cusum — Cusum plots and tests for binary variables

Description

cusum graphs the cumulative sum (cusum) of a binary (0/1) variable, yvar, against a (usually) continuous variable, xvar.

Quick start

Cusum statistics for binary variable y and graph of cumulative sum against values of x

cusum y x

Also generate cs to store the cumulative sum

cusum y x, generate(cs)

Set the seed first for reproducible results

set seed 87534690
cusum y x, generate(cs)

Cumulative sum of y against a variable containing fitted values yhat

cusum y x, yfit(yhat)

Menu

Statistics > Other > Quality control > Cusum plots and tests for binary variables
Syntax

```
cusum 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><code>generate(newvar)</code></td>
<td>save cumulative sum in <code>newvar</code></td>
</tr>
<tr>
<td><code>yfit(fitvar)</code></td>
<td>calculate cumulative sum against <code>fitvar</code></td>
</tr>
<tr>
<td>nograph</td>
<td>suppress the plot</td>
</tr>
<tr>
<td>nocalc</td>
<td>suppress cusum test statistics</td>
</tr>
<tr>
<td><strong>Cusum plot</strong></td>
<td></td>
</tr>
<tr>
<td><code>connect_options</code></td>
<td>affect the rendition of the plotted line</td>
</tr>
<tr>
<td><strong>Add plots</strong></td>
<td></td>
</tr>
<tr>
<td><code>addplot(plot)</code></td>
<td>add plots to the generated graph</td>
</tr>
<tr>
<td><strong>Y axis, X axis, Titles, Legend, Overall</strong></td>
<td>any options other than by() documented in [G-3] <code>twoway_options</code></td>
</tr>
</tbody>
</table>

**Options**

- `generate(newvar)` saves the cusum in `newvar`.
- `yfit(fitvar)` calculates a cusum against `fitvar`, that is, the running sums of the “residuals” `fitvar` minus `yvar`. Typically, `fitvar` is the predicted probability of a positive outcome obtained from a logistic regression analysis.
- `nograph` suppresses the plot.
- `nocalc` suppresses calculation of the cusum test statistics.
- `connect_options` affect the rendition of the plotted line; see [G-3] `connect_options`.
- `addplot(plot)` provides a way to add other plots to the generated graph. See [G-3] `addplot_option`.
- `twoway_options` are any of the options documented in [G-3] `twoway_options`, excluding by(). These include options for titling the graph (see [G-3] `title_options`) and for saving the graph to disk (see [G-3] `saving_option`).
Remarks and examples

The cusum is the running sum of the proportion of ones in the sample, a constant number, minus $yvar$, 

$$c_j = \sum_{k=1}^{j} f - yvar(k), \quad 1 \leq j \leq N$$

where $f = (\sum yvar)/N$ and $yvar(k)$ refers to the corresponding value of $yvar$ when $xvar$ is placed in ascending order: $xvar_{(k+1)} \geq xvar_k$. Tied values of $xvar$ are broken at random. If you want them broken the same way in two runs, you must set the random-number seed to the same value before giving the cusum command; see [R] set seed.

A U-shaped or inverted U-shaped cusum indicates, respectively, a negative or a positive trend of $yvar$ with $xvar$. A sinusoidal shape is evidence of a nonmonotonic (for example, quadratic) trend. cusum displays the maximum absolute cusum for monotonic and nonmonotonic trends of $yvar$ on $xvar$. These are nonparametric tests of departure from randomness of $yvar$ with respect to $xvar$. Approximate values for the tests are given.

Example 1

For the automobile dataset, auto.dta, we wish to investigate the relationship between foreign (0 = domestic, 1 = foreign) and car weight as follows:

```
. use https://www.stata-press.com/data/r16/auto
(1978 Automobile Data)
. cusum foreign weight
```

<table>
<thead>
<tr>
<th>Variable</th>
<th>Obs</th>
<th>Pr(1)</th>
<th>CusumL</th>
<th>zL</th>
<th>Pr&gt;zL</th>
<th>CusumQ</th>
<th>zQ</th>
<th>Pr&gt;zQ</th>
</tr>
</thead>
<tbody>
<tr>
<td>foreign</td>
<td>74</td>
<td>0.2973</td>
<td>10.30</td>
<td>3.963</td>
<td>0.000</td>
<td>2.92</td>
<td>0.064</td>
<td>0.475</td>
</tr>
</tbody>
</table>

The resulting plot, which is U-shaped, suggests a negative monotonic relationship. The trend is confirmed by a highly significant linear cusum statistic, labeled CusumL in the output above.

Some 29.73% of the cars are foreign (coded 1). The proportion of foreign cars diminishes with increasing weight. The domestic cars are crudely heavier than the foreign ones. We could have discovered that by typing table foreign, contents(mean weight), but such an approach does
not give the full picture of the relationship. The quadratic cusum (CusumQ) is not significant, so we do not suspect any tendency for the very heavy cars to be foreign rather than domestic. A slightly enhanced version of the plot shows the preponderance of domestic (coded 0) cars at the heavy end of the weight axis:

```stata
. label values foreign
. cusum foreign weight, s(none) recast(scatter) mlabel(foreign) mlabp(0)
```

<table>
<thead>
<tr>
<th>Variable</th>
<th>Obs</th>
<th>Pr(1)</th>
<th>CusumL</th>
<th>zL</th>
<th>Pr&gt;zL</th>
<th>CusumQ</th>
<th>zQ</th>
<th>Pr&gt;zQ</th>
</tr>
</thead>
<tbody>
<tr>
<td>foreign</td>
<td>74</td>
<td>0.2973</td>
<td>10.30</td>
<td>3.963</td>
<td>0.000</td>
<td>3.32</td>
<td>0.469</td>
<td>0.320</td>
</tr>
</tbody>
</table>

The example is, of course, artificial, because we would not really try to model the probability of a car being foreign given its weight.

---

**Stored results**

cusum stores the following in `r()`:

Scalars

- `r(N)` number of observations
- `r(prop1)` proportion of positive outcomes
- `r(cusuml)` cusum
- `r(zl)` test (linear)
- `r(P_zl)` p-value for test (linear)
- `r(cusumq)` quadratic cusum
- `r(zq)` test (quadratic)
- `r(P_zq)` p-value for test (quadratic)

**Acknowledgment**

cusum was 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.*
Reference


Also see

[R] **logistic** — Logistic regression, reporting odds ratios
[R] **logit** — Logistic regression, reporting coefficients
[R] **probit** — Probit regression