Subject: Re: 8.* graphics
Posted by Russell[1] on Tue, 17 Jan 2012 17:47:38 GMT
View Forum Message <> Reply to Message

UPDATE: I tried a variant of a trick by David F., but for the y-axis http://www.idlcoyote.com/tips/another_yaxis.html . For some kooky reason, as soon as I use the axis.pro (function not procedure), the colors and transparency immediately go away. GRRR!!!

R

On Jan 17, 12:22 pm, Russell <rryan....@gmail.com> wrote:
> Okay, so I'm trying my hand at the new graphics features in IDL 8,
> mostly because doing this in the classic direct graphics way is
> incredible painful and (according to the help pages) the new stuff
> should be able to knock this one out of the park. Famous last words,
> I know. So here's the problem:
>
> I'm trying to make a figure for an upcoming proposal where I want to
> show a series of transmission curves (as a function of wavelength)
> with the area under each curve shaded a different color. Many of
> these curves have small overlaps with adjacent bands, and I'd like to
> have the shading be the transparency (a la red+blue = purple). It
> seems that plot.pro (the function not the procedure) is ready and
> willing to do this, but I desperately need the x-axis to be displayed
> as a log (so xlog=1b). However! the shading and transparency is
> completely gone when I set xlog=1b! AAGGHH! Am I crazy, does anyone
> know anything about this?
>
> -Russell
>
> PS, Yes, I'm aware that I can simply take the logarithm of the axis
> and plot log(wavelength), but (1) I prefer the log-spaced tick marks
> and (2) it *SHOULD* work! switching this bit shouldn't affect the
> colors!

Subject: Re: 8.* graphics
Posted by David Fanning on Tue, 17 Jan 2012 17:53:15 GMT
View Forum Message <> Reply to Message

Russell writes:

> Okay, so I'm trying my hand at the new graphics features in IDL 8,
> mostly because doing this in the classic direct graphics way is
> incredible painful and (according to the help pages) the new stuff
> should be able to knock this one out of the park. Famous last words,

> I know. So here's the problem:
>
> I'm trying to make a figure for an upcoming proposal where I want to
> show a series of transmission curves (as a function of wavelength)
> with the area under each curve shaded a different color. Many of
> these curves have small overlaps with adjacent bands, and I'd like to
> have the shading be the transparency (a la red+blue = purple). It
> seems that plot.pro (the function not the procedure) is ready and
> willing to do this, but I desperately need the x-axis to be displayed
> as a log (so xlog=1b). However! the shading and transparency is
> completely gone when I set xlog=1b! AAGGHH! Am I crazy, does anyone
> know anything about this?

Well, one of, uh, several problems that hopefully will
be corrected in the next version. :-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: 8.* graphics
Posted by David Fanning on Tue, 17 Jan 2012 18:22:02 GMT
View Forum Message <> Reply to Message

Russell writes:

> UPDATE: I tried a variant of a trick by David F., but for the y-axis
> http://www.idlcoyote.com/tips/another_yaxis.html . For some kooky
> reason, as soon as I use the axis.pro (function not procedure), the
> colors and transparency immediately go away. GRRR!!!

Here is some code that worked surprisingly well!

```
pos = [0.125, 0.15, 0.9, 0.875]
x = findgen(101)
y1 = cgDemoData(17)
y2 = cgDemoData(17)
cgPlot, x, y1, /NoData, Position=pos, xtickformat='(A1)', $
   ytickformat='(A1)'
```

```
cgColorfill, [x[0],x,x[-1]], [!Y.Crange[0], y1,!Y.Crange[0]], $
   /data, color='yellow'
p1 = cgsnapshot()
cgerase
cgPlot, x, y2, /NoData, Position=pos, xtickformat='(A1)', $
   ytickformat='(A1)'
cgColorfill, [x[0],x,x[-1]], [!Y.Crange[0], y2,!Y.Crange[0]], $
   /data, color='olive'
p2 = cgsnapshot()
cgblendimage, p1, p2
cgPlot, x, y1, thick=2, /Noerase, Position=pos
cgplot, x, y2, /Overplot, thick=2
END
```

Your mileage may vary. For for 5 minutes work, it's not too bad. :-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: 8.* graphics
Posted by David Fanning on Tue, 17 Jan 2012 18:29:21 GMT
View Forum Message <> Reply to Message

David Fanning writes:

>
> Russell writes:
>
>> UPDATE: I tried a variant of a trick by David F., but for the y-axis
>> http://www.idlcoyote.com/tips/another_yaxis.html . For some kooky
>> reason, as soon as I use the axis.pro (function not procedure), the
>> colors and transparency immediately go away. GRRR!!!
>
> Here is some code that worked surprisingly well!
>
> pos = [0.125, 0.15, 0.9, 0.875]
> x = findgen(101)
> y1 = cgDemoData(17)
> y2 = cgDemoData(17)

```
>  cgPlot, x, y1, /NoData, Position=pos, xtickformat='(A1)', $
>     ytickformat='(A1)'
>  cgColorfill, [x[0],x,x[-1]], [!Y.Crange[0], y1,!Y.Crange[0]], $
>     /data, color='yellow'
>  p1 = cgsnapshot()
>  cgerase
>  cgPlot, x, y2, /NoData, Position=pos, xtickformat='(A1)', $
>     ytickformat='(A1)'
>  cgColorfill, [x[0],x,x[-1]], [!Y.Crange[0], y2,!Y.Crange[0]], $
>     /data, color='olive'
>  p2 = cgsnapshot()
>  cgblendimage, p1, p2
>  cgPlot, x, y1, thick=2, /Noerase, Position=pos
>  cgplot, x, y2, /Overplot, thick=2
>  END
```

Sorry, I meant to do that with log plots. :-)

```
pos = [0.125, 0.15, 0.9, 0.875]
x = findgen(101)+1
y1 = cgDemoData(17)
y2 = cgDemoData(17)
cgPlot, x, y1, /NoData, Position=pos, xtickformat='(A1)', $
   ytickformat='(A1)', /Xlog, xrange=[1,100]
cgColorfill, [x[0],x,x[-1]], [!Y.Crange[0], y1,!Y.Crange[0]], $
   /data, color='yellow'
p1 = cgsnapshot()
cgerase
cgPlot, x, y2, /NoData, Position=pos, xtickformat='(A1)', $
   ytickformat='(A1)', /XLog, xrange=[1,100]
cgColorfill, [x[0],x,x[-1]], [!Y.Crange[0], y2,!Y.Crange[0]], $
   /data, color='olive'
p2 = cgsnapshot()
cgblendimage, p1, p2
cgPlot, x, y1, thick=2, /Noerase, Position=pos, /XLog, xrange=[1,100]
cgplot, x, y2, /Overplot, thick=2, /XLog
END
```

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/

Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: 8.* graphics
Posted by David Fanning on Tue, 17 Jan 2012 18:33:27 GMT
View Forum Message <> Reply to Message

David Fanning writes:

> Sorry, I meant to do that with log plots. :-)

In case there is someone who doesn't have the Coyote Library
installed, here is what the plot looks like. :-)

http://www.idlcoyote.com/misc/log_blended_plot.png

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: 8.* graphics
Posted by lecacheux.alain on Tue, 17 Jan 2012 18:54:35 GMT
View Forum Message <> Reply to Message

On 17 jan, 18:22, Russell <rryan....@gmail.com> wrote:
> Okay, so I'm trying my hand at the new graphics features in IDL 8,
> mostly because doing this in the classic direct graphics way is
> incredible painful and (according to the help pages) the new stuff
> should be able to knock this one out of the park. Famous last words,
> I know. So here's the problem:
>
> I'm trying to make a figure for an upcoming proposal where I want to
> show a series of transmission curves (as a function of wavelength)
> with the area under each curve shaded a different color. Many of
> these curves have small overlaps with adjacent bands, and I'd like to
> have the shading be the transparency (a la red+blue = purple). It
> seems that plot.pro (the function not the procedure) is ready and
> willing to do this, but I desperately need the x-axis to be displayed

> as a log (so xlog=1b). However! the shading and transparency is
> completely gone when I set xlog=1b! AAGGHH! Am I crazy, does anyone
> know anything about this?
>
> -Russell
>
> PS, Yes, I'm aware that I can simply take the logarithm of the axis
> and plot log(wavelength), but (1) I prefer the log-spaced tick marks
> and (2) it *SHOULD* work! switching this bit shouldn't affect the
> colors!

What about this ?

```
x = 10^(findgen(100)*0.03)
b1 = exp(-(alog10(x)-1)^2/0.3)
b2 = 0.5*exp(-(alog10(x)-2)^2/0.2)

p1 = plot(x, b1, COLOR='red', /XTICKDIR, /YTICKDIR, /XLOG, THICK=3)
p2 = plot(x, /OVERPLOT, b2, COLOR='blue', THICK=3)
poly1 = polygon([x,x[0]], [b1,0], -0.01+fltarr(101), /DATA,
LINESTYLE=6, $
  /FILL_BACKGROUND, FILL_COLOR=!COLOR.DEEP_PINK, FILL_TRANSPARENCY=50)
poly2 = polygon([x,x[0]], [b2,0], -0.01+fltarr(101), /DATA,
LINESTYLE=6, $
  /FILL_BACKGROUND, FILL_COLOR=!COLOR.DEEP_SKY_BLUE,
FILL_TRANSPARENCY=50)
```

Subject: Re: 8.* graphics
Posted by Russell[1] on Tue, 17 Jan 2012 18:55:51 GMT
View Forum Message <> Reply to Message

On Jan 17, 1:33 pm, David Fanning <n...@dfanning.com> wrote:
> David Fanning writes:
>> Sorry, I meant to do that with log plots. :-)
>
> In case there is someone who doesn't have the Coyote Library
> installed, here is what the plot looks like. :-)
>
> http://www.idlcoyote.com/misc/log_blended_plot.png
>
> Cheers,
>
> David
>
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.

> Coyote's Guide to IDL Programming:http://www.idlcoyote.com/
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

WOW! Thanks a million David. I'm not too familiar with the cg* routines, other than knowing they're used in the Astro Library (I see many "% Compiled module:" statements a lot nowadays). I have a few routines that do what your quick thing did, in fact I learned how to do it from you: http://www.idlcoyote.com/code_tips/transpoly.html . More than anything, I wanted to give these new function graphics a try and see what the fuss was all about. In principle, I've seen a dozen or so things in the help files that look pretty cool, but in practice they're still so new that many (I would say easy) things are still not doable. I've spent a few hours now on this, and have decided that, while many of these new features are promising, their execution is still quite lacking. I guess every step forward is two steps backward --- back to square one!

Thanks again...
r

Subject: Re: 8.* graphics
Posted by David Fanning on Tue, 17 Jan 2012 19:05:23 GMT
View Forum Message <> Reply to Message

Russell writes:

> WOW! Thanks a million David. I'm not too familiar with the cg*
> routines, other than knowing they're used in the Astro Library (I see
> many "% Compiled module:" statements a lot nowadays).

What!? Seriously?

I can't imagine working in IDL without them. I don't think it's even possible. ;-)

The best thing about them is that they are actually programmable. By scientists, even. In fact, it's possible to easily turn your own routines into Coyote Graphics routines without much effort.

If you get a new idea for a graphic like this, you don't have to wait for the folks at Excelis to fix their stuff for you. You just spend 10 minutes or so writing a program that does it!

There are plenty of resources to help you learn more about them. ;-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: 8.* graphics
Posted by Russell[1] on Tue, 17 Jan 2012 19:56:31 GMT
View Forum Message <> Reply to Message

On Jan 17, 1:54 pm, alx <lecacheux.al...@wanadoo.fr> wrote:
> On 17 jan, 18:22, Russell <rryan....@gmail.com> wrote:
>
>
>
>
>
>
>
>
>
>
>
>> Okay, so I'm trying my hand at the new graphics features in IDL 8,
>> mostly because doing this in the classic direct graphics way is
>> incredible painful and (according to the help pages) the new stuff
>> should be able to knock this one out of the park. Famous last words,
>> I know. So here's the problem:
>
>> I'm trying to make a figure for an upcoming proposal where I want to
>> show a series of transmission curves (as a function of wavelength)
>> with the area under each curve shaded a different color. Many of
>> these curves have small overlaps with adjacent bands, and I'd like to
>> have the shading be the transparency (a la red+blue = purple). It
>> seems that plot.pro (the function not the procedure) is ready and
>> willing to do this, but I desperately need the x-axis to be displayed
>> as a log (so xlog=1b). However! the shading and transparency is
>> completely gone when I set xlog=1b! AAGGHH! Am I crazy, does anyone
>> know anything about this?
>
>> -Russell
>
>> PS, Yes, I'm aware that I can simply take the logarithm of the axis
>> and plot log(wavelength), but (1) I prefer the log-spaced tick marks

>> and (2) it *SHOULD* work! switching this bit shouldn't affect the
>> colors!
>
> What about this ?
>
> x = 10^(findgen(100)*0.03)
> b1 = exp(-(alog10(x)-1)^2/0.3)
> b2 = 0.5*exp(-(alog10(x)-2)^2/0.2)
>
> p1 = plot(x, b1, COLOR='red', /XTICKDIR, /YTICKDIR, /XLOG, THICK=3)
> p2 = plot(x, /OVERPLOT, b2, COLOR='blue', THICK=3)
> poly1 = polygon([x,x[0]], [b1,0], -0.01+fltarr(101), /DATA,
> LINESTYLE=6, $
> /FILL_BACKGROUND, FILL_COLOR=!COLOR.DEEP_PINK, FILL_TRANSPARENCY=50)
> poly2 = polygon([x,x[0]], [b2,0], -0.01+fltarr(101), /DATA,
> LINESTYLE=6, $
> /FILL_BACKGROUND, FILL_COLOR=!COLOR.DEEP_SKY_BLUE,
> FILL_TRANSPARENCY=50)

I like it. That should also fix one problem I had with the legend.
This is actually very close to my direct-graphics fix, so conceptually
I like it. Though, I still wish it just worked as advertised.

Thanks alx!
R
