Panel Data 2: Setting up the data

Richard Williams, University of Notre Dame, https://www3.nd.edu/~rwilliam/

Last revised March 19, 2018

We often have data where variables have been measured for the same subjects (or countries, or companies, or whatever) at multiple points in time. These are typically referred to as Panel Data or as Cross-Sectional Time Series Data. We need special techniques for analyzing such data, e.g. it would be a mistake to treat 200 individuals measured at 5 points in time as though they were 1,000 independent observations. Therefore, Stata has an entire manual and suite of XT commands devoted to panel data, e.g. `xtreg`, `xtlogit`, `xtpoisson`, etc. Some other commands, like `clogit`, can also sometimes be used. (Conversely, the `xt` commands can sometimes be used when you don't have panel data, e.g. you have data from students within a school. In such situations you might also use the `me`, mixed-effects, commands.)

In order to use these commands, though, the data set needs to be properly structured. This will sometimes require that the data be restructured from wide to long. In wide format, a data set has one record for each subject. This record has several variables, e.g. income1, income2, income3, where each of the income variables gives the value of income at a different time point. In long format, the data are restructured to have one record for each subject for each time point. I am going to give some examples of how to do this, but if in doubt be sure to read the Stata documentation for help on setting up your data.

Here is an example from Allison's 2009 book *Fixed Effects Regression Models*. Data are from the National Longitudinal Study of Youth (NLSY). The data set has 1151 teenage girls who were interviewed annually for 5 years beginning in 1979. Here is a listing of the values for the first three cases:

```
. version 13.1
. use https://www3.nd.edu/~rwilliam/statafiles/teenpov, clear
. rename inschool* school*
. list in 1/3
```

1.

id	pov1	mother1	spouse1	school1	hours1	pov2	mother2	spouse2	school2	hours2	pov3	mother3	spouse3	school3	hours3	pov4
22	1	0	0	1	21	0	0	0	1	15	0	0	0	1	3	0

mother4	spouse4	school4	hours4	age	black	pov5	mother5	spouse5	school5	hours5
0	0	1	0	16	0	0	0	0	1	0

2.

id	pov1	mother1	spouse1	school1	hours1	pov2	mother2	spouse2	school2	hours2	pov3	mother3	spouse3	school3	hours3	pov4
75	0	0	0	1	8	0	0	0	1	0	0	0	0	1	0	0

mother4	spouse4	school4	hours4	age	black	pov5	mother5	spouse5	school5	hours5
0	0	1	4	17	0	1	0	0	1	0

3.

id	pov1	mother1	spouse1	school1	hours1	pov2	mother2	spouse2	school2	hours2	pov3	mother3	spouse3	school3	hours3	pov4
92	0	0	0	1	30	0	0	0	1	27	0	0	0	1	24	1

mother4	spouse4	school4	hours4	age	black	pov5	mother5	spouse5	school5	hours5
1	0	0	31	16	0	1	1	0	0	0

The numbers at the ends of some variable names reflect the time period the variable refers to (1 = 1979, 2 = 1980, etc.) Variables without numbers in the names do not vary across time.

- id is the subject id number and is the same across each wave of the survey
- pov_t is coded 1 if the subject was in poverty during that time period, 0 otherwise.
- age is the age at the first interview.

- black is coded 1 if the respondent is black, 0 otherwise.
- $mother_t$ is coded 1 if the respondent currently has at least 1 child, 0 otherwise.
- $spouse_t$ is coded 1 if the respondent is currently living with a spouse, 0 otherwise.
- $school_t$ is coded 1 if the respondent is currently in school, 0 otherwise.
- $hours_t$ is the hours worked during the week of the survey.

The data are currently in wide format. There is one record per case with multiple variables representing values at different points in time. We need to get the data into long format instead. In Stata, we can do this with the `reshape` command.

```
. reshape long pov mother spouse school hours, i(id) j(year)
(note: j = 1 2 3 4 5)

Data                               wide   ->   long
-----------------------------------------------------------------------------
Number of obs.                     1151   ->    5755
Number of variables                  28   ->       9
j variable (5 values)                     ->   year
xij variables:
                     pov1 pov2 ... pov5   ->   pov
             mother1 mother2 ... mother5   ->   mother
             spouse1 spouse2 ... spouse5   ->   spouse
             school1 school2 ... school5   ->   school
                hours1 hours2 ... hours5   ->   hours
-----------------------------------------------------------------------------
```

The `reshape long` part of the command told Stata we wanted to reshape the data from wide to long. (There is also a `reshape wide` command for going from long to wide.) The variable list that followed was the list of variables (actually the stubnames of the variables) that varied across time (you should use a consistent naming convention, e.g. pov1, mother1, etc. pov79, mother79, pov80, mother80, would have also been ok. Be careful about doing something like inc2, inc79, inc80, inc81, where inc2 = income squared; Stata will think inc2 is another of the time-varying variables.) The variables not listed are those that do not vary across time; their values will be copied on to each of the new records for the case. `i(varlist)` specifies the variables whose unique values denote a logical observation. i() is required. In this case only `i(id)` was needed but in other cases multiple variables might define a case. `j(varname)` specifies the variable whose unique values denote a subobservation. Here is what the reshaped data for the first 3 (now 15) cases looks like.

```
. list in 1/15
```

	id	year	age	black	pov	mother	spouse	school	hours
1.	22	1	16	0	1	0	0	1	21
2.	22	2	16	0	0	0	0	1	15
3.	22	3	16	0	0	0	0	1	3
4.	22	4	16	0	0	0	0	1	0
5.	22	5	16	0	0	0	0	1	0
6.	75	1	17	0	0	0	0	1	8
7.	75	2	17	0	0	0	0	1	0
8.	75	3	17	0	0	0	0	1	0
9.	75	4	17	0	0	0	0	1	4
10.	75	5	17	0	1	0	0	1	0
11.	92	1	16	0	0	0	0	1	30
12.	92	2	16	0	0	0	0	1	27
13.	92	3	16	0	0	0	0	1	24
14.	92	4	16	0	1	1	0	0	31
15.	92	5	16	0	1	1	0	0	0

Each of the original cases now has 5 records, one for each year of the study. The value of year varies from 1 to 5. The values of age (age at first interview) and black have been duplicated on each of the 5 records. Instead of 5 poverty variables, we have 1, whose value can differ across the five records (e.g. the original value of pov2 for id 22 is now the value of pov for id 22 year 2). The same is true for the other time-varying variables.

The next thing we want to do is xtset the data. The `xtset` command tells Stata that these are Panel data. The usual format is

```
xtset panelvar
xtset panelvar timevar
```

That is, we must tell Stata what the panelvar is; in this case it is id. The timevar is optional and may or may not be necessary depending on our analysis. In the current case the timevar is year. `xtset` typed with no parameters tells us how the data are xtset.

```
. xtset id year
       panel variable:  id (strongly balanced)
        time variable:  year, 1 to 5
                delta:  1 unit

. xtset
       panel variable:  id (strongly balanced)
        time variable:  year, 1 to 5
                delta:  1 unit
```

NOTE (copied verbatim from the Stata 12 Manual): "The terms balanced and unbalanced are often used to describe whether a panel dataset is missing some observations. If a dataset does not contain a time variable, then panels are considered balanced if each panel contains the same number of observations; otherwise, the panels are unbalanced. When the dataset contains a time variable, panels are said to be strongly balanced if each panel contains the same time points, weakly balanced if each panel contains the same number of observations but not the same time points, and unbalanced otherwise."

A data set might be unbalanced because data are missing for some years. If you were, say, analyzing countries, it might even be that the country did not exist during some time periods. Strongly balanced data are best but my understanding is that Stata can generally do a good job with unbalanced data.

Once the data are xtset, several commands are available to us; see `help xt`. For example, you can use the `xtsum` command, which is similar to the `summarize` command but contains some additional information.

```
. xtsum

Variable         |      Mean   Std. Dev.       Min        Max |    Observations
-----------------+--------------------------------------------+----------------
id       overall |  6016.672   3298.064         22      12539 |     N =    5755
         between |             3299.211         22      12539 |     n =    1151
         within  |                   0   6016.672   6016.672 |     T =       5
                 |                                          |
year     overall |         3   1.414336          1          5 |     N =    5755
         between |                   0          3          3 |     n =    1151
         within  |             1.414336          1          5 |     T =       5
                 |                                          |
age      overall |  15.64639    1.04682         14         17 |     N =    5755
         between |             1.047184         14         17 |     n =    1151
         within  |                   0   15.64639   15.64639 |     T =       5
                 |                                          |
black    overall |  .5742832   .4944942          0          1 |     N =    5755
         between |             .4946661          0          1 |     n =    1151
         within  |                   0   .5742832   .5742832 |     T =       5
                 |                                          |
pov      overall |  .3768897    .484649          0          1 |     N =    5755
         between |             .3100424          0          1 |     n =    1151
         within  |             .3725925  -.4231103    1.17689 |     T =       5
                 |                                          |
mother   overall |  .1986099   .3989883          0          1 |     N =    5755
         between |             .3253864          0          1 |     n =    1151
         within  |             .2310605  -.6013901   .9986099 |     T =       5
                 |                                          |
spouse   overall |  .0992181   .2989806          0          1 |     N =    5755
         between |             .2206498          0          1 |     n =    1151
         within  |             .2018338  -.7007819   .8992181 |     T =       5
                 |                                          |
school   overall |  .6304083   .4827361          0          1 |     N =    5755
         between |               .32013          0          1 |     n =    1151
         within  |             .3614169  -.1695917   1.430408 |     T =       5
                 |                                          |
hours    overall |  8.671764   14.54341          0         90 |     N =    5755
         between |             9.363817          0       52.4 |     n =    1151
         within  |             11.13062  -43.72824   72.07176 |     T =       5
```

The different values for the standard deviations can sometimes be useful. For id, age and black, the within standard deviation is 0. This is because, within each subject, the value of these variables does not vary, i.e. for each of the five records the case has, the values of these variables are the same. For year, the between subjects standard deviation is 0. This is because all subjects have the same set of values on year. For poverty, the between and within standard deviations are nearly the same. This tells us that the variation in poverty across women is nearly equal to that observed within a woman over time. That is, if you were to draw two women randomly from the data, the difference in poverty is expected to be nearly equal to the difference for the same woman in two randomly selected years.

Some techniques, such as fixed effects models, work much better when there is a lot of within-subject variability (or conversely, they don't work well when subjects change little across time). As we will see, there are many advantages to fixed-effects models, but some types of data are friendlier to them than are others.