Subject: Re: Controlling axis labels in IDL plots?
Posted by davidf on Wed, 12 Apr 2000 07:00:00 GMT
View Forum Message <> Reply to Message

Rachel Howe (rhowe@noao.edu) writes:

> I wonder if anyone can help me with a graphics problem?
>
> I'm getting very finicky about my IDL plots. (I use direct graphics
> only, no objects --
> conceptually I'm still in the v3.6 era).
> I've learned to use the !p, !x and !y system variables, their associated
> keywords,
> and the axis command, to override most of the defaults, but one thing
> eludes me.
> Does anyone know how the distance of the axis label from the axis is
> determined in a single-panel 2d plot?
> There doesn't seem to be any way to control it directly, but it must be
> some arcane function of the plot area, position/margin settings, and
> character size. If anyone knows an algorithm, I'd be delighted to hear
> it.
> Or would it be easier to give up and just use
> XYOUTS to put the annotations where I want them?

Ooohh, someone as anal as I am about the looks of things!
Thank goodness, I had just about abandoned all hope. :-)

I don't think there is any way to control this. If
there is an algorithm (there must be), it's a screwy
one. I've learned to live with it in direct graphics,
but that distance from the Y axis in object graphics
drives me crazy. :-(

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Controlling axis labels in IDL plots?
Posted by steinhh on Thu, 13 Apr 2000 07:00:00 GMT

In article <MPG.135ee234b9aa2a5989acb@news.frii.com>
davidf@dfanning.com (David Fanning) writes:
>
> Rachel Howe (rhowe@noao.edu) writes:
[..]
>> Does anyone know how the distance of the axis label from the axis is
>> determined in a single-panel 2d plot?
>> There doesn't seem to be any way to control it directly, but it must be
>> some arcane function of the plot area, position/margin settings, and
>> character size. If anyone knows an algorithm, I'd be delighted to hear
>> it.
>> Or would it be easier to give up and just use
>> XYOUTS to put the annotations where I want them?
>
> Ooohh, someone as anal as I am about the looks of things!
> Thank goodness, I had just about abandoned all hope. :-)

I guess you're entitled to be a little bit "anal" if your plots are
appearing in the journal Science... See the first Report listed on
this page: http://www.sciencemag.org/content/vol287/issue5462/

What's your excuse, David?

Anyway, seems like Liam's suggestion is the best one can do without
using XYOUTS. Note that prepending a "!C" on the Y axis title will
push the text towards the axis, and likewise prepending one for the
X axis title will push it down.

Stein Vidar

Subject: Re: Controlling axis labels in IDL plots?
Posted by Liam E. Gumley on Thu, 13 Apr 2000 07:00:00 GMT

Rachel Howe wrote:
>
> I'm getting very finicky about my IDL plots. (I use direct graphics
> only, no objects --
> conceptually I'm still in the v3.6 era).
> I've learned to use the !p, !x and !y system variables, their associated
> keywords,
> and the axis command, to override most of the defaults, but one thing
> eludes me.
> Does anyone know how the distance of the axis label from the axis is

> determined in a single-panel 2d plot?
> There doesn't seem to be any way to control it directly, but it must be
> some arcane function of the plot area, position/margin settings, and
> character size. If anyone knows an algorithm, I'd be delighted to hear
> it.
> Or would it be easier to give up and just use
> XYOUTS to put the annotations where I want them?

Here's a quick and dirty method which can be used to push the X and Y axis labels further away from the axes using the !C (carriage return) font positioning code:

```
;- Set a reasonable character size and create data
device, set_character_size=[10, 12]
x = findgen(200) * 0.1
y = sin(x)

;- Default axis label positions
plot, x, y, xtitle='X AXIS', ytitle='Y AXIS'

;- Move X axis label down one character
IDL> plot, x, y, xtitle='!CX AXIS', ytitle='Y AXIS'

;- Move Y axis lable left one character
IDL> plot, x, y, xtitle='X AXIS', ytitle='Y AXIS!C'
```

I find this particularly useful when adding titles to maps which have a box grid:

```
window, /free
map_set, -30, 130, scale=40e6, $
  xmargin=[2, 2], ymargin=[2, 4], title='MAP TITLE!C'
map_continents
map_grid, /box, charsize=0.75
```

Unless you add the trailing !C to the title, it is displayed at the same height as the top row of longitude labels. If this method doesn't suit your needs, I think you will have to resort to manually positioned labels.

Cheers,
Liam.
http://cimss.ssec.wisc.edu/~gumley

Subject: Re: Controlling axis labels in IDL plots?
Posted by Martin Schultz on Thu, 13 Apr 2000 07:00:00 GMT

Rachel Howe wrote:
>
> I wonder if anyone can help me with a graphics problem?
>
> I'm getting very finicky about my IDL plots. (I use direct graphics
> only, no objects --
> conceptually I'm still in the v3.6 era).
> I've learned to use the !p, !x and !y system variables, their associated
> keywords,
> and the axis command, to override most of the defaults, but one thing
> eludes me.
> Does anyone know how the distance of the axis label from the axis is
> determined in a single-panel 2d plot?
> There doesn't seem to be any way to control it directly, but it must be
> some arcane function of the plot area, position/margin settings, and
> character size. If anyone knows an algorithm, I'd be delighted to hear
> it.
> Or would it be easier to give up and just use
> XYOUTS to put the annotations where I want them?
>
> Thanks in advance for any suggestions!
>
>
> --
> Rachel Howe
> National Solar Observatory, Tucson AZ
> rhowe@noao.edu

Rachel,

perhaps you can make use of my axlabel routine which I attach. I haven't used it that extensively and cannot guarantee for bug freeness, but it should at least provide you with some algorithm for finding label positions. I have only used it with Hershey fonts, and I have no idea how it behaves with Truetype or Hardware fonts. Any suggestions for improvements are welcome.

Regards,
Martin

--
[[[[[[[[[
[[Dr. Martin Schultz   Max-Planck-Institut fuer Meteorologie   [[
[[Bundesstr. 55, 20146 Hamburg   [[
[[phone: +49 40 41173-308   [[
[[fax: +49 40 41173-298   [[
[[martin.schultz@dkrz.de   [[

```
[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[[ [[[[[[[
;------------------------------------------------------------ --
; $Id$
;+
; NAME:
;       AXLABEL
;
; PURPOSE:
;       Put previously calculated axis labels onto the screen
;       at proper position. This routine was designed to work
;       together with LOGLEVELS to produce fancy log plots.
;       It involves several coordinate transformations in order
;       to be device independent and take into account the
;       character size. The user can specify a label format
;       and use 'external' formatting functions similar to
;       the [XYZ]TICKFORMAT keyword of PLOT.
;
; CATEGORY:
;       Plotting
;
; CALLING SEQUENCE:
;       AXLABEL,Value [,/XAxis] [,keywords]
;
; INPUTS:
;       VALUE -> A vector with the values to be labelled on the
;           axis.
;
; KEYWORD PARAMETERS:
;       /XAxis -> If set, the labels are placed on the X achis
;           rather than on the Y axis
;
;       /YAxis -> Place the labels on the Y axis (this is the default,
;           and this keyword is there for purely aesthetic reasons)
;
;       CHARSIZE -> The character size of the label
;
;       FORMAT -> An IDL format string (used as argument to the
;            STRING function) or the name of a function that returns
;            formatted labels. This function must accept three
;            arguments, the third of which is the current value
;            (see the online help to [XYZ]TICKFORMAT for more details).
;            AXLABEL always passes 0 to the first two arguments.
;
;       _EXTRA  keywords are passed on to XYOUTS (e.g. COLOR or
;            ORIENTATION). Note that the ALIGN keyword value is
;            determined automatically.
;
; OUTPUTS:
```

```
;       Axis labels without fuss.
;
; SUBROUTINES:
;       None.
;
; REQUIREMENTS:
;       A DATA coordinate system must be established by a previous
;       PLOT command.
;
; NOTES:
;       AXLABEL currently operates only on the left and bottom axes.
;
; EXAMPLE:
;         xrange = [0.3,3.0]   ; define axis range
;         yrange = [0.3,3.0]
;         plot,[1],xr=xrange,yr=yrange, $   ; do the plot
;          title='Logarithmic X axis, Logarithmic Y axis',$
;          xtickf='(a1)',ytickf='(a1)', /ylog,/xlog
;         ; important: turn the tick labeling off with ?tickformat='(A1)'
;         xlblv = loglevels(xrange)   ; get nice label values (0.5, 1., 2.)
;         ylblv = loglevels(yrange)
;         axlabel,xlblv, /xaxis       ; plot the labels
;         axlabel,ylblv, /yaxis
;
; MODIFICATION HISTORY:
;       mgs, 10 Sep 1999: VERSION 1.00
;       mgs, 23 Sep 1999: - bug fix for log-log plots
;
;-
; Copyright (C) 1999, Martin Schultz, Max-Planck-Institut f. Meteorologie
; This software is provided as is without any warranty
; whatsoever. It may be freely used, copied or distributed
; for non-commercial purposes. This copyright notice must be
; kept with any copy of this software. If this software shall
; be used commercially or sold as part of a larger package,
; please contact the author.
; Bugs and comments should be directed to martin.schultz@dkrz.de
; with subject "IDL routine axlabel"
 ;-------------------------------------------------------------- --


pro axlabel,value,Charsize=Charsize,XAxis=XAxis,YAxis=YAxis,   $
      Format=Format,_EXTRA=e


   ; Error catching
   if (N_Elements(VALUE) eq 0) then begin
      message,'Must supply at least one label value to AXLABEL!'
```

```
  endif

  ; Set default for CHARSIZE and FORMAT
  if (n_elements(CHARSIZE) EQ 0) then $
    CHARSIZE = 1.
  if (n_elements(FORMAT) EQ 0) then $
    FORMAT = '(f12.1)'

  if (keyword_set(XAxis)) then begin

    ; Get y position for label
    ; Subtract one character size
    PY = !Y.Window[0]
    PYOFF = CONVERT_COORD(1,!D.Y_CH_SIZE*CHARSIZE,/DEVICE,/TO_NORMAL)
    PY = PY - 1.05*PYOFF[1]
print,'X:PY:',py
    PY = REPLICATE(PY,N_Elements(VALUE))

    ; Convert data values to normalized x coordinates
    Y0 = !Y.CRANGE[0]
    if (!Y.TYPE eq 1) then $
      Y0 = 10.^Y0
    PX = CONVERT_COORD(VALUE,REPLICATE(Y0,N_Elements(VALUE)), $
                  /DATA,/TO_NORMAL)
    PX = PX[0,*]
print,'X:PX=',px

  endif else begin   ; Y axis label (default)

    ; Get x position for label
    PX = !X.Window[0] - 0.010
    PX = REPLICATE(PX,N_Elements(VALUE))

    ; Convert data values to normalized coordinates and
    ; subtract half the character size
    PYOFF = CONVERT_COORD(0,!D.Y_CH_SIZE*CHARSIZE,/DEVICE,/TO_NORMAL)
    X0 = !X.CRANGE[0]
    if (!X.TYPE eq 1) then $
      X0 = 10.^X0
    PY = CONVERT_COORD(REPLICATE(X0,N_Elements(VALUE)),VALUE,  $
                  /DATA,/TO_NORMAL)
    PY = PY[1,*]-0.5*PYOFF[1]
  endelse

  ; Format VALUE according to format string. If this string
  ; does not begin with '(', it is assumed that the user has passed
  ; a formatting function as for [XYZ]TICKFORMAT
  ; However, only the third (NUMBER) argument of this function is used
```

```
  if (STRPOS(FORMAT,'(') ne 0) then begin
    ValS = STRARR(N_Elements(VALUE))
    for j=0,N_Elements(VALUE)-1 do $
      ValS[j] = CALL_FUNCTION(FORMAT,0,0,VALUE[j])
  endif else $      ; apply format string directly
    ValS = STRING(VALUE,format=FORMAT)

  ValS = STRTRIM(ValS,2)

  XYOUTS,PX,PY,ValS,/NORMAL,align=1.-0.5*keyword_set(XAxis),  $
    charsize=CHARSIZE,_EXTRA=e

  return
end
```

File Attachments

1) axlabel.pro, downloaded 94 times