data formats

adam okulicz-kozaryn

adam.okulicz.kozaryn@gmail.com

this version: Tuesday 25th January, 2022 10:01

outline

misc

data types

commenting and syntax

import/export or read/write

outline

misc

data types

commenting and syntax

import/export or read/write

today's class

- still slow today; last one; next week will get fast!
  - if too slow for you, do extra things!
  - check out recommended materials, start working on final project, find more data, etc
- today, still basic so that everybody has the basics covered
  - everybody is at different level
  - if you are bored: help others, check extra materials, see help files and experiment with commands

outline

misc

data types

commenting and syntax

import/export or read/write

data basics

- dataset is a matrix
- cols are variables (var), rows are observations (obs; U/As)
- vars are characteristics of obs
- eg 'edu, 'age', and 'inc are vars and persons are obs
  - each row is a separate person

paths (location of a file on hd)

- eg.C:\Documents and Settings\myfile.txt
- if there is a blank in path, as above, needs quotes!: "C:\ Documents and Settings\ myfile.txt"
- avoid blanks: computers understand blank as a character
- and avoid special chars: all non-letters/numbers, eg $ % &
- special chars have special meaning for a computer!

finding the path

- Win: right-click the file$->$ properties
- Mac: ctrl-left-click the file $->$ get info

paths

- remember to write code that should run on other PCs
- and remember to cd first to desired directory:
- cd ????
- and then eg log using ps1.log
- as opposed to:

```
log using C:\Users\Documents\ASTATA\PS1.txt
```

- that won't run, because I do not have these dirs!
- and it's messy to repeat path for each reading/writing

data for today

- data we use is a subset of general social survey: `http://gss.norc.org/`
- a comprehensive social science data for the US
- whatever you study you are likely to find it in gss
- we will look today at income, education and gender across US regions

data types

- there are dozens of data types/formats/files
- a basic distinction:
  - software-specific binary files (.dta, .sas7bdat, .sav)
  - generic text files (.txt, .dat, .csv, .tab)
- just google it! eg 'stata read csv', 'stata export spss' etc

databases

- but wait, we have databases
- outside of academia, in the real world
- all data are in databases
- Oracle, MySQL, NoSQL, MsSQL, or even MsAccess
- we'll cover it in SQL class in 2nd part of the class
- sometimes you can use Stata
  - and always Python to pull directly from databases

internet

- but wait, we have internet
- popular internet data types: html, xml, json
- we'll discuss internet data when we do Python
- in the 2nd half of the class
- Python is best at dealing with internet data

outline

misc

data types

commenting and syntax

import/export or read/write

make comments in your code

- for each class we will have dofile with Stata code
- make comments in the electronic code files – you will run electronic files not the printout
- if you do not make comments, you will forget...
- use very handy keywords like “TODO”, “KLUDGE”, “BUG”, “LATER”, “FIXME”
  - then can ctrl-f for them in the dofile :)

commenting

- have preamble (notes, install packages, etc)
- `*comment`

```
/*comment
block */
```

stata command syntax and getting help

- <command> <variables> , <options>

  `sum var1 var2, detail`
- <variables> and <options> are optional
- command specific syntax is in help files,

  e.g. `help describe`
- `help` if you know command name, eg `help use`
  - esp options, examples, full pdf help

getting help using gui and google

- gui, eg to load/save, edit data, graphs, etc
- google: "stata"+"what you want to do"
  - eg "stata read excel"
- use google a lot! extremely useful!
- again, ucla website is the best:

  `https://stats.idre.ucla.edu/stata/`

tips

- if you did sth wrong, load data again and start over
  - (replication: you have dofile and can always start over)
- page -up and -down to get previous/next command in command window
- don't memorize commands but reuse and share code
- learn (naturally) abbreviations, e.g. d for describe
  - (they are underlined in help files)

packages/user-written commands

- to get them either google or `findit`;
  - say we want to load spss data e.g. `findit spss` and `help usespss`

outline

misc

data types

commenting and syntax

import/export or read/write

excel

- many people use it and you may need to import from there
- can save as csv and then insheet
- or just use gui to generate the code you need
- in some cases (as here) gui is useful to generate code
  - File-Import-Excel Spreadsheet
  - Worksheet: Cell Range: Import first row as variable names

saving

```
//good
use data1.dta
...
save data2.dta, replace

//bad
use data1.dta
...
outsheet data1.tab //loosing var/val labels,notes

//ugly
use data1.dta
...
save, replace //loosing  code in between
```