

graph twoway fpfit — Twoway fractional-polynomial prediction plots

[Description](#)
[Options](#)
[Quick start](#)
[Remarks and examples](#)
[Menu](#)
[Also see](#)
[Syntax](#)

Description

`twoway fpfit` calculates the prediction for *yvar* from estimation of a fractional polynomial of *xvar* and plots the resulting curve.

Quick start

Fractional-polynomial prediction plot of predicted *y* on *x* using `regress`

```
twoway fpfit y x
```

As above, but estimate a fractional polynomial of degree 4

```
twoway fpfit y x, estopts(degree(4))
```

Overlayed on a scatterplot of the observed values

```
twoway scatter y x || fpfit y x
```

As above, with lines for fractional polynomials of degrees 2 (default) and 4

```
twoway scatter y x || fpfit y x || fpfit y x, estopts(degree(4))
```

A separate graph area for each level of *catvar*

```
twoway scatter y x || fpfit y x, by(catvar)
```

As above, but with an additional graph area for all levels of *catvar* combined

```
twoway scatter y x || fpfit y x, by(catvar, total)
```

Overlay lines and data points for *catvar*==0 and *catvar*==1 in a single graph area

```
twoway scatter y x if catvar==0 || scatter y x if catvar==1 || ///
    fpfit y x if catvar==0 || fpfit y x if catvar==1
```

Specify `poisson` as the estimation command

```
twoway scatter y x || fpfit y x, estcmd(poisson)
```

Menu

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

Syntax

`twoway fpfit yvar xvar [if] [in] [weight] [, options]`

<i>options</i>	Description
<code>estcmd(<i>est_cmd</i>)</code>	estimation command; default is <code>regress</code>
<code>estopts(<i>est_opts</i>)</code>	specifies <i>est_opts</i> to estimate the fractional polynomial regression
<i>cline_options</i>	change look of predicted line
<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.

est_cmd may be `clogit`, `glm`, `intreg`, `logistic`, `logit`, `mlogit`, `nbreg`, `ologit`, `oprobit`, `poisson`, `probit`, `regress`, `rreg`, `stcox`, `stcrreg`, `streg`, or `xtgee`.

See [G-3] *cline_options*, [G-3] *axis_choice_options*, and [G-3] *twoway_options*.

Options `estcmd()` and `estopts()` are *unique*; see [G-4] **concept: repeated options**.

*aweight*s, *fweight*s, and *pweight*s are allowed. Weights, if specified, affect estimation but not how the weighted results are plotted. See [U] **11.1.6 weight**.

<i>est_opts</i>	Description
<code>degree(#)</code>	degree of fractional polynomial to fit; default is <code>degree(2)</code>
<code>noscaling</code>	suppress scaling of first independent variable
<code>noconstant</code>	suppress constant term
<code>powers(<i>numlist</i>)</code>	list of fractional polynomial powers from which models are chosen
<code>center(<i>cent_list</i>)</code>	specification of centering for the independent variables
<code>all</code>	include out-of-sample observations in generated variables
<code>log</code>	display iteration log
<code>compare</code>	compare models by degree
<i>display_options</i>	control column formats and line width
<i>other_est_opts</i>	other options allowed by <i>est_cmd</i>

cent_list is a comma-separated list with elements *varlist*:`{mean|#|no}`, except that the first element may optionally be of the form `{mean|#|no}` to specify the default for all variables.

Options

`estcmd(est_cmd)` specifies the estimation command to be used; `estcmd(regress)` is the default.

`estopts(est_opts)` specifies the options to estimate the fractional polynomial regression from which the curve will be predicted. Available *est_opts* are

`degree(#)` determines the degree of FP to be fit. The default is `degree(2)`, that is, a model with two power terms.

`noscaling` suppresses scaling of *xvar*₁ and its powers.

`noconstant` suppresses the regression constant if this is permitted by *est_cmd*.

`powers(numlist)` is the set of FP powers from which models are to be chosen. The default is `powers(-2,-1,-.5,0,.5,1,2,3)` (0 means log).

`center(cent_list)` defines the centering for the covariates *xvar*₁, *xvar*₂, ..., *xvarlist*. The default is `center(mean)`. A typical item in *cent_list* is *varlist*:`{mean|#|no}`. Items are separated by commas. The first item is special because *varlist*: is optional, and if omitted, the default is (re)set to the specified value (mean or # or no). For example, `center(no, age:mean)` sets the default to no and sets the centering for age to mean.

`all` includes out-of-sample observations when generating the best-fitting FP powers of *xvar*₁, *xvar*₂, etc. By default, the generated FP variables contain missing values outside the estimation sample.

`log` displays deviances and (for `regress`) residual standard deviations for each FP model fit.

`compare` reports a closed-test comparison between FP models.

display_options: `cformat(%fmt)`, `pformat(%fmt)`, `sformat(%fmt)`, and `nolstretch`; see [R] [estimation options](#).

other_est_opts are options appropriate to the *est_cmd*; see the documentation for that *est_cmd*. For example, for `stcox`, *other_est_opts* may include `efron` or some alternate method for handling tied failures.

cline_options specify how the prediction line is rendered; see [G-3] [cline_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

Remarks are presented under the following headings:

Typical use

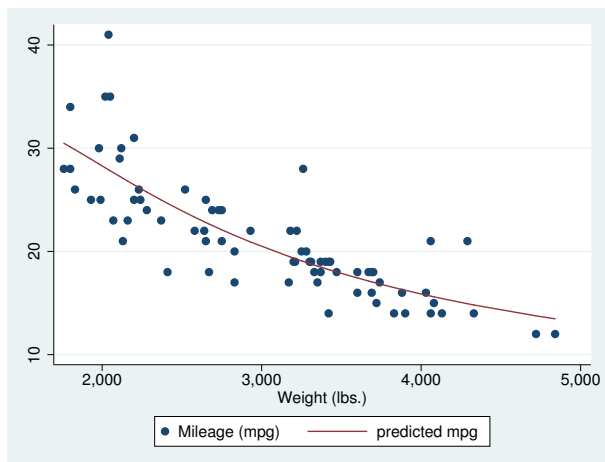
Cautions

Use with by()

Typical use

`twoway fpfit` is nearly always used in conjunction with other `twoway` plottypes, such as

```
. use http://www.stata-press.com/data/r14/auto  
(1978 Automobile Data)  
. scatter mpg weight || fpfit mpg weight
```



Cautions

Do not use `twoway fpfit` when specifying the *axis_scale_options* `yscale(log)` or `yscale(log)` to create log scales. Typing

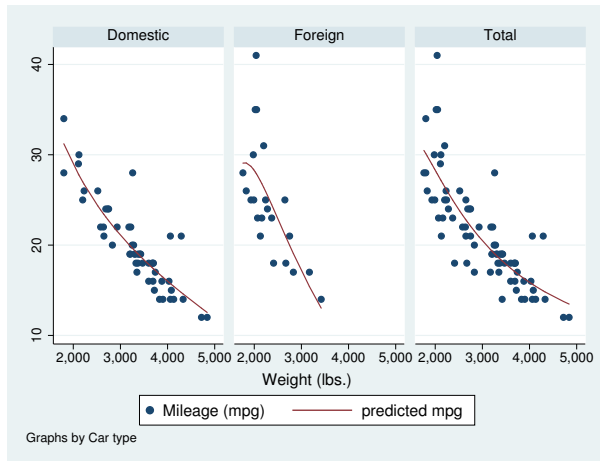
```
. scatter mpg weight, xscale(log) || fpfit mpg weight
```

will produce a curve that will be fit from a fractional polynomial regression of `mpg` on `weight` rather than `log(weight)`.

Use with by()

fpfit may be used with by() (as can all the twoway plot commands):

```
. scatter mpg weight || fpfit mpg weight ||, by(foreign, total row(1))
```



Also see

- [G-2] [graph twoway line](#) — Twoway line plots
- [G-2] [graph twoway fpfitci](#) — Twoway fractional-polynomial prediction plots with CIs
- [G-2] [graph twoway lfit](#) — Twoway linear prediction plots
- [G-2] [graph twoway qfit](#) — Twoway quadratic prediction plots
- [G-2] [graph twoway mband](#) — Twoway median-band plots
- [G-2] [graph twoway mspline](#) — Twoway median-spline plots