Subject: Re: rebin question
Posted by hradilv.nospam on Fri, 22 Mar 2002 17:11:17 GMT
View Forum Message <> Reply to Message

print, round(rebin(float([5,5,5,5,4]),1))

Hassle?

Maybe you could write a function. Which leads me to a new question:

Is it possible to define a function or procedure in IDL that can take
an arbitrary number of arguments, e.g.:

```
function my_rebin, a, arg1, arg2, ...

 return, round( rebin( float(a), arg1, arg2, ... ) )
end
```

On Fri, 22 Mar 2002 11:58:41 -0500, Jonathan Joseph <jj21@cornell.edu>
wrote:

> I figured I would use rebin to downsample an image by averaging the
> pixels in blocks of specified size. What I discovered, was that for
> integer type images, rebin averages the pixels, but then instead of
> rounding to the nearest integer, simply takes the integer part of
> the average. Hence:
>
> print, rebin([5,5,5,5,4], 1)
>
> gives the result of 4, not 5 which is what I would like. I suppose
> this is done for speed - to work around the problem, I need to convert
> to a floating point type, do the rebin, then round, then convert back
> to the proper integer type - a hassle.
>
> But, I would really like a more generic way of doing downsampling
> of this sort, without the high overhead of a loop. Apart from
> taking the mean of a block of pixels, I would also like the option
> of downsampling using the median of a block of pixels, or using the
> mean of a block of pixels disregarding the farthest outlier (or
> n outliers).
>
> Has anyone written IDL code to do downsampling in a more generalized
> way than rebin, or have any clever ideas about how to do it quickly?
>
> Thanks