Description

`lowess` carries out a locally weighted regression of `yvar` on `xvar`, displays the graph, and optionally saves the smoothed variable.

Warning: `lowess` is computationally intensive and may therefore take a long time to run on a slow computer. Lowess calculations on 1,000 observations, for instance, require performing 1,000 regressions.

Quick start

Locally weighted regression of `y1` on `x`
```
lowess y1 x
```

As above, but with a bandwidth of 0.4
```
lowess y1 x, bwidth(.4)
```

With running-mean smoothing
```
lowess y1 x, mean
```

Without the tricube weighting function
```
lowess y1 x, noweight
```

Generate a new variable `v` containing the smoothed values of `y1`
```
lowess y1 x, generate(v)
```

Adjust the mean of the smoothed values to equal the mean of the unsmoothed values
```
lowess y1 x, adjust
```

Lowess smoothing of categorical variable `y2` on `x` in terms of the log of the odds ratio
```
lowess y2 x, logit
```

Menu

Statistics > Nonparametric analysis > Lowess smoothing
Syntax

lowess yvar xvar [if] [in] [ , options ]

<table>
<thead>
<tr>
<th>options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Main</strong></td>
<td></td>
</tr>
<tr>
<td>mean</td>
<td>running-mean smooth; default is running-line least squares</td>
</tr>
<tr>
<td>noweight</td>
<td>suppress weighted regressions; default is tricube weighting function</td>
</tr>
<tr>
<td>bwidth(#)</td>
<td>use # for the bandwidth; default is bwidth(0.8)</td>
</tr>
<tr>
<td>logit</td>
<td>transform dependent variable to logits</td>
</tr>
<tr>
<td>adjust</td>
<td>adjust smoothed mean to equal mean of dependent variable</td>
</tr>
<tr>
<td>nograph</td>
<td>suppress graph</td>
</tr>
<tr>
<td>generate(newvar)</td>
<td>create newvar containing smoothed values of yvar</td>
</tr>
<tr>
<td><strong>Plots</strong></td>
<td></td>
</tr>
<tr>
<td>marker_options</td>
<td>change look of markers (color, size, etc.)</td>
</tr>
<tr>
<td>marker_label_options</td>
<td>add marker labels; change look or position</td>
</tr>
<tr>
<td><strong>Smoothed line</strong></td>
<td></td>
</tr>
<tr>
<td>lineopts(cline_options)</td>
<td>affect rendition of the smoothed line</td>
</tr>
<tr>
<td><strong>Add plots</strong></td>
<td></td>
</tr>
<tr>
<td>addplot(plot)</td>
<td>add other plots to generated graph</td>
</tr>
<tr>
<td><strong>Y axis, X axis, Titles, Legend, Overall, By</strong></td>
<td>any of the options documented in [G-3] twoway_options</td>
</tr>
</tbody>
</table>

yvar and xvar may contain time-series operators; see [U] 11.4.4 Time-series varlists.

Options

**Main**

mean specifies running-mean smoothing; the default is running-line least-squares smoothing.

noweight prevents the use of Cleveland’s (1979) tricube weighting function; the default is to use the weighting function.

bwidth(#) specifies the bandwidth. Centered subsets of bwidth() × N observations are used for calculating smoothed values for each point in the data except for the end points, where smaller, uncentered subsets are used. The greater the bwidth(), the greater the smoothing. The default is 0.8.

logit transforms the smoothed yvar into logits. Predicted values less than 0.0001 or greater than 0.9999 are set to 1/N and 1 − 1/N, respectively, before taking logits.

adjust adjusts the mean of the smoothed yvar to equal the mean of yvar by multiplying by an appropriate factor. This option is useful when smoothing binary (0/1) data.

nograph suppresses displaying the graph.

generate(newvar) creates newvar containing the smoothed values of yvar.
Plot

marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.

marker_label_options specify if and how the markers are to be labeled; see [G-3] marker_label_options.

Smoothed line

lineopts(cline_options) affects the rendition of the lowess-smoothed line; see [G-3] cline_options.

Add plots

addplot(plot) provides a way to add other plots to the generated graph; see [G-3] addplot_option.

Y axis, X axis, Titles, Legend, Overall, By

twoway_options are any of the options documented in [G-3] twoway_options. These include options for titling the graph (see [G-3] title_options), options for saving the graph to disk (see [G-3] saving_option), and the by() option (see [G-3] by_option).

Remarks and examples

By default, lowess provides locally weighted scatterplot smoothing. The basic idea is to create a new variable (newvar) that, for each yvar \( y_i \), contains the corresponding smoothed value. The smoothed values are obtained by running a regression of yvar on xvar by using only the data \((x_i, y_i)\) and a few of the data near this point. In lowess, the regression is weighted so that the central point \((x_i, y_i)\) gets the highest weight and points that are farther away (based on the distance \(|x_j - x_i|\)) receive less weight. The estimated regression line is then used to predict the smoothed value \(\hat{y}_i\) for \(y_i\) only. The procedure is repeated to obtain the remaining smoothed values, which means that a separate weighted regression is performed for every point in the data.

Lowess is a desirable smoother because of its locality—it tends to follow the data. Polynomial smoothing methods, for instance, are global in that what happens on the extreme left of a scatterplot can affect the fitted values on the extreme right.

Example 1

The amount of smoothing is affected by bwidth(#). You are warned to experiment with different values. For instance,
Now compare that with

```
. lowess h1 depth, bwidth(.4)
```

In the first case, the default bandwidth of 0.8 is used, meaning that 80% of the data are used in smoothing each point. In the second case, we explicitly specified a bandwidth of 0.4. Smaller bandwidths follow the original data more closely.

Example 2

Two `lowess` options are especially useful with binary (0/1) data: `adjust` and `logit`. `adjust` adjusts the resulting curve (by multiplication) so that the mean of the smoothed values is equal to the mean of the unsmoothed values. `logit` specifies that the smoothed curve be in terms of the log of the odds ratio:
With binary data, if you do not use the `logit` option, it is a good idea to specify `graph’s jitter()` option; see [G-2] `graph twoway scatter`. Because the underlying data (whether the car was manufactured outside the United States here) take on only two values, raw data points are more likely to be on top of each other, thus making it impossible to tell how many points there are. `graph’s jitter()` option adds some noise to the data to shift the points around. This noise affects only the location of points on the graph, not the lowess curve.

When you specify the `logit` option, the display of the raw data is suppressed.
lowess can be used for more than just lowess smoothing. Lowess can be usefully thought of as a combination of two smoothing concepts: the use of predicted values from regression (rather than means) for imputing a smoothed value and the use of the tricube weighting function (rather than a constant weighting function). lowess allows you to combine these concepts freely. You can use line smoothing without weighting (specify noweight), mean smoothing with tricube weighting (specify mean), or mean smoothing without weighting (specify mean and noweight).

Methods and formulas

Let \( y_i \) and \( x_i \) be the two variables, and assume that the data are ordered so that \( x_i \leq x_{i+1} \) for \( i = 1, \ldots, N - 1 \). For each \( y_i \), a smoothed value \( y_i^s \) is calculated.

The subset used in calculating \( y_i^s \) is indices \( i_\pm = \max(1, i-k) \) through \( i_\pm = \min(i+k, N) \), where \( k = \lfloor (N \times \text{bwidth} - 0.5)/2 \rfloor \). The weights for each of the observations between \( j = i_-, \ldots, i_+ \) are either 1 (noweight) or the tricube (default),

\[
  w_j = \left( 1 - \left( \frac{|x_j - x_i|}{\Delta} \right)^3 \right)^3
\]

where \( \Delta = 1.0001 \max(x_{i+} - x_i, x_i - x_{i-}) \). The smoothed value \( y_i^s \) is then the (weighted) mean or the (weighted) regression prediction at \( x_i \).

Acknowledgment

lowess is a modified version of a command originally written by Patrick Royston of the MRC Clinical Trials Unit, London, and coauthor of the Stata Press book *Flexible Parametric Survival Analysis Using Stata: Beyond the Cox Model*.

References


Also see

[R] `lpoly` — Kernel-weighted local polynomial smoothing

[R] `smooth` — Robust nonlinear smoother

[D] `ipolate` — Linearly interpolate (extrapolate) values