
Subject: Re: common scale colorbars

Posted by [Mark Piper](#) on Tue, 26 Apr 2011 15:05:23 GMT

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

There are a few tricks involved. Here's an example (sorry for the overly verbose comments):

```
pro ng_common_colorbar_ex, save=save  
compile_opt idl2
```

```
; Example functions to plot. The first (f1) has a range of [0,100], the  
; second (f2) a range of [0,70].
```

```
d = dist(41)
```

```
max1 = 100.0
```

```
max2 = 70.0
```

```
f1 = d / max(d) * max1
```

```
f2 = d / max(d) * max2
```

```
; Explicitly set 11 contour levels: [0, 10, 20, ... 100].
```

```
n_levels = 11
```

```
levels = findgen(n_levels)/(n_levels-1)*max1
```

```
; Make a step color table for the first contour plot. The color table STEP_CT
```

```
; is a [256,3] array, but there are only n_levels=11 distinct colors (to
```

```
; check, load & view the color table in XPALETTE). The indices into the color
```

```
; tables (both original and step) are contour levels interpolated to the
```

```
; range of color table indices (i.e., the byte range).
```

```
ct_number = 4
```

```
ct_indices = bytscl(levels)
```

```
loadct, ct_number, rgb_table=ct, /silent
```

```
step_ct = congrid(ct[ct_indices, *], 256, 3)
```

```
; Display the first function using the step color table and the
```

```
; interpolated indices.
```

```
c1 = contour(f1, $
```

```
    layout=[2,1,1], $
```

```
    c_value=levels, $
```

```
    rgb_table=step_ct, $
```

```
    rgb_indices=ct_indices, $
```

```
    /fill, $
```

```
    title='Max = ' + strtrim(max1,2), $
```

```
    window_title='Discrete Colorbar Example')
```

```
; Display the second function using the original color table and the
```

```
; interpolated indices.
```

```
c2 = contour(f2, $
```

```
    layout=[2,1,2], $
```

```
    /current, $
```

```
    c_value=levels, $
```

```
rgb_table=ct_number, $ ; compare with c1
rgb_indices=ct_indices, $
/fill, $
title='Max = ' + strftime(max2,2))

; Display colorbar with first contour plot as a target. It needs n_levels + 1
; ticks to make labels line up correctly.
tick_labels = [strftime(fix(levels), 2), "] ; append empty string
cb = colorbar( $
target=c1, $
ticklen=0, $
major=n_levels+1, $
tickname=tick_labels, $
font_size=8, $
position=[0.2, 0.06, 0.8, 0.09])

; Save result to a PNG file.
if keyword_set(save) then c1.save, 'ng_common_colorbar_ex.png'
end
```
