Subject: Re: Adding Text/Equations to Plots
Posted by David Fanning on Mon, 16 Mar 2009 19:26:49 GMT
View Forum Message <> Reply to Message

Heather writes:

> I'm hoping someone can help me figure out a way to add equations to a
> plot. I used POLY_FIT to find the coefficients of a line for my data,
> and plotted that line. I really want to add the equation of that line
> to my plot.
>
> I thought I could use legend.pro (a routine I found at
> http://astro.uni-tuebingen.de/software/idl/astrolib/plot/leg end.pro
> that I am already calling in my routine anyhow), but that requires the
> input "items" be a string. And I couldn't figure out how to create a
> string that would call the elements of the array in which the
> coefficients are.
> (This is what I tried:
> IDL> numbers=findgen(10)
> IDL> string="This is a number: numbers(6)"
> IDL> print, string
> This is a number: numbers(6)
>
> Clearly not what I want. And I can't do something like:
> IDL> string1="This is a number: "
> IDL> print, string1, numbers(6)
> This is a number: 6.00000
>
> because what it will actually look like to the legend.pro routine is:
> IDL> string2="This is a number: " numbers(6)
>
> string2="This is a number: " numbers(6)
> ^
> % Syntax error.
>
> Any advice would be greatly appreciated, but I'm fairly new to the
> world of IDL, so please use "simple" explanations!

"Simple explanations"!? Do you see what the world is
coming to?

OK, here is a simple explanation. What you want to
do is concatenate strings. That is a big word that
means "string them together like beads on a string".
Whoops! Two different meanings of "string" here. :-(

OK, looks like you maybe know what a string is in
computer-speak. What you need to know, in a nutshell,

is how to turn a number into a string. You do that,
believe it or not, with the STRING command. (Do you
see now how even simple explanations trip themselves up?)

Here is an example:

```
IDL> var_1 = 'dog'
IDL> number = 8
IDL> var_2 = 'something'
IDL> mystring = var_1 + ' ' + StrTrim(number,2) + ' ' + var_2
IDL> print, mystring
   dog 8 something
```

If you are really lucky, you can turn numbers into strings
with STRTRIM. If you are unlucky, you need to use the STRING
command, with (there is a good chance of this) the FORMAT
keyword set to a likely format.

```
number = 3.45096754
aString = String(number, FORMAT='(F5.2)')
```

The program Number_Formatter is handy for this sort of thing:

http://www.dfanning.com/programs/number_formatter.pro

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: Adding Text/Equations to Plots
Posted by Heather on Mon, 16 Mar 2009 19:33:37 GMT
View Forum Message <> Reply to Message

On Mar 16, 1:26 pm, David Fanning <n...@dfanning.com> wrote:
> Heather writes:
>> I'm hoping someone can help me figure out a way to add equations to a
>> plot. I used POLY_FIT to find the coefficients of a line for my data,
>> and plotted that line. I really want to add the equation of that line
>> to my plot.
>
>> I thought I could use legend.pro (a routine I found at

>> http://astro.uni-tuebingen.de/software/idl/astrolib/plot/leg end.pro
>> that I am already calling in my routine anyhow), but that requires the
>> input "items" be a string. And I couldn't figure out how to create a
>> string that would call the elements of the array in which the
>> coefficients are.
>> (This is what I tried:
>> IDL> numbers=findgen(10)
>> IDL> string="This is a number: numbers(6)"
>> IDL> print, string
>> This is a number: numbers(6)
>
>> Clearly not what I want. And I can't do something like:
>> IDL> string1="This is a number: "
>> IDL> print, string1, numbers(6)
>> This is a number: 6.00000
>
>> because what it will actually look like to the legend.pro routine is:
>> IDL> string2="This is a number: " numbers(6)
>
>> string2="This is a number: " numbers(6)
>> ^
>> % Syntax error.
>
>> Any advice would be greatly appreciated, but I'm fairly new to the
>> world of IDL, so please use "simple" explanations!
>
> "Simple explanations"!? Do you see what the world is
> coming to?
>
> OK, here is a simple explanation. What you want to
> do is concatenate strings. That is a big word that
> means "string them together like beads on a string".
> Whoops! Two different meanings of "string" here. :-(
>
> OK, looks like you maybe know what a string is in
> computer-speak. What you need to know, in a nutshell,
> is how to turn a number into a string. You do that,
> believe it or not, with the STRING command. (Do you
> see now how even simple explanations trip themselves up?)
>
> Here is an example:
>
> IDL> var_1 = 'dog'
> IDL> number = 8
> IDL> var_2 = 'something'
> IDL> mystring = var_1 + ' ' + StrTrim(number,2) + ' ' + var_2
> IDL> print, mystring
> dog 8 something

>
> If you are really lucky, you can turn numbers into strings
> with STRTRIM. If you are unlucky, you need to use the STRING
> command, with (there is a good chance of this) the FORMAT
> keyword set to a likely format.
>
>   number = 3.45096754
>   aString = String(number, FORMAT='(F5.2)')
>
> The program Number_Formatter is handy for this sort of thing:
>
> http://www.dfanning.com/programs/number_formatter.pro
>
> 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.")

I guess "simple" is a fairly relative classification. That actually
did it though. Thanks!

Subject: Re: Adding Text/Equations to Plots
Posted by Jean H. on Mon, 16 Mar 2009 19:45:11 GMT
View Forum Message <> Reply to Message

Heather wrote:

> IDL> numbers=findgen(10)
> IDL> string="This is a number: numbers(6)"
> IDL> print, string
> This is a number: numbers(6)

> Any advice would be greatly appreciated, but I'm fairly new to the
> world of IDL, so please use "simple" explanations!
> Thanks,
> Heather

Heather,
You can do string concatenation and conversion

print, "this is a number " + string(numbers[6])

you can also have more control... remove the blanks:

print, "this is a number " + strtrim(numbers[6],2)

or even add formats:
print, "this is a number " + string(numbers[6],format = '(I)')

and with no blanks

print, "this is a number " + strtrim(string(numbers[6],format = '(I)'),2)

Jean
PS: replace print, by text = if you want to save the string

Subject: Re: Adding Text/Equations to Plots
Posted by Jean H. on Mon, 16 Mar 2009 19:54:45 GMT
View Forum Message <> Reply to Message

doh, I hate when I receive the other poster messages 1/2 hour late...

so, sorry for the repetition..

Jean H. wrote:
> Heather wrote:
>
>> IDL> numbers=findgen(10)
>> IDL> string="This is a number: numbers(6)"
>> IDL> print, string
>> This is a number: numbers(6)
>
>> Any advice would be greatly appreciated, but I'm fairly new to the
>> world of IDL, so please use "simple" explanations!
>> Thanks,
>> Heather
>
> Heather,
> You can do string concatenation and conversion
>
> print, "this is a number " + string(numbers[6])
>
> you can also have more control... remove the blanks:
>
> print, "this is a number " + strtrim(numbers[6],2)
>
> or even add formats:
> print, "this is a number " + string(numbers[6],format = '(I)')
>

> and with no blanks
>
> print, "this is a number " + strtrim(string(numbers[6],format = '(I)'),2)
>
> Jean
> PS: replace print, by text = if you want to save the string

---

Subject: Re: Adding Text/Equations to Plots
Posted by Michael Galloy on Mon, 16 Mar 2009 20:16:43 GMT
View Forum Message <> Reply to Message

Jean H. wrote:
> Heather wrote:
>
>> IDL> numbers=findgen(10)
>> IDL> string="This is a number: numbers(6)"
>> IDL> print, string
>> This is a number: numbers(6)
>
>> Any advice would be greatly appreciated, but I'm fairly new to the
>> world of IDL, so please use "simple" explanations!
>> Thanks,
>> Heather
>
> Heather,
> You can do string concatenation and conversion
>
> print, "this is a number " + string(numbers[6])
>
> you can also have more control... remove the blanks:
>
> print, "this is a number " + strtrim(numbers[6],2)
>
> or even add formats:
> print, "this is a number " + string(numbers[6],format = '(I)')
>
> and with no blanks
>
> print, "this is a number " + strtrim(string(numbers[6],format = '(I)'),2)
>
> Jean
> PS: replace print, by text = if you want to save the string

I have been steering away from string concatenation and using C format
strings for almost everything lately:

IDL> print, numbers[6], format='(%"This is a number: %f.")'

I think the code is a more readable. Of course, then you have to learn
the available format codes (with C and Fortran versions to choose from).

The STRING command accept the same FORMAT if you want to pass the string
to a keyword of XYOUTS, PLOT, etc.

By the way, I hate that I have to type five extra characters to get
C-style format codes. Maybe a cformat option to COMPILE_OPT to switch
the default format to C-style? (Yes, I know "cformat" is seven characters.)

Mike
--
www.michaelgalloy.com
Associate Research Scientist
Tech-X Corporation
