sem and gsem option reliability( ) — Fraction of variance not due to measurement error

Description Syntax Option Remarks and examples Also see

Description

Option `reliability()` allows you to specify the fraction of variance not due to measurement error for measurement variables.

Syntax

{`sem`|`gsem`} ... [, ... `reliability(`*varname* # [*varname* # [...]])]

where *varname* is the name of an observed endogenous variable and # is the fraction or percentage of variance not due to measurement error:

```
. {sem|gsem} ..., ... reliability(x1 .8  x2 .9)
. {sem|gsem} ..., ... reliability(x1 80%  x2 90%)
```

Option

`reliability(`*varname* # [...]) specifies the reliability for variable *varname*. Reliability is bounded by 0 and 1 and is equal to

$$1 - \frac{\text{noise variance}}{\text{total variance}}$$

The reliability is assumed to be 1 when not specified.

Remarks and examples

See [SEM] **Example 24**.

Remarks are presented under the following headings:

Background
Dealing with measurement error of exogenous variables
Dealing with measurement error of endogenous variables
What can go wrong

Background

Option `reliability()` may be used with `sem` and may be used with `gsem` for Gaussian response variables with the identity link but only in the absence of censoring. We illustrate using `sem`, but we could just as well have used `gsem`.

Variables measured with error have attenuated path coefficients. If we had the model

```
. sem (y<-x)
```

and x were measured with error, then the estimated path coefficient would be biased toward 0. The usual solution to such measurement problems is to find multiple measurements and develop a latent variable from them:

```
. sem (x1 x2 x3<-X) (y<-X)
```

Another solution is available if we know the reliability of x. In that case, we can fit the model

```
. sem (x<-X) (y<-X), reliability(x .9)
```

The two solutions can even be combined:

```
. sem (x1 x2 x3<-X) (y<-X), reliability(x1 .9  x2 .8  x3 .9)
```

Even if you do not know the reliability, you can experiment using different but reasonable values for the reliability and thus determine the sensitivity of your estimation results to the measurement problem.

Dealing with measurement error of exogenous variables

Measurement error is most important when it occurs in exogenous variables, yet the `reliability()` option deals with measurement error of endogenous variables only. By creation of a latent variable, `reliability()` can deal with the measurement error of exogenous variables.

To fit the model `(y<-x)` where x is measured with error, you must introduce a latent variable corresponding to x measured without error. That is, the model `(y<-x)` can be converted into the model `(x<-X)` and `(y<-X)`:

$$x = \alpha_0 + \beta_0 X + e.x$$

$$y = \alpha_1 + \beta_1 X + e.y$$

To fit this model, you type

```
. sem (x<-X) (y<-X), reliability(x .9)
```

sem will introduce a normalization constraint, namely, that the path coefficient β_0 for `x<-X` is 1, but that is of no importance. What is important is that the estimate of that path coefficient β_1 of `y<-X` is the coefficient that would be obtained from `y<-x` were x measured without error.

In the above, we specified the measurement part of the model first. Be sure to do that. You might think you could equally well reverse the two terms so that, rather than writing

`(x<-X) (y<-X)` *(correct)*

you could write

`(y<-X) (x<-X)` *(incorrect)*

But you cannot do that unless you write

`(y<-X) (x<-X@1)` *(correct)*

because otherwise results are as if you typed

```
(y<-X@1) (x<-X)        (incorrect)
```

All of that is because `sem` places its normalization constraint from the latent variable to the first observed endogenous variable. There is no real error if the terms are interchanged except that you will be surprised by the coefficient of 1 for `y<-X` and (the reciprocal of) the coefficient of interest will be on `x<-X`.

See *How sem (gsem) solves the problem for you* in [SEM] **Intro 4** and see *Default normalization constraints* in [SEM] **sem**.

Dealing with measurement error of endogenous variables

When a variable would already be endogenous before you add the `reliability()` option, it really makes little difference whether you add the `reliability()` option. That is because endogenous variables are assumed to contain error, and if some of that error is measurement error, it is still just an error. Coefficients will be unchanged by the inclusion of the `reliability()` option.

Some variances and covariances will change, but the changes are offsetting in the calculation of path coefficients.

What will change are the standardized coefficients should you ask to see them. That is because the variances are changed.

What can go wrong

Consider a model of `y` on `x`. Say we fit the model with linear regression. If the R^2 of the fit is 0.6, then we know the reliability must be greater than 0.6. R^2 measures the fraction of variance of `y` that is explained by `x`, and the reliability of `x` measures the fraction of the variance of `x` that is not due to measurement error. Measurement error is assumed to be pure noise. It is just not possible that we could explain 0.6 of the variance of `y` by using a variable with reliability of, say, 0.5.

Well, in fact, it is because there is always a chance that the pure noise will correlate with `y`, too. Asymptotically, that probability vanishes, but in finite—especially small—samples, it could happen. Even so, the calculation of the corrected SEM estimates blows up.

If you have convergence problems, you need to check for this. Specify a larger value for the reliability. The problem is you cannot specify a value of 1, and large values such as 0.99999 can lead to a lack of identification. In most cases, you will be able to find a value in between, but the only way to be sure is to remove the `reliability()` option and, if necessary, simplify your model by removing any intermediary latent variables you had to add because of reliability.

If your model converges without reliability, then your measure of reliability is too low. At this point, we have little useful advice for you. Check whether you have the right value, of course. If you do, then there are two possibilities: either the experts who provided that estimate are wrong or you got unlucky in that the measurement error did just happen to correlate with the rest of your data. You will need to evaluate the chances of that for yourself. In any case, you can experiment with higher values of the reliability and at least provide an idea of the sensitivity of your estimates to differing assumptions.

Also see

[SEM] **sem** — Structural equation model estimation command

[SEM] **gsem** — Generalized structural equation model estimation command

[SEM] **gsem model description options** — Model description options

[SEM] **sem model description options** — Model description options

[SEM] **Example 24** — Reliability

Stata, Stata Press, Mata, NetCourse, and NetCourseNow are registered trademarks of StataCorp LLC. Stata and Stata Press are registered trademarks with the World Intellectual Property Organization of the United Nations. StataNow is a trademark of StataCorp LLC. Other brand and product names are registered trademarks or trademarks of their respective companies. Copyright © 1985–2025 StataCorp LLC, College Station, TX, USA. All rights reserved.

For suggested citations, see the FAQ on citing Stata documentation.