Subject: Re: cgBoxPlot not responding to xCharsize
Posted by David Fanning on Tue, 02 Jul 2013 13:21:43 GMT
View Forum Message <> Reply to Message

JP writes:

> It seems that cgBoxPlot is not responding to the xCharsize keyword. I tried the code below:
>
> PS_Start, 'cgHistoplot.png'
> !P.Multi=[0,3,1]
> labels=['var1','var2','var3','var4']
> cgdisplay, 1000, 300
> cgBoxPlot, randomn(seed, 4, 1000), labels=labels
> cgBoxPlot, randomn(seed, 4, 1000), labels=labels, xCharsize=0.5
> cgBoxPlot, randomn(seed, 4, 1000), labels=labels, xCharsize=0.25
> PS_End, resize=100, /png

Yes, I was using the XCharsize value incorrectly in this program. You can find an updated version here:

http://www.idlcoyote.com/programs/cgboxplot.pro

Unfortunately, this is not going to solve all your problems. :-)

The larger problem here is doing multiple plots with !P.Multi. This method of doing multiple plots is great as long as you are willing to surrender all control over plot position and character size and rely completely on the !P.Multi algorithm to set these for you. But, if you wish to control this aspect of your plots (as I do in cgBoxPlot), then chaos is certain to ensue.

In this particular case, since I am putting plot labels on with XYOutS, there will be a terrible mismatch between the size of the plot annotation, which is under the influence of !P.Multi, and the plot labels, which are not. Basically, you will never be able to get the two sizes to match! (Or, if you do, immediately call me with the algorithm you used!)

Download the updated program, then consider this code. Here is how you are doing things now.

```
!P.Multi=[0,3,1]
labels=['var1','var2','var3','var4']
cgdisplay, 1000, 300, wid=1
cgBoxPlot, randomn(seed, 4, 1000), labels=labels
cgBoxPlot, randomn(seed, 4, 1000), labels=labels, xCharsize=0.75
cgBoxPlot, randomn(seed, 4, 1000), labels=labels, xCharsize=1.25
```

```
!P.multi=0
```

You can see the plot annotations are all over the map!

In this case, it will make a LOT more sense to use cgLayout to set up your plot positions, which will allow you to control your plot annotations exactly. There will be no interference from !P.Multi.

See how much more sense this way of drawing the plots makes.

```
positions = cgLayout([3,1], xGap=5, oxMargin=[5,5], $
   oyMargin=[5,3])
labels=['var1','var2','var3','var4']
cgdisplay, 1000, 300
cgBoxPlot, randomn(seed, 4, 1000), labels=labels, $
   Position=positions[*,0]
cgBoxPlot, randomn(seed, 4, 1000), labels=labels, $
   Position=positions[*,1], /NoErase, XCharsize=0.75, $
   Charsize=0.75
cgBoxPlot, randomn(seed, 4, 1000), labels=labels, $
   Position=positions[*,2], /NoErase, XCharsize=1.25, $
   Charsize=1.25
```

Cheers,

David

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