Working with Stata: applied tricks and tips

Monday, 16 February 2015
Technical seminar series
Indaba Agricultural Policy Research Institute
Today’s exercise

• There are several objectives of today’s exercise:
  1. To familiarize IAPRI researchers with the World Bank’s World Development Indicators database and its potential applications in IAPRI’s research
  2. To provide some relevant applied examples of accessing this database in Stata via the \textit{wbopendata} command
  3. To review several important data manipulation techniques available in Stata:
     1. Alternative structures for panel data and the \textit{reshape} command
     2. Stata programming functions for
        1. Selecting sub-sets of data
        2. Looping over ranges
Resources for today’s exercise

Stata do-files:
• get_wbopendata_cpi.do
• get_wbopendata_growth_poverty.do
WB’s World Development Indicators

• The primary World Bank collection of development indicators, compiled from officially-recognized international sources. It presents the most current and accurate global development data available, and includes national, regional and global estimates.

• >1200 indicators

• Yearly time series, 1960 – 2015 (updated frequently)

• Available for countries and regional aggregates

WB’s World Development Indicators

Topics include:

- Agriculture & Rural Development
- Aid Effectiveness
- Climate Change
- Economy & Growth
- Education
- Energy & Mining
- Environment
- External Debt
- Financial Sector
- Gender
- Health
- Infrastructure
- Labor & Social Protection
- Poverty
- Private Sector
- Public Sector
- Science & Technology
- Social Development
- Trade
- Urban Development

Examples of available indicators:

- Internet users (per 100 people)
- Exports of goods and services (% of GDP)
- Unemployment, total (% of total labor force)
- Agriculture, value added (% of GDP)
- CO2 emissions (metric tons per capita)
- Poverty headcount ratio at nat’l poverty line (% of pop)

GDP per capita (current US$)
Exports of goods and services (% of GDP)
Foreign direct investment, net inflows (BoP, current US$)
GNI per capita, PPP (current international $)
GINI index
Literacy rate, adult total (% of people ages 15 and above)
Central government debt, total (% of GDP)
Inflation, GDP deflator (annual %)
The *wbopendata* module for Stata

- The *wbopendata* module connects to the World Bank Open Data API and provides direct access to the latest version of the WDI database through the Stata interface.
- To install:
  ```
  ssc install wbopendata
  ```
- After installation, for help:
  ```
  help wbopendata
  ```
- To activate the visual interface:
  ```
  db wbopendata
  ```
- *wbopendata* homepage:
  ```
  ```
Panel data structures

### Long:

<table>
<thead>
<tr>
<th>hhid</th>
<th>year</th>
<th>income</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2001</td>
<td>1079.177</td>
</tr>
<tr>
<td>1</td>
<td>2002</td>
<td>2242.663</td>
</tr>
<tr>
<td>1</td>
<td>2003</td>
<td>2722.033</td>
</tr>
<tr>
<td>2</td>
<td>2001</td>
<td>1117.656</td>
</tr>
<tr>
<td>2</td>
<td>2002</td>
<td>2212.599</td>
</tr>
<tr>
<td>2</td>
<td>2003</td>
<td>1848.009</td>
</tr>
<tr>
<td>3</td>
<td>2001</td>
<td>1672.03</td>
</tr>
<tr>
<td>3</td>
<td>2002</td>
<td>2681.245</td>
</tr>
<tr>
<td>3</td>
<td>2003</td>
<td>1572.694</td>
</tr>
</tbody>
</table>

### Wide:

<table>
<thead>
<tr>
<th>hhid</th>
<th>income2001</th>
<th>income2002</th>
<th>income2003</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1079.177</td>
<td>2242.663</td>
<td>2722.033</td>
</tr>
<tr>
<td>2</td>
<td>1117.656</td>
<td>2212.599</td>
<td>1848.009</td>
</tr>
<tr>
<td>3</td>
<td>1672.03</td>
<td>2681.245</td>
<td>1572.694</td>
</tr>
</tbody>
</table>
**reshape command**

- Provides way of reorganizing panel data from long to wide or vice versa

**Overview:**

<table>
<thead>
<tr>
<th>long</th>
<th>wide</th>
</tr>
</thead>
<tbody>
<tr>
<td>i  j  stub</td>
<td>i  stub1 stub2</td>
</tr>
<tr>
<td>1  1  4.1</td>
<td>1  4.1  4.5</td>
</tr>
<tr>
<td>1  2  4.5</td>
<td>2  3.3  3.0</td>
</tr>
<tr>
<td>2  1  3.3</td>
<td></td>
</tr>
<tr>
<td>2  2  3.0</td>
<td></td>
</tr>
<tr>
<td>+------------+ +------------+</td>
<td></td>
</tr>
</tbody>
</table>

To go from long to wide:

```
/ reshape wide stub, i(i) j(j)
```

To go from wide to long:

```
\ reshape long stub, i(i) j(j)
   j new variable
```
Selecting sub-sets of data

• The `inrange` and `inlist` functions to select observations that meet various criteria

**inrange** \((z, a, b)\)
Returns 1 if it is known that \(a < z < b\); otherwise, returns 0.
The following ordered rules apply:
- \(z > .\) returns 0.
- \(a > .\) and \(b = .\) returns 1.
- \(a > .\) returns 1 if \(z < b\); otherwise, it returns 0.
- \(b > .\) returns 1 if \(a < z\); otherwise, it returns 0.
  Otherwise, 1 is returned if \(a < z < b\).
  If the arguments are strings, "." is interpreted as "".

**inlist** \((z, a, b, \ldots)\)
Returns 1 if \(z\) is a member of the remaining arguments; otherwise, returns 0.
All arguments must be reals or all must be strings.
The number of arguments is between 2 and 255 for reals and between 2 and 10 for strings.

Examples:
```plaintext
reg y x1 x2 if inrange(farmsize, 0, 10)
keep if inlist(regioncode, "SSF")
drop if !inlist(countrycode, "ZMB", "KEN", "ETH", "MOZ", "MWI")
```
Looping structures

**foreach**: loop over elements of a list

```
foreach lname {in|of listtype} list {
    commands referring to `lname'
}
```

**Example**: loop over existing variables, summarizing each

```
foreach var of varlist pri-rep t* {
    quietly summarize `var'
    summarize `var' if `var' > r(mean)
}
```

**forvalues**: loop over consecutive values

```
forvalues lname = range {
    commands referring to `lname'
}
```

where range may be defined as #1(#d)#2, i.e. from #1 to #2 in steps of #d

**Example**: generate 100 uniform random variables named x1, x2, ..., x100.

```
forvalues i = 1(1)100 {
    generate x`i' = runiform()
}
```