data

adam okulicz-kozaryn

adam.okulicz.kozaryn@gmail.com

this version: Monday 11th September, 2023 14:12

outline

misc

data types

commenting and syntax

import/export or read/write

outline

misc

data types

commenting and syntax

import/export or read/write

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

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 pandas read csv', pandas export spss' etc

databases/sql; internet/api

- most data are in databases
  - Oracle, MySQL, NoSQL, MsSQL, etc
  - can use Python to pull directly from databases
- APIs
  - i'd just use API instead of database/sql
  - there is API section towards the end of pandas.ipynb
  - only if for some reason API doesnt work, i'd use database/sql connection–clunky
  - and can also scrap with beautifulsoup etc

outline

misc

data types

commenting and syntax

import/export or read/write

make comments in your code

- for each class we will have ipynb file
- 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 code

commenting

- have preamble (notes, install packages, etc)
- #comment

```
''' comment
block '''
```

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
```