
Subject: Explain Contour Plot to Me

Posted by [David Fanning](#) on Fri, 09 Sep 2011 19:39:46 GMT

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

Folks,

OK, forget labeling the color bar. I can live without that.

I would consider the week a triumph if someone could just explain to me, like I was a 10 year old child, how to use the IDL 8.1 contour function to produce a contour plot with four colors of my choosing. The colors will be used like this:

Data between 0.00 and 0.25 should be red.

Data between 0.25 and 0.50 should be blue.

Data between 0.50 and 0.75 should be green.

Data between 0.75 and 1.00 should be yellow.

Here is the program I am trying to run, using some ideas from Mark Piper's program from yesterday. You see two different ways of trying to get the Colorbar() colors right. One of which actually works. But neither of the Contour() plots have the right colors. The Coyote Graphics plot at the end is what I am trying to achieve.

PRO ContourTest

```
; Create a simple, random dataset for contouring:
```

```
data = RANDOMU(-3L, 9, 9)
```

```
LoadCT, 0
```

```
TVLCT, 255, 0, 0, 0
```

```
TVLCT, 0, 0, 255, 1
```

```
TVLCT, 0, 255, 0, 2
```

```
TVLCT, 255, 255, 0, 3
```

```
TVLCT, rr, gg, bb, /GET
```

```
rgb = Transpose([rr[0:3],[gg[0:3],[bb[0:3]]])
```

```
Help, rgb
```

```
rgb = Congrid(rgb, 3, 256)
```

```
levels =[0.0, 0.25, 0.5, 0.75]
```

```
; Contour function try 1.
```

```
w1 = Window(DIMENSIONS=[500, 400])
```

```
ctr=contour(data, /CURRENT, C_VALUE=levels, $
```

```

    POSITION=[0.1, 0.1, 0.9, 0.8], /FILL, $
    RGB_TABLE=rgb, $
    AXIS_STYLE=2)
names = String(levels,FORMAT='(F0.2)')

cb = Colorbar(TARGET=ctr, $
    POSITION=[0.1, 0.85, 0.9, 0.9])

rgb = Transpose([[[rr[0:3]], [gg[0:3]], [bb[0:3]]]])
Help, rgb

; Contour function try 2.
w2 = Window(DIMENSIONS=[500, 400])
ctr=contour(data, /CURRENT, C_VALUE=levels, $
    POSITION=[0.1, 0.1, 0.9, 0.8], /FILL, $
    RGB_TABLE=rgb, RGB_INDICES=Indgen(4), $
    AXIS_STYLE=2)
names = String(levels,FORMAT='(F0.2)')

cb = Colorbar(TARGET=ctr, $
    POSITION=[0.1, 0.85, 0.9, 0.9])

; Coyote graphics.
cgWindow, WXSize=500, WYSize=400
cgContour, data, LEVELS=levels, C_COLORS=Indgen(4), $
    POSITION=[0.1, 0.1, 0.9, 0.8], /FILL, /ADDCMD
cgColorBar, NCOLORS=4, RANGE=[0,1], FORMAT='(F0.2)', $
    DIVISIONS=4, /FIT, /ADDCMD
END

```

If you can get this to me before Nadal beats Roddick,
you will get bonus credit, which can be used to ask
function graphics questions at some time in the future. ;-)

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Explain Contour Plot to Me

Posted by [lecacheux.alain](#) on Sat, 10 Sep 2011 19:09:28 GMT

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

On 10 sep, 01:50, David Fanning <n...@dfanning.com> wrote:

> David Fanning writes:

>>> I had to expand the 4 colors into a full 256 color table with 257

>>> contour levels to make the color bar display correctly.

>

>> Well, that sort of boggles the mind, but OK, maybe

>> that's how you do it. Unfortunately, it begs the

>> question of how you are now going to overlay the

>> contour lines themselves on this filled contour

>> plot. This, of course, is what I really wanted,

>> although my previous example was reduced to the

>> basics just to get the colors right.

>

> OK, we are making a little progress here. But there

> are still a couple of problems. The color bar insists

> on querying the data to get its range. The range apparently

> cannot be set, so the labels on the color bar are wrong

> and don't correspond to the contour levels. Can anyone

> think of a way to fix these? (I could, of course, force

> the contour labels to be what I want them to be, but

> aside from this feeling like cheating, I can't do it

> because the TICKNAME keyword will crash my machine.)

>

> Also, I cannot seem to set a character size for the plot

> independently of the character size for the contour labels.

> Does anyone know how to do this? In this case, I would like

> the contour plot to have the same size annotation as the

> color bar, with the inside contour labels appearing smaller.

>

> Note that the C_USE_LABEL_ORIENTATION keyword seems to

> keep the contour labels right side up, rather than the

> jumbled, upside down default values. That's the good

> news. The bad news is that is a long keyword to have to

> type for every contour plot you want to create!

>

> Here is the code so far.

>

> ;-----

> PRO ContourTest

>

> ; Create a simple, random dataset for contouring:

> data = RANDOMU(-3L, 9, 9)

> LoadCT, 0

> TVLCT, 255, 0, 0, 0

> TVLCT, 0, 0, 255, 1

> TVLCT, 0, 255, 0, 2

> TVLCT, 255, 255, 0, 3

>

```

> levels =[0.25, 0.5, 0.75, 1.00]
>
> ; Contour function.
> tvlct, rgb, /get
> rgb = congrid(rgb[0:3, *], 256, 3)
> clevels = findgen(257) / 256.
>
> w = window(dimensions=[500, 400])
> ctr = contour(data, /current, c_value=clevles, $
>             position=[0.1, 0.1, 0.9, 0.8], /fill, $
>             rgb_table=rgb, rgb_indices=indgen(256), $
>             axis_style=2, font_size=10)
> ctrOver = contour(data, /current, c_value=levels, $
>                 color=cgColor('charcoal', /row, /triple), $
>                 /overplot, c_label_show=Replicate(1, 4), $
>                 c_use_label_orientation=1, font_size=7)
> cb = colorbar(target=ctr, $
>             position=[0.1, 0.90, 0.9, 0.95], $
>             major=5, border_on=1, font_size=10)
>
> ; Can we force the font size to be larger? Apparently, not. :-(
> ctr.font_size=10
>
> ; Coyote graphics.
> levels =[0.0, 0.25, 0.5, 0.75]
>
> cgWindow, WXSize=500, WYSize=400
> cgContour, data, LEVELS=levels, C_COLORS=Indgen(4), $
>     POSITION=[0.1, 0.1, 0.9, 0.8], /FILL, /ADDCMD
> cgContour, data, LEVELS=levels, C_COLOR='charcoal', LABEL=1, $
>     C_CHARSIZE=1.0, /OVERPLOT, /ADDCMD
> cgColorBar, NCOLORS=4, RANGE=[0,1], FORMAT='(F0.2)', $
>     DIVISIONS=4, /FIT, /ADDCMD, MINOR=5, XTICKLEN=1.0
> END
> ;-----
>
> And here is a little lagniappe (small gift). A program
> to clean up any and all graphics windows on your display.
>
> ;-----
> PRO CleanUp
>
> ; Function graphics windows.
> w = GetWindows()
> FOR j=0,N_Elements(w)-1 DO (w[j]).close
>
> ; Widget windows or Coyote Graphics windows.
> Widget_Control, /Reset

```

```

>
> ; IDL direct graphics windows.
> WHILE !D.Window GT -1 DO WDelete, !D.Window
>
> END
> ;-----
>
> Cheers,
>
> David
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming: http://www.dfanning.com/
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

```

David,

Your example illustrates two flaws in NG that ITTVIS should correct as soon as possible (or, possibly, for which I was not yet able to find any right answers in ITTVIS documentation ...).

The first one is the fact that the OVERPLOT keyword overrides the font and axis styles of the overplotted plot, which is a real nonsense. In your case, the font size attributed to the "ctrOver" plot does actually change the one of the "ctr" plot. My work around would be: drawing "ctrOver" first without any axis, then "ctr" (I get the correct font size, while keeping a smaller font for the isocontour labels), finally pushing ctrOver in front by using the "Order" method. The second flaw, even more annoying, is the way in which the NG "Colorbar" function defines its color range: by using the true min and max of the (Y or Z) values of the underlying object (here your DATA array, through TARGET=ctr keyword). You get 0.0134482 and 0.984703, instead of 0 and 1. Setting ZRANGE=[0,1] does not work better. My workaround would be: creating a new contour object (here "ctrFake") with modified data having the desired range, using it as the target of the colorbar, then pushing it back or deleting it.

Your code, modified accordingly, is given below.

But I agree that using NG in its present state is often more acrobatics than good programming !
alx.

```

;-----
PRO ContourTest

```

```

; Create a simple, random dataset for contouring:
data = RANDOMU(-3L, 9, 9)
LoadCT, 0
TVLCT, 255, 0, 0, 0
TVLCT, 0, 0, 255, 1

```

```

TVLCT, 0, 255, 0, 2
TVLCT, 255, 255, 0, 3

levels =[0.25, 0.5, 0.75, 1.00]

; Contour function.
tvlct, rgb, /get
rgb = congrid(rgb[0:3, *], 256, 3)
clevels = findgen(257) / 256.

w = window(dimensions=[500, 400])
ctrOver = contour(data, /current, $
    c_value=levels, $
    color=cgColor('charcoal', /row, /triple), $
    c_label_show=Replicate(1, 4), $
    c_use_label_orientation=1, $
    font_size=7, $
    axis_style=0)
ctr = contour(data, /current, c_value=clevels, /overplot, $
    position=[0.1, 0.1, 0.9, 0.8], /fill, $
    rgb_table=rgb, rgb_indices=indgen(256), $
    axis_style=2, font_size=10)
ctrOver.Order, /BRING_FORWARD

ctr.GetData, z, x, y
z[0] = 0
z[1] = 1
ctrFake = contour(z, /current, c_value=clevels, /overplot, $
    position=[0.1, 0.1, 0.9, 0.8], /fill, $
    rgb_table=rgb, rgb_indices=indgen(256), $
    axis_style=2, font_size=10)
cb = colorbar(target=ctr, $
    position=[0.1, 0.90, 0.9, 0.95], $
    major=5, border_on=1, font_size=10, TICKVALUES=[0,levels])
ctrFake.Delete ; or ctrFake.Order, /SEND_TO_BACK as well.

; Can we force the font size to be larger? Apparently, not. :(
ctr.font_size=7

;; Coyote graphics.
;levels =[0.0, 0.25, 0.5, 0.75]
;
;
;
;cgWindow, WXSize=500, WYSize=400
;cgContour, data, LEVELS=levels, C_COLORS=Indgen(4), $
; POSITION=[0.1, 0.1, 0.9, 0.8], /FILL, /ADDCMD
;cgContour, data, LEVELS=levels, C_COLOR='charcoal', LABEL=1, $

```

```
; C_CHARSIZE=1.0, /OVERPLOT, /ADDCMD
;cgColorBar, NCOLORS=4, RANGE=[0,1], FORMAT='(F0.2)', $
; DIVISIONS=4, /FIT, /ADDCMD, MINOR=5, XTICKLEN=1.0
END
;-----
```

Subject: Re: Explain Contour Plot to Me

Posted by [David Fanning](#) on Sun, 11 Sep 2011 13:59:30 GMT

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

alx writes:

- > Your example illustrates two flaws in NG that ITTVIS should correct as
- > soon as possible (or, possibly, for which I was not yet able to find
- > any right answers in ITTVIS documentation ...).
- > The first one is the fact that the OVERPLOT keyword overrides the font
- > and axis styles of the overplotted plot, which is a real nonsense. In
- > your case, the font size attributed to the "ctrOver" plot does
- > actually change the one of the "ctr" plot. My work around would be:
- > drawing "ctrOver" first without any axis, then "ctr" (I get the
- > correct font size, while keeping a smaller font for the isocontour
- > labels), finally pushing ctrOver in front by using the "Order" method.

I hadn't thought of overplotting **before** I had drawn the plot, but, as I say, I am having trouble adapting to the "new graphics" way of doing things and I am trying to be more flexible. I'll give this a try.

- > The second flaw, even more annoying, is the way in which the NG
- > "Colorbar" function defines its color range: by using the true min and
- > max of the (Y or Z) values of the underlying object (here your DATA
- > array, through TARGET=ctr keyword). You get 0.0134482 and 0.984703,
- > instead of 0 and 1. Setting ZRANGE=[0,1] does not work better. My
- > workaround would be: creating a new contour object (here "ctrFake")
- > with modified data having the desired range, using it as the target of
- > the colorbar, then pushing it back or deleting it.

Even though your (fake) color bar seems to be labeled correctly now, the colors don't line up with the annotations. I think we can safely conclude that it is impossible to create a color bar that actually represents the data in this very simple way.

If it were, say, Matlab graphics routines that behaved like this I am sure we would all agree that the software "doesn't friggin' work!" Since it's IDL, though, I guess I'll get out my checkbook and write a \$500 maintenance check

for another load of this manure.

Cheers,

David

P.S. Please understand, my criticism of the software does not imply criticism of the many fine and bright people working at ITTVIS. I do not know what is wrong there, though, that causes this kind of software to be produced. But ITTVIS needs to fix this, and quickly, or IDL as a product will be dead and gone.

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Seppure ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Explain Contour Plot to Me

Posted by [lecacheux.alain](#) on Sun, 11 Sep 2011 16:53:58 GMT

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

On 11 sep, 15:59, David Fanning <n...@dfanning.com> wrote:

> alx writes:

>> Your example illustrates two flaws in NG that ITTVIS should correct as
>> soon as possible (or, possibly, for which I was not yet able to find
>> any right answers in ITTVIS documentation ...).

>> The first one is the fact that the OVERPLOT keyword overrides the font
>> and axis styles of the overplotted plot, which is a real nonsense. In
>> your case, the font size attributed to the "ctrOver" plot does
>> actually change the one of the "ctr" plot. My work around would be:
>> drawing "ctrOver" first without any axis, then "ctr" (I get the
>> correct font size, while keeping a smaller font for the isocontour
>> labels), finally pushing ctrOver in front by using the "Order" method.

>

> I hadn't thought of overplotting *before* I had drawn the plot,
> but, as I say, I am having trouble adapting to the "new graphics"
> way of doing things and I am trying to be more flexible. I'll
> give this a try.

>

>> The second flaw, even more annoying, is the way in which the NG
>> "Colorbar" function defines its color range: by using the true min and
>> max of the (Y or Z) values of the underlying object (here your DATA
>> array, through TARGET=ctr keyword). You get 0.0134482 and 0.984703,
>> instead of 0 and 1. Setting ZRANGE=[0,1] does not work better. My
>> workaround would be: creating a new contour object (here "ctrFake")

>> with modified data having the desired range, using it as the target of
>> the colorbar, then pushing it back or deleting it.

>

> Even though your (fake) color bar seems to be labeled
> correctly now, the colors don't line up with the
> annotations. I think we can safely conclude that
> it is impossible to create a color bar that actually
> represents the data in this very simple way.

>

> If it were, say, Matlab graphics routines that behaved like
> this I am sure we would all agree that the software "doesn't
> friggin' work!" Since it's IDL, though, I guess I'll get
> out my checkbook and write a \$500 maintenance check
> for another load of this manure.

>

> Cheers,

>

> David

>

> P.S. Please understand, my criticism of the software does
> not imply criticism of the many fine and bright people working
> at ITTVIS. I do not know what is wrong there, though, that
> causes this kind of software to be produced. But ITTVIS
> needs to fix this, and quickly, or IDL as a product
> will be dead and gone.

>

> --

> David Fanning, Ph.D.

> Fanning Software Consulting, Inc.

> Coyote's Guide to IDL Programming:<http://www.dfanning.com/>

> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

> Even though your (fake) color bar seems to be labeled
> correctly now, the colors don't line up with the
> annotations.

The color bar should appear correctly after changing "cb =
colorbar(target=ctr, \$...)" to "cb = colorbar(target=ctrFake, \$...".
Sorry for the typo in my previous message.

By the way, let me say that I agree with what you said in your P.S.
But I do not think that the NG software is as bad as you are feeling:
NG offers, in principle, much more possibilities than DG and ...
Matlab's graphics ! However, some tuning by ITTVIS is certainly
urgently needed as well as, mainly, a much better documentation !
alain.

Subject: Re: Explain Contour Plot to Me

Posted by [David Fanning](#) on Sun, 11 Sep 2011 17:14:04 GMT

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

alx writes:

- > By the way, let me say that I agree with what you said in your P.S.
- > But I do not think that the NG software is as bad as you are feeling:
- > NG offers, in principle, much more possibilities than DG and ...
- > Matlab's graphics ! However, some tuning by ITTVIS is certainly
- > urgently needed as well as, mainly, a much better documentation !

I don't dispute any of this, except that NG software is not as bad as I think it is. I haven't even **started** to talk about map projections and widget programming. Believe me, it worse than what I have exposed so far. :-)

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Sepore ma de ni thui. ("Perhaps thou speakest truth.")
