Subject: Using color tables in function graphics
Posted by laura.hike on Thu, 12 Oct 2017 00:05:01 GMT
View Forum Message <> Reply to Message

I would like to plot multiple lines on the same set of axes and I would like each line to be a different color where the colors follow a specific IDL color table (e.g., rainbow and white = 39). In direct graphics, I could use

loadct, 39

then specify the colors using indices from 0 to 255. I can't find any way to do this in function graphics. Most functions only allow colors to be specified by names. The only exception I can find is using the rgb_table property with "plot," but this just applies the colors to the individual points in a line. Does anyone know a way to do this?

Thanks,

Laura

Subject: Re: Using color tables in function graphics
Posted by Jim Pendleton on Thu, 12 Oct 2017 03:36:27 GMT
View Forum Message <> Reply to Message

On Wednesday, October 11, 2017 at 6:05:02 PM UTC-6, laura...@gmail.com wrote:
> I would like to plot multiple lines on the same set of axes and I would like each line to be a different color where the colors follow a specific IDL color table (e.g., rainbow and white = 39). In direct graphics, I could use
>
> loadct, 39
>
> then specify the colors using indices from 0 to 255. I can't find any way to do this in function graphics. Most functions only allow colors to be specified by names. The only exception I can find is using the rgb_table property with "plot," but this just applies the colors to the individual points in a line. Does anyone know a way to do this?
>
> Thanks,
>
> Laura

One approach is to retrieve the color vectors from the specified color table, then apply the colors as RGB triplets like this:

```
IDL> loadct, 39
% LOADCT: Loading table Rainbow + white
IDL> tvlct, r, g, b, /get
IDL> p = plot(findgen(10), findgen(10), color = [r[100], g[100], b[100]])
IDL> p = plot(findgen(10), findgen(10)/2, color = [r[200], g[200], b[200]], /overplot)
```

In direct graphics with non-decomposed colors, this would be like

IDL> device, decomposed = 0
IDL> loadct, 39
IDL> plot, findgen(10), findgen(10), color = 100
IDL> oplot, findgen(10), findgen(10)/2, color = 200

Jim P.

Subject: Re: Using color tables in function graphics
Posted by Markus Schmassmann on Thu, 12 Oct 2017 09:27:08 GMT
View Forum Message <> Reply to Message

On 10/12/2017 05:36 AM, Jim P wrote:
> On Wednesday, October 11, 2017 at 6:05:02 PM UTC-6, laura...@gmail.com wrote:
>> I would like to plot multiple lines on the same set of axes and I
>> would like each line to be a different color where the colors follow a
>> specific IDL color table (e.g., rainbow and white = 39). In direct
>> graphics, I could use
>>
>> loadct, 39
>>
>> then specify the colors using indices from 0 to 255. I can't find
>> any way to do this in function graphics. Most functions only allow colors to
>> be specified by names. The only exception I can find is using the
>> rgb_table property with "plot," but this just applies the colors to the
>> individual points in a line. Does anyone know a way to do this?
> One approach is to retrieve the color vectors from the specified
> color table, then apply the colors as RGB triplets like this:
>
> IDL> loadct, 39
> % LOADCT: Loading table Rainbow + white
> IDL> tvlct, r, g, b, /get
> IDL> p = plot(findgen(10), findgen(10), color = [r[100], g[100], b[100]])
> IDL> p = plot(findgen(10), findgen(10)/2, color = [r[200], g[200], b[200]], /overplot)
>
> In direct graphics with non-decomposed colors, this would be like
>
> IDL> device, decomposed = 0
> IDL> loadct, 39
> IDL> plot, findgen(10), findgen(10), color = 100
> IDL> oplot, findgen(10), findgen(10)/2, color = 200
; use the VERT_COLORS keyword:

p=!null
for i=0,9 do p=plot(sin(!dpi*[0:2:.1]+i*.2), $

overplot=p,vert_colors=25*i,rgb_table=39)

Markus

Subject: Re: Using color tables in function graphics
Posted by laura.hike on Thu, 12 Oct 2017 18:26:57 GMT
View Forum Message <> Reply to Message

I like that -- thanks!

On Thursday, October 12, 2017 at 2:27:11 AM UTC-7, Markus Schmassmann wrote:
> On 10/12/2017 05:36 AM, Jim P wrote:
>> On Wednesday, October 11, 2017 at 6:05:02 PM UTC-6, laura...@gmail.com wrote:
>>> I would like to plot multiple lines on the same set of axes and I
>>> would like each line to be a different color where the colors follow a
>>> specific IDL color table (e.g., rainbow and white = 39). In direct
>>> graphics, I could use
>>>
>>> loadct, 39
>>>
>>> then specify the colors using indices from 0 to 255. I can't find
>>> any way to do this in function graphics. Most functions only allow colors to
>>> be specified by names. The only exception I can find is using the
>>> rgb_table property with "plot," but this just applies the colors to the
>>> individual points in a line. Does anyone know a way to do this?
>>
>
> p=!null
> for i=0,9 do p=plot(sin(!dpi*[0:2:.1]+i*.2), $
> overplot=p,vert_colors=25*i,rgb_table=39)
>
> Markus
