Exercise 5: Essential Data Processing

Launch Stata and begin by creating a new do-file called *Stata_Exercise5.do*. Add commands at the beginning to change the working directory to the Exercise 5 folder and load the *bl_combined1.dta* dataset. Remember to keep saving the do-file as you go along. Run through these exercises referring to chapter 6 in the module notes.

- ***Exercise 5.1:*** `generate` and `replace`

For each task check the distribution of the new variable and add a variable label:

  - Create a new variable named *pp* which is pulse pressure (Note: pulse pressure is defined as the difference between systolic and diastolic blood pressure).
  - Create a new variable named *bmi* which is body mass index (Note: BMI is defined as weight in kg divided by height in metres squared). Are there are any missing values. Why are the values missing?
  - We noted that *wc* contains measurements in metres and centimetres. Create a new variable named *wc_cm* containing waist circumference measured in centimetres.
  - Create a new variable named *obese* which is 1 if BMI>30 and 0 otherwise. Think carefully about the missing values.
  - Create a new variable named *hrate90* which is 1 if heart rate is greater than or equal to 90 BPM and 0 otherwise.
  - Create a new variable *hyper* which takes values 1 if *sbp*>140 or *dbp*>90. Again be careful if there are any missing values.
  - Create a new variable called *log_egfr* which is the natural log transformation of *egfr*.
  - Create a new variable called *log_bl_totbil* that is the natural log transformation of *bl_totbil*. Do not include the numeric missing values (8888 and 9999) in the transformation.

- ***Exercise 5.2:*** `destring` or `encode`

Convert the following variables from string to numeric. For each variable you need to decide which command (`destring` or `encode`) is appropriate.

  - *sex*
  - *lvef*
  - *race* (use order of White, Asian, Black, Other rather than alphabetical order)
  - *smoke* (think what might be a sensible order for the categories)

- ***Exercise 5.3:*** `recode`

  - The variable *bl_creat* has missing values recorded as real numbers 8888 and 9999. Recode these to Stata missing values (8888 to .a and 9999 to .b).
  - From *sbp* create a categorical variable *sbpcat* which has values 0, 1, 2 and 3 which mean <130 mmHg, 130-139 mmHg, 140-149 mmHg and 150+ mmHg respectively.
  - Create a variable *evsmoke* (Ever smoker) which takes the value 0 if the patient is a never smoker and 1 if the patient is an ex- or current smoker.

- Create a variable age70 which is 1 if age is 70 or above and 0 otherwise.
- The baseline medication variables (*diur*, *asp*, *arb*, *bblock*, *digox*) take values 1 and 2 where 2 is yes and 1 is no. Recode the values for these variables so that yes is 1 and no is 0. Try using the `foreach` command with `varlist` option.

- ***Exercise 5.4:*** `egen` with the `cut`, `rowmax`, or `rowtotal` functions:
  - Create a new categorical variable *bmi5* which cuts *bmi* in to 5 equal sized groups.
  - Create a new categorical variable *sbp5* which cuts *sbp* in to 5 equal sized groups.
  - Check the distribution of these two new variables. How well has the categorisation in to equal sized groups worked? Why has it not worked so well for *sbp*?
  - Create a new categorical variable *bmicat* which cuts *bmi* using the cut-points 22, 25 and 30. Use 12 and 60 as the minimum and maximum acceptable values.
  - Create a new variable *hrcat* which categorises *hrate* with cut-points at 60, 70, 80 and 90. Use values of 40 and 180 as the minimum and maximum acceptable values.
  - Create a new variable *stroke* which is 1 if the patient had any type of stroke (if any of *strisch*, *strhem*, *stremb* or *stroth* is yes) and 0 otherwise.
  - Create a new variable *strokemiss* which contains the number of missing values among the variables *strisch*, *strhem*, *stremb* and *stroth*.
  - Create a new variable named *cvd* (cardiovascular disease) which is equal to 1 if any of *miprev, angina, cabg, pci,* or *stroke* are 1 and 0 otherwise.
  - Create a new variable called *cvdnonmiss* which contains the number of non-missing values among *miprev, angina, cabg, pci,* or *stroke.*
  - Create a new variable *numdrgs* which is equal to the number of drugs (*diur*, *asp*, *arb*, *bblock*, *digox*) that each patient is taking.

- ***Exercise 5.5:*** `mvdecode` and missing values:
  - For the variables *bl_hb, bl_pot, bl_sodium* and *bl_totbil* change the numeric values 8888 and 9999 to .a and .b respectively.
  - Use the `misstable` command to investigate the number and patterns of missing values for the variables *bl_creat, bl_hb, bl_pot, bl_sodium* and *bl_totbil*.

Add a command to save the modified dataset as *bl_combined2*.