Applied Epidemiology I: Data clearance
A review of using Stata

Enoch Yi-Tung Chen

Department of Medical Epidemiology and Biostatistics, Karolinska Insitutet

October 17, 2020

Acknowledgements

This course material in data clearance is based on my learning from Anastasia Lam's teachings in last year's Applied Epidemiology I lab sessions, and readings from *A First Course in Probability and Statistics* by Goldsman and Goldsman [**?**], *Principles of Biostatistics* by Pagano and Gauvreau [**?**], and *Biostatistics I* by Gabriel and Frumento [**?**].

Outline

1. Get to know the data
   - Summarize
   - Describe
   - Codebook
   - List
2. Managing variables
   - Numeric and string
   - Keep/Drop
   - Label
   - Rename, recode, generate, replace
   - Sort, by, if, in
   - Operators
3. Managing datasets
   - Merge
   - Append

Get to know the data: Summarize

`summarize` gives summaries for all your variables, such as number of observations, mean, standard deviation, etc.

```
. sysuse cancer, clear
(Patient Survival in Drug Trial)

. keep if drug ==1 | drug == 2
(14 observations deleted)

. summarize age  // One variable only (age)

    Variable |        Obs        Mean    Std. Dev.       Min
-------------+---------------------------------------------------
         age |         34    56.41176    6.010686          47
```

Get to know the data: Describe

`describe` gives descriptions for all your variables, such as storage type and labels.

```
. describe  age

              storage   display value
variable name   type    format  label variable label
------------------------------------------------------------------------------
age               byte    %8.0g         Patient's age at start of exp
```

Get to know the data: Codebook

codebook is a combination of summarize and describe and will give a detailed summary of all your variables, including mean, sd, range, percentiles, missing, frequency, etc.

```
. codebook  age

-------------------------------------------------------------------------------------------------------------------
age                                                                    Patient's age at start of exp.
-------------------------------------------------------------------------------------------------------------------

                  type:  numeric (byte)

                 range:  [47,67]                      units:  1
         unique values:  15                       missing .:  0/34

                  mean:   56.4118
             std. dev:   6.01069

           percentiles:        10%       25%       50%       75%       90%
                                49        51        56        61        65
```

Get to know the data: List

`list` lists the observations of specified variables.

```
. list      age if age < 50

     +-----+
     | age |
     |-----|
 12. |  49 |
 15. |  49 |
 18. |  49 |
 25. |  49 |
 33. |  47 |
     +-----+
```

Managing variables: Numeric and string

Numeric: byte, integer, long, float, double – all types of numeric variables that just differ based on min and max length
String: character variables with a certain length (*str#*)

Managing variables: Keep/Drop

s

Managing variables: Label

- `label` helps you keep track of your dataset and variables, and helps others understand your data.
- `label define` to a variable (usually the one you defined)
- `label values` attaches the labels defined using.

```
. label variable drug "1=placebo, 2=mild, 3=strong"
```

Variables

	Name	Label
☑	studytime	Months to death or end of exp.
☑	died	1 if patient died
☑	drug	1=placebo, 2=mild, 3=strong

```
. label define drug 1 "placebo" 2 "mild" 3 "strong"
. label values drug drug
```

	studytime	died	drug
19	22	1	placebo
20	23	1	placebo
21	6	1	mild
22	6	0	mild
23	7	1	mild
24	9	0	mild
25	10	0	mild
26	11	0	mild

Managing variables: Rename, recode, generate, replace

- `rename` changes the name of a variable.

```
. rename died death
```

- `recode` changes variable values.

```
. recode drug (3=4)
```

- `generate` creates a new variable.

```
. generate placebo = 1 if drug == 1
```

- `replace` replaces existing variables (or variable values).

```
. replace placebo = 0 if drug != 1
```

Managing variables: Sort, by, if, in

s

Managing variables: Operators

Operator	**Purpose**	**Example**
==	Evaluates if true/false	summarize if sex==1
~= or !=	Indicates 'not equal'	summarize if sex!=0
<, <=	Less than (equal to) or greater	summarize if age<35
>, >=	than (equal to)	
&	Indicates 'and'	summarize outcome if sex==1 & age>=60
|	Indicates 'or'	gen x=1 if a==1 & (b==1 | c==1)

Managing datasets: Merge

`merge` adds new variables from a second dataset to your existing dataset. (Make the dataset wider)

```
. use cancer_st, clear // cancer dataset contains only studytime an
(Patient Survival in Drug Trial)

. merge 1:1 id using cancer_drug12.dta

    Result                           # of obs.
    -----------------------------------------
    not matched                             0
    matched                                34  (_merge==3)
    -----------------------------------------
```

Managing datasets: Append

append adds new observations to existing variables in your current dataset. (Make the dataset longer)

```
. use cancer_drug12, clear
(Patient Survival in Drug Trial)

. append using cancer_drug3.dta // append patients using drug 3
```

Summary statistics: Operators

`summarize`

Summary statistics: Generate/Replace

```
summarize
```

Summary statistics: Missing

```
summarize
```

Summary statistics: Missing

```
summarize
```