Subject: Log axes using object graphics
Posted by Brad Gom on Tue, 20 Jun 2000 07:00:00 GMT
View Forum Message <> Reply to Message

I've been using object graphics to make a plot object that uses the mouse for zooming, data picking, etc. (in the style of David Fanning's XPlot routine). It works great for linear axes, but now I want to add the option of having logarithmic axes. It sort of works, but tends to hang the session more often than do something predictable.

The problem I'm having is that when the axes are set to log mode, they return the log of the axis ranges in the CRange parameters. I sort of understand how to properly scale 'normal' graphics objects, but now I'm lost.
For example, if I set the range of an IDLgrAxis object to the following:
axis=obj_new('IdlgrAxis', range=[5d,50000d], /log, /exact)
and then retrieved the 'actual' range later for coordinate conversion :
axis->getproperty,crange=crange
crange would contain [0.698970, 4.69897] instead of [5,50000].

My question is how to I set the coordinate convertion factors in all my other graphic objects? Do I use the real range, or the log of the range? Do I do the following?
xs = Norm_range(crange)
Plot->SetProperty, XCoord_Conv=xs, xrange=crange

or the following?
realrange=10d ^ crange
xs = Norm_range(realrange)
Plot->SetProperty, XCoord_Conv=xs, xrange=realrange

get my drift?

What about when data-picking and other times when you have to convert the values- how do you know whether to use the real range or the log of the range?

Thanks,

Brad

Subject: Re: Log axes using object graphics
Posted by Brad Gom on Wed, 21 Jun 2000 07:00:00 GMT
View Forum Message <> Reply to Message

I've blundered around a bit, and I think I've sorted out one problem and

found a more devious one.

In general, I think the method to plot logarithmic axis is as follows:

Set the log parameter of the IDLgrAxis object to 1
Set the range parameters of the IDLgrAxis objects to the actual range you want (eg [10,10000])
Retrieve the crange values from the axis (which will be the log of the range you passed in)
Use these crange values to calculate the coordinate conversion factors.
Set the xcoord_conv parameters of the axis and plot objects to the calculated factors.
Set the range parameters of the plot to the crange values retrieved from the axes.

Fine, this is more or less like the linear plot case, with the exception that you have to keep in mind that you pass the 'actual' range into the axis objects, but use the CRANGE values for the plot objects and coordinate convertion factors.

Now, the problem is that the machine hangs when I try to plot a logarithmic range that is approximately one order of magnitude and spans only one power of 10 -ie [500,5000]

Try the following procedure called 'Objtest.pro'. First, execute it as is. Then try un-commenting the 2nd line. On my machine it hangs at "thisWindow->Draw, thisView"

Any ideas?

Thanks,
Brad

```
FUNCTION BGNorm_range, range, Position=position
 IF (N_Elements(position) EQ 0) THEN position = [0d, 1d] ELSE $
  position=double(position)
 range = double(range)
 scale = [((position[0]*range[1])-(position[1]*range[0])) / $
  (range[1]-range[0]), (position[1]-position[0])/(range[1]-range[0])]
RETURN, scale
END

pro objtest
 xrange=[10,1000]
; xrange=[500,5000]    ;this is the range that crashes the Draw method.
```

```
yrange=[-2,2]
xlog=1
ylog=0
x=findgen(10000)+2
y=sin(x)

xAxis1 = Obj_New("IDLgrAxis", 0, Ticklen=0.025, Minor=4, $
  color=[255,255,255], Range=xrange, Location=[1000, 0 ,0],$
  /Exact, log=xlog, name='xaxis1')
xAxis2 = Obj_New("IDLgrAxis", 0, Ticklen=0.025, Minor=4, $
  /NoText, color=[255,255,255], Range=xrange, TickDir=1, $
  Location=[1000, 1, 0], /Exact, log=xlog, name='xaxis2')
yAxis1 = Obj_New("IDLgrAxis", 1, Ticklen=0.025, Minor=4,$
  color=[255,255,255], Range=yrange, Location=[0, 1000, 0], $
  /Exact, log=ylog, name='yaxis1')
yAxis2 = Obj_New("IDLgrAxis", 1, Ticklen=0.025, Minor=4, $
  /NoText, color=[255,255,255], Range=yrange, TickDir=1,$
  Location=[1, 1000, 0], /Exact, log=ylog, name='yaxis2')

xAxis1->GetProperty, CRange=xrange
yAxis1->GetProperty, CRange=yrange

xs = BGNorm_range(xrange)
ys = BGNorm_range(yrange)
xAxis1->SetProperty, XCoord_Conv=xs
xAxis2->SetProperty, XCoord_Conv=xs
yAxis1->SetProperty, YCoord_Conv=ys
yAxis2->SetProperty, YCoord_Conv=ys

;use log values for the plot
if xlog ne 0 then x=alog10(x)
if ylog ne 0 then y=alog10(y)
p1 = Obj_New("IDLgrPLOT", x, y,name='p1',XCoord_Conv=xs,$
  YCoord_Conv=ys,color=[255,255,255], xrange=xrange,$
  yrange=yrange)

thisModel = Obj_New('IDLgrModel')
thisModel->add,xAxis1
thisModel->add,xAxis2
thisModel->add,yAxis1
thisModel->add,yAxis2
thisModel->add,p1

Container1 = Obj_New('IDL_Container')

thisWindow=Obj_new('IDLgrWindow')
thisView = Obj_New('IDLgrView', Viewplane_Rect=[-.2, -.2, 1.3, 1.3], $
  Location=[0,0], Color=[0,0,0])
```

```
 thisView->Add, thisModel
 Container1->add,thisView
 Container1->add,thisWindow
 Container1->add,thisModel

print,'drawing'
 thisWindow->Draw, thisView
print,'done'

 result=dialog_message('Hit OK to continue',/info)
 obj_destroy,Container1
end
```

Subject: Re: Log axes using object graphics
Posted by Brad Gom on Fri, 23 Jun 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Hi Ben,

thanks for the test. It turns out that the problem is a divide by zero error in the object graphics code, caused when there is only one major tick mark in the range. "Floating illegal operand" doesn't show up on my PC, but it hints at the problem.

Here is the reply I got from Karl at RSI:

> It is a bug in IDL 5.3.
>
> When you change the range so that only one tick is needed on the axis,
> the bug shows up. There is a div by 0 problem in there and the code
> goes into an infinite loop.
>
> I'll open a bug report for you.
>
> As a work-around, you might try specifying MAJOR=2 to force two ticks
> on the axis. I tried this in your program and got really nice
> results.
>
> Happily, the problem does not occur in IDL 5.4. I've changed much of
> the axis support for another reason and apparently exorcised this
> gremlin in the process.
>
Thanks

Brad

Subject: Re: Log axes using object graphics
Posted by Ben Tupper on Fri, 23 Jun 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Brad Gom wrote:

> Now, the problem is that the machine hangs when I try to plot a logarithmic
> range that is approximately one order of magnitude and spans only one power
> of 10 -ie [500,5000]
>
> Try the following procedure called 'Objtest.pro'. First, execute it as is.
> Then try un-commenting the 2nd line. On my machine it hangs at
> "thisWindow->Draw, thisView"
>
>

Hello,

I tried the code on:

```
IDL> help,!version,/str
** Structure !VERSION, 5 tags, length=40:
   ARCH            STRING    'sparc'
   OS              STRING    'sunos'
   OS_FAMILY       STRING    'unix'
   RELEASE         STRING    '5.2'
   BUILD_DATE      STRING    'Oct 30 1998'
```

First, I tried it as written ... no problem.

```
IDL> .COMPILE '/disk1/tupper/pemaquid/objtest.pro'
% Compiled module: BGNORM_RANGE.
% Compiled module: OBJTEST.
IDL> objtest
drawing
done
```

Then I uncommented the specified line. It works without hanging, but the non-fatal error message that follows
appeared. By stepping through I found that the error message occurs at the thisWindow->Draw, thisView statement.

```
IDL> .COMPILE '/disk1/tupper/pemaquid/objtest.pro'
% Compiled module: BGNORM_RANGE.
% Compiled module: OBJTEST.
IDL> objtest
drawing
done
% Program caused arithmetic error: Floating illegal operand
```

I'm not sure if this helps, but perhaps it narrows down the problem.

Ben

--
Ben Tupper

Bigelow Laboratory for Ocean Science
tupper@seadas.bigelow.org

pemaquidriver@tidewater.net