

graph twoway area — Twoway line plot with area shading[Description](#)
[Options](#)[Quick start](#)
[Remarks and examples](#)[Menu](#)
[Also see](#)[Syntax](#)

Description

`twoway area` displays (y,x) connected by straight lines and shaded underneath.

Quick start

Area plot with shading between the line for y versus x and 0, sorting on values of x

```
twoway area y x, sort
```

As above, but with shading between the line of y versus x and the y axis

```
twoway area y x, sort horizontal
```

Fill to 100 rather than 0

```
twoway area y x, sort base(100)
```

Label values on the y axis from 0 to 50 in units of 10, and change the angle of the labels

```
twoway area y x, sort ylabel(0(10)50, angle(0))
```

Specify maroon as the line and fill color, and set fill-color intensity to 50%

```
twoway area y x, sort color(maroon) fintensity(50)
```

Add the title “My Title” to the graph

```
twoway area y x, sort title("My Title")
```

Menu

Graphics > Twoway graph (scatter, line, etc.)

Syntax

```
twoway area yvar xvar [if] [in] [, options]
```

<i>options</i>	Description
<code>vertical</code>	vertical area plot; the default
<code>horizontal</code>	horizontal area plot
<code>cmissing(y n)</code>	missing values do not force gaps in area; default is <code>cmissing(y)</code>
<code>base(#)</code>	value to drop to; default is 0
<code>nodropbase</code>	programmer's option
<code>sort</code>	sort by <i>xvar</i> ; recommended
<i>area_options</i>	change look of shaded areas
<i>axis_choice_options</i>	associate plot with alternative axis
<i>twoway_options</i>	titles, legends, axes, added lines and text, by, regions, name, aspect ratio, etc.

Option `base()` is *rightmost*; `vertical`, `horizontal`, `nodropbase`, and `sort` are *unique*; see [G-4] **Concept: repeated options**.

Options

`vertical` and `horizontal` specify either a vertical or a horizontal area plot. `vertical` is the default. If `horizontal` is specified, the values recorded in *yvar* are treated as *x* values, and the values recorded in *xvar* are treated as *y* values. That is, to make horizontal plots, do not switch the order of the two variables specified.

In the `vertical` case, shading at each *xvar* value extends up or down from 0 according to the corresponding *yvar* values. If 0 is not in the range of the *y* axis, shading extends up or down to the *x* axis.

In the `horizontal` case, shading at each *xvar* value extends left or right from 0 according to the corresponding *yvar* values. If 0 is not in the range of the *x* axis, shading extends left or right to the *y* axis.

`cmissing(y|n)` specifies whether missing values are to be ignored when drawing the area or if they are to create breaks in the area. The default is `cmissing(y)`, meaning that they are ignored. Consider the following data:

	y1	y2	x
1.	1	2	1
2.	3	5	2
3.	5	4	3
4.	.	.	.
5.	6	7	5
6.	11	12	8

Say that you graph these data by using `twoway area y1 y2 x`. Do you want a break in the area between 3 and 5? If so, you type

```
. twoway area y1 y2 x, cmissing(n)
```

and two areas will be drawn, one for the observations before the missing values at observation 4 and one for the observations after the missing values.

If you omit the option (or type `cmismissing(y)`), the data are treated as if they contained

	y1	y2	x
1.	1	2	1
2.	3	5	2
3.	5	4	3
4.	6	7	5
5.	11	12	8

meaning that one contiguous area will be drawn over the range (1,8).

`base(#)` specifies the value from which the shading should extend. The default is `base(0)`, and in the above description of options `vertical` and `horizontal`, this default was assumed.

`nodropbase` is a programmer's option and is an alternative to `base()`. It specifies that rather than the enclosed area dropping to `base(#)`—or `base(0)`—it drops to the line formed by (y_1, x_1) and (y_N, x_N) , where (y_1, x_1) are the y and x values in the first observation being plotted and (y_N, x_N) are the values in the last observation being plotted.

`sort` specifies that the data be sorted by `xvar` before plotting.

`area_options` set the look of the shaded areas. The most important of these options is `color` (*colorstyle*), which specifies the color and opacity of both the area and its outline; see [G-4] *colorstyle* for a list of color choices. See [G-3] *area_options* for information on the other *area_options*.

`axis_choice_options` associate the plot with a particular y or x axis on the graph; see [G-3] *axis_choice_options*.

`twoway_options` are a set of common options supported by all `twoway` graphs. These options allow you to title graphs, name graphs, control axes and legends, add lines and text, set aspect ratios, create graphs over `by()` groups, and change some advanced settings. See [G-3] *twoway_options*.

Remarks and examples

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

Remarks are presented under the following headings:

Typical use
Advanced use
Cautions

Typical use

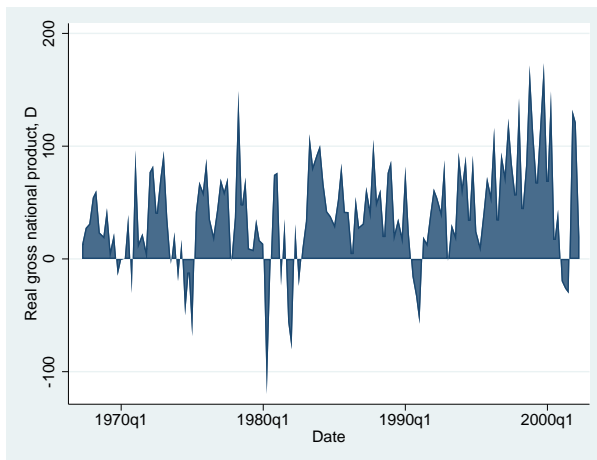
We have quarterly data recording the U.S. GNP in constant 1996 dollars:

```
. use https://www.stata-press.com/data/r17/gnp96  
(1996 gross national product)  
. list in 1/5
```

	date	gnp96
1.	1967q1	3631.6
2.	1967q2	3644.5
3.	1967q3	3672
4.	1967q4	3703.1
5.	1968q1	3757.5

In our opinion, the area under a curve should be shaded only if the area is meaningful:

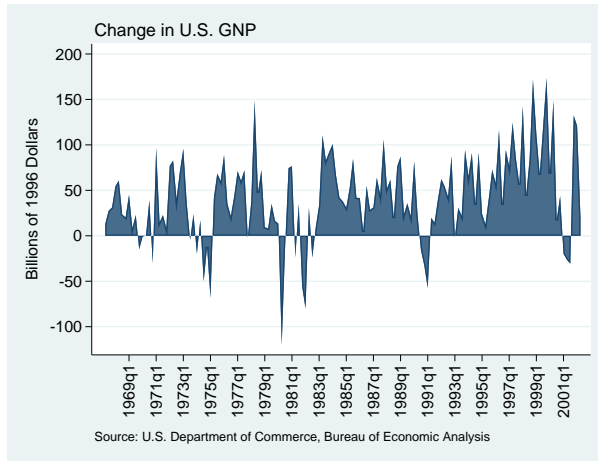
```
. use https://www.stata-press.com/data/r17/gnp96, clear  
(1996 gross national product)  
. twoway area d.gnp96 date
```



Advanced use

Here is the same graph, but greatly improved with some advanced options:

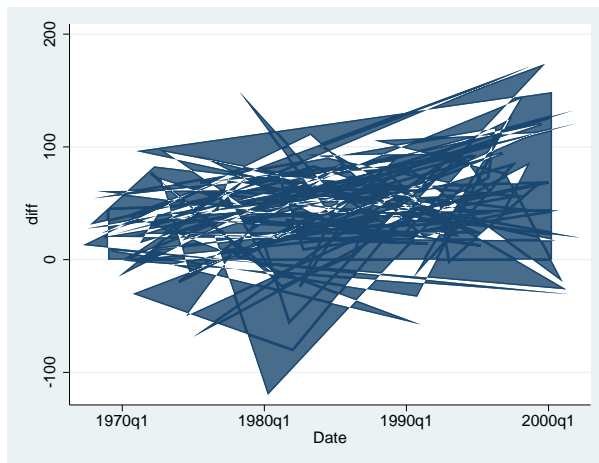
```
. twoway area d.gnp96 date, xlabel(36(8)164, angle(90))
  ylabel(-100(50)200, angle(0))
  ytitle("Billions of 1996 Dollars")
  xtitle("")
  subtitle("Change in U.S. GNP", position(11))
  note("Source: U.S. Department of Commerce,
        Bureau of Economic Analysis")
```



Cautions

Be sure that the data are in the order of *xvar*, or specify *area*'s *sort* option. If you do neither, you will get something that looks like modern art:

```
. use https://www.stata-press.com/data/r17/gnp96, clear  
  (1996 gross national product)  
. generate d = d.gnp96  
  (1 missing value generated)  
. generate u = runiform()  
. sort u                               (put in random order)  
. twoway area d date
```



Also see

[G-2] [graph twoway scatter](#) — Twoway scatterplots

[G-2] [graph twoway dot](#) — Twoway dot plots

[G-2] [graph twoway dropline](#) — Twoway dropped-line plots

[G-2] [graph twoway histogram](#) — Histogram plots

[G-2] [graph twoway spike](#) — Twoway spike plots

[G-2] [graph bar](#) — Bar charts