ECON B314
Economics of Social Policy
Bryn Mawr College

Stata Lab 1

Prof. June Kim

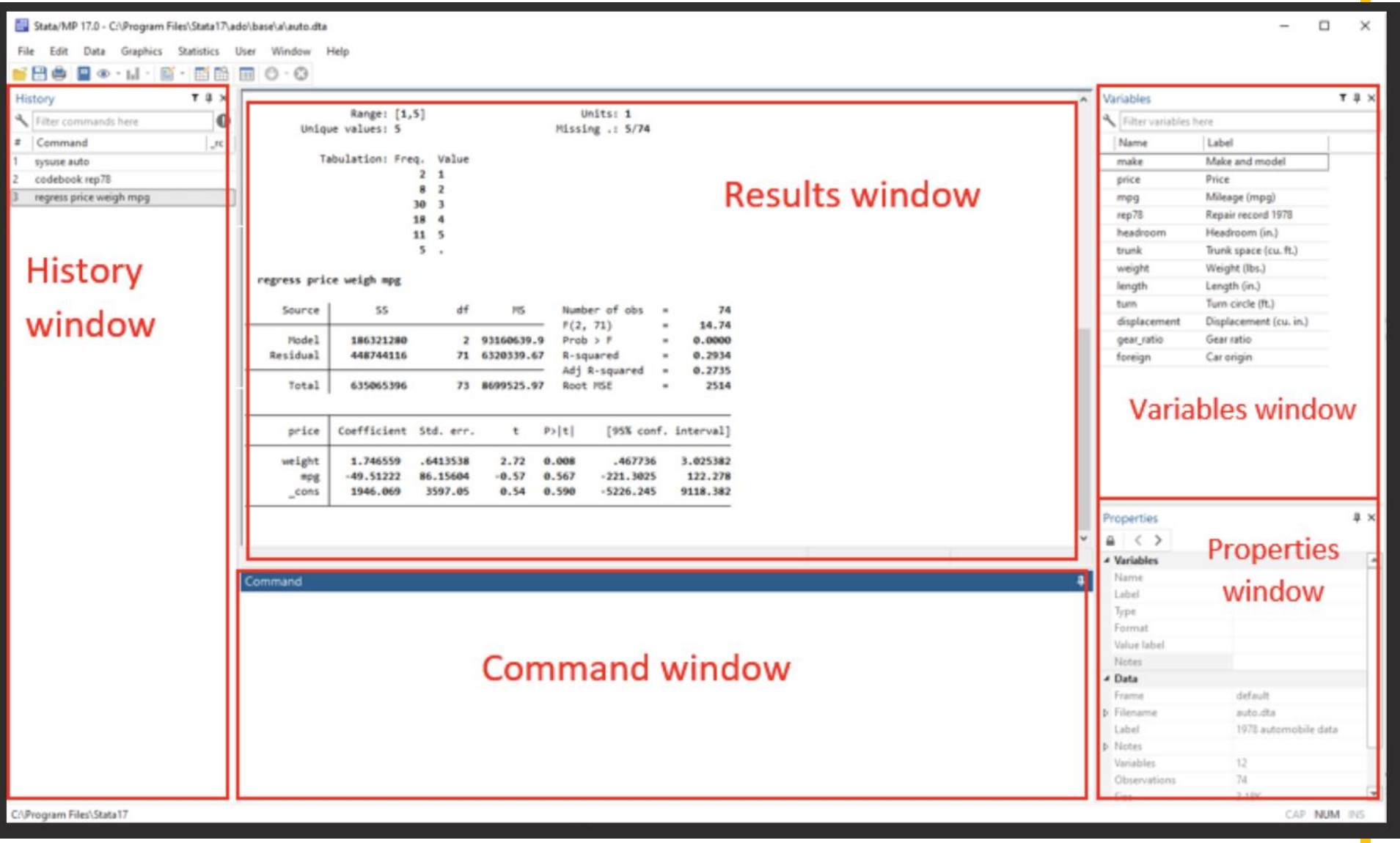
Introduction to Stata

1. The Stata Screen
2. Entering a database into Stata
3. Describing data
4. Creating / Editing data
5. T-test
6. For-loop
7. Writing up your own do-file!

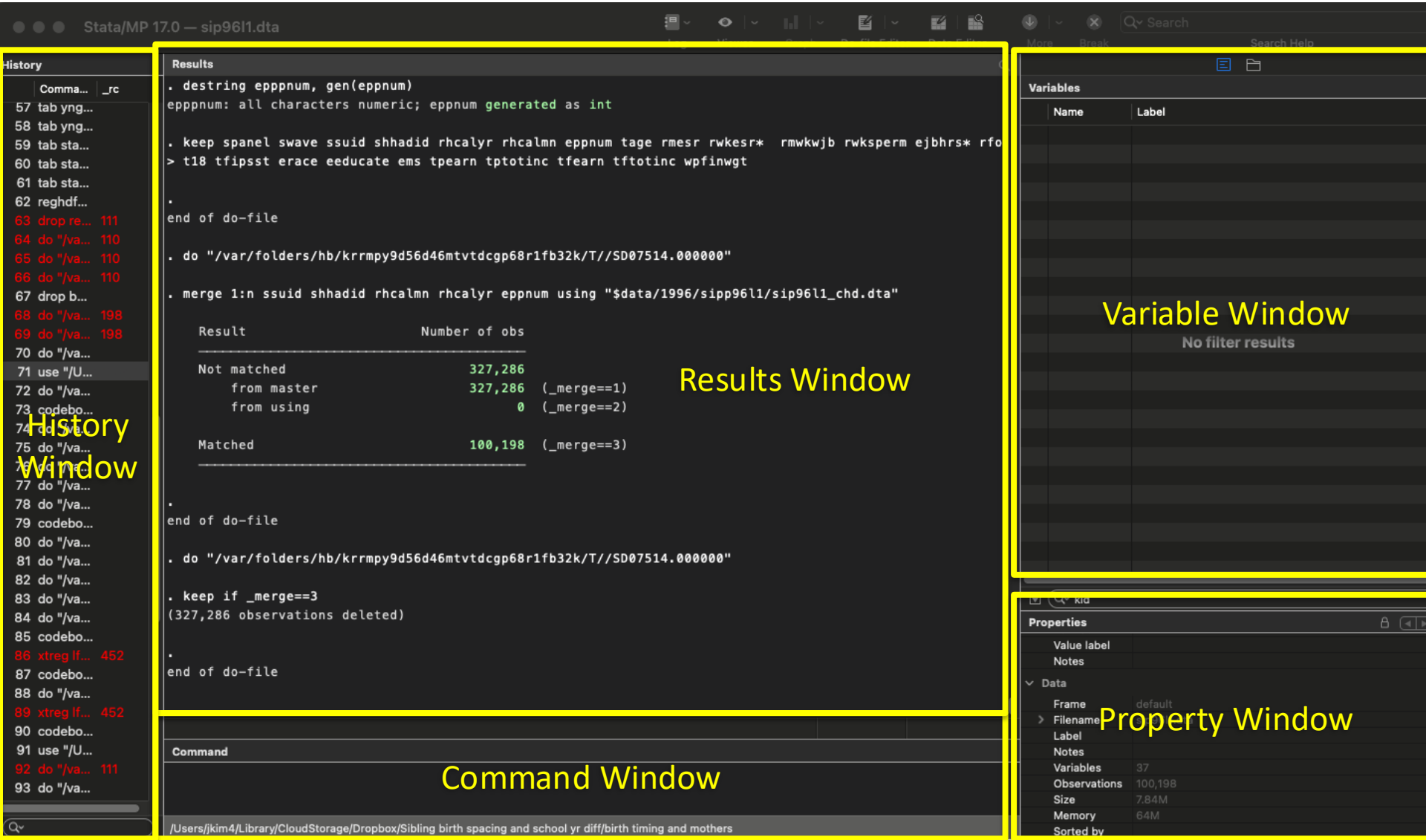
Introduction to Stata

1. The Stata Screen
2. Entering a database into Stata
3. Describing data
4. Creating / Editing data
5. T-test
6. For-loop
7. Writing up your own do-file!

1. The Stata Screen (Window)



1. The Stata Screen (Mac)



1. The Stata Screen

- Command window
  - Where you write commands and execute them (by pressing the Enter key)
- Results window
  - Where the output (or error messages) resulting from the commands is displayed
- Variables window
  - Where the variables you are working with are listed
- History window
  - Displays the history of the commands you have used

Introduction to Stata

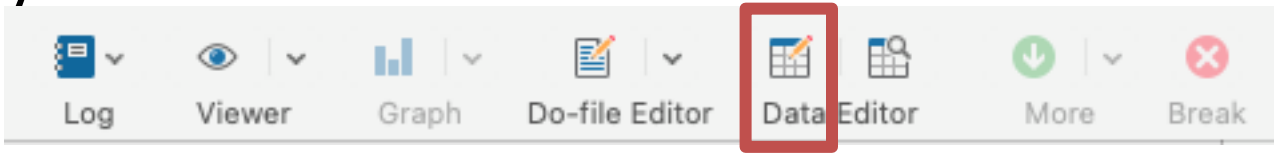
1. The Stata Screen
2. Entering a database into Stata
3. Describing data
4. Creating / Editing data
5. T-test
6. For-loop
7. Writing up your own do-file!

2-1. Importing excel data into Stata

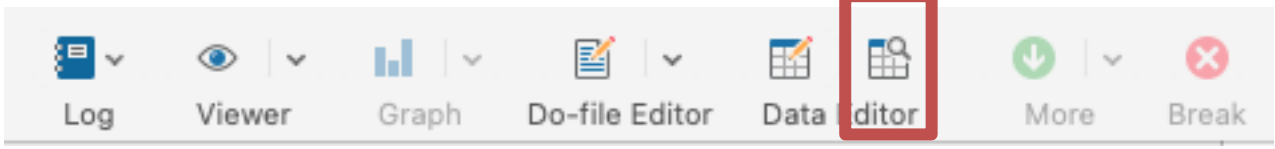
- From Moodle>Week 2> data, download a excel data set: **Student survey result.xlsx**

Method 1) Stata>File>Import>Excel Spreadsheet>Browse the **Student survey result.xlsx**>Select Worksheet: Data > Click “import first row as variable names”>OK

Method 2) Copy the entire excel sheet>Stata data editor (edit) >paste it

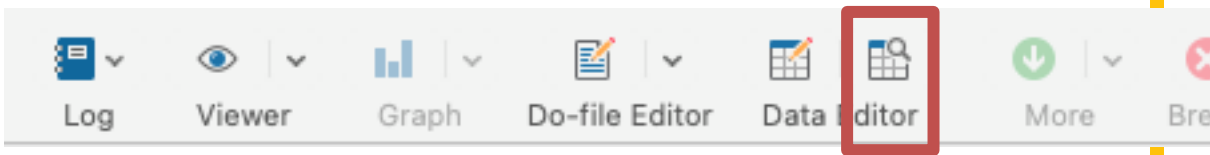


- You can see the data by typing “bro” or by clicking the icon



2-2. Entering existing data into Stata

- The databases we use in Stata have the extension ".dta"
- From Moodle>Week 2> data, download a stata data set: students.dta
- Open the dataset into Stata
- You can see the data by typing "bro" or by clicking the icon



Introduction to Stata

1. The Stata Screen
2. Entering a database into Stata
3. Describing data
4. Creating / Editing data
5. T-test
6. For-loop
7. Writing up your own do-file!

3. Describing data

- Writing a command:

```
command_name variable1 variable2 …, option
```

Example: `summarize age, detail`

Example: `summarize age gender`

- For some commands, if you don't specify variable names, Stata will execute the command for all the variables in your dataset.

Example: `summarize`

Example: `summarize, detail`

3. Describing data

- `describe (des)`
  - Display a table describing the variables of your dataset (name, labels and type)

- `codebook`
  - Display pretty much all the information of the variable: range, number of missing values, mean, sd, and percentile.
  - E.g.) `codebook age`

- `summarize(sum)`
  - Display summary statistics for the variables: number of observations, mean, sd, Min, Max
  - You will have more summary statistics (percentiles for example) using the option `detail` : E.g.) `sum weightlb, detail`

- `tabulate (tab)`
  - Display the frequency table for the variable (how many times each value of the variables is observed in the dataset)
  - E.g.) `tab  gender`

3. Describing data

- `tab var1, sum (var2)`
  - Tabulate by variable 1, summarize variable 2

  E.g.1) `tab gender, sum(sns_number)`

  : For each gender, summarize number of SNS using

  E.g.2) `tab sns_number, sum(workout)`

  : by number of SNS using, summarize days of workout

  Does SNS usage motivate you to workout more???

Introduction to Stata

1. The Stata Screen
2. Entering a database into Stata
3. Describing data
4. Creating / Editing data
5. T-test
6. For-loop
7. Writing up your own do-file!

if option

- You can restrict the use of your commands to certain observations of your dataset that satisfy one or more conditions

- Format of a command line

```
command variable if condition1 condition2, options;
```

- Be careful with the syntax of conditions (big source of error messages) :
  - Equal to ==
  - Not equal to !=
  - Higher (less) than > (<)
  - Higher (less) or equal to >= (<=)
  - Condition 1 AND condition 2 : `condition1 & condition2`
  - Condition 1 OR condition 2 : `condition1 | condition2`

- Example : detailed summary statistics of earnings for female :

```
summarize weightlb if gender=="M", detail
```

Creating & Editing variables

- Names and labels
  - Change a variable name :

    ```
    rename travel countries
    ```

  - Give a label to a variable :

    ```
    label variable foot "height_foot"
    label variable inch "height_inch"
    ```

- Generating a new variable: **gen**

(Note the difference in uses of "=" and "==")

E.g.) generate the dummy variable female

```
gen female=1 if gender=="F"
```

- Modifying an existing variable : **replace**

```
replace female=0 if gender=="M"
```

- Above two commands can be done with just one line!

```
gen female=(gender=="F")
```

Introduction to Stata

1. The Stata Screen
2. Entering a database into Stata
3. Describing data
4. Creating / Editing data
5. T-test
6. For-loop
7. Writing up your own do-file!

5. T-test using Stata

`ttest [outcome], by(group)`

- E.g.) Test whether number of SNS(s) using is significantly different across gender

`ttest sns_number, by(female)`

- You can also use the following commands to compare outcome by group:
  - `tab female, sum(sns_number)`
  - `bysort female: sum sns_number`
- But, `ttest` command actually implements the hypothesis testing and provides whether they are statistically different

$$H_0: \mu_1 - \mu_2 = 0 \quad \text{vs} \quad H_a: \mu_1 - \mu_2 \neq 0$$

(where μ_1: # SNS for female, μ_2: # SNS for male)

Introduction to Stata

1. The Stata Screen
2. Entering a database into Stata
3. Describing data
4. Creating / Editing data
5. T-test
6. For-loop
7. Writing up your own do-file!

6. For-loop

For repeated commands, we can make it loop so that we don't write the same command again and again!

E.g.) Test whether a variety of outcomes are significantly different across gender

```
foreach var in sns_number  workout work workhrs {
   ttest `var', by(female)
}
```

E.g.) generate a new variable denoting "yes" for 1, "no" for 0 for some dummy variables:

```
foreach var in work car foreign livewpar {
   gen `var'_yn="yes" if `var'==1
   replace `var'_yn="no" if `var'==0
}
```

Introduction to Stata

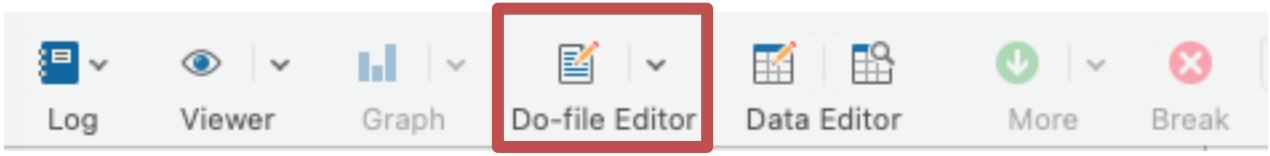
1. The Stata Screen
2. Entering a database into Stata
3. Describing data
4. Creating / Editing data
5. T-test
6. For-loop
7. Writing up your own do-file!

Why do-file?

- A do-file contains a sequence of commands that can be processed automatically

- Select all commands in your History window and right click to select “Send to Do-file Editor”

- To create a new do-file, click on the Do-file Editor icon in the toolbar. Save the do-file that you have created in your working directory.



- You can run **all** the commands it contains by clicking on “Do” icon or Ctrl+d
- Or run just **one** command by highlighting the part of the command and Ctrl+d

Set working directory

- Notice how the “use” statement with the filepath appears.
- Copy the entire filepath and designate it as your working directory:

```
cd “H:/user/Econ304/Data/"
```

- Now you don’t have to have a long line of code with the entire working path.

- It is also a good practice to start adding the “`clear all`” before you open any dataset, to clear any existing data in memory.

```
clear all
use ”students.dta", clear
```

global and local macros

- **`Global`**
  - used to define file paths or list of variables <u>until Stata closes</u>
  - Reference: `$name`

e.g.)

```
global data "/Users/jkim4/Dropbox/SIPP/Newresearch”
use "$data/sipp1996.dta”

global demo age sns_number car
reg bmi workout $demo
```

- **`local`**
  - used to define file paths or list of variables <u>temporarily within the specific do-file</u>
  - Reference: `` `name' ``

e.g.)

```
local myvars “age workhrs travel sns_number”
summarize `myvars'
```

commenting

- To make a comment in do-file, use * at the beginning of a command line in your do-file.
  - to comment a whole section of your do file, use /* before the first line and */ after the last line of your commands
  - useful if you don't want to execute a series of command but want to keep them for notes
- "help" command in Stata is very useful !
  - Just type help + the name of a command, and see what happens.

logging

- Also, begin a new log file. Similarly, copy the code that starts the log file into your do file above your use statement. You should add the "replace" line as well to prevent errors, because otherwise you will get an error every time. It is good practice to put both the using and the close statements at this time (**log close is the very last line of code in the do file**).

```
log using "econ304_lab1.log", replace

.
.
<<lines of code>>
.
.
log close
```

- Note that the log file is overwritten each time.
- All text will be written in between the "log" lines.

Bonus: Practice Merging Two Datasets

1. Download the excel file named "UKCPR_National_Welfare_Data_1980_2023_July25update.xlsx"
2. Read this file into Stata
3. Keep variables `state_fips year Unemploymentrate PovertyRate StateMinimumWage`
4. Save this new data with name "UKCPR.dta"
5. Clear all and Open a new dataset: "ACS female head 2000-22.dta" from Moodle and Use this as master data
6. Merge this master data with "UKCPR.dta" using two keys: **year** and **state_fips**

```
merge X:X KEY VARIABLE(S) using DATA DIRECTORY
```

Bonus: Practice Merging Two Datasets

3 types of merge in Stata:

1) m:m

both master data and using data contain multiple observations for each combination of key variables

2) 1:m

master data has one observation for each combination of key variables, while using data does not.

3) m:1

using data has one observation for each combination of key variables, while master data does not.

Which merge type do we need?

“Master” data: ACS female head 2000-22.dta

“Using” data: UKCPR.dta

- “Master” data has many observations for, say, a combination of year 2000 and state Pennsylvania

- “Using” data has just one single observation for the combination of year 2000 and state Pennsylvania (one single measure for Unemployment rate, Poverty rate, Minimum wage for 2000 in PA)

Bonus: Practice Merging Two Datasets

```
. merge m:1 year state_fips using "/Users/jkim4/Library/CloudStorage/Dropbox/Bryn Mawr/Fall 2025/Econ o
>  Social Policy/Data/UKCPR.dta"

    Result                      Number of obs
    ------------------------------------------
    Not matched                         1,072
        from master                         0  (_merge==1)
        from using                      1,072  (_merge==2)

    Matched                         1,730,516  (_merge==3)
    ------------------------------------------
```

- What is _merge==1?
- _merge==2?
- _merge==3?