Title

> *axis_scale_options* — Options for specifying axis scale, range, and look

Description

The *axis_scale_options* determine how axes are scaled (arithmetic, log, reversed), the range of the axes, and the look of the lines that are the axes.

Quick start

Put the y axis on a natural-log scale
    graph_command ... , ... yscale(log)

Reverse the order of the ticks and labels on the x axis
    graph_command ... , ... xscale(reverse)

Make the width of the line for the x axis thick
    graph_command ... , ... xscale(lwidth(thick))

Same as above, but specify line width as 18 point
    graph_command ... , ... xscale(lwidth(18pt))

Turn off the x axis
    graph_command ... , ... xscale(off)

Syntax

axis_scale_options are a subset of *axis_options*; see [G-3] **axis_options**.

axis_scale_options	Description
yscale(*axis_suboptions*)	how y axis looks
xscale(*axis_suboptions*)	how x axis looks
tscale(*axis_suboptions*)	how t (time) axis looks
zscale(*axis_suboptions*)	how contour legend axis looks

The above options are *merged-implicit*; see [G-4] **Concept: repeated options**.

axis_suboptions	Description
axis(*#*)	which axis to modify; $1 \leq \# \leq 9$
[no]log	use natural-logarithmic scale
[no]reverse	reverse scale to run from max to min
range(*numlist*)	expand range of axis
range(*datelist*)	expand range of t axis (tscale() only)
off and on	suppress/force display of axis
fill	allocate space for axis even if off
alt	move axis from left to right or from top to bottom
fextend	extend axis line through plot region and plot region's margin
extend	extend axis line through plot region
noextend	do not extend axis line at all
noline	do not draw axis line
line	force drawing of axis line
titlegap(*size*)	margin between axis title and tick labels
outergap(*size*)	margin outside axis title
lstyle(*linestyle*)	overall style of axis line
lcolor(*colorstyle*)	color and opacity of axis line
lwidth(*linewidthstyle*)	thickness of axis line
lpattern(*linepatternstyle*)	axis pattern (solid, dashed, etc.)

Options

yscale(*axis_suboptions*), xscale(*axis_suboptions*), and tscale(*axis_suboptions*) specify the look
of the y, x, and t axes. The t axis is an extension of the x axis. Inside the parentheses, you
specify *axis_suboptions*.

zscale(*axis_suboptions*); see *Contour axes—zscale()* below.

Suboptions

axis(*#*) specifies to which scale this axis belongs and is specified when dealing with multiple y or x axes; see [G-3] *axis_choice_options*.

log and nolog specify if the scale should be transformed to the natural log or should be arithmetic. nolog is the usual default, so log is the option. See *Obtaining log scales* under *Remarks and examples* below.

reverse and noreverse specify whether the scale should run from the maximum to the minimum or from the minimum to the maximum. noreverse is the usual default, so reverse is the option. See *Obtaining reversed scales* under *Remarks and examples* below.

range(*numlist*) specifies that the axis be expanded to include the numbers specified. Missing values, if specified, are ignored. See *Specifying the range of a scale* under *Remarks and examples* below.

range(*datelist*) (tscale() only) specifies that the axis be expanded to include the specified dates; see [U] **11.1.9 datelist**. Missing values, if specified, are ignored. See [TS] **tsline** for examples.

off and on suppress or force the display of the axis. on is the default and off the option. See *Suppressing the axes* under *Remarks and examples* below.

fill goes with off and is seldom specified. If you turned an axis off but still wanted the space to be allocated for the axis, you could specify fill.

alt specifies that, if the axis is by default on the left, it be on the right; if it is by default on the bottom, it is to be on the top. The following would draw a scatterplot with the y axis on the right:

    . scatter yvar xvar, yscale(alt)

fextend, extend, noextend, line, and noline determine how much of the line representing the axis is to be drawn. They are alternatives.

  noline specifies that the line not be drawn at all. The axis is there, ticks and labels will appear, but the line that is the axis itself will not be drawn.

  line is the opposite of noline, for use if the axis line somehow got turned off.

  noextend specifies that the axis line not extend beyond the range of the axis. Say that the axis extends from -1 to $+20$. With noextend, the axis line begins at -1 and ends at $+20$.

  extend specifies that the line be longer than that and extend all the way across the plot region. For instance, -1 and $+20$ might be the extent of the axis, but the scale might extend from -5 to $+25$, with the range $[-5, -1)$ and $(20, 25]$ being unlabeled on the axis. With extend, the axis line begins at -5 and ends at 25.

  fextend specifies that the line be longer than that and extend across the plot region and across the plot region's margins. For a definition of the plot region's margins, see [G-3] *region_options*. If the plot region has no margins (which would be rare), fextend means the same as extend. If the plot region does have margins, extend would result in the y and x axes not meeting. With fextend, they touch.

  fextend is the default with most schemes.

titlegap(*size*) specifies the margin to be inserted between the axis title and the axis tick labels; see [G-4] *size*.

outergap(*size*) specifies the margin to be inserted outside the axis title; see [G-4] *size*.

lstyle(*linestyle*), lcolor(*colorstyle*), lwidth(*linewidthstyle*), and lpattern(*linepatternstyle*) determine the overall look of the line that is the axis; see [G-4] **Concept: lines**.

Remarks and examples

axis_scale_options are a subset of *axis_options*; see [G-3] **axis_options** for an overview. The other appearance options are

axis_label_options	(see [G-3] **axis_label_options**)
axis_title_options	(see [G-3] **axis_title_options**)

Remarks are presented under the following headings:

> Use of the yscale( ) and xscale( )
> Specifying the range of a scale
> Obtaining log scales
> Obtaining reversed scales
> Suppressing the axes
> Contour axes—zscale()

Use of the yscale( ) and xscale( )

yscale() and xscale() specify the look of the y and x axes. Inside the parentheses, you specify *axis_suboptions*, for example:

```
. twoway (scatter ...) ... , yscale(range(0 10) titlegap(1))
```

yscale() and xscale() may be abbreviated ysc() and xsc(), suboption range() may be abbreviated r(), and titlegap() may be abbreviated titleg():

```
. twoway (scatter ...) ... , ysc(r(0 10) titleg(1))
```

Multiple yscale() and xscale() options may be specified on the same command, and their results will be combined. Thus the above command could also be specified

```
. twoway (scatter ...) ... , ysc(r(0 10)) ysc(titleg(1))
```

Suboptions may also be specified more than once, either within one yscale() or xscale() option, or across multiple options, and the rightmost suboption will take effect. In the following command, titlegap() will be 2, and range() 0 and 10:

```
. twoway (scatter ...) ... , ysc(r(0 10)) ysc(titleg(1)) ysc(titleg(2))
```

Specifying the range of a scale

To specify the range of a scale, specify the $\{y\,|\,x\}$scale(range(*numlist*)) option. This option specifies that the axis be expanded to include the numbers specified.

Consider the graph

```
. scatter yvar xvar
```

Assume that it resulted in a graph where the y axis varied over 1–100 and assume that, given the nature of the y variable, it would be more natural if the range of the axis were expanded to go from 0 to 100. You could type

```
. scatter yvar xvar, ysc(r(0))
```

Similarly, if the range without the yscale(range()) option went from 1 to 99 and you wanted it to go from 0 to 100, you could type

```
. scatter yvar xvar, ysc(r(0 100))
```

If the range without `yscale(range())` went from 0 to 99 and you wanted it to go from 0 to 100, you could type

```
. scatter yvar xvar, ysc(r(100))
```

Specifying missing for a value leaves the current minimum or maximum unchanged; specifying a nonmissing value changes the range, but only if the specified value is outside the value that would otherwise have been chosen. `range()` never narrows the scale of an axis or causes data to be omitted from the plot. If you wanted to graph *yvar* versus *xvar* for the subset of *xvar* values between 10 and 50, typing

```
. scatter yvar xvar, xsc(r(10 50))
```

would not suffice. You need to type

```
. scatter yvar xvar if xvar >=10 & xvar <=50
```
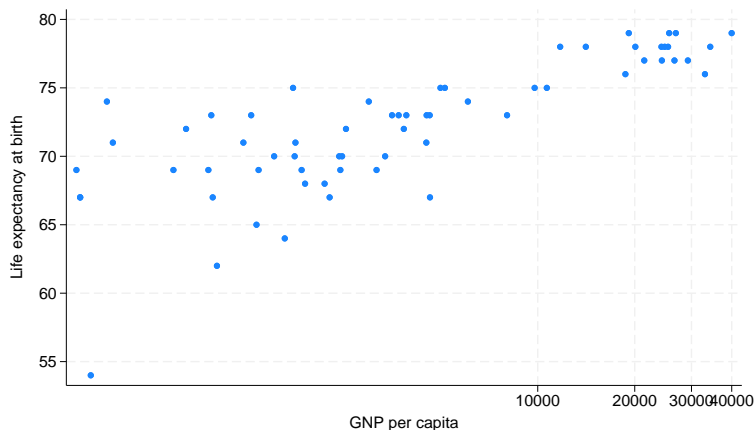
Obtaining log scales

To obtain log scales specify the $\{\,y\mid x\,\}$`scale(log)` option. Ordinarily when you draw a graph, you obtain arithmetic scales:

```
. use https://www.stata-press.com/data/r18/lifeexp, clear
(Life expectancy, 1998)
. scatter lexp gnppc
```
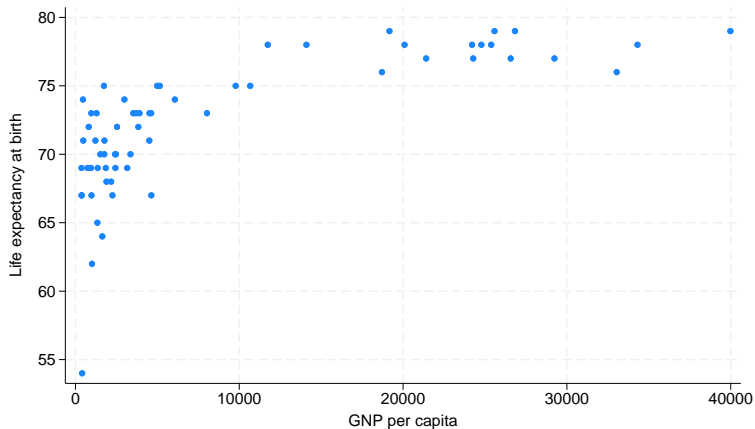
To obtain the same graph with a log x scale, we type

```
. scatter lexp gnppc, xscale(log)
```

We obtain the same graph as if we typed

```
. generate log_gnppc = log(gnppc)
. scatter lexp log_gnppc
```

The difference concerns the labeling of the axis. When we specify $\{y \mid x\}$scale(log), the axis is labeled in natural units. Here the overprinting of the 30,000 and 40,000 is unfortunate, but we could fix that by dividing gnppc by 1,000.

Obtaining reversed scales

To obtain reversed scales—scales that run from high to low—specify the $\{y \mid x\}$scale(reverse) option:

```
. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. scatter mpg weight, yscale(rev)
```
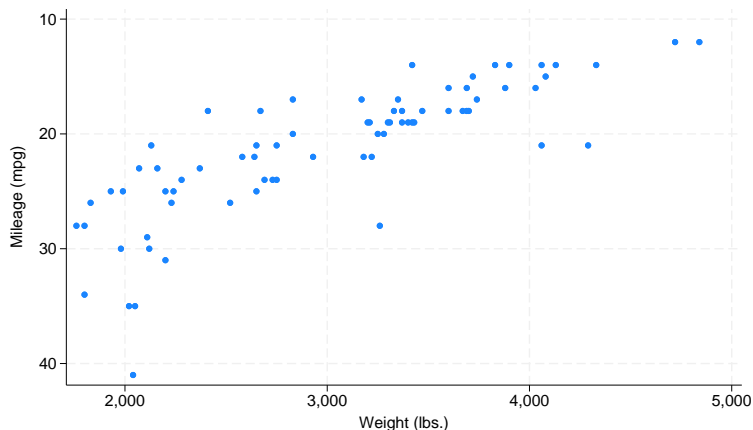
Suppressing the axes

There are two ways to suppress the axes. The first is to turn them off completely, which means that the axis line is suppressed, along with all of its ticking, labeling, and titling. The second is to simply suppress the axis line while leaving the ticking, labeling, and titling in place. The first is done by { y | x }scale(off) and the second by { y | x }scale(noline).
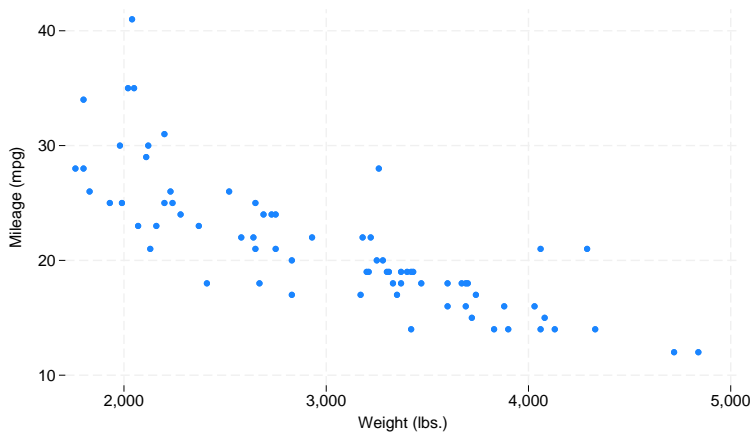
For example, below we eliminate the axes:

```
. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. scatter mpg weight, yscale(off) xscale(off)
```
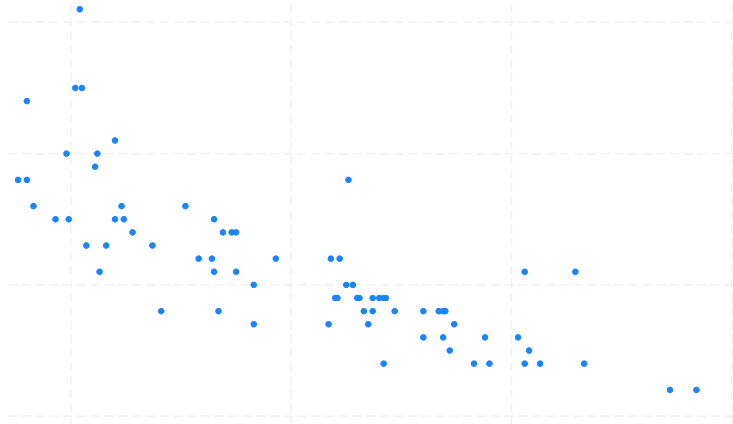

To eliminate the lines that are the axes while leaving in place the labeling, ticking, and titling, type

```
. scatter mpg weight, yscale(noline) xscale(noline)
```


Rather than using { y | x }scale(noline), you may specify { y | x }scale(lstyle(noline)) or { y | x }scale(lstyle(none)). They all mean the same thing.

Contour axes—zscale()

The zscale() option is unusual in that it applies not to axes on the plot region, but to the axis that shows the scale of a contour legend. It has effect only when the graph includes a twoway contour plot; see [G-2] **graph twoway contour**. In all other respects, it acts like xscale(), yscale(), and tscale().

References

Cox, N. J. 2008. Stata tip 59: Plotting on any transformed scale. *Stata Journal* 8: 142–145.

———. 2012. Speaking Stata: Transforming the time axis. *Stata Journal* 12: 332–341.

———. 2018. Speaking Stata: Logarithmic binning and labeling. *Stata Journal* 18: 262–286.

Also see

[G-3] *axis_options* — Options for specifying numeric axes

[G-3] *axis_label_options* — Options for specifying axis labels

[G-3] *axis_title_options* — Options for specifying axis titles

[G-3] *region_options* — Options for shading and outlining regions and controlling graph size

[TS] **tsline** — Time-series line plots