

## Household data

Here's an example (from Stata's listserv) of the challenges in manipulating data for households & members:

Date: Mon, 28 Feb 2005 00:59:19 -0500  
From: "Kakatua Kutta" <[ambush@earthling.net](mailto:ambush@earthling.net)>  
Subject: st: data manipulation problem

Dear stata maestros,

I have a following data set

nh	pid	sid	age	educ
1	1	2	34	3
1	2	1	29	2
1	3	.	21	2
1	4	.	27	1
2	1	3	44	12
2	2	.	23	9
2	3	1	31	11
2	4	.	19	2
2	5	.	27	3
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.

nh is the household identification no and pid is individual identification within each household. Sid is the spouse identification no for those people who are married within a household. So for nh=1, it means pid 1 and 2 are married to each other and for nh=2, pid 1 and 3 are married to each other.

What I want to do is to make a datasets consisting of household members married to each other. That is, in the new data set I want to keep , for nh=1, pid 1 and 2 and for nh=2, pid 1 and 3. So what I am trying to find that for each nh, take those pid for which sid equals pid from others observations.

Is there anyway to do it rather than going through each observation individually?

-----  
In addition to yesterday's response (which suggested keeping respondents with non-missing sid values) ...

It sounds as if you eventually wish to (a) create within-household partnership identifiers, and then (b) attribute values for one spouse to the other spouse. This is relatively straightforward, but there are many potential complications, including e.g. (i) a person having a spouse (legitimate value for spouse id), but spouse is non-respondent (not in the data set); (ii) spouse id and person id numbers do not always differ by one.

Here follows an illustration of how to do (a) and (b) using British Household Panel Survey wave 1 variable names:

ahid: household identifier (like nh in your data)  
apno: person identifier within HHs (like your pid)  
ahgspn: spouse identifier within HH (= 0 for persons with no spouse; like your sid, except that your sid has missing values if no spouse)

\* create partnership identifier within each HH (missing if no partner)

```
// "one-line" solution uses cond(x,a,b) function:  
// cond(x,a,b) = a if x is "true"; = b if false  
// [See Manual under functions]  
// NB cond(x,a,b,c) as above, but evaluates to c if x missing
```

```
ge apartnum = cond(apno < ahgspn, apno, ahgspn) if ahgspn > 0
```

```
// alternative "two line" solution is perhaps easier to read
```

```
gen apartnum2 = apno if apno < ahgspn & ahgspn > 0  
replace apartnum2 = ahgspn if ahgspn < apno & ahgspn > 0
```

\* Create spousal variables

```
foreach v of varlist ajbsemp ajbft asex aage amastat ///  
    amlstat ajbstat pid {
```

```
// Use cond() again. Within each partnership, max # cases = 2.  
// For case #1 in partnership, set var value = value for case #2  
// (and missing if missing case #2, i.e. non-resp spouse)  
// For case #2 in partnership (where there is one), set var value  
// = value for case #1  
// Case #2 has _n==2 (_n indexes observation number within  
// partnership within household)  
// Since max # cases in partnership = 2, if "_n==2" false, then must  
// refer to case #1, or be missing
```

```
bysort ahid apartnum: ge s`v' = cond(_n==2, `v'[1], `v'[2], .) if apartnum < .
```

```
// A slightly longer -- but perhaps more readable -- equivalent would be:  
// bysort ahid apartnum: ge s`v' = `v'[2] if apartnum < . & _n == 1  
// bysort ahid apartnum: replace s`v' = `v'[1] if apartnum < . & _n == 2
```

```
}
```

The main apparent difference between the BHPS case and yours is that (1) your spid appears to be missing (".") rather than 0 for respondents with no spouse, and (2) your other variables are age and educ, rather than ajbsemp, etc. Adapt the above code accordingly

As in many things, the trick here is use of Stata's wonderful "by group" capabilities

Stephen

=====  
Professor Stephen P. Jenkins <[stephenj@essex.ac.uk](mailto:stephenj@essex.ac.uk)>  
Institute for Social and Economic Research (ISER)  
University of Essex, Colchester CO4 3SQ, UK  
Phone: +44 1206 873374. Fax: +44 1206 873151.  
<http://www.iser.essex.ac.uk>

- \*
- \* For searches and help try:
- \* <http://www.stata.com/support/faqs/res/findit.html>
- \* <http://www.stata.com/support/statalist/faq>
- \* <http://www.ats.ucla.edu/stat/stata/>

-----  
Date: Tue, 1 Mar 2005 10:41:07 +0100 (CET)  
From: "Jens Lauritsen" <[jl@epidata.dk](mailto:jl@epidata.dk)>  
Subject: st: Re: st Data Manupulation

For the "find mate" problem of:

nh	pid	sid	age	educ
1	1	2	34	3
1	2	1	29	2
1	3	.	21	2
1	4	.	27	1
2	1	3	44	12
2	2	.	23	9
2	3	1	31	11
2	4	.	19	2
2	5	.	27	3

A combined strategy of `egen, by` etc would do this type of rearrangement, but it can be very useful to control the logic yourself by comparison of one record with the next using `recordidentifier varname[_n]` and comparing to prior or next record with index `[_n-1]` or `[_n+1]`

Often the problem is however that the data are not consistent due to data entry error or misreading of entry material.

For the situation at hand one strategy could be something along the lines of:

```
* assuming you only consider couples:  
use datafilename  
drop if sid == . //as suggested yesterday.  
sort nh pid sid
```

```
* assert quality of data:
```

```
gen str10 test = string(nh)+"-" + string(pid) + "-" + string(sid)
test = trim(test)
assert test != test[_n-1]
* if the assert failed then do a listing of the records which failed.
e.g. list if test == test[_n-1]
* and fix the data.
```

```
* now generate an identifier.
gen coupleid = _n //record number
replace coupleid =coupleid[_n-1] if nh == nh[_n-1]
```

```
* generate variables of interest e.g. age difference:
gen agedif = age - age[_n+1] if coupleid == coupleid[_n+1]
```

```
*make a lookup table of this:
keep nh pid coupleid agedif
```

```
sort nh pid
save lookupfile, replace
```

```
* put back the variables to original file:
use datefilename
sort nh pid
merge nh pid using lookupfile
```

```
* do whatever you like of analysis, but remember to count each couple
only once, e.g.
keep if coupleid != coupleid[_n-1]
summ agedif
```

Jens Lauritsen  
Odense University Hospital, Denmark  
[www.epidata.dk](http://www.epidata.dk)