Syntax

```
proportion varlist [if] [in] [weight] [ , options ]
```

**options**

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
</tr>
</tbody>
</table>
| stdize(varname) | variable identifying strata for standardization  
| stdweight(varname) | weight variable for standardization  
| nodstdescale | do not rescale the standard weight variable  
| nolabel | suppress value labels from varlist  
| missing | treat missing values like other values  
| if/in/over | group over subpopulations defined by varlist; optionally, suppress group labels  
| SE/Cluster |
| vce(vcetype) | vcetype may be analytic, cluster clustvar, bootstrap, or jackknife  
| Reporting |
| level(#) | set confidence level; default is level(95)  
| citype(logit | method to compute limits of confidence intervals; default is citype(logit)  
| noheader | suppress table header  
| nolegend | suppress table legend  
| display_options | control column formats and line width  
| coeflegend | display legend instead of statistics  

bootstrap, jackknife, mi estimate, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands. If vce(bootstrap) and vce(jackknife) are not allowed with the mi estimate prefix; see [MI] mi estimate. Weights are not allowed with the bootstrap prefix; see [R] bootstrap. vce() and weights are not allowed with the svy prefix; see [SVY] svy. fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight. coeflegend does not appear in the dialog box. See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.
proportion produces estimates of proportions, along with standard errors, for the categories identified by the values in each variable of `varlist`.

Options

**Model**

- `stdize(varname)` specifies that the point estimates be adjusted by direct standardization across the strata identified by `varname`. This option requires the `stdweight()` option.
- `stdweight(varname)` specifies the weight variable associated with the standard strata identified in the `stdize()` option. The standardization weights must be constant within the standard strata.
- `nostdrescale` prevents the standardization weights from being rescaled within the `over()` groups. This option requires `stdize()` but is ignored if the `over()` option is not specified.
- `nolabel` specifies that value labels attached to the variables in `varlist` be ignored.
- `missing` specifies that missing values in `varlist` be treated as valid categories, rather than omitted from the analysis (the default).

**if/in/over**

- `over(varlist [, nolabel])` specifies that estimates be computed for multiple subpopulations, which are identified by the different values of the variables in `varlist`.

When this option is supplied with one variable name, such as `over(varname)`, the value labels of `varname` are used to identify the subpopulations. If `varname` does not have labeled values (or there are unlabeled values), the values themselves are used, provided that they are nonnegative integers. Noninteger values, negative values, and labels that are not valid Stata names are substituted with a default identifier.

When `over()` is supplied with multiple variable names, each subpopulation is assigned a unique default identifier.
- `nolabel` requests that value labels attached to the variables identifying the subpopulations be ignored.

**SE/Cluster**

- `vce(vcetype)` specifies the type of standard error reported, which includes types that are derived from asymptotic theory (`analytic`), that allow for intragroup correlation (`cluster clustvar`), and that use bootstrap or jackknife methods (`bootstrap`, `jackknife`); see [R] `vce_option`.

- `vce(analytic)`, the default, uses the analytically derived variance estimator associated with the sample proportion.

**Reporting**

- `level(#)`; see [R] `estimation options`. 
proportion — Estimate proportions

proportion (logit|normal) specifies how to compute the limits of confidence intervals.

proportion (logit), the default, uses the logit transformation to compute the limits of confidence intervals.

proportion (normal) uses the normal approximation to compute the limits of confidence intervals.

noheader prevents the table header from being displayed. This option implies nolegend.

nolegend prevents the table legend identifying the subpopulations from being displayed.

display_options: cformat(%fmt) and nolstretch; see [R] estimation options.

The following option is available with proportion but is not shown in the dialog box: coeflegend; see [R] estimation options.

Remarks and examples

Example 1

We can estimate the proportion of each repair rating in auto2.dta:

```
. use http://www.stata-press.com/data/r13/auto2
(1978 Automobile Data)
. proportion rep78
```

```
Proportion estimation
Number of obs = 69

<table>
<thead>
<tr>
<th>Proportion</th>
<th>Std. Err.</th>
<th>[95% Conf. Interval]</th>
</tr>
</thead>
<tbody>
<tr>
<td>rep78</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Poor</td>
<td>.0289855</td>
<td>.0203446</td>
</tr>
<tr>
<td>Fair</td>
<td>.115942</td>
<td>.0388245</td>
</tr>
<tr>
<td>Average</td>
<td>.4347826</td>
<td>.0601159</td>
</tr>
<tr>
<td>Good</td>
<td>.2608696</td>
<td>.0532498</td>
</tr>
<tr>
<td>Excellent</td>
<td>.1594203</td>
<td>.0443922</td>
</tr>
</tbody>
</table>
```

Here we use the missing option to include missing values as a category of rep78:

```
. proportion rep78, missing
```

```
Proportion estimation
Number of obs = 74

<table>
<thead>
<tr>
<th>_prop_6: rep78 = .</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proportion</td>
</tr>
<tr>
<td>rep78</td>
</tr>
<tr>
<td>Poor</td>
</tr>
<tr>
<td>Fair</td>
</tr>
<tr>
<td>Average</td>
</tr>
<tr>
<td>Good</td>
</tr>
<tr>
<td>Excellent</td>
</tr>
<tr>
<td>_prop_6</td>
</tr>
</tbody>
</table>
```

stata.com
Example 2

We can also estimate proportions over groups:

```
proportion rep78, over(foreign)
```

<table>
<thead>
<tr>
<th>Over</th>
<th>Proportion</th>
<th>Std. Err.</th>
<th>[95% Conf. Interval]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Poor</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Domestic</td>
<td>0.0416667</td>
<td>0.0291477</td>
<td>0.0100299 0.1572433</td>
</tr>
<tr>
<td>Foreign</td>
<td>(no observations)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fair</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Domestic</td>
<td>0.1666667</td>
<td>0.0543607</td>
<td>0.0839032 0.3039797</td>
</tr>
<tr>
<td>Foreign</td>
<td>(no observations)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Domestic</td>
<td>0.5625</td>
<td>0.0723605</td>
<td>0.4169211 0.6980553</td>
</tr>
<tr>
<td>Foreign</td>
<td>0.1428571</td>
<td>0.0782461</td>
<td>0.0444941 0.3736393</td>
</tr>
<tr>
<td>Good</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Domestic</td>
<td>0.1875</td>
<td>0.0569329</td>
<td>0.0986718 0.3272601</td>
</tr>
<tr>
<td>Foreign</td>
<td>0.4285714</td>
<td>0.1106567</td>
<td>0.2333786 0.6488451</td>
</tr>
<tr>
<td>Excellent</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Domestic</td>
<td>0.0416667</td>
<td>0.0291477</td>
<td>0.0100299 0.1572433</td>
</tr>
<tr>
<td>Foreign</td>
<td>0.4285714</td>
<td>0.1106567</td>
<td>0.2333786 0.6488451</td>
</tr>
</tbody>
</table>
Stored results

proportion stores the following in e():

Scalars

- e(N) number of observations
- e(N_over) number of subpopulations
- e(N_stdize) number of standard strata
- e(N_clust) number of clusters
- e(k_eq) number of equations in e(b)
- e(df_r) sample degrees of freedom
- e(rank) rank of e(V)

Macros

- e(cmd) proportion
- e(cmdline) command as typed
- e(varlist) varlist
- e(stdize) varname from stdize()
- e(stdweight) varname from stdweight()
- e(weight) weight type
- e(weight_expression) weight expression
- e(title) title in estimation output
- e(cluster) name of cluster variable
- e(over) varlist from over()
- e(over_labels) labels from over() variables
- e(over_namelist) names from e(over_labels)
- e(namelist) proportion identifiers
- e(label_id) labels from #th variable in varlist
- e(vce) vcetype specified in vce()
- e(vcetype) title used to label Std. Err.
- e(properties) b V
- e(estat_cmd) program used to implement estat
- e(marginsnotok) predictions disallowed by margins

Matrices

- e(b) vector of proportion estimates
- e(V) (co)variance estimates
- e(N) vector of numbers of nonmissing observations
- e(N_stdsum) number of nonmissing observations within the standard strata
- e(p_stdize) standardizing proportions
- e(error) error code corresponding to e(b)

Functions

- e(sample) marks estimation sample

Methods and formulas

Proportions are means of indicator variables; see [R] mean.

Confidence intervals

Confidence intervals for proportions are calculated using a logit transform so that the endpoints
lie between 0 and 1. Let \( \hat{p} \) be an estimated proportion and \( \hat{s} \) be an estimate of its standard error. Let

\[
f(\hat{p}) = \ln \left( \frac{\hat{p}}{1 - \hat{p}} \right)
\]

be the logit transform of the proportion. In this metric, an estimate of the standard error is

\[
\frac{\hat{s}}{\hat{p}(1 - \hat{p})}
\]
Thus a 100(1 − α)% confidence interval in this metric is

\[ \ln \left( \frac{\hat{p}}{1 - \hat{p}} \right) \pm \frac{t_{1-\alpha/2,\nu} \hat{s}}{\hat{p}(1 - \hat{p})} \]

where \( t_{1-\alpha/2,\nu} \) is the \( (1 - \alpha/2) \)th quantile of Student’s \( t \) distribution with \( \nu \) degrees of freedom. The endpoints of this confidence interval are transformed back to the proportion metric by using the inverse of the logit transform

\[ f^{-1}(y) = \frac{e^y}{1 + e^y} \]

Hence, the displayed confidence intervals for proportions are

\[ f^{-1} \left\{ \ln \left( \frac{\hat{p}}{1 - \hat{p}} \right) \pm \frac{t_{1-\alpha/2,\nu} \hat{s}}{\hat{p}(1 - \hat{p})} \right\} \]

References


Also see

[R] proportion postestimation — Postestimation tools for proportion

[R] mean — Estimate means

[R] ratio — Estimate ratios

[R] total — Estimate totals

[MI] estimation — Estimation commands for use with mi estimate

[SVY] direct standardization — Direct standardization of means, proportions, and ratios

[SVY] poststratification — Poststratification for survey data

[SVY] subpopulation estimation — Subpopulation estimation for survey data

[SVY] svy estimation — Estimation commands for survey data

[SVY] variance estimation — Variance estimation for survey data

[U] 20 Estimation and postestimation commands