Subject: Re: simple questions: position and strings
Posted by sterne on Mon, 17 May 1993 23:33:58 GMT
View Forum Message <> Reply to Message

In article <May15.223357.92566@yuma.ACNS.ColoState.EDU> flatau@lamar.ColoState.EDU (Piotr Flatau) writes:

> 3. How do I get 3 plots on one page but such that the are together
> (common x-axis at the top of the first, and bottom of the next plot).
> I don't think it can be done with !p.multi and margin because of
> overlapping labels.

The following procedure will deal with an arbitrary number of plots with a common x-axis. They are plotted one below the other, to form a "ladder" of graphs. It will automatically assign an amount of "space" between "rungs" according to the range of each dataset, or you can do the calculation yourself and pass it in with the VINPOS keyword. It will optionally round the range of each dataset to some reasonable number. For this it uses a function ROUNDUP which is also included.

This ladder plot format is particularly convenient for plotting things which are sums of functions, i.e. of the form

T(x) = Sum_i(f_i(x))

You can plot the total T at the top or the bottom of the ladder and then display the constituent f_i components on the same x-axis, but displaced in y. Looks much less cluttered than overlaying dotted and dashed curves etc.

Hope this helps,
Phil

```
;+
; NAME:     LADDER
; PURPOSE: Graph one or more X-Y plots in separate vertical graphs with a
;         common X-axis to produce a "ladder" of X-Y graphs.
; CATEGORY: Display
; CALLING SEQUENCE: LADDER, A [, B, round=round, vinpos=vinpos,
;                          vpos=vpos,htotal=htotal, round = round]
;
; INPUTS:
;         A = an array or a vector.  If a vector, it specifies the values
;         of the X-axis for all the plots, and an input array B should
;         also be provided.  If A is an array, then A(*,0) is taken as
```

```
;       the vector specifying the X-axis and A(*,i) contains the
;       y-values for the i-th X-Y plot.
;
; OPTIONAL INPUT PARAMETERS:
;       B = an array which must be present if A is a vector.
;       B(*, i) contains the y-values FOR plot i-1.
;
; KEYWORD PARAMETERS:
;       ROUND: By default (round = 0), do not round up y-axis data
;       ranges to convenient values for each graph in the ladder.
;       Calling LADDER with the /ROUND keyword will round up the
;       data range for each graph in the ladder using ROUNDUP.
;
;       VINPOS:  Vector of dimension nplot+1 (nplot=number of plots)
;       specifying the vertical position of each plot in normalized
;       coordinates.  For plot i, vinpos(i-1) specifies the top and
;       vinpos(i) the bottom of the graph in normalized coordinates.
;       If VINPOS is not supplied, or is zero, appropriate values
;       are determined and returned through the VPOS keyword to
;       facilitate subsequent annotation.
;
;       VPOS:  On output, a vector specifying the vertical position of
;       each plot in normalized coordinates.  See VINPOS for more info.
;
;       HTOTAL: On output, contains the total height in DATA coords
;       of the entire ladder of plots.  Provided for use in subsequent
;       annotation.
;
; COMMON BLOCKS: None
; SIDE EFFECTS: Changes display.
; RESTRICTIONS: Needs function ROUNDUP to make convenient y-axis ranges.
; PROCEDURE: Check for consistency in input.  Determine height segment
;          for each plot, or use VPOS, if supplied.  Plot graphs.
; MODIFICATION HISTORY: Written by P.A. Sterne 6/1/1991 (sterne1@llnl.gov)
;                   PAS: Added optional vinpos. 5/17/93
;-

PRO ladder, A, B, vinpos = vinpos, vpos = vpos, htotal = htotal, $
          round = round
  CASE n_params() OF
    1: BEGIN
      index = size(A)
      m = index(1)
      n = index(2)
      nplot = n-1
      B = A(*, 1:n-1)
      A1 = A(*, 0)
    END
```

```
    2: BEGIN
      x = size(a)
      IF x(0) NE 1 THEN BEGIN   ; check: 1st argument array or vector?
        print, $
          'ladder: first parameter must be a vector of X-coordinates'
        RETURN
      ENDIF

      index = size(B)
      nplot = index(2)
      IF index(1) NE x(1) THEN BEGIN ; check: dimensions A and B match?
        print, $
          'ladder: mismatch in dimensions input parameters:'
        print, 'ladder: A(nx),B(nx,nplot)'
        RETURN
      ENDIF
      A1 = A
    END
  ENDCASE

  xrange = [min (A1), max (A1)]
  yst = 1
  xtickn = replicate('', 30)              ; If there is one plot, the
;                                  ; x tickmarks will appear.
  IF nplot GT 1 THEN xtickn = replicate(' ', 30)
;                                  ; Suppress x-tickmarks
  yminor = -1                          ; Suppress y minor tickmarks

  p = fltarr(nplot)
  h = fltarr(nplot)
  FOR i = 0, nplot-1 DO BEGIN
    h(i) = max (b(*, i))
  ENDFOR
  hnorm = h
  IF KEYWORD_SET(round) THEN BEGIN
    hn = sqrt(h/max(h))             ; magnify smaller sections
    hnorm = roundup(hn*max(h))      ; make nice upper limits
  ENDIF
  htotal = total(hnorm)
  FOR i = 0, nplot-1 DO BEGIN      ; set the fraction of the total
    p(i) = [hnorm(i)/htotal*0.8]  ; graph area that each graph will
  ENDFOR                           ; get, normalized units.
  hmax = max(hnorm)
  pmax = max(p)
  psc = hmax/pmax

  IF KEYWORD_SET(vinpos) THEN BEGIN    ; using vinpos
    vpos = vinpos
```

```
  ENDIF ELSE BEGIN          ; set min,max for each graph (normal coords.)
    vpos = fltarr(nplot+1)
    vpos(0) = .9
    FOR i = 1, nplot DO BEGIN
      vpos(i) = [vpos(i-1)-p(i-1)]
    ENDFOR
  ENDELSE

  IF p(0) GE .25 THEN yticks = 0 ; large rung, ytickmarks at default
  IF (p(0) LT .25) AND (p(0) GT .15) THEN yticks = 2
;                               ; smaller graph, reduce yticks
  IF p(0) LE .15 THEN yticks = 1       ; even smaller, reduce again

  plot, A1, B(*, 0), pos = [.2, vpos(1), 0.8, vpos(0)], $
    yrange = psc*p(0)*[0, 1], ystyle = yst, xtickname = xtickn, $
    yticks = yticks, yminor = yminor
  IF nplot GT 1 THEN BEGIN
    FOR i = 1, nplot-1 DO BEGIN
      IF i EQ nplot-1 THEN xtickn = replicate('', 30)
      IF p(i) GE .25 THEN yticks = 0
      IF (p(i) LT .25) AND (p(0) GT .15) THEN yticks = 2
      IF p(i) LE .15 THEN yticks = 1
      plot, A1, B(*, i), pos = [.2, vpos(i+1), .8, vpos(i)], $
        /noerase, yrange = psc*p(i)*[0, 1], ystyle = yst, $
        xtickname = xtickn, yticks = yticks, yminor = yminor
    ENDFOR
  ENDIF
  RETURN
END

function roundup, a
;
; NAME:
;  ROUNDUP
;
; FUNCTION:
;
; Round numbers up to the next highest "convenient" numbers.
; Mantissas for convenient numbers are given in increasing
; order in the array allowv.  The numbers in the array can be positive,
; negative or zero.  Positive numbers are rounded up, negative numbers
; are rounded down to a larger absolute value, and zero elements are
; left alone.
;
; INPUT:
;   a : Array of numbers to be rounded up (down for -ve numbers)
;
; OUTPUT:
```

```
;         b : Array of convenient output numbers.
;
;
arrayck = size(a)
if(arrayck(0) lt 1 ) then begin    ; check that
   print,'function roundup:  input must be an array' ; a is an
   return,0      ; array
endif
;
allowv =[1.2,1.5,2,2.5,3,4,5,6,8,10,12] ; output mantissas
incr = 1.05    ; multiply each input mantissa
;     ; by incr
;
index = where(a ne 0,count)  ; all non-zero elts. of a
if(count eq 0) then begin
   print,'all elements of the array a are zero'
   return,a
endif
;
al0 = a(index)     ; non-zero elts only
signa = al0/abs(al0)    ; sign of elts.
;
l10 = alog10(abs(al0))
expon = fix(l10)
indx1 = where(l10 lt 0,count1)
if count1 ne 0 then expon(indx1) = expon(indx1) - 1; correct exponent for
;       ; abs(a) < 1
;
rf10 = incr*10^(l10 - expon)                  ; 1 <= rf10 <= 10
n = size(allowv)
b1 = rf10
for i = 0,n(1)-1 do begin    ; work backwards
   indx = where(rf10 lt allowv(n(1)-1-i),count)  ; through allowv
   if count gt 0 then b1(indx) = allowv(n(1)-1-i)  ; set b1 = allowv
endfor       ; value if
;       ; rf10 < allowv
;
b1 = b1*10^float(expon)*signa   ; restore exponent & sign
b = a
b(index) = b1     ; put back zero elts
return,b
end
```

--

---------- ---------
Philip Sterne | sterne@dublin.llnl.gov
Lawrence Livermore National Laboratory | Phone   (510) 422-2510
Livermore, CA 94550 | Fax   (510) 422-7300