Subject: Re: Problem with logarithmic axes using AXIS IDL command
Posted by Russell Ryan on Wed, 03 Apr 2013 15:25:15 GMT
View Forum Message <> Reply to Message

On Wednesday, April 3, 2013 12:34:43 AM UTC-4, Jeremy Bailin wrote:
> On 4/2/13 9:03 PM, hibou21@gmail.com wrote:
>
>> I am having a strange IDL problem involving logarithmic axes. I am trying to add a logarithmic axis on the right hand side of a plot using the AXIS command. The function that relates the values on the left-hand-side to those on the right-hand-side is:
>
>>
>
>> (90.0/3.85)*(10.0d^(!Y.CRANGE/2.50d) - (90.0/3.85)
>
>> (The value 90.0/3.85 ~= 23.4)
>
>>
>
>> When I try to plot (90.0/3.85)*(10.0d^(!Y.CRANGE/2.50d), without subtracting the final number, IDL does what it's supposed to do.
>
>>
>
>> However, when I try to plot my full expression:
>
>> (90.0/3.85)*(10.0d^(!Y.CRANGE/2.50d) - (90.0/3.85)
>
>> IDL gives the error:
>
>> % AXIS: Warning: Infinite plot range.
>
>>
>
>> I played around with it a bit and it turns out you can subtract up to about 14.8 alright, but once you try subtracting more than that, it stops working and gives the error.
>
>>
>
>> Here is a sample plot that you can use to reproduce the error:
>
>> IDL> plot, [1,2,3], [1,2,3], yr=[-0.5,4.5],ysty=9
>
>> IDL> AXIS, YAXIS = 1, /ylog, YRANGE = (90.0/3.85)*(10.0d^(!Y.CRANGE/2.50d))-15, /ysty, /save
>
>>
>

>> I've tried resetting !Y.Type to both 0 and 1, as suggested online, but to no effect. I've also tried inputting the YRANGE values manually [-8.6269713,1451.5886], but it doesn't make a difference. I'm really stumped by this...
>
>>
>
>> My IDL information: IDL Version 7.0, Mac OS X (darwin i386 m32). (c) 2007, ITT Visual Information Solutions
>
>>
>
>
>
> There are a few problems here, and they mainly boil down to your use of
>
> /YLOG.
>
>
>
> /YLOG can be used to have a uniform logarithmic axis. But that's clearly
>
> not what you have, because of that offset. And because /YLOG expects the
>
> YRANGE to be in terms of the actual number (not the logarithm), when it
>
> sees a negative value, it dies since you're effectively telling it that
>
> you want the y axis to extend down to -infinity.
>
>
>
> What I generally do when I have an arbitrary transformation between two
>
> different axis ranges is to stick the tick marks on manually. I can't
>
> tell based on what you've said what the actual minimum value that you
>
> want plotted is, but say it is 0.1 and you want tick marks at [1, 10,
>
> 100, 1000, 10000]:
>
>
>
> ytick_values = [1, 10, 100, 1000, 10000]
>
> c_offset = 90.0/3.85
>
> ; inverse transformation of what you gave, so this will

```
>
>  ; take RH axis values and turn them into LH axis values:
>
>  ytick_transformed = 2.5*alog10((ytick_values + c_offset)/c_offset)
>
>  ; make an axis that matches the original LH y axis by using the same
>
>  ; y range and forcing the y style to be correct, but puts RH axis
>
>  ; labels at the appropriate LH locations
>
>  axis, yaxis=1, yrange=!y.crange, ystyle=1, $
>
>    yticks=n_elements(ytick_values)+1, ytickv=ytick_transformed, $
>
>    ytickname=string(ytick_values,format='(F0.0)')
>
>
>
>  -Jeremy.
```

Yeah, I agree with Jeremy. But for good measure, read David Fanning's webpage on irregular tick marks. Of course, all this info can just as easily be used for regular tick marks, or ones which are some odd mapping from one to the other.

http://www.idlcoyote.com/tips/irregular_tick_spacing.html

Good Luck,
Russell
