

mi expand — Expand mi data

[Syntax](#) [Menu](#) [Description](#) [Options](#)
[Remarks and examples](#) [Also see](#)

Syntax

```
mi expand [=]exp [if] [, options]
```

<i>options</i>	Description
<code>generate(newvar)</code>	create <i>newvar</i> ; 0 = original, 1 = expanded
<code>noupdate</code>	see [MI] noupdate option

Menu

Statistics > Multiple imputation

Description

`mi expand` is `expand` (see [\[D\] expand](#)) for `mi` data. The syntax is identical to `expand` except that `in range` is not allowed and the `noupdate` option is allowed.

`mi expand` replaces each observation in the dataset with n copies of the observation, where n is equal to the required expression rounded to the nearest integer. If the expression is less than 1 or equal to missing, it is interpreted as if it were 1, meaning that the observation is retained but not duplicated.

Options

`generate(newvar)` creates new variable *newvar* containing 0 if the observation originally appeared in the dataset and 1 if the observation is a duplication.

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

Remarks and examples

stata.com

`mi expand` amounts to performing `expand` on $m = 0$, then duplicating the result on $m = 1$, $m = 2$, ..., $m = M$, and then combining the result back into `mi` format. Thus if the requested expansion specified by *exp* is a function of an imputed, passive, varying, or super-varying variable, then it is the values of the variable in $m = 0$ that will be used to produce the result for $m = 1$, $m = 2$, ..., $m = M$, too.

Also see

[MI] **intro** — Introduction to mi

[D] **expand** — Duplicate observations