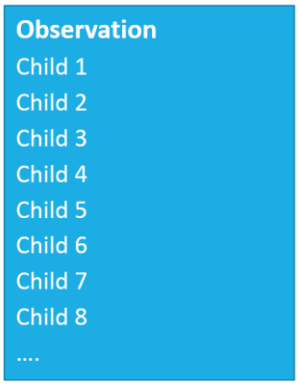
Stata: Merge and append

Topics: Merging datasets, appending datasets

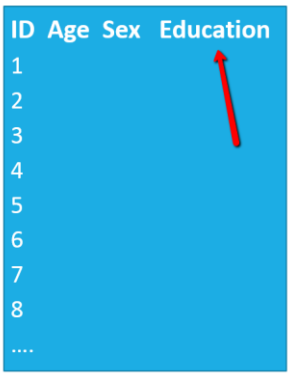


1. Terms

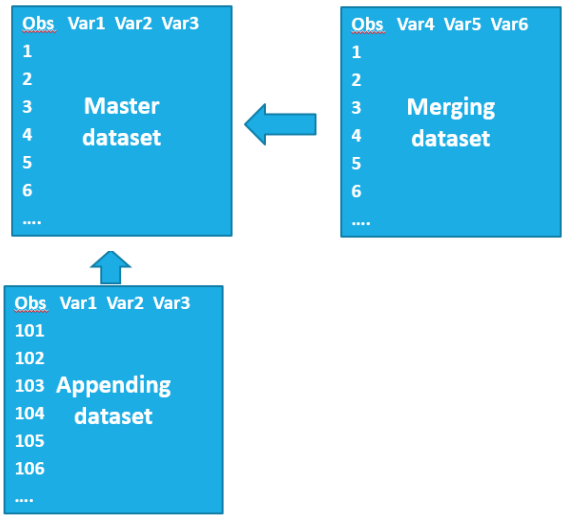
There are several situations when working with large population datasets that you need to append or merge datasets.

Let us clarify a few terms first.

Observations are the rows in the dataset. Observations are related to the unit of analysis in the dataset. For example, observations might be women, or households, or children.



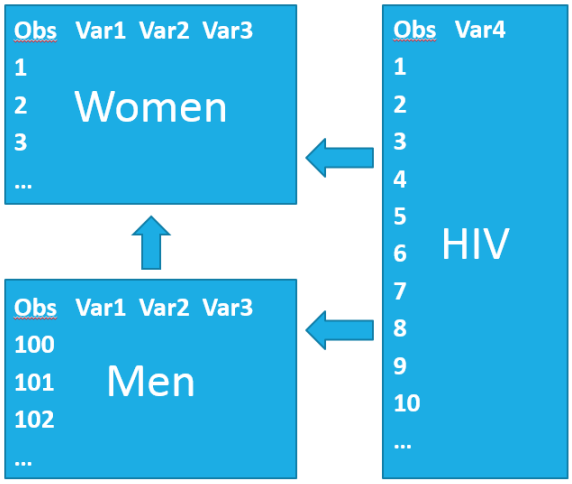
Variables are the columns in the dataset. Each variable contains a unique piece of information about each observation.



The **master dataset**, is the main dataset. The master dataset usually is organized by the unit of analysis – for example the master dataset would be women if we were answering a research question about pregnancies.

The **merging** dataset contains the variables that we are adding to the master dataset.

The **appending** dataset contains the observations that we are adding to the master dataset.



Here is one situation with DHS data that requires both merge and append: Let us say that we will analyze HIV infection status and several demographic and social covariates in both men and women. In the DHS data, women covariates, men covariates, and HIV test results (for men and women) are stored in three separate files. So we have to first append men and women demographic and social data, then merge HIV status.

2. Append

Let us look at this scenario using 2010 Rwanda DHS data.

2a. Prepare data for append

```
*** PREAPRE MEN DATA ***
use "C:\Dana\Project\data\RWMR61FL", clear
keep mv001 mv002 mv003 mv012
rename mv001 v001
rename mv002 v002
rename mv003 v003
rename mv012 v012
gen long id=((1000+v001)*10000)+(v002*100)+v003
gen gender = "man"
save "C:\Dana\Project\data\men", replace
count //6329 men
```

Before we append the women and men datasets, we open each original dataset, make some changes to facilitate the `append`, and then `save` modified versions of each file.

With the men's individual recode file open, we make the following changes:

First, we `keep` only those variables that we need for this analysis.

Second, we ensure that we have the same variables in both datasets, and that the variables share a common variable name. Let us use the `rename` statement to change the variables names in the male recode file (which start with "mv") to start with "v" like the woman's file.

Third, we create a unique numerical ID for each person. For DHS data, a person ID is derived from the cluster, household, and household member line number. In this example, these are variables v001, v002, and v003, respectively.

Fourth, we `generate` a new variable called "gender" since we are about to combine data from men and women.

```
*** PREPARE WOMEN DATA ***
use "C:\Dana\Project\data\RWIR61FL", clear
keep v001 v002 v003 v012
gen long id=((1000+v001)*10000)+(v002*100)+v003
gen gender = "woman"
save "C:\Dana\Project\data\women", replace
count //13,671 women
```

We perform these steps first in the men dataset and `save` it. Then we repeat these steps with the women's individual recode dataset – creating variables for person ID and gender. Let us check the `count` of observations in each file and make a note, so we can check our work later.

2b. Append statement

```
*** APPEND WOMEN <--MEN ***
use "C:\Dana\Project\data\women", clear
append using "C:\Dana\Project\data\men"
tab gender
count // 20,000 women and men
save "C:\Dana\Project\data\menwomen", replace
```

Now that both datasets have the same variable names, we are ready to append them. The append statement itself is quite easy to use. First, open the master dataset, in this case the woman file that we just created.

With the master dataset open, type `append using` and the path and name of the men dataset in quotes.

```
. tab gender
```

gender	Freq.	Percent	Cum.
man	6,329	31.65	31.65
woman	13,671	68.36	100.00
Total	20,000	100.00	

We check that the data appended correctly by tabulating the variable gender. We see that the file contains 6,329 men and 13,671 women as expected. Then we save a new permanent dataset in our project\data folder called "menwomen.dta".

3. Merge

3a. Prepare data for merge

```
*** PREAPRE HIV DATA ***
use "C:\Dana\Project\data\RWAR61FL", clear
gen long id=((1000+hivclust)*10000)+(hivnumb*100)+hivline
keep id hiv03 hiv05
save "C:\Dana\Project\data\hiv", replace
```

Before merging HIV status to the combined menwomen dataset, we ensure that the HIV dataset has a unique person ID variable to link observations in the HIV dataset with observations in the combined menwomen dataset. We `keep` only the variables that we need, and `save` the file.

3b. Merge statement

```
*** MERGE MENWOMEN <--HIV ***
use "C:\Dana\Project\data\menwomen", clear
merge 1:1 id using "C:\Dana\Project\data\hiv"
```

The syntax for merge is simple. Start by opening the master dataset, in this case the menwomen dataset that we just saved.

After the `merge` statement, we must specify the type of merge (1:1, m:1, or 1:m), the variable name that is common in both datasets, and then type `using` and the path and name of the merging dataset in quotes.

1:1 one observation master = one observation merging

m:1 many observations master (kids)= one observation merging (mother)

1:m one observation master = many observations merging

3c. Merge types (1:1, m:1, 1:m)

There are three types of merges that are generally performed with survey data: one-to-one, many-to-one, and one-to-many. Our example of merging HIV status to person is an example of a one-to-one merge. One-to-one means that there is one observation in the master dataset for each observation in the merging dataset. It is possible that some observations will not match, but there are not duplicated IDs in either dataset.

If we were merging mother data onto kid data, then we would have a many-to-one merge because there are many kids per woman. A one-to-many merge would be the opposite.

3d. Investigating unmatched data

When we use the merge statement, Stata automatically generates a new variable in the dataset called _merge, and provides a summary of that variable in the output window. The _merge variable always has 3 categories: 1 is assigned to unmatched observations from the master dataset, 2 is assigned to unmatched observations in the using dataset, and 3 is assigned to matched observations.

Variables

Variable	Label
v001	cluster number
v002	household number
v003	respondent's line number
v012	respondent's current age
id	
gender	
hiv03	blood test result
hiv05	sample weight
_merge	

```
Result                           # of obs.
-------------------------------------------
not matched                         7,026
    from master                     6,752  (_merge==1)
    from using                        274  (_merge==2)

matched                            13,248  (_merge==3)
-------------------------------------------
```

The output window tells us that 13,248 men and women had an HIV test result. And there were 6,752 men and women without an HIV result, and 274 HIV results for whom we have no other survey information.

Before moving on, it is essential to understand why we have any unmatched IDs. In this case, I investigated these unmatched records using the `browse` statement, by comparing the merged data to the household roster dataset, and by reading the final DHS Report to check the response rates for these different datasets. Ultimately I determined that the 274 people tested for HIV with no matching survey main data refused or had incomplete responses. The 7,026 people who responded to the main survey but did not get tested for HIV were expected per the sample design because only every other household was selected for HIV testing.

```
*** INVESTIGATE UNMATCHED ***
br
sort id

use "C:\Dana\Project\data\RWPR61FL", clear
rename hv001 v001
rename hv002 v002
rename hv003 v003
rename hvidx person
rename hv105 age
rename hv104 gender2
keep v001 v002 v003 age gender2 person
gen long id = ((1000+v001)*10000) + (v002*100) + person
save "C:\Dana\Project\data\hhmembers", replace

use "C:\Dana\Project\data\menwomen", clear
merge 1:1 id using "C:\Dana\Project\data\hiv"
drop _merge
merge 1:1 id using "C:\Dana\Project\data\hhmembers"
br
sort id
tab age gender2 if v001==.
```

```
Result                           # of obs.
-----------------------------------------
not matched                         7,026
    from master                     6,752  (_merge==1)
    from using                        274  (_merge==2)

matched                            13,248  (_merge==3)
-----------------------------------------
```

```
*** SAVE MERGED DATASET ***
use "C:\Dana\Project\data\menwomen", clear
merge 1:1 id using "C:\Dana\Project\data\hiv"
keep if _merge==2 | _merge==3
drop _merge

save "C:\Dana\Project\data\menwomenhiv"
```

3e. Keeping observations

Since our research question is about HIV, and because we have a sampling weights for everyone tested for HIV (the HIV sample weight is `hiv05`), we decide to keep the 13,248 matching records, plus the 274 HIV test results that did not have matching observations in the main women's or men's surveys. Ultimately, these 274 records will "fall out" of the analysis due to missing data, but it is important to keep them in the dataset so that Stata can calculate variance estimates correctly based on the total observations per the sample design. When we are done with the _merge variable, we drop it to keep the dataset clean, then save the final merged dataset.

If it is unclear why we kept _merge==2, check out the Survey Analysis in Stata video at populationsurveyanalysis.com.