help for simex  
(SJ3-4: st0049, st0050, st0051)

Simulation Extrapolation

    simex depvar [label] [label] ... 
     (label) [label] ... 
     [indepvars] [label] ... 
     (label) [label] ... 
     [if exp] [in range] [bstrap brep(#) 
     ltolerance(#) iterate(#) family(familyname) link(linkname) 
     suinit(matrixname) theta(matrixname) linear|rational nlep(#) 
     nleps(#) [cmd:median srep(#) btrim(#) saving(filename) replace 
     seed(#) scale(x2|dev|#) message(#)]

where familyname is one of

    gaussian | igaussian | binomial [varnameN|#N] | poisson | nbinomial [#k] | gamma

and linkname is one of

    identity | log | logit | probit | cloglog | opower # | power # | nbinomial | loglog | logc

and label:varlist describes a variable measured with error. The label is
for the unknown measurement error covariate (label cannot be the same as
an existing variable in the data set). varlist is a list of variables
with the replicate measurements for the unknown label covariate (see
comments for restrictions).

by ... : may be used with simex; see help by.

simex No predict is implemented.

Description

simex fits generalized linear models for measurement error data using
IRLS (maximum quasi-likelihood) and is similar in syntax to rcal. The
command is implemented by Stata's plug-in mechanism. simex allows one or
more (see comments) covariates measured with errors and uses simulation
extrapolation to estimate the missing covariates. It will allow replicate
data or a user specified measurement error covariance matrix. It supports
a very fast internal bootstrap (different from the regular Stata bootstrap
command).

Options

bstrap(#) specifies that the bootstrap estimate of variance be used.

brep(#) specifies the number of bootstrap samples to consider in forming
the bootstrap estimate of variance. The default is brep(199).

ltolerance(#) specifies the convergence criterion for the change in
deviance between iterations; ltolerance(1e-6) is the default.

iterate(#) specifies the maximum number of iterations allowed in fitting
the model; iterate(100) is the default. You should seldom need to
specify iterate()!

family(familyname) specifies the distribution of depvar; family(gaussian)
is the default.
link(linkname) specifies the link function; the default is the canonical link for the family() specified.

suunit(matrixname) Specify the measurement error covariance matrix. This is calculated from the replications in the measurement error variables if it is not specified.

theta(matrixname) The thetas we will use for our simex. The default is theta=(0, .5, 1, 1.5, 2)

linear|rational The default extrapolation is quadratic regression. Choose linear to use linear regression extrapolation or (cm: rational) to use the rational extrapolant (see nlrep, nleps and comments).

nlrep(#) When using the rational extrapolant, the maximum number of iterations the optimizer will use.

nleps(#) When using the rational extrapolant, the convergence criteria (tolerance) we use in our optimizer.

median Use the median instead of the default mean of the simulated estimators (see srep).

srep(#) Number of replications (simulations) for each theta.

btrim(#) Percent bootstrap trimming. The default is btrim(.02).

saving(filename) Save the bootstrap results to the specified file.

replace Replace the existing 'bootstrap results' file if it exists.

seed(#) specify the seed for the random number generator used for simex. This enables for identical simex runs. This option is generally not specified.

scale(x2|dev|#) overrides the default scale parameter. By default, scale(1) is assumed for discrete distributions (binomial, Poisson, negative binomial) and scale(x2) for continuous distributions (Gaussian, gamma, inverse Gaussian).

scale(x2) specifies the scale parameter be set to the Pearson chi-squared (or generalized chi-squared) statistic divided by the residual degrees of freedom.

scale(dev) sets the scale parameter to the deviance divided by the residual degrees of freedom. This provides an alternative to scale(x2) for continuous distributions and over- or under-dispersed discrete distributions.

scale(#) sets the scale parameter to #.

message(#) The message or debug level from the plug-in module. The default is message(2)).

Special comments on multiple measurement error covariates

The number of replications for a covariate measured with error can vary across observations. When two or more measurement error covariates exist, they must all have the same number of replications across observations.

Special comments on standard errors
By default simex will not calculate standard errors. The only supported standard errors are via the bootstrap. Use the 'bstrap' and 'brep' options, but note that this calculation can take considerable time. An estimated time to completion is printed if the bootstrap will require more than 30 seconds.

**Special comments on the rational extrapolant**

The rational extrapolant requires the fitting of a non-linear curve on estimated coefficients using very few data points (the number of thetas). This can encounter numerical difficulties and produce and error message. We suggest the default quadratic extrapolant.

**Plotting the effect of measurement error on parameter estimates**

Use the simexplo to view the effect of measurement error on parameter estimates. simexploit will plot the effect of measurement error on parameter estimate. It gives a visual representation on how the parameter estimates are derived. The line shows the extrapolation back to \(-1\). Note that if the rational extrapolant was used no extrapolant line is drawn. See the examples below.

**Remarks**

The allowed link functions are

<table>
<thead>
<tr>
<th>Link function</th>
<th>glm option</th>
</tr>
</thead>
<tbody>
<tr>
<td>identity</td>
<td>link(identity)</td>
</tr>
<tr>
<td>log</td>
<td>link(log)</td>
</tr>
<tr>
<td>logit</td>
<td>link(logit)</td>
</tr>
<tr>
<td>probit</td>
<td>link(probit)</td>
</tr>
<tr>
<td>complementary log-log</td>
<td>link(cloglog)</td>
</tr>
<tr>
<td>odds power</td>
<td>link(opower #)</td>
</tr>
<tr>
<td>power</td>
<td>link(power #)</td>
</tr>
<tr>
<td>negative binomial</td>
<td>link(nbinomial)</td>
</tr>
<tr>
<td>log-log</td>
<td>link(loglog)</td>
</tr>
<tr>
<td>log-compliment</td>
<td>link(logc)</td>
</tr>
</tbody>
</table>

The allowed distribution families are

<table>
<thead>
<tr>
<th>Family</th>
<th>glm option</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gaussian(normal)</td>
<td>family(gaussian)</td>
</tr>
<tr>
<td>Inverse Gaussian</td>
<td>family(igaussian)</td>
</tr>
<tr>
<td>Bernoulli/binomial</td>
<td>family(binomial)</td>
</tr>
<tr>
<td>Poisson</td>
<td>family(poisson)</td>
</tr>
<tr>
<td>Negative binomial</td>
<td>family(nbinomial)</td>
</tr>
<tr>
<td>Gamma</td>
<td>family(gamma)</td>
</tr>
</tbody>
</table>

Reasonable combinations of `family()` and `link()` are

<table>
<thead>
<tr>
<th></th>
<th>id</th>
<th>log</th>
<th>logit</th>
<th>probit</th>
<th>clog</th>
<th>pow</th>
<th>opower</th>
<th>nbinomial</th>
<th>loglog</th>
<th>logc</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gaussian</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>inv. Gau.</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>binomial</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>Poisson</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>neg. bin.</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>gamma</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
</tbody>
</table>
Note: Nonstandard combinations other than those marked out above are allowed, but the user is responsible for seeing that the data fit the combination and for the interpretation of the results.

If you specify `family()` but not `link()`, you obtain the canonical link for the family:

<table>
<thead>
<tr>
<th>family</th>
<th>default link()</th>
</tr>
</thead>
<tbody>
<tr>
<td>family(gaussian)</td>
<td>link(identity)</td>
</tr>
<tr>
<td>family(igaussian)</td>
<td>link(power -2)</td>
</tr>
<tr>
<td>family(binomial)</td>
<td>link(logit)</td>
</tr>
<tr>
<td>family(poisson)</td>
<td>link(log)</td>
</tr>
<tr>
<td>family(nbinomial)</td>
<td>link(log)</td>
</tr>
<tr>
<td>family(gamma)</td>
<td>link(power -1)</td>
</tr>
</tbody>
</table>

Examples

```
.* generate some data
.set obs 1000
.gen x1 = uniform()
.gen x2 = uniform()
.gen x3 = uniform()
.gen err = invnorm(uniform())
.gen y = 1+2*x1+3*x2+4*x3+err

.* estimate with x3 known
.qvf y x1 x2 x3, bstrap

.* simulate measurement error covariate
.gen a1 = x3 + .3*invnorm(uniform())
.gen a2 = x3 + .3*invnorm(uniform())

.* estimate x1, x2 & w3 using simex & plot the extrapolation
.simex (y=x1 x2) (w3: a1 a2), bstrap
.simexplot w3

. eret list

.mat theta=(0,.5,1,1.5,2,2.5,3,3.5)
.simex (y=x1 x2) (w3: a1 a2), bstrap theta(theta) median
.simexplot w3

See rcal for further examples of options.
```

**Also see**

Online: help for `qvf, simex, simexplot`