Title

stata.com

Intro 11 — Fitting models with summary statistics data (sem only)

Description Remarks and examples Reference Also see

Description

In textbooks and research papers, the data used are often printed in summary statistic form. These summary statistics include means, standard deviations or variances, and correlations or covariances. These summary statistics can be used in place of the underlying raw data to fit models with `sem`.

Summary statistics data (SSD) are convenient for publication because of their terseness. By not revealing individual responses, they do not violate participant confidentiality, which is sometimes important.

Support for SSD is provided by `sem` but not by `gsem`.

Remarks and examples

stata.com

Remarks are presented under the following headings:

Background

The structural equation modeling estimator is a function of the first and second moments—the means and covariances—of the data. Thus it is possible to obtain estimates of the parameters of an SEM by using means and covariances. One does not need the original dataset.

In terms of `sem`, one can create a dataset containing these summary statistics and then use that dataset to obtain fitted models. The `sem` command is used just as one would use it with the original, raw data.

How to use sem with SSD

To use `sem` with SSD,

1. Enter the summary statistics by using the `ssd` command. How you do that is the topic of an upcoming section.
2. Save the data just as you would any dataset, namely, with the `save` command.
3. Use the `sem` command just as you would ordinarily. You `use` the SSD if they are not already in memory, and no special syntax or options are required by `sem`, except

a. Do not use `sem`'s `if` *exp* or `in` *range* qualifiers. You do not have the raw data in memory and so you cannot select subsets of the data.

b. If you have entered summary statistics for groups of observations (for example, males and, separately, females), use `sem`'s `select()` option if you want to fit the model with a subset of the groups. That is, where you would ordinarily type

```
. sem ... if sex==1, ...
```

you instead type

```
. sem ..., ... select(1)
```

Where you would ordinarily type

```
. sem ... if region==1 | region==3, ...
```

you instead type

```
. sem ..., ... select(1 3)
```

See [SEM] **Example 3**.

What you cannot do with SSD

With SSD in memory,

1. You cannot obtain Satorra–Bentler standard errors and the Satorra–Bentler scaled χ^2 test, which you would normally do by specifying `sem` option `vce(sbentler)`.
2. You cannot obtain robust standard errors, which you would normally do by specifying `sem` option `vce(robust)`.
3. You cannot obtain clustered standard errors, which you would normally do by specifying `sem` option `vce(cluster` *clustvar*`)`.
4. You cannot obtain survey-adjusted results, which you would normally do by specifying the `svy:` prefix in front of the `sem` command.
5. You cannot obtain bootstrap or jackknife standard errors, which you would normally do by specifying `sem` option `vce(bootstrap)` or `vce(jackknife)`.
6. You cannot obtain VCE estimates from the observation-level outer product of the gradients, which you would normally do by specifying `vce(opg)`.
7. You cannot use weights, which you would normally do by specifying, for instance, `[fw=`*varname*`]`.
8. You cannot restrict the estimation sample with `if` *exp* or `in` *range*.
9. You cannot fit the model by using maximum likelihood with missing values or by using the asymptotic distribution free method, which you would normally do by specifying `method(mlmv)` or `method(adf)`.

Entering SSD

Entering SSD is easy. You need to see an example of how easy it is before continuing: see [SEM] **Example 2**.

What follows is an outline of the procedure. Let us begin with the data you need to have. You have

1. The names of the variables. We will just call them `x1`, `x2`, and `x3`.
2. The number of observations, say, 74.
3. The correlations, say,

```
 1
-0.8072    1
 0.3934   -0.5928    1
```

   or you may have the covariances,

```
33.4722
-3.6294    0.6043
 1.0374   -0.2120    0.2118
```

4. The variances: 33.4722, 0.6043, and 0.2118.

   Or the standard deviations: 5.6855, 0.7774, and 0.4602.

   Or neither.

   If you have the covariances in step 3, you in fact have the variances—they are just the diagonal elements of the covariance matrix—but the software will not make you enter the values twice.

5. The means: 21.2973, 3.0195, and 0.2973.

   Or not.

With that information at hand, do the following:

1. Start with no data in memory:

```
. clear all
```

2. Initialize the SSD by stating the names of the variables:

```
. ssd init x1 x2 x3
```

The remaining steps can be done in any order.

3. Set the number of observations:

```
. ssd set obs 74
```

4. Set the covariances:

```
. ssd set cov 33.4722 \ -3.6294 .6043 \ 1.0374 -.2120 .2118
```

   Or the correlations:

```
. ssd set cor  1 \ -.8072 1 \ .3934 -.5928 1
```

5. If you set covariances in step 4, skip to step 6. Otherwise, if you have them, set the variances:

```
. ssd set var 33.4722 .6043 .2118
```

   Or set the standard deviations:

```
. ssd set sd  5.6855 .7774 .4602
```

6. Set the means if you have them:

```
. ssd set means 21.2973 3.0195 .2973
```

7. If at any point you become confused as to what you have set and what remains to be set, type

```
. ssd status
```

8. If you want to review what you have set, type

```
. ssd list
```

9. If you make a mistake, you can repeat any `ssd set` command by adding the `replace` option to the end. For instance, you could reenter the means by typing

```
. ssd set means 21.2973 3.0195 .2973, replace
```

10. Save the dataset just as you would with any dataset:

```
. save mydata
```

You are now ready to use `sem` with the SSD. With the SSD in memory, you issue the `sem` command just as you would if you had the raw data:

```
. sem ...
```

Entering SSD for multiple groups

You can enter summary statistics for groups of the data. Perhaps you have summary statistics not for the data as a whole, but for males and for females, or for the young, for the middle-aged, and for the old.

Let's pretend you have the following data:

The young:			
observations:	74		
correlations:	1		
	−0.8072	1	
	0.3934	−0.5928	1
standard deviations:	5.6855	0.7774	0.4602
means:	21.2973	3.0195	0.2973
The middle-aged:			
observations:	141		
correlations:	1		
	−0.5721	1	
	0.3843	−0.4848	1
standard deviations:	4.9112	0.7010	0.5420
means:	38.1512	5.2210	0.2282
The old:			
observations:	36		
correlations:	1		
	−0.8222	1	
	0.3712	−0.3113	1
standard deviations:	6.7827	0.7221	0.4305
means:	58.7171	2.1511	0.1623

The commands for entering these summary statistics are

```
. ssd init x1 x2 x3
. ssd set obs   74
. ssd set cor    1    \  -.8072  1    \ .3934 -.5928 1
. ssd set sd     5.6855   .7774  .4602
. ssd set means 21.2973  3.0195  .2973
. ssd addgroup agecategory
. ssd set obs  141
. ssd set cor    1    \  -.5721  1    \ .3843 -.4848 1
. ssd set sd     4.9112   .7010  .5420
. ssd set means 38.1512  5.2210  .2282
. ssd addgroup
. ssd set obs   36
. ssd set cor    1    \  -.8222  1    \ .3712 -.3113 1
. ssd set sd     6.7827   .7221   .4305
. ssd set means 58.7171  2.1511   .1623
. save mygroupdata
```

The general procedure is as follows:

1. Enter the summary statistics for the first group just as outlined in the previous section.
2. Next add a group by typing

   . `ssd addgroup` *newgroupvar*

   In that one command, you are telling `ssd` that the summary statistics you entered in step 1 were for a group you are now calling *newgroupvar*, and in particular they were for *newgroupvar* = 1. You are also telling `ssd` that you now want to enter the summary statistics for the next group, namely, *newgroupvar* = 2.
3. Enter the summary statistics for the second group in the same way you entered them for the first group, just as outlined in the previous section.
4. If you have a third group, add it by typing

   . `ssd addgroup`

   In this case, you are telling `ssd` only one thing: that you now want to enter data for the next group, namely, *newgroupvar* = 3.
5. Enter the summary statistics for the third group in the same way you entered them for the second group, and just as outlined in the previous section.
6. If you want to add more groups, continue in the same way. Declare the next group of data by typing

   . `ssd addgroup`

   and then enter the data by using the `ssd set` command.
7. If you mistakenly add a group and wish to rescind that, type

   . `ssd unaddgroup`

8. If you wish to go back and modify the values entered for a previous group, put the group number between `ssd set` and what is being set—for instance, type `ssd set 2 observations` . . . —and specify the `replace` option. For instance, to reenter the correlations for group 1, type

   . `ssd set 1 correlations` *values*`, replace`

What happens when you do not set all the summary statistics

You are required to set the number of observations and to set the covariances or the correlations. Setting the variances (standard deviations) and setting the means are optional.

1. If you set correlations only, then

   a. Means are assumed to be 0.

   b. Standard deviations are assumed to be 1.

   c. You will not be able to pool across groups if you have group data.

   As a result of (a) and (b), the parameters `sem` estimates will be standardized even when you do not specify `sem`'s `standardized` reporting option. Estimated means and intercepts will be 0.

   Concerning (c), we need to explain. This concerns group data. If you type

   ```
   . sem ...
   ```

   then `sem` fits a model with all the data. `sem` does that whether you have raw data or SSD in memory. If you have SSD with groups—say, males and females or age groups 1, 2, and 3—`sem` combines the summary statistics to obtain the summary statistics for the overall data. It is only possible to do this when covariances and means are known for each group. If you set correlations without variances or standard deviations and without means, the necessary statistics are not known and the groups cannot be combined. Thus if you type

   ```
   . sem ...
   ```

   you will get an error message. You can still estimate using `sem`; you just have to specify on which group you wish to run `sem`, and you do that with the `select()` option:

   ```
   . sem ..., select(#)
   ```

2. If you set correlations and means,

   a. Standard deviations are assumed to be 1.

   b. You will not be able to pool across groups if you have group data.

   This situation is nearly identical to situation 1. The only difference is that estimated means and intercepts will be nonzero.

3. If you set correlations and standard deviations or variances, or if you set covariances only,

   a. Means are assumed to be 0.

   b. You will not be able to pool across groups if you have group data.

   This situation is a little better than situation 1. Estimated intercepts will be 0, but the remaining estimated coefficients will not be standardized unless you specify `sem`'s `standardized` reporting option.

Labeling SSD

You may use the following commands on SSD, and you use them in the same way you would with an ordinary dataset:

1. `rename` *oldvarname newvarname*
   You may rename the variables; see [D] **rename**.
2. `label data "`*dataset label*`"`
   You may label the dataset; see [D] **label**.

3. `label variable` *varname* "*variable label*"
   You may label variables; see [D] **label**.
4. `label values` *groupvarname valuelabelname*
   You may place a value label on the group variable; see [D] **label**. The group variable always takes on the values 1, 2, ...
5. `note:` *my note*
   `note` *varname*`:` *my note*
   You may places notes on the dataset or on its variables; see [D] **notes**.

Do not modify the SSD except by using the `ssd` command. Most importantly, do not drop variables or observations.

Making summary statistics from data for use by others

If you have raw data and wish to make the summary statistics available for subsequent publication, type

```
. ssd build varlist
```

where *varlist* lists the variables you wish to include in the dataset. The SSD will replace the raw data you had in memory. The full syntax is

```
. ssd build varlist if exp in range
```

so you may specify `if` and `in` to restrict the observations that are included.

For instance, to build an SSD for variables `occ_prestige`, `income`, and `social_status`, type

```
. ssd build occ_prestige income social_status
```

If you wish to build the dataset to include separate groups for males and females, type

```
. ssd build occ_prestige income social_status, group(sex)
```

However the `sex` variable was coded in your original data, the two sexes will be now be coded 1 and 2 in the resulting SSD. Which sex is 1 and which is 2 will correspond to however `sort` would have ordered `sex` in its original coding. For instance, if variable `sex` took on values "`male`" and "`female`", the resulting variable `sex` would take on values 1 corresponding to female and 2 corresponding to male.

Once you have built the SSD, you can describe it and list it:

```
. ssd describe
. ssd list
```

See [SEM] **Example 25**.

Reference

Acock, A. C. 2013. *Discovering Structural Equation Modeling Using Stata*. Rev. ed. College Station, TX: Stata Press.

Also see

[SEM] **Intro 10** — Fitting models with survey data

[SEM] **Intro 12** — Convergence problems and how to solve them

[SEM] Example 2 — Creating a dataset from published covariances

[SEM] Example 3 — Two-factor measurement model

[SEM] Example 19 — Creating multiple-group summary statistics data

[SEM] Example 25 — Creating summary statistics data from raw data

[SEM] **sem option select( )** — Using sem with summary statistics data

[SEM] **ssd** — Making summary statistics data (sem only)