axis_choice_options — Options for specifying the axes on which a plot appears

DescriptionSyntaxOptionsRemarks and examplesReferenceAlso see

Description

The axis_choice_options determine the y and x axis (or axes) on which the plot is to appear.

Syntax

axis_choice_options	Description
yaxis(#[#])	which y axis to use, $1 \le \# \le 9$
$\underline{\mathtt{xax}}\mathtt{is}(\#[\#\dots])$	which x axis to use, $1 \le \# \le 9$

yaxis() and xaxis() are unique; see [G-4] Concept: repeated options.

These options are allowed with any of the *plottypes* (scatter, line, etc.) allowed by graph twoway; see [G-2] graph twoway.

Options

yaxis(#[#...]) and xaxis(#[#...]) specify the y or x axis to be used. The default is yaxis(1) and xaxis(1).

Typically, yaxis() and xaxis() are treated as if their syntax is yaxis(#) and xaxis(#)—that is, just one number is specified. In fact, however, more than one number may be specified, and specifying a second is sometimes useful with yaxis(). The first y axis appears on the left, and the second (if there is a second) appears on the right. Specifying $yaxis(1 \ 2)$ allows you to force there to be two identical y axes. You could use the one on the left in the usual way and the one on the right to label special values.

Remarks and examples

Options yaxis() and xaxis() are used when you wish to create one graph with multiple axes. These options are specified with twoway's scatter, line, etc., to specify which axis is to be used for each individual plot.

Remarks are presented under the following headings:

Usual case: one set of axes Special case: multiple axes due to multiple scales yaxis(1) and xaxis(1) are the defaults Notation style is irrelevant yaxis() and xaxis() are plot options Specifying the other axes options with multiple axes Each plot may have at most one x scale and one y scale Special case: Multiple axes with a shared scale

Usual case: one set of axes

Normally, when you construct a twoway graph with more than one plot, as in

. scatter y1 y2 x

or equivalently,

. twoway (scatter y1 x) (scatter y2 x)

the two plots share common axes for y and for x.

Special case: multiple axes due to multiple scales

Sometimes you want the two y plots graphed on separate scales. Then you type

```
. twoway (scatter gnp year, c(l) yaxis(1))
(scatter r year, c(l) yaxis(2))
```

yaxis(1) specified on the first scatter says, "This scatter is to appear on the first y axis." yaxis(2) specified on the second scatter says, "This scatter is to appear on the second y axis."

The result is that two y axes will be constructed. The one on the left will correspond to gnp and the one on the right to r. If we had two x axes instead, one would appear on the bottom and one on the top:

```
. twoway (scatter year gnp, c(l) xaxis(1))
(scatter year r, c(l) xaxis(2))
```

You are not limited to having just two y axes or two x axes. You could have two of each:

. twoway (scatter y1var x1var, c(1) yaxis(1) xaxis(1))
 (scatter y2var x2var, c(1) yaxis(2) xaxis(2))

You may have up to nine y axes and nine x axes, although graphs become pretty well unreadable by that point. When there are three or more y axes (or x axes), the axes are stacked up on the left (on the bottom). In any case, you specify yaxis (#) and xaxis (#) to specify which axis applies to which plot.

Also, you may reuse axes:

```
. twoway (scatter gnp year, c(l) yaxis(1))
      (scatter nnp year, c(l) yaxis(1))
      (scatter r year, c(l) yaxis(2))
      (scatter r2 year, c(l) yaxis(2))
```

The above graph has two y axes, one on the left and one on the right. The left axis is used for gnp and nnp; the right axis is used for r and r2.

The order in which we type the plots is not significant; the following would result in the same graph,

. twoway (scatter gnp year, c(l) yaxis(1))
 (scatter r year, c(l) yaxis(2))
 (scatter nnp year, c(l) yaxis(1))
 (scatter r2 year, c(l) yaxis(2))

except that the symbols, colors, and *linestyles* associated with each plot would change.

yaxis(1) and xaxis(1) are the defaults

In the first multiple-axis example,

```
. twoway (scatter gnp year, c(l) yaxis(1))
(scatter r year, c(l) yaxis(2))
```

xaxis(1) is assumed because we did not specify otherwise. The command is interpreted as if we had typed

```
. twoway (scatter gnp year, c(l) yaxis(1) xaxis(1))
      (scatter r year, c(l) yaxis(2) xaxis(1))
```

Because yaxis(1) is the default, you need not bother to type it. Similarly, because xaxis(1) is the default, you could omit typing it, too:

```
. twoway (scatter gnp year, c(1))
(scatter r year, c(1) yaxis(2))
```

Notation style is irrelevant

or

Whether you use the ()-binding notation or the ||-separator notation never matters. You could just as well type

. scatter gnp year, c(1) || scatter r year, c(1) yaxis(2)

yaxis() and xaxis() are plot options

Unlike all the other axis options, yaxis() and xaxis() are options of the individual plots and not of twoway itself. You may not type

. scatter gnp year, c(1) || scatter r year, c(1) ||, yaxis(2)

because twoway would have no way of knowing whether you wanted yaxis(2) to apply to the first or to the second scatter. Although it is true that how the axes appear is a property of twoway—see [G-3] *axis_options*—which axes are used for which plots is a property of the plots themselves.

For instance, options ylabel() and xlabel() are options that specify the major ticking and labeling of an axis (see [G-3] *axis_label_options*). If you want the x axis to have 10 ticks with labels, you can type

```
. scatter gnp year, c(l) ||
scatter r year, c(l) yaxis(2) ||, xlabel(#10)
```

and indeed you are "supposed" to type it that way to illustrate your deep understanding that xlabel() is a twoway option. Nonetheless, you may type

```
. scatter gnp year, c(1) ||
scatter r year, c(1) yaxis(2) xlabel(#10)
. scatter gnp year, c(1) xlabel(#10) ||
scatter r year, c(1) yaxis(2)
```

because twoway can reach inside the individual plots and pull out options intended for it. What twoway cannot do is redistribute options specified explicitly as twoway back to the individual plots.

Specifying the other axes options with multiple axes

Continuing with our example,

```
. scatter gnp year, c(1) ||
scatter r year, c(1) yaxis(2) ||
, xlabel(#10)
```

say that you also wanted 10 ticks with labels on the first y axis and 8 ticks with labels on the second y axis. You type

```
. scatter gnp year, c(1) ||
scatter r year, c(1) yaxis(2) ||
, xlabel(#10) ylabel(#10, axis(1)) ylabel(#8, axis(2))
```

Each of the other axis options (see [G-3] *axis_options*) has an axis(#) suboption that specifies to which axis the option applies. When you do not specify the suboption, axis(1) is assumed.

As always, even though the other axis options are options of twoway, you can let them run together with the options of individual plots:

```
. scatter gnp year, c(1) ||
scatter r year, c(1) yaxis(2) xlabel(#10) ylabel(#10, axis(1))
ylabel(#8, axis(2))
```

Each plot may have at most one x scale and one y scale

Each scatter, line, connected, etc.—that is, each plot—may have only one y scale and one x scale, so you may not type the shorthand

. scatter gnp r year, c(l l) yaxis(1 2)

to put gnp on one axis and r on another. In fact, yaxis(1 2) is not an error—we will get to that in the next section—but it will not put gnp on one axis and r on another. To do that, you must type

```
. twoway (scatter gnp year, c(l) yaxis(1))
(scatter r year, c(l) yaxis(2))
```

which, of course, you may type as

. scatter gnp year, c(l) yaxis(1) || scatter r year, c(l) yaxis(2)

The overall graph may have multiple scales, but the individual plots that appear in it may not.

Special case: Multiple axes with a shared scale

It is sometimes useful to have multiple axes just so that you have extra places to label special values. Consider graphing blood pressure versus concentration of some drug:

```
. scatter bp concentration
```

Perhaps you would like to add a line at bp = 120 and label that value specially. One thing you might do is

. scatter bp concentration, yaxis(1 2) ylabel(120, axis(2))

The ylabel(120, axis(2)) part is explained in [G-3] *axis_label_options*; it caused the second axis to have the value 120 labeled. The option yaxis(1 2) caused there to be a second axis, which you could label. When you specify yaxis() (or xaxis()) with more than one number, you are specifying that the axes be created sharing the same scale.

To better understand what yaxis(12) does, compare the results of

. scatter bp concentration

with

```
. scatter bp concentration, yaxis(1 2)
```

In the first graph, there is one y axis on the left. In the second graph, there are two y axes, one on the left and one on the right, and they are labeled identically.

Now compare

. scatter bp concentration

with

. scatter bp concentration, xaxis(1 2)

In the first graph, there is one x axis on the bottom. In the second graph, there are two x axes, one on the bottom and one on the top, and they are labeled identically.

Finally, try

. scatter bp concentration, yaxis(1 2) xaxis(1 2)

In this graph, there are two y axes and two x axes: left and right, and top and bottom.

Reference

Wiggins, V. L. 2010. Stata tip 93: Handling multiple y axes on twoway graphs. Stata Journal 10: 689-690.

Also see

- [G-3] axis_label_options Options for specifying axis labels
- [G-3] axis_options Options for specifying numeric axes
- [G-3] axis_scale_options Options for specifying axis scale, range, and look
- [G-3] axis_title_options Options for specifying axis titles

Stata, Stata Press, Mata, NetCourse, and NetCourseNow are registered trademarks of StataCorp LLC. Stata and Stata Press are registered trademarks with the World Intellectual Property Organization of the United Nations. StataNow is a trademark of StataCorp LLC. Other brand and product names are registered trademarks or trademarks of their respective companies. Copyright © 1985–2025 StataCorp LLC, College Station, TX, USA. All rights reserved.



For suggested citations, see the FAQ on citing Stata documentation.