
Subject: Re: compare 2-d array with vector

Posted by [Helder Marchetto](#) on Mon, 24 Sep 2012 08:20:54 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Monday, September 24, 2012 9:32:17 AM UTC+2, Danxia wrote:

> Hi all, I'm having problem when comparing a 2-d array with a given vector. For example, if giving a 2-d array

>

> a=[[1,1,1],

>

> [2,2,2],

>

> [3,3,3],

>

> [4,4,4]]

>

> and a vector b=[1,2],

>

> how can I get all the subscripts of array a that's equal to any element in vector b, which in this example is [0,1,2,3,4,5]. Please let me know if I failed to make this clear. Looking forward to any reply. Thank you very much!

Hi,

I think you should go for the histogram function. It is counter-intuitive at first, but is very fast and works well.

Example on how to start out:

```
IDL> a=[[1,1,1],[2,2,2],[4,4,4],[5,5,5]]
```

```
IDL> b = [2,4]
```

```
IDL> ha = histogram(a, REVERSE_INDICES=ra, OMAX=oMax, OMIN=oMin)
```

```
IDL> hb = histogram(b, MAX=oMax, MIN=oMin)
```

```
IDL> print, ha,hb
```

```
      3      3      0      3      3
      0      1      0      1      0
```

If you then loop for non zero values in the product of ha and hb then you can trace back the common values. If you use the reverse indices, you can find out where such values were in the original array.

Something like this (you have to make some extra controls for non existing matches and similar):

```
IDL> CommonPos = where(ha*hb)
```

```
IDL> FOR I=0,N_ELEMENTS(CommonPos)-1 DO PRINT,
```

```
r[r[CommonPos[I]]:r[CommonPos[I]+1]-1]
```

```
      3      4      5
      6      7      8
```

Use array_indices if you want the positions in two dimensions.

Hope it helps,

Helder
