Introduction To Data Management Using Stata: Exercises

This version: April 1, 2010

HINT: Use ‘cd’ (change directory) command to go to the directory with files and avoid typing paths to data (Answers to exercises assume that you are in correct directory).

Exercise 1 : Import/Export

1. Load gss.dta. Use ‘describe’ and ‘summarize’ to see if it looks OK
2. Load gss.csv (you can figure out from **.csv** extension that it is Comma Separated Values, but you would never know if it had extension **.dat** or **.raw**. To figure out delimiter open file in text editor and check out the delimiter)
3. Load gss.tab
4. gss.dat (This is fixed format, and here is dictionary: v1 cols 10-11; v2 cols 20-22, v3 cols 30-32, v4 cols 40-41, v5 cols 50-52, v6 cols 60-62, v7 cols 70-75) Note: v7 is a string variable!

Exercise 2 : Labels

1. Let’s import data using any of the import commands (eg. gss.csv) Do NOT use gss.dta that is already labeled.
2. Rename v1 to marital, label it with “marital status” and label values as follows: 1”married” 2”widowed” 3”divorced” 4”separated” 5”never married”

Exercise 3 : Variables

1. For this exercise use gss.dta.
2. Generate age^2 from age.

3. Generate a divorced/separated dummy variable that will take on value 1 if a person is either divorced or separated and 0 otherwise
4. Generate a variable that is a deviation from income's mean ($x - \bar{x}$)
5. Generate a variable showing average income for each region
6. Change storage type of income variable into string and name it inc_str and then change it back into number and name it inc_num
7. Generate numeric codes for regions

Exercise 4 : Observations

1. For this exercise use gss.dta.
2. Create a new dataset using 'collapse' by region that has mean income, mean happiness, mean education, number of people who are married and number of females. Hint: to get number of married and females first generate respective dummy variables and then use 'sum' option with 'collapse'.

Exercise 5: Combine/Reorganize

1. For this exercise use gss.dta.
2. First let's create id, a unique identifier for each observation, and save two datasets: One with id and region; Another with id and income. Then merge these two datasets.
3. Let's crate another two datasets. One that contains first 50 observations, another that contains the rest observation. And then let's append them.
4. Finally, let's create new variable: income_in_previous_year which is 10% smaller than respective income for this year. Then, reshape dataset to long format on income. Hint: remember to have similar prefix on both, eg. 'inc' and different suffix, eg. '1' and '2'.

Solution to Exercise 1 : Import/Export

1. load gss.dta. Use ‘describe’ and ‘summarize’ to see if it looks OK

```
use gss.dta, clear
save mygss.dta, replace
```

2. load gss.csv (you can figure out from **.csv** extension that it is Comma Separated Values, but you would never know if it had extension **.dat** or **.raw**. To figure out delimiter open file in text editor and check out the delimiter)

```
insheet using gss.csv, clear
save mygss.dta, replace
```

3. load gss.tab

```
insheet using gss.tab, clear
save mygss.dta, replace
```

4. gss.dat (This is fixed format, and here is dictionary: v1 cols 10-11; v2 cols 20-22, v3 cols 30-32, v4 cols 40-41, v5 cols 50-52, v6 cols 60-62, v7 cols 70-75) Note: v7 is a string variable!

```
        infix v1 10-11 v2 20-22 v3 30-32 v4 40-41 v5 50-52 v6 60-62 str v7 70-75
using gss.dat, clear
        edit /*it turns out that first line is missing*/
        drop in 1/*drop the first line*/
        save mygss.dta, replace
```

Solution to Exercise 2 : Labels

1. Let’s import data using any of the import commands (eg. gss.csv) Do NOT use gss.dta that is already labeled.

2. Rename v1 to marital, label it with “marital status” and label values as follows: 1”married” 2”widowed” 3”divorced” 4”separated” 5”never married”

```
insheet using gss.csv, clear


ren v1 marital
la var marital ”marital status”
la def marital_lbl 1”married” 2”widowed” 3”divorced” 4”separated”
5”never married”
la val marital marital_lbl
codebook marital
```

Solution to Exercise 3 : Variables

1. For this exercise use gss.dta.

   ```
   use gss.dta, clear
   ```

2. Generate age^2 from age.

   ```
   gen age2=age∧2
   ```

3. Generate a divorced/separated dummy variable that will take on value 1 if a person is either divorced or separated and 0 otherwise

   ```
   tab marital
   tab marital, nola

   gen divsep=.
   replace divsep=1 if marital==3|marital==4
   replace divsep=0 if marital<3 |marital==5
   ```

   or use recode:

   ```
   recode marital (1 2 5=0) (3 4=1), gen(divsep)
   ```

4. Generate a variable that is a deviation from income's mean $(x - \bar{x})$

   ```
   egen avg_inc=mean(inc)
   gen dev_inc=inc-avg_inc
   ```

5. Generate a variable showing average income for each region

   ```
   bys region: egen mean_income=mean(inc)
   ```

6. Change storage type of income variable into string and name it inc_str and then change it back into number and name it inc_num

```
tostring inc, gen(inc_str)
destring inc_str, gen(inc_num)
```

7. Generate numeric codes for regions

```
encode region, gen(region_numeric)
```

Solution to Exercise 4 : Observations

1. For this exercise use gss.dta.

```
use gss.dta, clear
```

2. Create a new dataset using ‘collapse’ by region that has mean income, mean happiness, mean education, number of people who are married and number of females. Hint: to get number of married and females first generate respective dummy variables and then use ‘sum’ option with ‘collapse’.

```
recode marital (1=1) (2 3 4 5 =0), gen(married)
recode sex (1=0) (2=1), gen(female)
collapse age educ inc happy (sum) married (sum) female, by(region)
```

Solution to Exercise 5: Combine/Reorganize

1. For this exercise use gss.dta.

```
use gss.dta, clear
```

2. First let’s create id, a unique identifier for each observation, and save two datasets: One with id and region; Another with id and income. Then merge these two datasets.

```
gen id=_n
keep id region
save gss1.dta, replace

use gss.dta, clear
gen id=_n
keep id inc
merge id using gss1.dta, sort
tab _merge
```

3. Let's crate another two datasets. One that contains first 50 observations, another that contains the rest observation. And then let's append them.

```
use gss.dta, clear
keep in 1/50
save gss1.dta, replace

use gss.dta, clear
keep in 51/l
append using gss1.dta
edit
```

4. Finally, let's create new variable: income_in_previous_year which is 10% smaller than respective income for this year. Then, reshape dataset to long format on income. Hint: remember to have similar prefix on both, eg. 'inc' and different suffix, eg. '1' and '2'.

```
use gss.dta, clear
gen inc1=inc
gen inc2=.9*inc
drop inc /*need to drop this one because we want to have only two inc variables*/
gen id=_n
reshape long inc, i(id) j(period)
edit
```