Lab 1 – 07/10/19 – INTRODUCTION TO STATA

PART 1

INTERFACE

We start by exploring the interface of Stata. It is composed by a few windows:

Results: it displays the results of our analysis
Command: window where we can enter our commands
History: window with a list of all the previously used commands
Variables/Properties: windows describing the variables of the data set under analysis and their properties
Project manager: tool for organizing and navigating Stata files

There are **three ways** to interact with Stata:

1. **Menu system and dialog boxes**
2. **Command window**
3. **Do-file editor**

MENU SYSTEM AND STATA SYNTAX and COMMAND WINDOW

We start first by working with the menu system and dialog boxes, which will be helpful in learning some Stata syntax as well. This is the easiest approach to use at the beginning. In order to see some examples, we start by uploading some data set. We do this by typing, in the **commands** window, the command

```
webuse fuel
```

which allows us to load one of the data sets (in this case the one called *fuel*) available at the url

https://www.stata-press.com/data/r16/.

Do you want to know more about the command *webuse*? (or in general about any command) Type in the command window

```
help webuse
```

Once the data set is loaded we will see in the **variables** window the list of variables composing the data set (in this case mpg1 and mpg2). By clicking on each variable we can see and edit its properties on the **properties** window.

Now that a data set is loaded we can first of all visualize the data by using the data editor icon. Two options are available:

data editor (edit) – which allows us to edit entries of the data set data editor (browse) – if we do not want to risk to edit the data by mistake

Next we want to run the function *summarize* to produce a summary of the variables of the data set *fuel*. We can do this by following the path

Statistics tab => Summaries, Tables and Tests => Summary and descriptive statistics => Summarize

Press the "?" icon at the bottom left of the page to have more info on the command. We find syntax, options, examples. If want to have more information on the same command we can click on the link to view the complete PDF manual (which contains more worked examples as well).

Once we open a function (such as *summarize*) we can use the dialog box to choose the options we want to apply and run the function. The dialog box will create the command for us. We first choose the *standard display* option in the dialog box and observe that in the **results** window the command used was

```
summarize
```

We try then to type the same in the command windows and obtain the same result, which is a summary of the whole data set.

We repeat the same procedure by using the menu system and select the option *display additional statistics*, in the **results** window we can see now a richer list of summaries for the *fuel* data set. We also learn that the syntax for obtaining this is

```
summarize, details
```

What comes after a comma in Stata syntax is known as *option* for the main command.

If we want to produce summary only for one (or some) variables of the data set, we can instead use the command

```
summarize mpg1
```

```
summarize mpg2
```

```
summarize mpg1 mpg2
```

and if we want to visualize additional summaries for the same variables we will add the *detail* option to the same commands

```
summarize mpg1, detail
```

```
summarize mpg2, detail
```

```
summarize mpg1 mpg2, detail
```

By looking at the menu system, we can see that there exists a huge variety of statistical commands beside the one we considered for this first example. JA second example we try to produce a boxplot for the variables of the data set *fuel*. We again use the menu system and follow the path

Graphics => Boxplot

In the dialog box we chose the orientation of the plot and the variable (e.g. horizontal and mpg1). Then we produce the plot and see, in the **commands** window, that the corresponding command is

```
graph hbox mpg1
```

Once we have learnt this syntax we input a new command to produce vertical box plot (box rather than hbox) to compare the two variables

```
graph box mpg1 mpg2
```

Do we need to recall a function that we have used before? We can give a look in the **history** window. If we want to re-run the same command we can simply click on it.

IMPORTING DATA SETS

To run our first example we ran the command

```
webuse fuel
```

What other data sets are available in the Stata repository?

If use the menu system and follow the path

File => Example datasets

we see that available data sets are collected under two main headings. First those installed with Stata, second those which are not installed but are made available with the Stata manuals which are available online. We can try to click on both headings and when we locate the relevant data set we can press *describe* or *use*. Importantly, for the data sets associated with Stata manuals we need an internet connection as they are not already installed but they need to be downloaded. The syntax corresponding to the actions we just did by using the menu system are

```
sysuse auto // for the data sets which are already installed with Stata
webuse surface // for the data sets available with Stata manuals
```

Next, we want to work with a data set which is not already available in Stata. The data set we consider for this example is available (together with many others) at the University of California Irvine repository at the link

https://archive.ics.uci.edu/ml/datasets.php

The data set is related to the two-year historical log corresponding to years 2011 and 2012 from Capital Bikeshare system, Washington D.C., USA, and contains also weather and seasonal information.

You can find the file in the e-learning page of the course: it is in .xlsx format and it is composed by two sheets (*by hour* and *by day*) referring to the same data aggregated either by the hour or by the day. You will also find a text file with a description of the data set and of its variables.

Once the data set is downloaded and stored in a folder we have three ways to load it in Stata.

1. By doing copy and paste (not recommended, specially if the data set is heavy)

We do this for the data aggregated by the day.

2. By importing the file (and by specifying the format of the original file) with the menu system

We do this for the data aggregated by the hour. We use the menu bar and follow the path

File => Import => excel spreadsheet

We select the data set in the folder where we stored it and follow the dialog box to make sure we import it as we want (in this case we have to check *import first row as variable names* and make sure we are importing the *by hour* sheet).

When we do this, we can learn the syntax from the **results** window. This suggests the third way to import the dataset.

3. We can import the file directly from the **command** window by typing

```
import excel "/Users/bernardo/Documents/teaching/BICOCCA/advanced statistics/labs/LAB 1/datasets/Bike-Sharing-Dataset/bike-sharing-data.xlsx", sheet("by hour") firstrow clear
```

where, clearly, the location of the file in your computer must be changed.

Rather than dealing with the complete path to the dataset, it might be convenient to change the working directory. First of all, to figure out the current working directory, use the command

```
pwd
```

Recall that if you save a file without specifying any path, Stata will save it in the working directory.

To change directory we can use the Menu system and follow the path

File => Change working directory

and select the folder where we store the data set. Once this is done, in order to import the data set we can avoid specifying any path and use the shorter command

```
import excel “bike-sharing-data.xlsx", sheet("by hour") firstrow clear
```

We might want to edit or clean the dataset. As an example we are going to add a new variable which counts, at each hour, the proportion of casual rental bikes over the total number of rental bikes (casual + registered). We follow the path

Data => Create or change data => create new variable

and then, within the dialog box we specify how we want to define the new variable and its type. In this case we can choose a *float* type, call the new variable *prop_casual* and define it as *casual/cnt.* The corresponding syntax is

```
generate prop_casual = casual / cnt
```

We can also change the properties of the variables by using the **properties** window. For example, if we want to display less decimal digits for the newly created variable *prop_casual*, we can select it, click on the three dots on the right to open the dialog box and choose a *fixed numeric type* with two digits right of decimal. The corresponding syntax is

```
format %9.2f prop_casual
```

We can then save the data set in .dta format.
At the beginning of this quick introduction we mentioned three ways to interact with Stata. So far we have used the **Menu system and dialog boxes** and **Command window**. Next, we are going to talk about the third one, which is the **Do-file editor**.

USING DO-FILES

Do-files are scripts which store sequences of commands that will be executed in order. They have some clear advantages over using the Menu system or the **command** window.

- If you have to use repeatedly the same commands (for example on different datasets), then a Do-file might save a lot of time.
- They help keeping track of how exactly a data set was processed.
- It is easy to share your work with other people.

We are going to start a new Do-file by clicking on the Do-file editor icon. In order to store in the Do-file the commands we have already run to import and edit the data set, we can copy such commands from the **history** window and paste them on the Do-file.

PART 2 – DO IT ON YOUR OWN

For this part you can either work alone or in small groups.

The goal of this part is for you to import, edit and analyse a dataset. You might use the Menu system but the ultimate goal is to produce a working Do-file which executes all the needed commands.

The dataset, called wine_quality_red, is related to the red variant of the Portuguese "Vinho Verde" wine. It is available at the University of California Irvine repository and it can be downloaded from the e-learning page (this time is a .csv file).

Here is a list of tasks you should complete.

1. Import the data set (make sure the right delimiter is selected)
2. Display an overall summary of the dataset
3. Display a summary, with more details, for the variable alcohol
4. Produce a histogram for the variable alcohol
5. Produce a scatter plot (twoway grahp – scatter) by selecting sulphates as x-variable and alcohol as y-variable
6. Produce a pie chart for the variable quality.