Subject: Re: Approximate convolution - for loop problem
Posted by Sam on Sun, 21 Dec 2008 20:01:39 GMT
View Forum Message <> Reply to Message

Hi David, unfortunately shift() does not do the business for me, as
these two examples below show. So I'm still a bit stumped here.

```
; Array operation I'm trying to execute.
a=[1.,2.,3.,4.]
for ii=1,3 do a[ii] += 0.5*a[ii-1]
print,a
 1.00000     2.50000     4.25000     6.12500

; Attempt to perform this operation with shift()
a=[1.,2.,3.,4.]
a += 0.5*shift(a,-1)
print,a
 2.00000     3.50000     5.00000     4.50000
```

On Dec 21, 7:03 pm, David Fanning <n...@dfanning.com> wrote:
> samuel.le...@gmail.com writes:
>> Hello everyone, I'm trying to execute a 1-d convolution of an array,
>> signal.
>
>> Using an analytic approximation, obtaining the convolved bolometer
>> signal, bolo_signal, at time step ii, is given by the following:
>
>> nsamp=n_elements(signal)
>> const1 = exp(-tsamp/taubolo)
>> const2 = 1.-const1
>
>> bolo_signal = const2*signal
>> for ii= 1L,nsamp-1L do begin
>>   bolo_signal[ii] += const1*bolo_signal[ii-1]
>> endfor
>
>> where tsamp and taubolo are scalars. Is there any way to avoid the for
>> loop in this case? The hope is to speed up the execution.
>
> I think this gives you the same results:
>
>   bolo_signal += const1 * shift(bolo_signal,-1)
>

> Cheers,
>
> David
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming:http://www.dfanning.com/
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")