
Subject: Re: Polar plots

Posted by [rsimpson](#) on Wed, 28 Aug 1996 07:00:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

Don't know if this is exactly what you want, but it might help.

```
; Program Overview: polar_plot
; Polar_axis defines and displays a polar diagram representation, and
; then plots the data.
```

```
; J.Stupples , March 1994
```

```
; parameters: polardat - The polar data to be displayed.
```

```
; keywords: Charsize - The character size of the numeric data.
;           The default is 1.
```

```
; Color      - The color for the entire plot. The default
;           is the foreground color.
```

```
; Data_Range - Specifies the polar angle through which
;           the data is to be viewed. The default
;           is 0 - 360.
```

```
; Degrees    - Labels the polar axis in degrees.(switch)
```

```
; Font       - Specifies software or hardware font.
```

```
; Gridstyle  - The gridstyle pattern the axis is to be
;           made up from.
```

```
; Linestyle  - The linestyle in which the data is to be
;           displayed.
```

```
; Nodata     - Will draw an empty axis (switch)
```

```
; Radians    - Labels the polar axis in radians.(switch)
```

```
; Subtitle   - Prints a subtitle under the plot.
```

```
; Tickformat - Allows axis variables to be displayed
;           using fortran type formatting.
```

```
; Title      - Prints a title on the plot diagram.
```

```
; Xgridstyle - The gridstyle for the x-axis.
```

```
; Xminor     - The number of minor tickmarks on the
;           x-axis. The default is 36.
```

```
; Xstyle     - Specifies style of x-axis.
```

```
; Xticks     - The number of major tickmarks on the
;           x-axis. The default is 8.
```

```
; Ygridstyle - The gridstyle for the y-axis.
```

```
; Ynozero    - Prohibits the minimum value of the y-axis
;           from being zero.
```

```
; Yrange     - Specifies the range of the y-axis.
```

```
; Ystyle     - Specifies the style of the y-axis.
```

```
; Yticks     - The number of major tick marks on the
;           y-axis.
```

```
; *****
```

PRO Draw_labels,degrees,radians

IF degrees or radians THEN BEGIN

label = ['90','180','270']

position = [64,0,-3,-66,-70,0]

IF radians THEN BEGIN

label = ['!4p!3/2','!4p','!33!4p!3/2']

position = [64,0,-1,-66,-73,0]

ENDIF

FOR count = 0 , 4 ,2 DO BEGIN

xyouts, position(count), position(count+1), \$

label(((count +2)/2)-1)

ENDFOR

ENDIF

END

```
; *****
```

FUNCTION Find_num_of_ticks,polardat,yrange,yticks,ynozero,ystyle,ytic kv

tick_6 = 0

num_ticks = yticks

IF yticks EQ 0 THEN BEGIN

PLOT, polardat>(max (polardat) - 30.0),xstyle = 4,\$

ystyle = (ystyle or 4), /nodata, yrange = yrange,\$

yticks = yticks, ytickv = ytickv, ynozero = ynozero

range = !Y.Crange(1) - !Y.Crange(0)

IF (range lt 100) OR (range ge 1000) THEN BEGIN

func = '/'

IF range lt 100 THEN func='*'

REPEAT BEGIN

command = 'range = range' + func + '10'

a_dummy = execute (command)

ENDREP UNTIL (range ge 100) and (range lt 1000)

ENDIF

FOR i = 8, 3, -1 DO BEGIN

IF (range mod i EQ 0) and ((range / i) mod 10 EQ 0) THEN BEGIN

num_ticks = i

IF num_ticks EQ 6 THEN tick_6 = 1

ENDIF

ENDFOR

ENDIF

IF tick_6 AND num_ticks EQ 3 THEN num_ticks = 6

RETURN, num_ticks

END

.*****

FUNCTION draw_circles, inc, num_of_circles, angle,color,\$
 subtitle, title, y_gridstyle, xstyle, ygridstyle, ystyle

 IF ygridstyle NE 0 THEN y_gridstyle = ygridstyle
 num_angles = 361
 circle = fltarr (num_angles, /nozero)
 circle (*) = inc * num_of_circles
 blank_line = fltarr(2,num_of_circles)
 blank_line (0,*) = 0.0
 num_of_blanks = num_of_circles / 2
 size_of_blank = 8
 angle_of_blank = (360 * size_of_blank)/(2 * !Dpi * circle(0))
 blank_circle = fltarr(361 - angle_of_blank - 1.0)
 blank_circle(*) = circle(0)
 blank_angle = ((findgen(360 - angle_of_blank) / 360) * 2 * !Dpi)+\$
 (!Dpi / 2)+(!Dpi / (360 / angle_of_blank))
 xsize = (float (!D.X_Vsize))
 ysize = (float (!D.Y_Vsize))
 smalldim = xsize < ysize
 xdim = 0.45 * smalldim / xsize
 ydim = 0.45 * smalldim / ysize

 PLOT, (blank_circle + 7), blank_angle, /polar, xstyle = 5, ystyle = 5, \$
 position = [0.5 - xdim, 0.5 - ydim, 0.5 + xdim, 0.5 + ydim],\$
 color = color, subtitle = subtitle, title = title, /nodata

 count2 = -1

 FOR count = (circle(0)-(inc * num_of_circles)), \$
 ((circle(0) - inc) + 0.0001), inc DO BEGIN

 count2 = count2 + 1

 IF (count2 mod 2 EQ 0) AND (num_of_blanks GT 0) THEN BEGIN

 blank_line (1,count2) = (circle(0) - count - 3)

 angle_of_blank = fix((360 * size_of_blank) / \$
 (2 * !Dpi * (circle(0) - count)))

 blank_circle = fltarr(num_angles - angle_of_blank - 1)

 blank_circle(*) = circle(0)

 blank_angle = ((findgen(360 - angle_of_blank) / 360) * 2 * !Dpi) + \$
 (!Dpi / 2) + (!Dpi / (360 / angle_of_blank))

 OPLOT , blank_circle - count, blank_angle, /polar, color = color, \$
 linestyle = y_gridstyle, nodata = (ystyle and 4)

 num_of_blanks = num_of_blanks - 1

 ENDIF else BEGIN

 blank_line (1,count2) = (circle(0) - count - inc + 3)

 OPLOT , circle - count, angle, /polar, color = color, \$

```

        linestyle = y_gridstyle, nodata = (ystyle and 4)
    ENDELSE
ENDFOR
count2 = count2 + 1
blank_line(1,(((count2 / 2) * 2) - 1)) = 0
IF xstyle ne 4 THEN BEGIN
    FOR count = 0 , (n_elements(blank_line) / 2) - 2 , 2 DO BEGIN
        PLOTS,blank_line(*,count:count + 1),color = color,$
            linestyle = y_gridstyle
    ENDFOR
ENDIF
RETURN , circle
END

,*****
,

PRO Draw_lines , num_of_lines, circle, inc, num_of_ticks, max_val,$
    min_val, num_of_circles, color, gridstyle,$
    xgridstyle, xstyle

IF xgridstyle GT 0 THEN gridstyle = xgridstyle
line_angles = findgen (num_of_lines)
line_angles = line_angles * (360 / float(num_of_lines))
line_angles = (line_angles / 360) * 2 * !Dpi
lines = fltarr(2, num_of_lines * 2)
FOR count = 2, ((num_of_lines - 1) * 2), 2 DO BEGIN
    line_length = 1.05
    IF (line_angles(count / 2) + 0.1) mod (!Dpi / (num_of_ticks / 2))$
        GT 0.15 THEN BEGIN
        line_length = 1
    ENDIF
    IF (line_angles(count / 2) + 0.1) mod (!Dpi / 6) GT 0.15 THEN BEGIN
        x_centre = (circle(0) - (inc * (num_of_circles - 1))) *$
            sin(line_angles(count / 2))
        y_centre = (circle(0) - (inc * (num_of_circles - 1))) *$
            cos(line_angles(count / 2))
    ENDIF else BEGIN
        x_centre = 0.0
        y_centre = 0.0
    ENDELSE
    lines(0,count) = x_centre
    lines(1,count) = y_centre
    lines(0,count+1) = (line_length * (num_of_circles * inc)) *$
        sin(line_angles(count / 2))
    lines(1,count+1) = (line_length * (num_of_circles * inc)) *$
        cos(line_angles(count / 2))
    IF xstyle ne 4 THEN BEGIN
        PLOTS,lines(*,count:count+1), color = color, linestyle = gridstyle
    
```

```

ENDIF
ENDFOR
END

.*****
,

PRO Draw_values , num_of_circles, inc, y_vals, charsize, color, font,$
    xstyle

factor = 1
if !D.Name EQ "PS" then factor = 30.5555

x_size = !D.X_size / factor
y_size = !D.Y_size / factor
x_char_size = !D.X_ch_size / factor
y_char_size = !D.Y_ch_size / factor

IF charsize EQ -10 THEN BEGIN
    charsize = float(x_size < y_size) / 470
ENDIF

y_offset = (y_char_size * charsize) / (charsize / 0.11)

num_of_values = num_of_circles / 2
for count = 0, num_of_values - 1 do begin
    x_offset = (strlen(y_vals(count)) * x_char_size / $
        (charsize / 0.15)) * charsize
    y_pos = (num_of_circles - (count * 2)) * inc
    xyouts , - x_offset, y_pos - y_offset, y_vals(count),$
        charsize = charsize
endfor

END

.*****
,

FUNCTION extract_y_vals,y_vals, tickformat

dummy = y_vals
count1 = -1
FOR count = n_elements(y_vals) , 1, -2 DO BEGIN
    count1 = count1 +1
    dummy(count1) = y_vals(count - 1)
ENDFOR
polar_vals = dummy(0:count1 - 1)
polar_vals = string (polar_vals , format = tickformat)
RETURN , polar_vals
END

```

```
*****
```

```
PRO polar, polardat, charsize = charsize, color = color, font = font,$  
    gridstyle = gridstyle, linestyle = linestyle, $  
    nodata = nodata, subtitle = subtitle, title = title,$  
    xgridstyle = xgridstyle,xminor = xminor, xstyle = xstyle,$  
    xticks = xticks, ygridstyle = ygridstyle, ynozero = ynozero,$  
    yrange = yrange, ystyle = ystyle, yticks = yticks,$  
    ytickv = ytickv, degrees = degrees, radians = radians,$  
    data_range = data_range, tickformat = tickformat
```

```
IF (not keyword_set(charsize)) THEN charsize = -10  
IF (not keyword_set(color)) THEN color = !P.Color  
IF (not keyword_set(font)) THEN font = 0  
IF (not keyword_set(gridstyle)) THEN gridstyle = 0  
IF (not keyword_set(linestyle)) THEN linestyle = 0  
IF (not keyword_set(nodata)) THEN nodata = 0  
IF (not keyword_set(subtitle)) THEN subtitle = "  
IF (not keyword_set(title)) THEN title = "  
IF (not keyword_set(xgridstyle)) THEN xgridstyle = 0  
IF (not keyword_set(xminor)) THEN xminor = 36  
IF (not keyword_set(xstyle)) THEN xstyle = 0  
IF (not keyword_set(xticks)) THEN xticks = 8  
IF (not keyword_set(ygridstyle)) THEN ygridstyle = 0  
IF (not keyword_set(ynozero)) THEN ynozero = 0  
IF (not keyword_set(yrange)) THEN yrange = [0,0]  
IF (not keyword_set(ystyle)) THEN ystyle = 0  
IF (not keyword_set(yticks)) THEN yticks = 0  
IF (not keyword_set(ytickv)) THEN ytickv = [0,0]  
IF (not keyword_set(degrees)) THEN degrees = 0  
IF (not keyword_set(radians)) THEN radians = 0  
IF (not keyword_set(data_range)) THEN data_range = [0,360]  
IF (not keyword_set(tickformat)) THEN tickformat = "
```

```
PLOT, polardat>(max (polardat) - 30.0), ystyle = (ystyle or 4),$  
    xstyle = 4, yticks = yticks, yrange = yrange, ytickv = ytickv,$  
    /nodata, ynozero = ynozero
```

```
polardat1 = polardat - !Y.Crange(0)  
min_val = !Y.Crange(0)  
max_val = !Y.Crange(1)  
start = data_range(0)  
finish = data_range(1)  
data_angle = (reverse(findgen(n_elements (polardat)))) /$  
    n_elements (polardat) * (finish-start)) + 90 - finish  
data_angle = (data_angle / 360) * 2 * !Dpi
```

```

angle = (findgen (360) / (361) * 2.0 * !Dpi) + (!Dpi / 2)
angle = reverse(angle)

num_y_ticks = Find_num_of_ticks (polardat, yrange, yticks, ynozero,$
                                ystyle,ytickv)

y_step = fix((((max_val - min_val) / (num_y_ticks))))
y_vals = strtrim (string (fix (findgen (num_Y_ticks + 1)) * y_step), 2)
y_vals = strtrim (fix ((y_vals + min_val)), 2)

polar_vals = Extract_y_vals(y_vals, tickformat)

num_of_circles = num_y_ticks
inc = (max_val - min_val) / num_of_circles
scale_factor = float(60 / (inc * num_of_circles))
inc = inc * scale_factor
xstyle = xstyle and 4
line_gridstyle = gridstyle

circle = Draw_circles(inc, num_of_circles, angle, color,subtitle,$
                    title, gridstyle, xstyle, ygridstyle, ystyle)

Draw_lines , xminor, circle, inc, xticks, max_val, min_val,$
            num_of_circles, color, line_gridstyle, xgridstyle, xstyle
Draw_values , num_of_circles, inc, polar_vals, charsize, color, font,$
            xstyle

Draw_labels , degrees, radians

OPLOT , (polardat1 * scale_factor) > 0.0, data_angle, /polar,$
        color = color, linestyle = linestyle, nodata = nodata
END
--
Richard Simpson
Farnborough, Hants, Uk          Fax: 01252 392118
rsimpson@ewrcsdra.demon.co.uk

```
