Plot Data Using

The format of data within a file can be selected with the **using** option. An explicit scanf string can be used, or simpler column choices can be made.

plot "datafile"

```
{ using {<ycol> |
<xcol>:<ycol> |
<xcol>:<ycol>:<ydelta> |
<xcol>:<ycol>:<width> |
<xcol>:<ycol>:<xdelta> |
<xcol>:<ycol>:<ylo>:<yhi> |
<xcol>:<ycol>:<xlo>:<xhi> |
<xcol>:<ycol>:<xdelta>:<ydelta> |
<xcol>:<ycol>:<ydelta>:<width> |
<xcol>:<ycol>:<ylo>:<yhi>:<width> |
<xc>:<yc>:<xlo>:<xhi>:<ylo>:<yhi>}
{"<scanf string>"}}...
```

splot "datafile"

```
{ using {<xcol>:<ycol>:<zcol>}
{" <scanf string> "}}...
```

<xcol>, <ycol>, and <zcol> explicitly select the columns to plot from a space or tab separated multicolumn data file. If only <ycol> is selected for **plot**, <xcol> defaults to 1. If only <zcol> is selected for **splot**, then only that column is read from the file. An <xcol> of 0 forces <ycol> to be plotted versus its coordinate number. <xcol>, <ycol>, and <zcol> can be entered as constants or expressions. Expressions enclosed in parentheses can be used to compute a column data value from all numbers in the input record.

If errorbars (see also **plot errorbars**) are used for **plot**s, xdelta or ydelta (for example, a +/- error) should be provided as the third column, or (x,y)low and (x,y)high as third and fourth columns. These columns must follow the x and y columns. If errorbars in both directions are wanted then xdelta and ydelta should be in the third and fourth columns, respectively, or xlow, xhigh, ylow, yhigh should be in the third, fourth, fifth, and sixth columns, respectively.

Scanf strings override any <xcol>:<ycol>(:<zcol>) choices, except for ordering of input, e.g.,

plot "datafile" `using 2:1 "%f%*f%f"`

causes the first column to be y and the third column to be x.

If the scanf string is omitted, the default is generated based on the <xcol>:<ycol>(:<zcol>) choices. If the **using** option is omitted, "%f%f" is used for **plot** ("%f%f%f%f" or "%f%f%f%f%f%f" for **errorbar plot**s) and "%f%f%f" is used for **splot**.

plot "MyData" `using "%*f%f%*20[^\n]%f" w lines`

Data are read from the file "MyData" using the format "%*f%f%*20[^\n]%f". The meaning of this format is: "%*f" ignore the first number, "%f" then read in the second and assign to x, "%*20[^\n]" then ignore 20 non-newline characters, "%f" then read in the y value.

Plot With Errorbars

Error bars are supported for 2-d data file plots by reading one to four additional columns specifying ydelta, ylow and yhigh, xdelta, xlow and xhigh, xdelta and ydelta, or xlow, xhigh, ylow, and yhigh respectively. No support exists for error bars for **splot**s.

In the default situation, gnuplot expects to see three to six numbers on each line of the data file, either (x, y, ydelta), (x, y, ylow, yhigh), (x, y, xdelta), (x, y, xlow, xhigh), (x, y, xdelta, ydelta), or (x, y, xlow, xhigh, ylow, yhigh). The x coordinate must be specified. The order of the numbers must be exactly as given above. Data files in this format can easily be plotted with error bars:

plot "data.dat" with errorbars (or yerrorbars)

plot "data.dat" with xerrorbars

plot "data.dat" with xyerrorbars

The error bar is a line plotted from (x, ylow) to (x, yhigh) or (xlow, y) to (xhigh, y). If ydelta is specified instead of ylow and yhigh, ylow=y-ydelta and yhigh=y+ydelta are derived. The values for xlow and xhigh are derived similarly from xdelta. If there are only two numbers on the line, yhigh and ylow are both set to y and xhigh and xlow are both set to x. To get lines plotted between the data points, **plot** the data file twice, once with errorbars and once with lines.

If x or y autoscaling is on, the x or y range will be adjusted to fit the error bars.

Boxes may be drawn with y error bars using the **boxerrorbars** style. The width of the box may be either set with the "set boxwidth" command, given in one of the data columns, or calculated automatically so each box touches the adjacent boxes. Boxes may be drawn instead of the cross drawn for the **xyerrorbars** style by using the **boxxyerrorbars** style.

x,y,ylow & yhigh from columns 1,2,3,4	`plot "data.dat" us 1:2:3:4 w errorbars`
x from third, y from second, xdelta from 6	`plot "data.dat" using 3:2:6 w xerrorbars`
x,y,xdelta & ydelta from columns 1,2,3,4	`plot "data.dat" us 1:2:3:4 w xyerrorbars`

Plot Ranges

The optional range specifies the region of the plot that will be displayed.

Ranges may be provided on the **plot** and **splot** command line and affect only that plot, or in the **set xrange**, **set yrange**, etc., commands, to change the default ranges for future plots.

[{<dummy-var>={<xmin>:<xmax>}] { [{<ymin>:<ymax>}] }

where <dummy-var> is the independent variable (the defaults are x and y, but this may be changed with **set dummy**) and the min and max terms can be constant expressions.

Both the min and max terms are optional. The ':' is also optional if neither a min nor a max term is specified. This allows '[ ]' to be used as a null range specification.

Specifying a range in the **plot** command line turns autoscaling for that axis off for that plot. Using one of the **set** range commands turns autoscaling off for that axis for future plots, unless changed later. (See **set autoscale**).

This uses the current ranges	`plot cos(x)`
This sets the x range only	`plot [-10:30] sin(pi*x)/(pi*x)`
This sets both the x and y ranges	`plot [-pi:pi] [-3:3] tan(x), 1/x`
sets only y range, & turns off autoscaling on both axes	`plot [ ] [-2:sin(5)*-8] sin(x)**besj0(x)`
This sets xmax and ymin only	`plot [:200] [-pi:]  exp(sin(x))`
This sets the x, y, and z ranges	`splot [0:3] [1:4] [-1:1] x*y`