Subject: Removing specific elements in an array
Posted by thtran296 on Tue, 27 Jun 2017 20:17:00 GMT
View Forum Message <> Reply to Message

Hello everybody,
So I have an array that looks like this:
x = [4,5,17,18,30,38,50,51,70]

The goal is to get rid of the element that is only 1 unit larger than the previous one.
The ideal output should look like:
x = [4,17,30,38,50,70]   (since 5,18, and 51 are deleted)

I have tried all sorts of loops but it is not working. Here's my own attempt.

```
y = [4,5,8,15,30,31,45,70]
y2 = [y,0]
for i = 0,6 do begin
  if y2(i) ne 0 then begin
    if y2(i+1) - y2(i) le 1 then begin
      remove, i, y
    endif
  endif
endfor
```

Please help. I appreciate it.

Subject: Re: Removing specific elements in an array
Posted by Michael Galloy on Tue, 27 Jun 2017 21:32:40 GMT
View Forum Message <> Reply to Message

On 6/27/17 2:17 PM, thtran296@gmail.com wrote:

```
>  Hello everybody,
>  So I have an array that looks like this:
>  x = [4,5,17,18,30,38,50,51,70]
>
>  The goal is to get rid of the element that is only 1 unit larger than the previous one.
>  The ideal output should look like:
>  x = [4,17,30,38,50,70]          (since 5,18, and 51 are deleted)
>
>  I have tried all sorts of loops but it is not working. Here's my own attempt.
>  y = [4,5,8,15,30,31,45,70]
>  y2 = [y,0]
>  for i = 0,6 do begin
>     if y2(i) ne 0 then begin
>       if y2(i+1) - y2(i) le 1 then begin
>         remove, i, y
>       endif
>     endif
```

> endfor
>
> Please help. I appreciate it.
>

How about this?

```
IDL> x = [4, 5, 17, 18, 30, 38, 50, 51, 70]
IDL> remove_ind = where(x[1:*] - x[0:-2] eq 1, count) + 1L
IDL> keep_ind = mg_complement(remove_ind, n_elements(x))
IDL> print, remove_ind
           1           3           7
IDL> print, keep_ind
           0           2           4           5           6           8
IDL> print, x[remove_ind]
       5      18      51
IDL> print, x[keep_ind]
       4      17      30      38      50      70
```

MG_COMPLEMENT is available here:

https://github.com/mgalloy/mglib/blob/master/src/indices/mg_ complement.pro

But it is fairly short:

```
function mg_complement, indices, n, count=ncomplement
  compile_opt strictarr, strictarrsubs

  all = bytarr(n)
  valid_indices = where(indices ge 0 and indices lt n, n_valid)
  if (n_valid gt 0L) then all[indices[valid_indices]] = 1B
  return, where(all eq 0B, ncomplement)
end
```

Mike
--
Michael Galloy
www.michaelgalloy.com
Modern IDL: A Guide to IDL Programming (http://modernidl.idldev.com)

Subject: Re: Removing specific elements in an array
Posted by gombgg on Tue, 27 Jun 2017 21:57:44 GMT
View Forum Message <> Reply to Message

Mike beat me to it. My response is equivalent:

You shouldn't need loops.

You didn't define what you want to do with the first element, which has no predecessor. You also didn't specify whether your output array is allowed to have neighbouring elements that differ by one. (eg, what should happen if x=[5,10,11,11,11,20] ?)

So long as the last element isn't one less than the first element, this will work:

```
result=x[where((x-shift(x,1)) ne 1,/null)]
```

If you want to make sure the last element doesn't mess things up, this should work:

```
inds=where((x-shift(x,1)) ne 1,count)
if count eq 0 then result=x[0] else begin
  if inds[0] ne 0 then inds=[0,inds]  ;make sure the first element is included
  result=x[inds]
endelse
```