**joinby — Form all pairwise combinations within groups**

### Syntax

```
joinby [varlist] using filename [ , options]
```

### options

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>When observations match:</strong></td>
<td></td>
</tr>
<tr>
<td>update</td>
<td>replace missing data in memory with values from <code>filename</code></td>
</tr>
<tr>
<td>replace</td>
<td>replace all data in memory with values from <code>filename</code></td>
</tr>
<tr>
<td><strong>When observations do not match:</strong></td>
<td></td>
</tr>
<tr>
<td><code>unmatched(none)</code></td>
<td>ignore all; the default</td>
</tr>
<tr>
<td><code>unmatched(both)</code></td>
<td>include from both datasets</td>
</tr>
<tr>
<td><code>unmatched(master)</code></td>
<td>include from data in memory</td>
</tr>
<tr>
<td><code>unmatched(using)</code></td>
<td>include from data in <code>filename</code></td>
</tr>
<tr>
<td><code>_merge(varname)</code></td>
<td><code>varname</code> marks source of resulting observation; default is <code>_merge</code></td>
</tr>
<tr>
<td>nolabel</td>
<td>do not copy value-label definitions from <code>filename</code></td>
</tr>
</tbody>
</table>

`varlist` may not contain `strL`s.

### Menu

Data > Combine datasets > Form all pairwise combinations within groups

### Description

`joinby` joins, within groups formed by `varlist`, observations of the dataset in memory with `filename`, a Stata-format dataset. By `join` we mean to form all pairwise combinations. `filename` is required to be sorted by `varlist`. If `filename` is specified without an extension, `.dta` is assumed.

If `varlist` is not specified, `joinby` takes as `varlist` the set of variables common to the dataset in memory and in `filename`.

Observations unique to one or the other dataset are ignored unless `unmatched()` specifies differently. Whether you load one dataset and join the other or vice versa makes no difference in the number of resulting observations.

If there are common variables between the two datasets, however, the combined dataset will contain the values from the master data for those observations. This behavior can be modified with the `update` and `replace` options.
Options

update varies the action that joinby takes when an observation is matched. By default, values from the master data are retained when the same variables are found in both datasets. If update is specified, however, the values from the using dataset are retained where the master dataset contains missing.

replace, allowed with update only, specifies that nonmissing values in the master dataset be replaced with corresponding values from the using dataset. A nonmissing value, however, will never be replaced with a missing value.

unmatched:none | both | master | using) specifies whether observations unique to one of the datasets are to be kept, with the variables from the other dataset set to missing. Valid values are

- none: ignore all unmatched observations (default)
- both: include unmatched observations from the master and using data
- master: include unmatched observations from the master data
- using: include unmatched observations from the using data

_merge(varname) specifies the name of the variable that will mark the source of the resulting observation. The default name is _merge(_merge). To preserve compatibility with earlier versions of joinby, _merge is generated only if unmatched is specified.

label prevents Stata from copying the value-label definitions from the dataset on disk into the dataset in memory. Even if you do not specify this option, label definitions from the disk dataset do not replace label definitions already in memory.

Remarks and examples

The following, admittedly artificial, example illustrates joinby.

Example 1

We have two datasets: child.dta and parent.dta. Both contain a family_id variable, which identifies the people who belong to the same family.

```
. use http://www.stata-press.com/data/r13/child
   (Data on Children)
. describe
Contains data from http://www.stata-press.com/data/r13/child.dta
   obs: 5  Data on Children
   vars: 4  11 Dec 2012 21:08
   size: 30

variable name storage display value variable label
      type format label
family_id  int   %8.0g      Family ID number
  child_id  byte  %8.0g      Child ID number
             %8.0g
             %8.0g
x2         int   %8.0g

Sorted by: family_id
```
We want to join the information for the parents and their children. The data on parents are in memory, and the data on children are posted at http://www.stata-press.com. child.dta has been sorted by family_id, but parent.dta has not, so first we sort the parent data on family_id:
. sort family_id
. joinby family_id using http://www.stata-press.com/data/r13/child
. describe
Contains data
    obs:     8  Data on Parents
    vars:    6
    size:   136

Variable name  type  format  label       variable label
family_id      int   %8.0g  Family ID number
parent_id      float %9.0g  Parent ID number
x1            float %9.0g
x3            float %9.0g
child_id      byte   %8.0g  Child ID number
x2            int    %8.0g

Sorted by:
    Note: dataset has changed since last saved
. list, sepby(family_id) abbrev(12)

<table>
<thead>
<tr>
<th>family_id</th>
<th>parent_id</th>
<th>x1</th>
<th>x3</th>
<th>child_id</th>
<th>x2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1025</td>
<td>12</td>
<td>27</td>
<td>721</td>
<td>1</td>
<td>300</td>
</tr>
<tr>
<td>1025</td>
<td>12</td>
<td>27</td>
<td>721</td>
<td>4</td>
<td>275</td>
</tr>
<tr>
<td>1025</td>
<td>12</td>
<td>27</td>
<td>721</td>
<td>3</td>
<td>320</td>
</tr>
<tr>
<td>1025</td>
<td>11</td>
<td>20</td>
<td>643</td>
<td>4</td>
<td>275</td>
</tr>
<tr>
<td>1025</td>
<td>11</td>
<td>20</td>
<td>643</td>
<td>1</td>
<td>300</td>
</tr>
<tr>
<td>1025</td>
<td>11</td>
<td>20</td>
<td>643</td>
<td>3</td>
<td>320</td>
</tr>
<tr>
<td>1026</td>
<td>13</td>
<td>30</td>
<td>760</td>
<td>2</td>
<td>280</td>
</tr>
<tr>
<td>1026</td>
<td>14</td>
<td>26</td>
<td>668</td>
<td>2</td>
<td>280</td>
</tr>
</tbody>
</table>

1. family_id of 1027, which appears only in child.dta, and family_id of 1030, which appears only in parent.dta, are not in the combined dataset. Observations for which the matching variables are not in both datasets are omitted.

2. The x1 variable is in both datasets. Values for this variable in the joined dataset are the values from parent.dta—the dataset in memory when we issued the joinby command. If we had child.dta in memory and parent.dta on disk when we requested joinby, the values for x1 would have been those from child.dta. Values from the dataset in memory take precedence over the dataset on disk.

Acknowledgment

joinby was written by Jeroen Weesie of the Department of Sociology at Utrecht University, The Netherlands.
Reference

Baum, C. F. 2009. *An Introduction to Stata Programming*. College Station, TX: Stata Press.

Also see

[D] **append** — Append datasets

[D] **cross** — Form every pairwise combination of two datasets

[D] **fillin** — Rectangularize dataset

[D] **merge** — Merge datasets

[D] **save** — Save Stata dataset

[U] 22 Combining datasets