mi stsplit — Split and join time-span records for mi data

Description

`mi stsplit` and `mi stjoin` are `stsplit` and `stjoin` for `mi` data; see [ST] **stsplit**. Except for the addition of the `noupdate` option, the syntax is identical. Except for generalization across m, the results are identical.

Your `mi` data must be `stset` to use these commands. If your data are not already `stset`, use `mi stset` rather than the standard `stset`; see [MI] **mi XXXset**.

Menu

Statistics > Multiple imputation

Syntax

To split at designated times

```
mi stsplit newvar [ if ], { at(numlist) | every(#) } [ options ]
```

options	Description
Main	
* `at(`*numlist*`)`	split at specified analysis times
* `every(#)`	split when analysis time is a multiple of #
`after(`*spec*`)`	use time since *spec* instead of analysis time for `at()` or `every()`
`trim`	exclude observations outside of range
`noupdate`	see [MI] **noupdate option**
`nopreserve`	programmer's option

* `at()` or `every()` is required.

`nopreserve` is not included in the dialog box.

To split at failure times

```
mi stsplit [ if ], at(failures) [ options ]
```

options	Description
Main	
* `at(failures)`	split at times of observed failures
`strata(`*varlist*`)`	perform splitting by failures within stratum, strata defined by *varlist*
`riskset(`*newvar*`)`	create risk-set ID variable
`noupdate`	see [MI] **noupdate option**
`nopreserve`	programmer's option

* `at()` is required.

`nopreserve` is not included in the dialog box.

To join episodes

```
mi stjoin [ , options ]
```

options	Description
Main	
`censored(`*numlist*`)`	values of failure that indicate no event
`noupdate`	see [MI] **noupdate option**

Options

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

See [ST] **stsplit** for documentation on the remaining options.

Remarks and examples

You should never use `stsplit`, `stjoin`, or any other heavyweight data management command with `mi` data. Instead, you should use their `mi` counterparts, such as `mi stsplit`. Heavyweight commands are commands that make sweeping changes to the data rather than simply deleting some observations, adding or dropping some variables, or changing some values of existing variables. `stsplit` and `stjoin` are examples of heavyweight commands (see [ST] **stsplit**).

Also see

[MI] **Intro** — Introduction to mi

[ST] **stsplit** — Split and join time-span records

[MI] **mi XXXset** — Declare mi data to be svy, st, ts, xt, etc.