Subject: Re: transforming an array where some values can't
Posted by S Penzes on Thu, 05 Feb 1998 08:00:00 GMT
View Forum Message <> Reply to Message

How about

```
function xform, input
temp = alog(input)
temp(where(finite(temp) EQ 0))=1e-7
return, temp
end
```

Works for me.

bowler@eisner.decus.org wrote:

> I'm trying to write a function that will transform an arbitrary array by taking
> hte natural log of every element in the array.
>
> What I have is
>
>
```
>         function xform, input
>         temp = alog(input)  ; I realize I could combine these 2 statements
>         return, temp        ; but it makes debugging easier
>         end
```
>
> Unfortunately, I can't guarentee that there won't be some elements that are
> zero and thus invalid arguments to alog. what's the "most efficient" way to
> take the alog of any element that's greater than 0 and set the value of ony
> that are less than or equal to 0 to some small value (1e-7 for example)?
>
> TIA,
> Bruce

--
Steven Penzes (Steven.nospamPenzes@dres.dnd.ca)
Note: remove "nospam" from Reply-To and .signature

Subject: Re: transforming an array where some values can't
Posted by meron on Thu, 05 Feb 1998 08:00:00 GMT
View Forum Message <> Reply to Message

In article <1998Feb5.094807.1@eisner>, bowler@eisner.decus.org writes:
> I'm trying to write a function that will transform an arbitrary array by taking

> hte natural log of every element in the array.
>
> What I have is
>
> function xform, input
> temp = alog(input) ; I realize I could combine these 2 statements
> return, temp   ; but it makes debugging easier
> end
>
> Unfortunately, I can't guarentee that there won't be some elements that are
> zero and thus invalid arguments to alog. what's the "most efficient" way to
> take the alog of any element that's greater than 0 and set the value of ony
> that are less than or equal to 0 to some small value (1e-7 for example)?
>
Just do
 function xform, input
 eps = ..some small value here, can make it input parameter too ..
 temp = alog(input > eps) ; I realize I could combine these 2 statements
 return, temp   ; but it makes debugging easier
 end

Mati Meron | "When you argue with a fool,
meron@cars.uchicago.edu | chances are he is doing just the same"

Subject: Re: transforming an array where some values can't
Posted by offenbrg on Fri, 06 Feb 1998 08:00:00 GMT
View Forum Message <> Reply to Message

bowler@eisner.decus.org writes:

> I'm trying to write a function that will transform an arbitrary array by taking
> hte natural log of every element in the array.

> What I have is

> function xform, input
> temp = alog(input) ; I realize I could combine these 2 statements
> return, temp   ; but it makes debugging easier
> end

> Unfortunately, I can't guarentee that there won't be some elements that are
> zero and thus invalid arguments to alog. what's the "most efficient" way to
> take the alog of any element that's greater than 0 and set the value of ony
> that are less than or equal to 0 to some small value (1e-7 for example)?

I think what you want is this:

```
function xform, input
temp = alog(input > 1e-7)
return, temp
end
```

The ">" operator in this case sets a floor on the values of "input"...any
value less than 1e-7 is replaced with 1e-7.

Good luck,
Joel
--
"...And I am unanimous in this" - Mrs. Slocumbe

Joel D Offenberg	Joel.D.Offenbrg.1@gsfc.nasa.gov
Hughes STX, NASA/GSFC/LASP & STScI	UIT, NGST programmer & sysadmin

---

Subject: Re: transforming an array where some values can't
Posted by alpha on Sun, 08 Feb 1998 08:00:00 GMT
View Forum Message <> Reply to Message

offenbrg@fondue.gsfc.nasa.gov (Joel D. Offenberg) writes:
> bowler@eisner.decus.org writes:

>> What I have is

```
>>  function xform, input
>>  temp = alog(input)  ; I realize I could combine these 2 statements
>>  return, temp        ; but it makes debugging easier
>>  end
```

>> Unfortunately, I can't guarentee that there won't be some elements that are
>> zero and thus invalid arguments to alog. what's the "most efficient" way to
>> take the alog of any element that's greater than 0 and set the value of ony
>> that are less than or equal to 0 to some small value (1e-7 for example)?

> I think what you want is this:

```
>  function xform, input
>  temp = alog(input > 1e-7)
>  return, temp
>  end
```

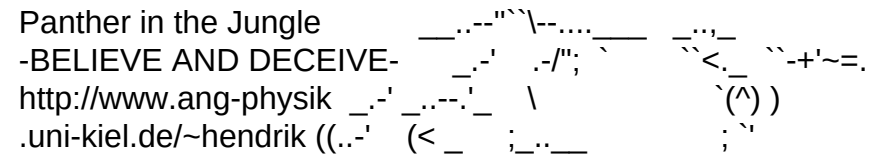
> The ">" operator in this case sets a floor on the values of "input"...any
> value less than 1e-7 is replaced with 1e-7.

RSI told us, that the alo10 error was corrected in release 5.1!

so wait

Panther

--
Panther in the Jungle
-BELIEVE AND DECEIVE-
http://www.ang-physik
.uni-kiel.de/~hendrik



Subject: Re: transforming an array where some values can't
Posted by Evilio del Rio on Mon, 09 Feb 1998 08:00:00 GMT
View Forum Message <> Reply to Message

On 6 Feb 1998, Joel D. Offenberg wrote:

> bowler@eisner.decus.org writes:
>
>> I'm trying to write a function that will transform an arbitrary array by taking
>> hte natural log of every element in the array.
>
>> What I have is
>
>> function xform, input
>> temp = alog(input) ; I realize I could combine these 2 statements
>> return, temp ; but it makes debugging easier
>> end
>
>> Unfortunately, I can't guarentee that there won't be some elements that are
>> zero and thus invalid arguments to alog. what's the "most efficient" way to
>> take the alog of any element that's greater than 0 and set the value of ony
>> that are less than or equal to 0 to some small value (1e-7 for example)?
>
IDL softly handles any math exception (IEEE standard) so you don't need to
worry about the small/negative values for alog() :

```
IDL> print,alog(0.0)
 -inf
IDL> print,alog(-1.0)
 nan0x2000000
```

You just need to handle special case with the function FINITE(). However,
many of the standard plot/calculation routines can treat naN values as
missing data, try for example:

IDL> plot,alog(randomn(seed,100))

Cheers,

Evilio Jose del Rio Silvan Institut de Ciencies del Mar
E-mail: edelrio@icm.csic.es URL: http://www.ieec.fcr.es/~evilio/
"Anywhere you choose,/ Anyway, you're gonna lose"- Mike Oldfield