Stata Tutorial: Session III

Data Management

- Importing data
- Specialized download commands
- Clean & reshape data
- Merging datasets

Goals:

1) Read Data into Stata
2) Clean & Reformat Data
3) Combine data from difference sources
4) Do it all via .do files

Why #4?

- Proper research methodology
- Correct errors later on

1) Read Data into Stata

- Start with Excel or .csv (comma separated variable) formats
- Several other formats supported

Start with menu: `File > Import`

- Specify sheet
- Specify data range
- Specify 1st row as variable names

Excel:

```
import excel "FH.xls", cellrange(A4:CI22) firstrow clear
```

Text:

```
import delimited "EMDAT.csv", clear
```

2) Specialized Download Commands

`wbopendata` World Development Indicators (WDI)

```
ssc install wbopendata  /*once to install*/
wbopendata, indicator(SP.POP.GROW; NY.GDP.MKTP.CD) long clear
```

`freduse` Federal Reserve Economic Database (FRED)

```
ssc install freduse  /*once to install*/
freduse UNRATE CPIAUCSL, clear
```

`wid` World Wealth & Income Database (WID)

```
ssc install wid  /*once to install*/
wid, indicators(shweal) areas(FR US) perc(p90p100 p99p100)
     year(1950/2015) ages(992) pop(j) clear
```

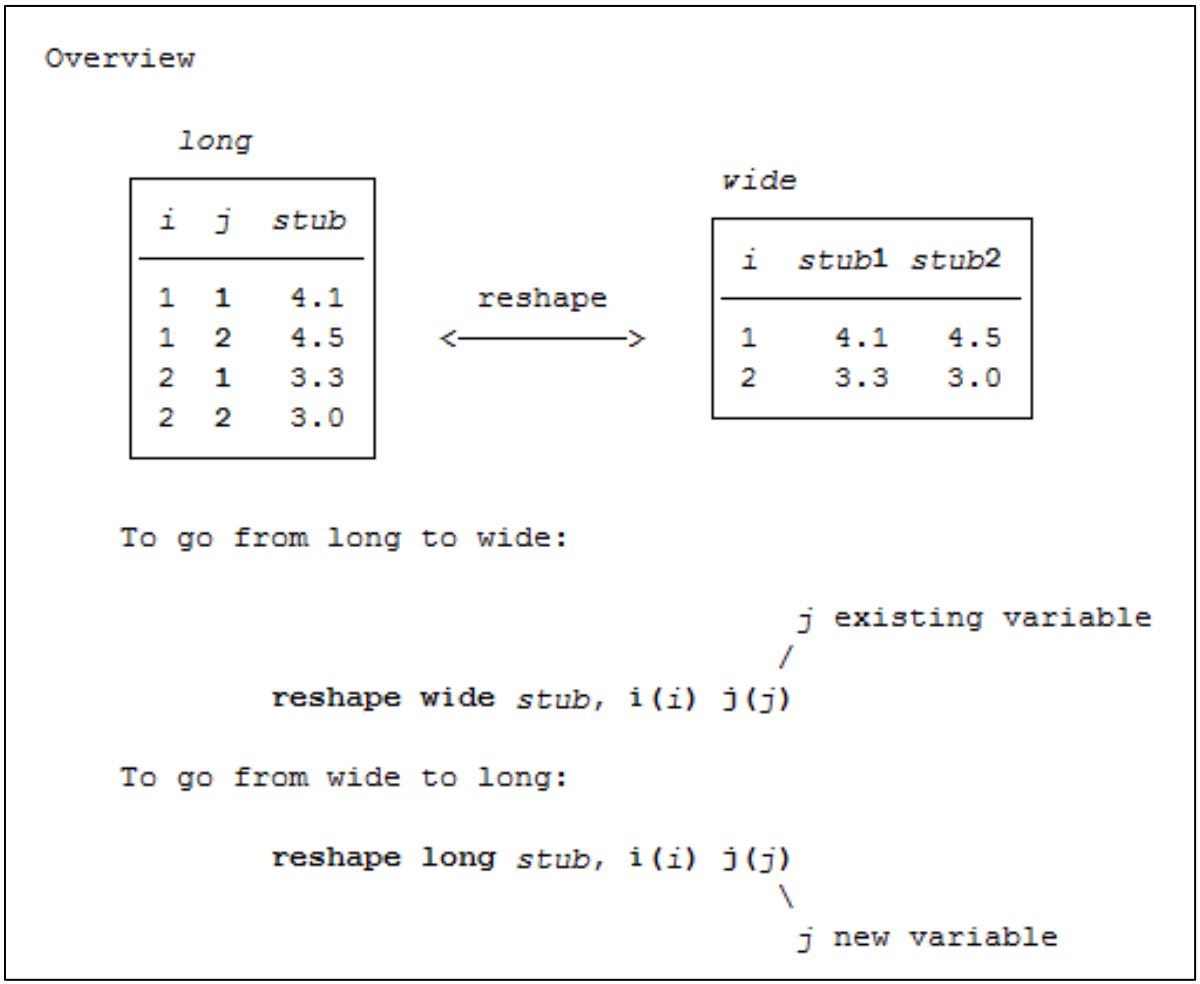
Example: `"aid data.do"`

3) Clean & Reformat Data

- Reshape if needed
- Drop unwanted variables / observations
- Identify & correct errors in data
- Collapse to average, sum, etc., as needed

3) Clean & Reformat Data

Reshape:



Options:

```
string
```

Placing # with @:

```
@stub
ctry@var
```

3) Clean & Reformat Data

Dropping variables/observations

drop GDP Pop

drop if year<2001

keep year country growth

keep if year>2000

Text variables – “Strings”

`destring`	convert from text to number
`ltrim / rtrim`	trim blank spaces from left
`substr`	keep part of longer string of text
`subinstr`	replace part of string
`string`	converts # to text

3) Clean & Reformat Data

Dates: Stata counts from January 1, 1960

- dates before then are negative #s
- count normally in days but can be in seconds, months, etc.

`date` converts text to date

```
gen mydate=date(adate,"MDY")
```

`year` extracts year from date

```
gen myyear=year(mydate)
```

`month` extracts month of year from date

```
gen mymonth=month(mydate)
```

`day` extracts day of month from date

```
gen myday=day(mydate)
```

`quarter` extracts fiscal quarter from date

```
gen myquarter=quarter(mydate)
```

3) Clean & Reformat Data

Cleaning might need loops to repeat tasks:

```
foreach i in var1 var2 var3 {
   replace ‘i’=0 if missing(`i’)
}

forvalue i=1/3 {
   replace var`i’=0 if missing(var`i’)
}
```

Another useful commands:

```
bysort country: gen count=_n
egen country_mean=mean(GDP), by(country)
```

3) Clean & Reformat Data

Collapse Command

- Collapse a large data set into smaller data set by
  - sum across group
  - average across group
  - count non-missing elements group
- Examples

```
collapse (mean) income, by(year state)
collapse (max) max_y=income (min) min_y=income, by(state)
collapse (count) income (sum) unemployed, by(state)
```

4) Merging Datasets

- Procedure to match names for merging
- Merge command

Example: `Import data.do`

```
merge 1:1 varlist using filename [, options]

merge 1:1 country year using wdi.dta
merge 1:1 country year using wdi,dta nogen
merge 1:m country using wdi,dta keep(master match)
merge 1:1 country year using wdi.dta, keepusing(GDP)
```