Subject: Re: chunk indexing like
Posted by pgrigis on Fri, 08 Jan 2010 16:16:33 GMT
View Forum Message <> Reply to Message

On Jan 8, 9:12 am, Wox <s...@nomail.com> wrote:
> On Fri, 08 Jan 2010 14:08:45 +0100, Wox <s...@nomail.com> wrote:
>> Hi IDLers,
>
>> I know how to get
>> from [2,3,1,0,5] to [0,0,1,1,1,2,4,4,4,4,4] (i.e. chunk indexing)
>> but how do I get
>> from [2,3,1,0,5] to [0,1,0,1,2,0,0,1,2,3,4]
>> without loops that is.
>
>> I have been juggling with histogram and total(.../cumulative) but I
>> can figure this one out.
>
>> Any ideas?
>
> Ok, I got something working:
>
> n=[2,3,1,0,5]
> h=histogram(total(n>0,/CUMULATIVE,/int)-1,/BINSIZE,MIN=0,REVERSE_INDICES=ri)
> nh=n_elements(h)
> chinkind=ri[0:nh-1]-ri[0]
> ind2=where([1,chinkind[1:*]-chinkind[0:nh-2]] ne 0)
> print,lindgen(nh)-ind2[chinkind]
>
> IDL> 0 1 0 1 2 0 0 1 2 3 4

I wonder how does the IDL-magic version above compares
with the more boring for-loop version below in execution
speed?
I.e. is it worth to go the histogram-way, rewarding
but fraught with danger? ;)

Ciao,
Paolo

```
n=[2,3,1,0,5]

res=intarr(total(n))
indexarr=findgen(max(n))
i2=0

.run
FOR i=0,n_elements(n)-1 DO BEGIN
  IF n[i] GE 1 THEN BEGIN
```

```
    res[i2:i2+n[i]-1]=indexarr[0:n[i]-1]
    i2=i2+n[i]
  ENDIF
ENDFOR
end
```