Subject: Re: nearest number

Posted by bressert@gmail.com on Sun, 06 Aug 2006 16:58:53 GMT

View Forum Message <> Reply to Message

I thought about the problem more and realized a simple solution:

```
smallest = min(abs(array - floating_value))
index = where((array - floating_value) eq smallest, count)
```

That should work for me without any problems I think

Cheers,

Eli

bressert@gmail.com wrote:

Subject: Re: nearest number

Posted by James Kuyper on Sun, 06 Aug 2006 19:34:12 GMT

View Forum Message <> Reply to Message

bressert@gmail.com wrote:

```
>  I thought about the problem more and realized a simple solution:
>
>  smallest = min(abs(array - floating_value))
>  index = where((array - floating_value) eq smallest, count)
```

Almost: that should be "abs(array-floating_value) eq smallest".

I'd be inclined to pull out the common code, and write it as

```
diff = abs(array-floating_value)
index = where(array eq min(array))
```

Subject: Re: nearest number

Posted by Timm Weitkamp on Mon, 07 Aug 2006 13:13:59 GMT

View Forum Message <> Reply to Message

kuyper@wizard.net wrote:

```
>  bressert@gmail.com wrote:
>>  I thought about the problem more and realized a simple solution:
>>
>>  smallest = min(abs(array - floating_value))
>>  index = where((array - floating_value) eq smallest, count)
```

>
> Almost: that should be "abs(array-floating_value) eq smallest".
>
> I'd be inclined to pull out the common code, and write it as
>
> diff = abs(array-floating_value)
> index = where(array eq min(array))

Or simply,
  void = min(abs(array - floating_value), index)

Timm

Subject: Re: nearest number
Posted by Benjamin Hornberger on Mon, 07 Aug 2006 13:18:50 GMT
View Forum Message <> Reply to Message

bressert@gmail.com wrote:
> Hello,
>
> Im writing a bit of IDL code where a large array of floating numbers is
> compared to one floating number. The aim of comparing the array to the
> floating number to find which the values "closest" match.

IDL provides the VALUE_LOCATE function for this purpose. The array must
be monotonic, but of course you can sort it if necessary.

Benjamin

Subject: Re: nearest number
Posted by btt on Mon, 07 Aug 2006 13:52:34 GMT
View Forum Message <> Reply to Message

bressert@gmail.com wrote:
> Hello,
>
> Im writing a bit of IDL code where a large array of floating numbers is
> compared to one floating number. The aim of comparing the array to the
> floating number to find which the values "closest" match. I had the
> idea that I could write an algorithm that starts with the following
>
> ===========
> index = where(array eq floating_value, count)
>
> if (size(index, /n_elements) eq 1) AND (index eq -1) then begin ...

> ===========
>
>
> This bit would find the index numbers of the array that would be
> exactly the same to the floating number. If there is no number the
> where function returns a -1. From this point, I could try and round off
> the floating number and numbers in the array with smaller decimal
> places until there is a match. Would this be the best approach? Or is
> there already a built code in IDL or from someone else that does this?
> Any help is appreciated. Thanks!

Hi,

Check out the following thread - JD Smith discusses a couple of
approaches to solve this kind of problem.

http://tinyurl.com/e6ke7

Cheers,
Ben

Subject: Re: nearest number
Posted by mchinand on Mon, 07 Aug 2006 15:09:14 GMT

View Forum Message <> Reply to Message

In article <1154956439.920334.266030@m73g2000cwd.googlegroups.com>,
Timm Weitkamp <timm.weitkamp@iss.fzk.de> wrote:
> kuyper@wizard.net wrote:
>> bressert@gmail.com wrote:
>>> I thought about the problem more and realized a simple solution:
>>>
>>> smallest = min(abs(array - floating_value))
>>> index = where((array - floating_value) eq smallest, count)
>>
>> Almost: that should be "abs(array-floating_value) eq smallest".
>>
>> I'd be inclined to pull out the common code, and write it as
>>
>> diff = abs(array-floating_value)
>> index = where(array eq min(array))
>
> Or simply,
>   void = min(abs(array - floating_value), index)
>

I was going to post the same thing last night, but this optional argument to MIN will only return
the index of the first element that is the minimum of the array even if there are multiple

occurrences of the minimum value. Which may be alright, depending on what the original poster wants.

--Mike

--
Michael Chinander
m-chinander@uchicago.edu
Department of Radiology
University of Chicago

Subject: Re: nearest number
Posted by James Kuyper on Mon, 07 Aug 2006 16:24:26 GMT

View Forum Message <> Reply to Message

kuyper@wizard.net wrote:
> bressert@gmail.com wrote:
>> I thought about the problem more and realized a simple solution:
>>
>> smallest = min(abs(array - floating_value))
>> index = where((array - floating_value) eq smallest, count)
>
> Almost: that should be "abs(array-floating_value) eq smallest".
>
> I'd be inclined to pull out the common code, and write it as
>
> diff = abs(array-floating_value)
> index = where(array eq min(array))

Agh! That should, of course, have been

    index = where(diff eq min(diff))