

cat_axis_label_options — Options for specifying look of categorical axis labels

Description
Also see

Quick start

Syntax

Options

Remarks and examples

Description

The *cat_axis_label_options* determine the look of the labels that appear on a categorical *x* axis produced by `graph bar`, `graph hbar`, `graph dot`, and `graph box`; see [G-2] **graph bar**, [G-2] **graph dot**, and [G-2] **graph box**. These options are specified inside `label()` of `over()`:

```
. graph ..., over(varname, ... label(cat_axis_label_options) ...)
```

The most useful *cat_axis_label_options* are `angle()`, `alternate`, `labcolor()`, and `labsize()`.

Quick start

Increase size of categorical axis labels by 20% for plots drawn over categorical variable `catvar`
`graph_command ... , over(catvar, label(labsize(*1.2)))`

Place the categorical axis labels at a 45-degree angle
`graph_command ... , over(catvar, label(angle(45)))`

Alternate the placement of labels to increase space between adjacent labels
`graph_command ... , over(catvar, label(alternate))`

Make the categorical axis labels green
`graph_command ... , over(catvar, label(labcolor(green)))`

Suppress the display of labels on the categorical axis
`graph_command ... , over(catvar, label(nolabels))`

Note: Categorical axis-label options can be used only with `graph bar`, `graph box`, `graph dot`, or `graph hbar`.

Syntax

<i>cat_axis_label_options</i>	Description
<code>nolabels</code>	suppress axis labels
<code>ticks</code>	display axis ticks
<code>angle(<i>anglestyle</i>)</code>	angle of axis labels
<code>alternate</code>	offset adjacent labels
<code>tistyle(<i>tickstyle</i>)</code>	labels and ticks: overall style
<code>labgap(<i>relativesize</i>)</code>	labels: margin between tick and label
<code>labstyle(<i>textstyle</i>)</code>	labels: overall style
<code>labsize(<i>textsizestyle</i>)</code>	labels: size of text
<code>labcolor(<i>colorstyle</i>)</code>	labels: color of text
<code>tlength(<i>relativesize</i>)</code>	ticks: length
<code>tposition(<u>outside</u> <u>crossing</u> <u>inside</u>)</code>	ticks: position/direction
<code>tlstyle(<i>linestyle</i>)</code>	ticks: linestyle of
<code>tlwidth(<i>linewidthstyle</i>)</code>	ticks: thickness of line
<code>tlcolor(<i>colorstyle</i>)</code>	ticks: color of line

Options

`nolabels` suppresses display of category labels on the axis. For `graph bar` and `graph hbar`, the `nolabels` option is useful when combined with the `blabel()` option used to place the labels on the bars themselves; see [G-3] *blabel_option*.

`ticks` specifies that ticks appear on the categorical *x* axis. By default, ticks are not presented on categorical axes, and it is unlikely that you would want them to be.

`angle(anglestyle)` specifies the angle at which the labels on the axis appear. The default is `angle(0)`, meaning horizontal. With vertical bar charts and other vertically oriented charts, it is sometimes useful to specify `angle(90)` (vertical text reading bottom to top), `angle(-90)` (vertical text reading top to bottom), or `angle(-45)` (angled text reading top left to bottom right); see [G-4] *anglestyle*.

Unix users: if you specify `angle(-45)`, results will appear on your screen as if you specified `angle(-90)`; results will appear correctly when you print.

`alternate` causes adjacent labels to be offset from one another and is useful when there are many labels or when labels are long. For instance, rather than obtaining an axis labeled,

```

|-----|-----|-----|-----|
|   |   |   |   |
ResearchDevelopmentMarketing Sales

```

with `alternate`, you obtain

```

|-----|-----|-----|-----|
|   |   |   |   |
Research           Marketing
Development                               Sales

```

`tistyle(tickstyle)` specifies the overall look of labels and ticks; see [G-4] *tickstyle*. Here the emphasis is on labels because ticks are usually suppressed on a categorical axis. The options documented below will allow you to change each attribute of the label and tick, but the *tickstyle* specifies the starting point.

You need not specify `tstyle()` just because there is something you want to change about the look of labels and ticks. You specify `tstyle()` when another style exists that is exactly what you desire or when another style would allow you to specify fewer changes to obtain what you want.

`labgap(relativesize)`, `labstyle(textstyle)`, `labsize(textsizestyle)`, and `labcolor(colorstyle)` specify details about how the labels are presented. Of particular interest are `labsize(textsizestyle)`, which specifies the size of the labels, and `labcolor(colorstyle)`, which specifies the color of the labels; see [G-4] [textsizestyle](#) and [G-4] [colorstyle](#) for a list of text sizes and color choices. Also see [G-4] [relativesize](#) and [G-4] [textstyle](#).

`tlength(relativesize)` specifies the overall length of the ticks; see [G-4] [relativesize](#).

`tposition(outside | crossing | inside)` specifies whether the ticks are to extend *outside* (from the axis out, the usual default), *crossing* (crossing the axis line, extending in and out), or *inside* (from the axis into the plot region).

`tlstyle(linestyle)`, `tlwidth(linewidthstyle)`, and `tlcolor(colorstyle)` specify other details about the look of the ticks. Ticks are just lines. See [G-4] [concept: lines](#) for more information.

Remarks and examples

[stata.com](http://www.stata.com)

You draw a bar, dot, or box plot of `empcost` by `division`:

```
. graph ... empcost, over(division)
```

Seeing the result, you wish to make the text labeling the divisions 20% larger. You type:

```
. graph ... empcost, over(division, label(labsize(*1.2)))
```

Also see

[G-2] [graph bar](#) — Bar charts

[G-2] [graph box](#) — Box plots

[G-2] [graph dot](#) — Dot charts (summary statistics)