Stata: A Brief Introduction

1. Statistical Packages

- There are many statistical packages (Stata, SPSS, SAS, Splus, etc.)

- Statistical packages can be used for
  - Analysis
  - Data Manipulation
  - Data Management

- Different statistical packages have different strengths

- We will use Stata.
  - It is a relatively easy package to learn.
  - Stata has an easy spreadsheet-style data representation
  - In Stata, you work a little bit at a time, accumulating results, rather than writing one big program and doing everything at once.
  - Stata keeps good records of your actions.
  - It is relatively cheap (both in cost and memory).
    - The Stata GradPlan (an educational plan) allows you to get Intercooled Stata with a small User's Guide for < $200. This can be purchased as the Matthews Medical Bookstore, located in the 1830 Building.
    - A complete set of Reference Manuals is available at additional cost (but not necessary for this course).
  - Stata license is perpetual – if you buy it, you own it forever.
  - Stata is available in the Hygiene computer labs (W3017 and W3025) and in the Hampton House lab. To access, follow the path: Start -> Programs -> Stata -> Stata8

2. Stata's Windows

- Toolbar: Provides buttons for common tasks

- Command Window: Where the commands are typed

- Results: Shows all the output (i.e. the results) from previous commands
- Review: Shows previous commands
- Variables: Shows all variables in the dataset
- Status Bar: Shows the current working directory
- Graph: Appears when graphs are created

3. Command Line Interface (CLI)

A command is issued in the command window and Stata

- Attempts to figure out the meaning of the command
- Executes the command if it is entered properly
- Issues an error message (in red) if it cannot figure out what to do, or if there is something which could inadvertently change the dataset

If Stata gives you an error, it will provide you with a return code, r(#). You can find out more about the error by typing: `lookup rc #`

4. Graphical User Interface (GUI)

Stata 8 offers a wide variety of menu-driven commands. There is nothing you can do with menus that cannot be done with commands, but the menus allow you to carry out tasks without knowing the commands or their syntax. However, we will be providing commands so that you will become familiar with them. The commands will be useful for constructing longer and more complicated programs.

5. Keeping Track of What You Do: Log Files

- Log files write a copy of the Results window into a text file
    - As you work in Stata, you can view the results of the most recent commands, but the Results window does **not** save all the results of all your commands.
    - The graph window is **not** written into the log file.

- Comments may be added by starting a command with an asterisk (*).
- Log files are a necessity, since they provide a record of exactly what has been done with the data.

- **The first thing you should do after starting Stata is to open a log file.**

- To open/create a log file:

- Click the Log button on the toolbar (it has a picture of a scroll on it).
- In the "Save as type" menu, select Formatted Log (*.smcl) or Log(*.log).

Give the log file a name.

- A message will be issued in the results window indicating that your log has been created.

- One can easily import information from a log file into a word processor:
    - Open the log file in a word processor (Notepad or Word)
    - Highlight a selection from the log file, copy and paste into a report.

Note: The log file is a text file that only prints nicely in MS Word using certain fonts, like "courier 8".

Note: You may also save your log file as a Formatted Log (*.smcl), which is the default for Stata8. If you view this log file within Stata, the log appears exactly as it did in the Results window and any error messages are printed in red. If you open this type of log file in a word processor, you will see formatting commands. You must first "Translate" the Formatted Log file into a text file. Go to File -> Log -> Translate, then enter the Formatted Log file name as the "Input File" and give a name for the text version "Output File".

*The two types of log files serve the same purpose. The difference is that *.smcl files keep fonts and color, *.log files are plain text, ready to be edited into *.do files. Note that .smcl files can be translated into *.log files using the translate command.

6. Opening a Dataset

There are many ways of getting data into Stata.

- If you have a Stata dataset (*.dta):
    - Click the Open button
    - Or, File -> Open
    - Or, Control-O

  Note the 'use ….' Statement in the Results window which gives the full name and path to the dataset.

- If you have a Stata Dataset (*.dta) you can double click on it to open it.

- Enter the data directly within Stata. This is useful for small datasets.
    - Click on the Editor button
    - Enter the data

      Tab to move to the right
      Return to move down
      Don't use arrow keys – they move the cursor without saving the data.

  - To make changes in the editor permanent, click the Preserve button. This cannot be undone.
  - To undo all changes since the last preserve, click the Restore button. This cannot be undone.

- Use data translation software (i.e. StatTransfer) to create a Stata dataset.
- Copy the data from a spreadsheet, and paste it into the Stata editor.

Stata can use only **one** dataset at a time. If you try to open a dataset and …

- There is no dataset in use, everything is fine.
- There is a dataset in use which has been changed since the last save, Stata will put up a warning.
- There is dataset in use which has not been changed since the last save, then Stata will clear this dataset, and bring in the requested dataset.

7. Exploring the Dataset

- This should be the first task when a new dataset is received.
  - There may be unexpected data
  - There may be missing data
  - There may be miscoded data
- **Never** use a dataset without first looking at it and understanding its contents.

- **Viewing the Data: Browser and Editor**

  Both of these buttons allow us to look at the data as a table.

  - The Browser is better for exploring, since it prevents any inadvertent changes.

  Click the Browse button.

  - Notice the spreadsheet-like layout.
  - The rows are observations.
  - The columns are variables.
  - The top row gives the name of the variable, the observation number, and the value of the current cell.

  When browsing, data may not be changed.

- **The `list` command:**
  - `list` will list all the observations and all the variables in the data

- `list` *`varlist`* will list all the observations for those variables listed in *`varlist`*

- **The `codebook` command**

The `codebook` command gives details about both structure and contents.

- `codebook` gives details about every variable in the dataset
- `codebook` *`variable name`* gives details about the one variable listed.

- **The `summarize` command**

The `summarize` command gives standard statistics for each variable. Note, string/character variables will be listed as having no observations, since they have no *numerical* observations.

8. Good Dataset Management

We want to make the dataset usable in case

- it is passed on to another researcher
- you put it away for several months

To this end, we should

- Add notes about the data
- Give good variable names and labels
- Carefully label encoded variables

- **Data Labels**

A data label is a description of the entire dataset.

- `label data` *`"put in a data label"`* provides a description which will show up in the results from `describe` and shows on the screen when data are opened in Stata.
- Label var varname "content"

- **Notes**

The `notes` command allows us to add additional notes about the dataset or about specific variables in the dataset.

- `notes:` *`put a note about dataset here`* adds notes about the dataset

- `notes` *`varname: put a note about variable here`*
  adds notes about the specified variable

The notes may be viewed by typing: `notes`

- **Variable Names**

Variable names may contain letters of the alphabet, underscores (_), and numbers but

- cannot start with a number
- should not start with an underscore

Variable names may have up to 32 characters. Stata will abbreviate variable names to 8 or 12 characters.

Stata is case-sensitive. The variable `thisvar` is different than the variable `THISVAR.`

To rename a variable

- at command line, type: `rename` *`oldname newname`*
- in editor: double-click in proper column, and edit variable name

- **Variable Labels**

Variable labels allow us to give an informative variable description. These labels will appear in the variables window, and will be printed as labels on tables and graphs.

- at command line, type: `label variable` *`varname “description of variable here”`*
- in editor: double-click in proper column, edit label

- **Value Labels**

Value labels are text labels for numerically encoded categorical variables.

- `label define` *`lblname # “tag” # “tag”`*
- `label values` *`varname lblname`*

Note: one value label may be attached to many variables

9. Graphs

Graphical displays can be easily made by Stata. You should complete the **Self-Practice**, which will give a more detailed explanation of some of the graphing commands available in Stata. The `graph` command (with appropriate options specified) will make:

- histograms
- scatterplots

- box plots

The following commands will produce basic graphs.

- `histogram varname` -> histogram
- `twoway scatter var1 var2` -> scatterplot
- `graph box var1` -> box plot

You may consult the Stata8 *Graphics Manual* for more information on:

- axis title
- figure titles
- specifying symbol and line types
- specifying axis tick placement

NOTE: You must save your graphs because they are not written to the log file. You can use the following option to save your graphs as a Windows Metafile for later use in a document:

- at command line: `graph varname, saving(filename.wmf, replace)`
- pull-down menus: File -> Save Graph -> *filename.*wmf

Alternatively, you may immediately copy/paste graphs into a MS Word document.
Edit -> Copy Graph -> go to MS Word document -> Edit -> Paste Special -> Picture

10. Exiting Stata

Before you exit a Stata session, you should save changes to your Stata dataset.

- File -> Save will replace the dataset on the disk
- File -> Save as … saves the dataset under a new name

When working with a dataset, you should always keep a copy of the original dataset as a backup, so you don't accidentally delete critical data.

When you exit Stata, your log file will automatically close.

11. Getting Help

We will get back to this in the Self-Practice section.

12. Self-Practice

Suppose that you are studying the incidence of pneumonia in East Baltimore over the last two years, our observations might consist of tabulations of the number of visits to the Emergency Room at Johns Hopkins Hospital over that time period. Then some *variables* might consist of each patient's height, age, gender, weight and whether or not a person is experiencing pain or has a cough. Some of the data might look like this:

Age	Gender	Height	Weight	Pain	Cough
18	1	4.1	109	1	0
11	0	4.3	97	1	1
22	0	7.2	222	1	0
25	1	4.9	101	1	0
16	1	6.3	190	1	0
37	1	6.5	212	1	0
76	1	5.8	156	0	1
19	0	6.3	176	1	1
40	1	6.0	187	0	1
12	0	3.4	84	0	1

In the data above, a value of 1 indicates Male and 0 indicates Female for Gender, for both Pain and Cough, 1 indicates the symptom is present and 0 indicates that the symptom is absent.

Perform the tasks described below. You will learn about data entry into Stata and also more about the graphics options in Stata.

1. Open a log file in Stata. Pay attention to where the log file is going to be written (pathname), this will allow you to view your log file after you have finished this exercise.

2. Enter the above data into the editor.

3. Double click on the variable name box in the editor to assign meaningful variable names.

4. Label each variable

5. Create labels for the values of gender, pain and cough.

   For instance, if we want to assign a label for present (1) and absent (0) of the variables, which I called pain or cough, I could use the following commands.

   ```
   label define symptom 1 “present” 0 “absent”
   label values pain symptom
   label values cough symptom
   ```

   You create a label for the gender of the patients.

6. Use the `codebook` and `summarize` commands to get familiar with the output.

7. Suppose in our pneumonia example we want to make a graph of the patient's *height*. Use the following command:

   ```
   histogram height
   ```

   Height is italicized since you may be using a different variable name, please enter the appropriate name into Stata. This command will create a histogram (bar chart) of the height variable.

8. Now obtain a boxplot. Use the following:

   ```
   graph box height
   ```

   `graph` is the command, box indicates the type of graph, *`height`* is the variable.

9. Perhaps we would like to plot the relationship between height and weight of the pneumonia patients. We can use the `graph` command and list two variables to obtain a scatterplot. Use the following:

   ```
   twoway scatter height weight
   ```

   The plot produced is a scatterplot and you will see that the height variable is on the vertical (y) axis and the weight variable is on the horizontal (x) axis.

   Note: All the above operations also can be performed by using the *Graphics menu.*

10. Get more details on `graph` by using the Help menu. The syntax is hard to understand at first sight. But you can always scroll down the examples to find helpful information.

11. Close your log file and open it in NOTEPAD or MS Word.