Title

stata.com

mi add — Add imputations from another mi dataset

Description

`mi add` adds the imputations from the using dataset on disk to the end of the master dataset in memory.

Menu

Statistics > Multiple imputation

Syntax

`mi add` *varlist* `using` *filename* [`,` *options*]

options	Description
`assert(master)`	assert all observations found in master
`assert(match)`	assert all observations found in master and in using
`noupdate`	see [MI] **noupdate option**

`collect` is allowed; see [U] **11.1.10 Prefix commands**.

Notes:

1. Jargon:
   match variables = *varlist*, variables on which match performed
   master = data in memory
   using = data on disk (*filename*)
2. Master must be `mi set`.
3. Using must be `mi set`.
4. *filename* must be enclosed in double quotes if *filename* contains blanks or other special characters.

Options

`assert(`*results*`)` specifies how observations are expected to match. If results are not as you expect, an error message will be issued and the master data left unchanged.

`assert(master)` specifies that you expect a match for every observation in the master, although there may be extra observations in the using that `mi add` is to ignore.

`assert(match)` specifies that you expect every observation in the master to match an observation in the using and vice versa.

The default is that the master may have observations that are missing from the using and vice versa. Only observations in common are used by `mi add`.

`noupdate` in some cases suppresses the automatic `mi update` this command might perform; see [MI] **noupdate option**.

Remarks and examples

stata.com

Think of the result produced by `mi add` as being

Result	Source
$m = 0$	$m = 0$ from master
$m = 1$	$m = 1$ from master
$m = 2$	$m = 2$ from master
.	.
.	.
$m = M_{\text{master}}$	$m = M_{\text{master}}$ from master
$m = M_{\text{master}} + 1$	$m = 1$ from using
$m = M_{\text{master}} + 2$	$m = 2$ from using
.	.
.	.
$m = M_{\text{master}} + M_{\text{using}}$	$m = M_{\text{using}}$ from using

That is, the original data in the master remain unchanged. All that happens is the imputed data from the using are added to the end of the master as additional imputations.

For instance, say you discover that you and a coworker have been working on the same data. You have added $M = 20$ imputations to your data. Your coworker has separately added $M = 17$. To combine the data, type something like

```
. use mydata
. mi add patientid using karensdata
(17 imputations added; M=37)
```

The only thing changed in your data is M. If your coworker's data have additional variables, they are ignored. If your coworker has variables registered differently from how you have them registered, that is ignored. If your coworker has not yet registered as imputed a variable that you have registered as imputed, that is noted in the output. You might see

```
. use mydata
. mi add patientid using karensdata
(17 imputations added; M=37)
(imputed variable grade not found in using data;
    added imputations contain m=0 values for that variable)
```

Stored results

`mi add` stores the following in `r()`:

Scalars

`r(m)`	number of added imputations
`r(unmatched_m)`	number of unmatched master observations
`r(unmatched_u)`	number of unmatched using observations

Macros

`r(imputed_f)`	variables for which imputed found
`r(imputed_nf)`	variables for which imputed not found

Also see

[MI] **Intro** — Introduction to mi

[MI] **mi append** — Append mi data

[MI] **mi merge** — Merge mi data