

> To do this, try:

> 1) sz=size(input)
> 2) summing=total(input,1,/nan)
> 3) summing=rebin(reform(summing,1,sz[2],sz[3]),sz[1],sz[2],sz[3])
> 4) output=input/summing
> 5) badval=where(output eq 0, ct)
> 6) if ct ne 0 then output[badval]=0

> Explanation:

in previous post. The next newbie to happen along can try this:

> the correct diagrams (and answers) will now be:

> (k=0) i=0 i=1
> j=0: 0 1
> j=1: 2 3

> (k=1)
> j=0: 4 5
> j=1: 6 7

```
IDL> input=FLTARR(2,2,2)
IDL> ; first index is the column!
IDL> input[0,0,0]=0
IDL> input[1,0,0]=1
IDL> input[0,1,0]=2
IDL> input[1,1,0]=3
IDL> input[0,0,1]=4
IDL> input[1,0,1]=5
IDL> input[0,1,1]=6
```

```
IDL> input[1,1,1]=7
```

FWIW the order of the subscripts is the reverse of the digits in binary counting, which was the order I used the first time I tried to set this up :-)

```
IDL> PRINT, input
```

```
0.00000  1.00000
2.00000  3.00000
```

```
4.00000  5.00000
6.00000  7.00000
```

```
IDL> sz=SIZE(input)
```

```
IDL> summing=TOTAL(input,1,/nan)
```

```
IDL> summing=REBIN(REFORM(summing,1,sz[2],sz[3]),sz[1],sz[2],sz[3 ])
```

I'm still reading

http://www.dfanning.com/tips/rebin_magic.html

Scary stuff :-)

```
IDL> output=input/summing
```

```
IDL> badval=WHERE(output eq 0, ct)
```

```
IDL> IF ct ne 0 THEN output[badval]=0
```

```
IDL> PRINT, output
```

```
0.00000  1.00000
0.400000  0.600000
```

```
0.444444  0.555556
0.461538  0.538462
```

```
>      0  1
```

```
>      2/5 3/5
```

```
>      4/9 5/9
```

```
>      6/13 7/13
```

Indeed! thanks, Tom Roche <Tom_Roche@pobox.com>
