Project 1: Ambulatory Care Medical Data

McCourt School of Public Policy, Georgetown University

- This project will be on-going for the next three weeks
- Remember to save your do-file each week so you can pick up where you left off

---

Week 1: Review

- Review Stata commands from last semester
- Practice working on a large project in a do-file

Questions

1.1. Download the data set and documentation from Blackboard. Save the data set and documentation in a new folder dedicated to this project. As our project gets more complex, it will be important to keep all related files in the same folder. Take a few minutes to look at the documentation before opening the data set. - What does one line of the data set represent? - What types of variables are in this data set?

1.2. Open Stata and start a new do-file for this project - The first command in your new do-file should change the working directory to your dedicated project folder. - You can get this command from the drop-down menu `File >> Change Working Directory`. - Add a `use` command to open up the data set. - Save your do-file and make sure it runs before moving on.

~~~
```
cd "C:\Users\hoya99\Desktop\project1"
use NAMCS2010.dta, clear
```
~~~

> Tip: Another command you may want at the beginning of your do-file is `set more off` (why?).

1.3. Use your basic descriptive commands to answer some questions about the data. These questions can all be answered with `summarize` , `tabulate` , and `histogram`. You might have to use some options, which are described on the help page for each command. The commands that you use to answer these questions should be in your do-file. - What is the average age of patients in this data set? - What is the median value of age in this data set? - What is the most common age of patients in this data set? - In descending order, what are the five most common ages of patients in this data set? - Generate a histogram showing the distribution of the age of patients in this data set. - Generate a second histogram treating age as a discrete variable and add a normal density line to the graph for comparison. - Does it look like age is distributed normally in this sample? - Does `AGE` have any observations with values of missing (`.`)? - What is the breakdown of patients by race? - Is there a relationship between sex and race?

> Tip: Some people don't like typing capital letters all the time in Stata. Use the command `rename * , lower` to change all variable names to lower case. For more advanced uses of rename type `help rename group`.

1.4. We are interested in the relationship between tobacco use, weight, and blood pressure. Young children generally do not use tobacco, so we want to exclude them from this analysis. - We want to create a sample composed of patients age 18 and older. - Write a command to drop patients if their age is less than 18 years old. - (Alternatively, keep patients if their age is 18 years old or greater.) - Your data set should only have 3,885 observations after this step. - For more information, see `help keep`.

1.5. We want to learn about the height and weight of patients in this data set. Find the two variables that give "Height in inches" and "Weight in pounds". hint: Don't scroll through the variable window searching for these variables. Instead you can - Use the Filter Bar on the variable window - Use the `lookfor` command to search variable names and labels - Search the documentation

1.6. We want to learn more about the maximum and minimum values of the weight and height variables. - These are both labeled numeric variables. - How can you tell that they are labeled numeric variables? - You need to find the underlying numeric value of "Blank" before you can replace it with Stata missing. - Try running a tabulate command without value labels. - To see all the labeled values for a variable, first find the label name with `describe`. - Then get the label values with `label list`. - For both weight and height, recode the "Blank" entries to Stata missing. - What are the maximum and minimum values of the weight and height variables?

> In the future, you should check every variable that you use for numeric values that should be recoded to Stata missing. You can recode numeric values to Stata missing for multiple variables simultaneously using `mvdecode`

1.7. Variable transformations - For BMI, recode "Missing data" or "Not calculated" as Stata missing value `.` - Use a natural log function to create three new variables containing the logs of height, weight, and bmi. - For more information on using functions and function names, see `help functions` - Label your new variables appropriately. - Summarize the three original and three log variables in a single table. - For each variable, verify that the original and log versions have the same number of observations.

1.8. If-statements and indicators - Create a new binary variable equal to one if a person is a current tobacco user. - Use a two-way tabulation to verify your variable creation. - Make sure your two-way tabulation displays missing values. - Create a new binary variable if a person is overweight, defined as having bmi of 27 or greater. - Verify your variable creation with another two-way tabulation, displaying missing values. - Create and verify a final binary variable equal to one if a person is both a current tobacco user and overweight. - How many patients in this data are both overweight and current tobacco users?

1.9. Regressions and testing - Run a regression of systolic blood pressure on age, height, weight, and bmi. - Verify that this regression uses 1,464 observations. - How would you test the null hypothesis that the coefficient on weight is zero in this regression? - Another way to run tests after a regression uses the `test` command. - Try testing each of the individual coefficients from the regression. - Compare the resulting p-values. - `test` is a postestimation command, and will always use the results from the most recently run regression.

1.10. Challenge question: - The equation to calculate BMI is (weight in kg) divided by (height in meters squared). - Create a new variable calculating BMI from the component height and weight variables. - How does your new variable compare to the existing BMI variable? - Create a new variable showing the difference between the original bmi variable and your new calculation. - Examine these three variables in the browse window: `browse bmi bmi2 diffbmi` - Does your calculation match the existing variable? - What are the maximum and minimum values of diffbmi? - Generate a histogram of diffbmi. - The original bmi variable only contains integer values. - Use a math function to make your version of bmi match the original variable. - Replace diffbmi with the new difference in variables. - Do they match?

> You can compare two variables without generating a third, difference variable: `compare bmi bmi2`