Subject: rebinning two arrays? or re: finding exclusive elements between 2 not-quite identical arrays?
Posted by havok2063 on Fri, 28 Nov 2014 16:54:10 GMT
View Forum Message <> Reply to Message

So this post is related to my original post here, but it's not quite working out.
https://groups.google.com/forum/#!topic/comp.lang.idl-pvwave /CtiGTuNqD10

...

So I have two arrays where I need to extract the elements in one array not in the other, but the elements are not quite exact to one another.

A = [309, 313, 318, 322, 327, 331, 336, 340]
B = [305, 309, 314, 318, 323, 327, 332, 336, 341, 345]

B is the true array, and A is the "found" array that is missing some elements. I want to find all elements missing from A that exist in B.

The solution I adopted, from the previous post, is something like

```
diff = 2 ; offset between integers
temp = value_locate(A,B)
missloc = where(B-A[temp] gt diff and A[temp+1]-B gt diff, misscount)
```

This mostly works, however it does not work when the elements that are missing are the edges, i.e. the first or last element.

I can't seem to get value_locate to chunk the two together the way I want. Is there a way to rebin A so that the elements in A that are the same match those in B, e.g. 309=309, 313 changes to 314, 322 changes to 323, etc? So the elements match up, so then I could do a simple thing like this

```
missloc = where(histogram(A, omin=om) eq 0 and histogram(B, min=om) ne 0) + om
```

However, I tested this out by forcing the like elements to be the same, and this also does not work when the missing elements are the first and last elements. It only seems to work when the missing elements are internal, i.e when there is at least 1 common element first.

Any suggestions on a more robust solution for either problem?

Thanks, Brian

Subject: Re: rebinning two arrays? or re: finding exclusive elements between 2 not-quite identical arrays?
Posted by Heinz Stege on Fri, 28 Nov 2014 18:49:37 GMT

Hi Brian.

On Fri, 28 Nov 2014 08:54:10 -0800 (PST), Brian Cherinka wrote:

>
> So this post is related to my original post here, but it's not quite working out.
> https://groups.google.com/forum/#!topic/comp.lang.idl-pvwave /CtiGTuNqD10
>
> ...
>
> So I have two arrays where I need to extract the elements in one array not in the other, but the elements are not quite exact to one another.
>
> A = [309, 313, 318, 322, 327, 331, 336, 340]
> B = [305, 309, 314, 318, 323, 327, 332, 336, 341, 345]
>
> B is the true array, and A is the "found" array that is missing some elements. I want to find all elements missing from A that exist in B.
>
> The solution I adopted, from the previous post, is something like
>
> diff = 2 ; offset between integers
> temp = value_locate(A,B)
> missloc = where(B-A[temp] gt diff and A[temp+1]-B gt diff, misscount)
>
> This mostly works, however it does not work when the elements that are missing are the edges, i.e. the first or last element.
>
> [...]

The missing elements on the left edge were recognized by the additional two lines in my draft. But not the missing elements on the right edge. This is true.

However I would not use the histogram function to overcome this. I would go a more easy way, which I expect to be both, faster and less memory consuming. It is only a little change to the code above:

```
A = [309, 313, 318, 322, 327, 331, 336, 340]
B = [305, 309, 314, 318, 323, 327, 332, 336, 341, 345]
diff = 2 ; offset between integers
temp = value_locate(A,B)
missloc = where((B-A[temp] gt diff and A[temp+1]-B gt diff) or $
  a[0]-b gt diff or $   ; elements on the left edge
  b-a[n_elements(a)-1] gt diff, $   ; right edge
  misscount)
if misscount ge 1 then print,b[missloc]
```

Good luck again, Heinz

Subject: Re: rebinning two arrays? or re: finding exclusive elements between 2 not-quite identical arrays?
Posted by havok2063 on Fri, 28 Nov 2014 19:05:50 GMT
View Forum Message <> Reply to Message

Hi Heinz,

Thanks so much. That's perfect. Sorry I missed that other line in your original post. Thanks for pointing it out. Things make much more sense now. Thanks for your help.

Cheers, Brian

On Friday, November 28, 2014 1:49:23 PM UTC-5, Heinz Stege wrote:
> Hi Brian.
>
> On Fri, 28 Nov 2014 08:54:10 -0800 (PST), Brian Cherinka wrote:
>
>>
>> So this post is related to my original post here, but it's not quite working out.
>> https://groups.google.com/forum/#!topic/comp.lang.idl-pvwave /CtiGTuNqD10
>>
>> ...
>>
>> So I have two arrays where I need to extract the elements in one array not in the other, but the elements are not quite exact to one another.
>>
>> A = [309, 313, 318, 322, 327, 331, 336, 340]
>> B = [305, 309, 314, 318, 323, 327, 332, 336, 341, 345]
>>
>> B is the true array, and A is the "found" array that is missing some elements. I want to find all elements missing from A that exist in B.
>>
>> The solution I adopted, from the previous post, is something like
>>
>> diff = 2 ; offset between integers
>> temp = value_locate(A,B)
>> missloc = where(B-A[temp] gt diff and A[temp+1]-B gt diff, misscount)
>>
>> This mostly works, however it does not work when the elements that are missing are the edges, i.e. the first or last element.
>>
>> [...]
>
> The missing elements on the left edge were recognized by the

> additional two lines in my draft. But not the missing elements on the
> right edge. This is true.
>
> However I would not use the histogram function to overcome this. I
> would go a more easy way, which I expect to be both, faster and less
> memory consuming. It is only a little change to the code above:
>
> A = [309, 313, 318, 322, 327, 331, 336, 340]
> B = [305, 309, 314, 318, 323, 327, 332, 336, 341, 345]
> diff = 2 ; offset between integers
> temp = value_locate(A,B)
> missloc = where((B-A[temp] gt diff and A[temp+1]-B gt diff) or $
> a[0]-b gt diff or $; elements on the left edge
> b-a[n_elements(a)-1] gt diff, $; right edge
> misscount)
> if misscount ge 1 then print,b[missloc]
>
> Good luck again, Heinz
