NAME

plt − make 2-D plots

SYNOPSIS

plt [*data-spec*] [*data-file*] [[*xcol*] *ycol*] [*options ...*] [**-T lw | lwcat** [*lwcat-options*]]

DESCRIPTION

This man page is intended as a supplement to the command-line help provided by **plt** itself (using the **-h** option, see below). If you have not previously used **plt**, please look at the *plt Tutorial and Cookbook*, which is included in the **plt** package (see *SOURCES* below).

plt is a non-interactive (command line-driven) plotting utility. **plt** can produce publication-quality 2D plots in PostScript from easily-produced text or binary data files, and can also create screen plots under the X Window System.

All data presented to **plt** must be organized in rows and columns. Columns are numbered beginning with zero, and each column contains values for a variable that can be used as an abscissa (x coordinate), ordinate (y coordinate), or (with appropriate options described below) a grey level, color, or other plot attributes. Rows are numbered beginning with one, and each row contains a value for each column. Within a *data-file*, values are always arranged in row-major order (all elements of row 1, followed by all elements of row 2, etc.).

Usually, data must be in text form in order for **plt** to read them. Each non-empty, non-comment line (row) in the input should contain a value for each column that will be plotted; if there are additional values or other extra text at the end of a row, it will be ignored. Columns can be separated by any number of spaces or tabs. Commas and single or double quotation marks can also be used as column separators with current versions of **plt**, though not with older versions. It is not necessary to line up the values in each row. There may also be spaces or tabs at the beginning of a line, and these will also be ignored.

If no *data-file* is specified, **plt** reads data from its standard input. The command-line arguments *xcol* and *ycol* specify the column numbers for the abscissas and ordinates respectively. If only one column number is specified, it is taken as *ycol*, and **plt** generates a series of abscissas automatically. If the *data-file* contains no more than two columns, both *xcol* and *ycol* may be omitted.

By default, **plt** reads all rows of the *data-file* and scales the x and y axes so that all data can be plotted. An optional *data-spec*, a string beginning with a colon (:), can be used to select a subset of the rows in the *data-file*. For details on using a *data-spec*, and for information about reading binary data files using **plt**, see the *plt Tutorial and Cookbook*.

plt recognizes a large number of *options* for controlling and customizing plots. To see a summary of all options, run ‘‘**plt -h**’’; if this command is followed by one or more strings (which should not begin with hyphens), **plt** prints one-line summaries of all options beginning with those strings only.

plt can read its options from command-line arguments, from a *format file* (specified using the **-f** option), or from a *format string* (supplied on the command line, following the **-F** option). When using format files or format strings, omit the hyphen (-) before each option.

Options

Following is a brief summary of **plt**’s options. Note that many options require arguments. **plt** chooses a suitable default for most such arguments if the argument is supplied as ‘**-**’. See the *plt Tutorial and Cookbook* for further details.

-p *plot-styles*

Specify style(s) for data plots. Available *plot-styles* include ‘**c**’, ‘**C**’, ‘**e+***c*’, ‘**e-***c*’, ‘**e:***c*’, ‘**E+***n*’, ‘**E-***n*’, ‘**E:***n*’, ‘**f**’, ‘**i**’, ‘**l**’, ‘**m**’, ‘**n**’, ‘**N**’, ‘**o**’, ‘**O**’, ‘**s***c*’, ‘**S***n*’, and ‘**t**’.

-s *elements*

Suppress *elements* of output. Elements that can be suppressed include ‘**e**’ (erasing the screen or beginning a new page before plotting), ‘**a**’ (anything associated with axes), ‘**x**’ (anything associated with the x axis), ‘**y**’ (anything associated with the y axis), ‘**g**’ (the grid), ‘**m**’ (x and y axis tick marks), ‘**n**’ (x and y tick mark numbers), ‘**t**’ (x and y axis labels and plot title), ‘**l**’ (user-supplied labels), ‘**p**’ (data plots), and ‘**f**’ (‘‘figures’’ -- boxes, line segments, arrows, and legends). In addition, these *elements* modify the effects of any other elements that follow: '**X**' (restrict effects to x axis), '**Y**' (restrict effects to y axis), and '**A**' (apply effects to both axes); and the element '**C**' reenables all elements.

-X *xmin xmax*
Set the x-axis range (see also **-xa**).

-Y *ymin ymax*
Set the y-axis range (see also **-ya**).

-t *title* Set the title for the plot (enclose *title* in quotes if it contains whitespace or begins with '(' or '**[**').

-T *type* Specify the output *type*, which may be **xw** (X11 window, the default under Unix or Linux and not available under MS-Windows), or **lw** (PostScript, the default under MS-Windows).

-g *grid-mode*
Specify the grid style, which may be **in**, **out** (default), **both**, **none**, **sym** (make symmetric axes at top and right), **grid** (extend major ticks across the entire plot), **xgrid**, **ygrid**, or **sub** (extend all ticks across the entire plot).

-h [*option-prefix ...*]
Show help on options beginning with *option-prefix* (which should not begin with '**-**'). If *option-prefix* is omitted, show help on all options.

Within the next group of options, those with upper-case names ('**-A**', '**-B**', ...) use *window coordinates* between (0,0) and (1,1); those with lower-case names ('**-a**', '**-b**', ...) use *data coordinates*.

-a *x0 y0 x1 y1*
Draw an arrow to *(x0,y0)* from *(x1,y1)*.

-A *xw0 yw0 xw1 yw1*
Draw an arrow to *(xw0,yw0)* from *(xw1,yw1)*.

-b *x0 y0 x1 y1*
Draw a box with opposite corners at *(x0,y0) and (x1,y1)*.

-B *xw0 yw0 xw1 yw1*
Draw a box with opposite corners at *(xw0,yw0) and (xw1,yw1)*.

-c *x0 y0 x1 y1*
Connect points *(x0,y0)* and *(x1,y1)*.

-C *xw0 yw0 xw1 yw1*
Connect points *(xw0,yw0)* and *(xw1,yw1)*.

-d *x0 y0 x1 y1*
Draw a dark (filled) box with opposite corners at *(x0,y0) and (x1,y1)*.

-D *xw0 yw0 xw1 yw1*
Draw a dark (filled) box with opposite corners at *(xw0,yw0) and (xw1,yw1)*.

-l *x y tbc label-string*
Print *label-string* at *(x,y)*. The *tbc* argument is a two-character text box coordinate that specifies how the label is to be positioned relative to *(x,y)*; the default (**CC**) centers the string at *(x,y)*.

-L *xw yw tbc label-string*
As for **-l**, but using window coordinates *(xw,yw)*.

-w *configuration subwindow*
Confine the plot to a predefined window, specified by the arguments. *configuration* specifies the number of subwindows (panels), using one of '**m**' (1), '**b**' (2), or '**q**' (4), and *subwindow*' specifies which panel is to be plotted (0 or 1 for '**m**'; 0, 1, or 2 for '**b**'; or 0, 1, 2, 3, or 4 for '**q**'). In each case, subwindow 0 creates the frame of the entire plot, and the other subwindows refer to regions where data can be plotted. Use this option with '**-o**' or '**-s e**' to create multi-panel plots in stages without starting a new page or erasing the window before starting each new stage.

-W *xp0 yp0 xp1 yp1*
: Define the region of the page in which to plot. The arguments are *page coordinates*; the page coordinates (0,0) and (1,1) correspond to the lower left and upper right corners of the page.

-f *format-file*
: Read options from the specified *format-file*.

-fa *format-file*
: Record the current axis parameters as options in the specified *format-file* (for use with a later **plt** command). The previous contents of *format-file*, if any, will be overwritten.

-F *format-string*
: Read options from the specified *format-string*.

-o Suppress all output except data plots.

-cz *xfrom xincr*
: Generate abscissas, beginning with *xfrom* (default: 0) and incrementing by *xincr* (default: 1) at each step.

-ex Don't exclude points outside axis limits.

-hl *x y tbc n file*
: Print the next *n* (default: 1000) lines of the specified *file* as a label, placing the reference point for the first line of the label at data coordinates *(x,y)*. The *tbc* argument is defined as for **-l** and is applied to each line of the label. The *file* is opened when first used by **-hl** or **-vl**, and remains open, so that successive **-hl** or **-vl** options referring to the same *file* read and print successive lines. At most **MAXLABELFILES** (defined in **plt.h**, currently 6) *file*s of label strings can be open at once.

-vl *x y tbc n file*
: As for **-hl**, but print the label in a vertical orientation (rotated 90 degrees counterclockwise).

-le *linenumber plotnumber* [*text*]
: Define the specified *linenumber* in the legend (see also **-lp**). Line numbers in the legend begin with 0 (the top line); plot numbers also begin with 0 (these refer to the data plots, and are used here to determine the line style for the entry's sample plot segment). The *text* is printed to the right of the sample plot segment. To create an entry with more than one line of text, use additional **-le** options with different *linenumber*s as necessary, omitting the *plotnumber* (use '**-**') for all but the first. If the same data are plotted more than once in a single figure to create an overlay (for example, using symbols over line segments), an overlaid legend entry can be created using additional **-le** options with the same *linenumber* and different *plotnumbers*, omitting the *text* for all but the first.

-lp *xw0 yw0* [*boxscale* [*seglength* [*opaque*]]]
: Define the window coordinates *(xw0, yw0)* of the upper left corner of the plot legend text, and other attributes for the plot legend (key). **plt** determines the size of the box it draws around the legend, but the calculated width of the box is multiplied by *boxscale*. The *seglength* option specifies the length of the sample plot segments, as a fraction of the x-axis length (default: 0.05). If *opaque* is '**yes**' (default), the background of the legend is opaque white; otherwise, the background is transparent (any previously drawn material remains visible through the legend box). Unless a **-lp** option is provided, no legend is printed.

-lx [*base* [*subticks*]]
: Draw a logarithmic x-axis; *base* is the base of the logarithms (default: 10), and *subticks* is either '**yes**' or '**no**'. If the axis has a small number of major ticks, **plt** draws subticks by default; use the *subticks* argument to change **plt**'s default behavior.

-ly [*base* [*subticks*]]
: Draw a logarithmic y-axis.

-tf *file* [*tbc*]
Load the text string array from the specified *file*. Each line of the *file* defines an element of the string array; using plot styles **c** or **t**, these strings can be plotted in the same manner as data points. The optional *tbc* specifies how the positions of the strings are to be modified when they are printed, in the same way as for **-l**; by default, the strings are centered on the coordinates specified for them.

-ts "*string0 string1 ...***"** [*tbc*]
Load the text string array from the quoted argument (whitespace separates strings in the array) rather than from a file; otherwise, this option is the same as **-tf**.

-fs "*string0 string1 ...***"**
Load the font string array from the quoted argument. Using appropriate plot style (**-p**) options, the strings can be used to change the font, line style (solid, dotted, dashed, etc.), or drawing color.

-x *string*
Set the x-axis title to *string* (which must be quoted if this option is used on the command line or if *string* begins with '(' or '[').

-xa *xmin xmax tick fmt tskip ycross*
Specify the x-axis range (as *xmin* to *xmax*); the interval between x-axis tick marks; the format, *fmt*, in which to print the numbers (e.g., "**%.3f**", "**%.2e**"; any format that **printf(3)** can use for printing floating-point numbers is acceptable); the number of ticks per labelled tick, *tskip*; and *ycross*, the point on the y-axis that the x-axis should cross, in y-units. Any of these parameters may be supplied as "-", which causes **plt** to choose a reasonable value based on the input data.

-xe *xmin-error xmax-error*
Use this option to specify the amount by which the x-axis range is allowed to exceed the range of x-values in the input data, when **plt** determines the x-axis range automatically.

-xm *tick-base*
Make x-axis ticks be multiples of the specified *tick-base*.

-xo *x-axis-offset*
Move the x-axis down by *x-axis-offset* (expressed as a fraction of the y-axis length).

-xr Draw the x-axis at the top of the plot

-xt *x label* [*tick-size*]
Add an extra labelled tick at the specified *x* position, and label it with the specified *label* (which may be any string). The optional *tick-size* argument specifies the length of the added tick, as a fraction of the default length for labelled ticks (e.g., a value of 1.5 makes the added tick 50 longer than the standard size).

-xts *x* [*tick-size*]
Force a labelled tick to appear on the x-axis at the specified *x* (the positions of the other labelled x-ticks are adjusted accordingly). *tick-size* is defined as for **-xt**.

-y *string*
Set the y-axis title to *string* (see **-x**).

-ya *ymin ymax tick fmt tskip xcross*
Set up the y-axis (see **-xa**).

-ye *ymin-error ymax-error*
Set the allowable error in the y-axis range (see **-xe**).

-ym *tick-base*
Make y-axis ticks be multiples of the specified *tick-base*.

-yo *y-axis-offset*
Move the y-axis to the left by *y-axis-offset* (expressed as a fraction of the x-axis length).

-yr Draw the y-axis at the right edge of the plot.

-yt *y label* [*tick-size*]
Add an extra labelled tick at the specified *y* position (see **-xt**).

-yts *y* [*tick-size*]
Force a labelled tick to appear on the y-axis at the specified *y* (see **-xts**).

-dev *pterm option*
Process *option* only if the value of **PTERM** is *pterm*. The **-dev** option may be useful in scripts that produce screen or printed plots in different formats.

-sf *name specification*
Create a new font group with the specified *name* and set its specifications (font, point size, color/grey level, line width, and line style). See the chapter titled *Colors, Line Styles, and Fonts* in the *plt Tutorial and Cookbook* for details.

-ch *height-factor width-factor*
Modify the height and width of all characters printed in the plot by the specified factors.

-size *fscl width height left-margin bottom-margin*
Specify the size and position of the plot on the page. The *width*, *height*, *left-margin*, and *bottom-margin* are specified in *inches* (1 inch = 25.4 mm). *fscl* is a factor applied to the point size of all printed characters, *independently* of the scaling applied to the rest of the plot. This option is effective for printed plots only.

Screen and printed plots

By default, **plt** makes an X11 screen plot. To make a printed plot, use the option **-T lw**, and pipe the output of **plt** to **lwcat**. Under Unix or Linux, **lwcat** uses the standard **lpr** print spooler to send **plt**'s output in PostScript format to the default printer. Under MS-Windows, or using **lwcat**'s **-gv** option under Unix or Linux, the PostScript output is displayed on-screen using GhostScript (**GSView** under MS-Windows, or **gv** otherwise; these programs can save the output in a file or send it to a printer).

EXAMPLES

Create a text file with the following contents:

```
0  0  0
1  1  1
2  4  8
3  9 27
4 16 64
```

and call the file *powers*. Plot the first column vs. the second by:

plt powers 0 1 -t "Squares of small integers" -x "Integer" -y "Square"

The same file can be used to generate a number of different plots, by choosing different columns. To plot the third column vs. the first, try:

plt powers 2 0 -t "Marshmallows" -x "Mass (kg)" -y "Height (m)"

The *plt Tutorial and Cookbook* contains many more examples.

FILES

/usr/local/lib/ps/plt.pro
PostScript prolog for plots printed using **lwcat**

SEE ALSO

plt Tutorial and Cookbook (a book-length introduction to **plt**, included in the **plt** source package)

AUTHORS

plt was originally written by Paul Albrecht, and is currently maintained by George B. Moody (**george@mit.edu**).

SOURCES

http://www.physionet.org/physiotools/plt/