Stats Bootcamp – Part 1
Using Stata and R for Data Analysis

Federico Vegetti

Central European University

fede.vegetti@gmail.com

University of Zagreb

1-2 February 2016

Goals for these 2 days

1. Understand the interface of Stata and RStudio
2. Familiarize with Stata and R syntax
3. Learn how to read, compute, transform and write data with Stata and R
4. Learn how to explore and visualize data with Stata and R
5. Learn how to figure things out when you reach an obstacle

Structure

- 6 learning sessions, 1 wrap-up/troubleshooting session
- A bit of "frontal" lecture, a lot of practice
- Day 1: We get accustomed to using statistical software for data analysis, with a focus on Stata (easier)
- Day 2: We dip into R (a bit more difficult, but more rewarding)

Software

- Stata (ask in case)
- R (`http://www.r-project.org/`)
- RStudio (`http://www.rstudio.com/`)

Data Analysis

What is data analysis?

- Data analysis is the set of operations that we perform to **learn** something from the data
- Learning can be:
  - **Deductive**: we have a theory, we need evidence to support it
  - We want to learn whether our understanding of a *general* phenomenon might be correct
  - E.g. Academic research
  - **Inductive**: we want to discover patterns in the data
  - We want to learn about the presence of a *specific* phenomenon we previously ignored
  - E.g. Data mining

Quantitative Data Analysis

- The phenomenon we care about is measurable
- We measure it many, many times (hence the term "quantitative")
- We use statistics to summarize all the measurements
- Since we are not calculators, we need software to perform statistics

The Data

- Our stored, repeated measurements are what we call "data"
- They are organized in 2 dimensions:
  1. **Observations**: the instances of measurement, what our theory focuses on (e.g. individuals, political parties)
  2. **Variables**: the phenomena we are interested in (e.g. policy position, vote choice)
- This structure (observations x variables) is called "matrix"

Data Matrix

	yrsurv	country	weight	weight_l	weight_a	gender	age	age9c	hhsize	equalinc	nbgoodc	nbstatus
1	2009	United S	1.781835	1.47237	1.710366	Female	29	25-34	5	No	Yes	Yes
2	2009	United S	1.147147	.9479129	1.12215	Female	52	45-54	3	.	.	.
3	2009	United S	.6465383	.	.7453732	Female	66	65-120	2	No	No	Yes
4	2009	United S	.6419132	.530427	.6566575	Female	64	55-64	1	No	Yes	Yes
5	2009	United S	.4885322	.4036849	.5856816	Male	58	55-64	1	.	.	.
6	2009	United S	1.008012	.8329423	.9696192	Male	48	45-54	5	.	.	.
7	2009	United S	.4528215	.	.4027303	Male	80	65-120	1	.	.	.
8	2009	United S	1.008012	.8329423	.9696192	Male	50	45-54	2	.	Yes	Yes
9	2009	United S	.4821386	.	.4288043	Male	65	65-120	1	Yes	Yes	Yes
10	2009	United S	1.739558	1.437435	1.689984	Female	31	25-34	5	.	.	.
11	2009	United S	.4528215	.	.4027303	Male	85	65-120	2	.	.	.
12	2009	United S	.8386272	.6929762	.8747435	Male	52	45-54	3	.	.	.
13	2009	United S	.5832038	.4819142	.6120074	Male	56	55-64	2	Yes	No	No
14	2009	United S	.5686522	.4698899	.6399806	Female	64	55-64	1	Yes	No	No
15	2009	United S	.8521174	.7041234	.8789882	Male	47	45-54	4	No	Yes	No
16	2009	United S	.5179352	.	.4606411	Male	67	65-120	2	.	.	.
17	2009	United S	.9887486	.8170248	.9784866	Male	54	45-54	2	No	Yes	Yes
18	2009	United S	.9506454	.7855393	.9721146	Female	52	45-54	1	Yes	Yes	No
19	2009	United S	1.43613	1.186706	1.342454	Male	37	35-44	4	.	.	.
20	2009	United S	.6419132	.530427	.6566575	Female	60	55-64	1	No	Yes	Yes
21	2009	United S	.5856579	.	.6751862	Female	80	65-120	1	Yes	Yes	Yes
22	2009	United S	1.782808	1.473173	1.687814	Female	39	35-44	4	Yes	Yes	Yes
23	2009	United S	1.008012	.8329423	.9696192	Male	47	45-54	1	.	.	.
24	2009	United S	.5179352	.	.4606411	Male	77	65-120	1	.	.	.
25	2009	United S	.7439813	.6147682	.7191215	Male	57	55-64	1	.	.	.
26	2009	United S	.5885136	.	.6784784	Female	77	65-120	1	.	.	.

How Stata and R can help us

- Stata and R do (mostly) the same job
  - They are both used to explore, manipulate, summarize, visualize, and model data (AKA, do data analysis)
- However, they are very different
  - Stata makes it very easy to do a lot of things, but it's not too flexible when tasks get more advanced
  - R requires a lot of basic skills to do even the simplest tasks, but once you master it it will bring you anywhere

Coding

- Whether you use Stata or R, you will need to write **code**
- What does "code" mean?
  - Write your commands in a language that the software can understand, called "syntax"
- If you are serious with data analysis, you can't escape using syntax
- But it's a common struggle, and there are many online resources to help you
- Stata's syntax is easier than R's syntax

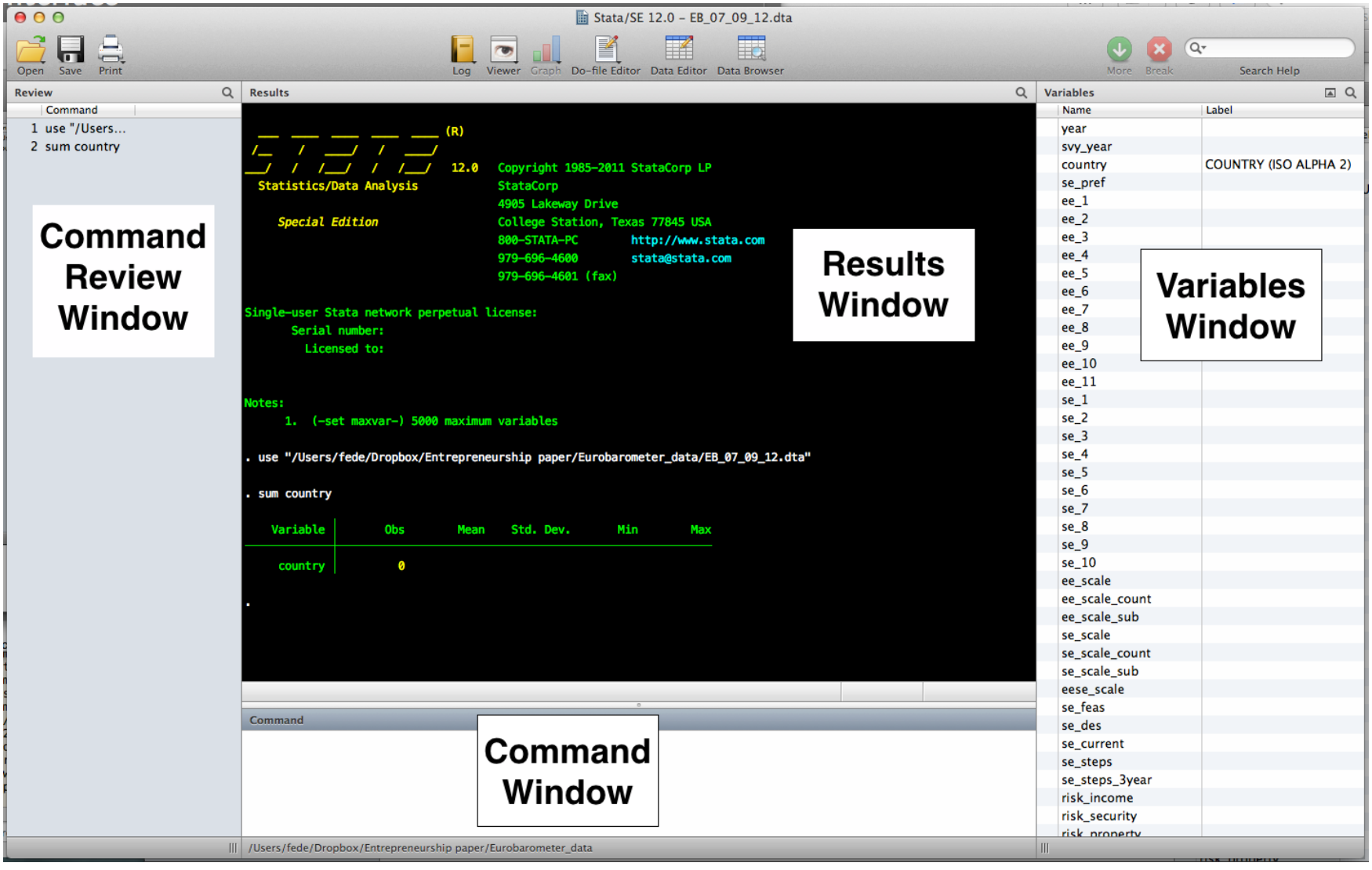
Some online resources

- Stata
  - Official manual (very well organized and accessible online)
    `http://www.stata.com/features/documentation/`
  - Statalist (several tips and tricks)
    `http://www.statalist.org/forums/`
  - UCLA (useful statistics course done in Stata)
    `http://www.ats.ucla.edu/stat/stata/modules/`
- R
  - CrossValidated (Q&A for stats/data analysis)
    `http://stats.stackexchange.com/`
  - Stack Overflow (Q&A for programming)
    `http://stackoverflow.com/`
  - R-Bloggers (some interesting applications)
    `http://www.r-bloggers.com/`
  - A million others
- Either way, Google is your friend!

About Stata

- Probably the most user-friendly statistical software
- 4 versions
  - MP: up to 20 BN observations, parallel processing
  - SE: 2+ BN observations, 32,000+ variables
  - IC: 2+ BN observation, 2,000+ variables
  - Small Stata: 1,200 observations, 99 variables
- SE/IC are the most common
- Can handle point & click as well as complex syntax
- Can work with matrix programming language (Mata) at the cost of increased difficulty

Stata interface



Stata file types

- `.dta`: a data file
- `.do`: a syntax file
- `.gph`: a graph produced by Stata
- `.smcl`: a record of your session

do-files

- Most of what you do in Stata, is done using "do-files"
- do-files are empty text files where you write the same code that you would write in the command window
- However, you can write more complex, multi-line programs
- Moreover, you can save your syntax and replicate your data manipulations/analyses in the future
- Everything that you do to the data and with the data should be done through do-files
- Working with do-files is often called "batch mode"

Stata syntax

Commands in Stata are like in the English language

- First you have the verb: "*Buy...*"
- Then you have the object: "*...the milk...*"
- Then you can specify the conditions under which the command applies: "*...if you leave the office before 7*"
- Moreover, you sometimes use adverbs to specify how an action should be performed: "*Come, quickly!*"

Stata syntax

Beside a few exceptions (that we will see) Stata syntax is divided in four parts:

1. *Command*: What action do you want performed?
2. *Names of Variables, Files, or other Objects*: On what things is the command to be performed?
3. *Qualifier on Observations*: On which observations should the command be performed?
4. *Options*: What special things should be done in executing the command?

The only part that has to be there all the time is obviously the first. In most of the cases, the second part will be there as well.

Learn by doing!

- The rest of the class (besides a few exceptions in the following slides) will be practical
- We will be working on the file called "*script.do*" that you can find in the folder "*Stata_intro*"
- We will use data from the Comparative Manifesto Project (CMP)
- Our main goal is to understand how Stata syntax works, and how we can explore and manipulate datasets

The CMP data

- CMP is a very popular dataset among political scientists
- The CMP project analyses the content of **party manifestos**, and turn it into quantitative data
- The information in the CMP data tells us how much parties emphasize different topics in their program
- Currently it covers 55 countries, with observations going back to the 1920s
- The unit of observations are **political parties**
- The codebook in the folder will help you understand what the variables are about

Variable types in Stata

Stata deals with 2 types of data: **numeric** and **string**.

Numeric data

- byte: integer between -127 and 100
- int: integer between -32,767 and 32,740
- long: integer between -2,147,483,647 and 2,147,483,620
- float: real number with about 8 digits of accuracy
- double: real number with about 16 digits of accuracy

The default when you create a new variable is "float".

Variable types in Stata

- A "string" variable contains characters – like a word or an alphanumeric code.
- You can not perform calculation with string variables
- Mostly used to define categories in your data
- The variable type defines the maximum number of characters it can contain.

String data

- `str1`: 1 character
- `str2`: 2 characters
- `str2045`: 2045 characters
- `strL`: 2000000000 characters

Common Operators

Arithmetic	**String**	**Relational**	**Logical**
+ addition	+ concatenation	> greater than	& and
− subtraction	*n duplication	< less than	| or
* multiplication		>= greater or equal	! not
/ division		<= less or equal	
^ raise to power		== equal	
		!= not equal	

From Stata to R

Similarities between Stata and R

- Both require you to learn their language in order to be used at all (R) or properly (Stata)
- Both allow you to transform, visualize, and model your data
- Both allow you to save your syntax and keep it for future replication
- Both have many user-written extensions that enhance what they can do (R has more)
- Both can produce graphs of publishable quality
- Both have great online support

From Stata to R

Why Stata

- Stata language is more intuitive than R: it makes it very easy to perform basic to rather advanced statistical tasks

Why R

- R is able to perform *a lot* of statistical analyses that are not available in Stata
- R is more versatile
- R is one of the standards for data science in the private sector
- R is free

Features of R

- R language is **object oriented**
  - Objects are entities identified by a name and a content
  - You can put many different things into objects numbers, words, datasets, functions, graphs, etc.
  - Think of Stata's "macros", but more general
- R can load data written by Stata, SPSS, Excel, and whatever other data format is available
- R can be used through several different graphical user interfaces (GUIs). We will focus here on RStudio

Installing R on your computer

`https://www.r-project.org/`

The CRAN Comprehensive R Archive Network

- `https://cran.r-project.org/mirrors.html`
- From here you can download the installer and most of the additional packages
- You may have to choose a *mirror* i.e. a server located possibly close to where you are

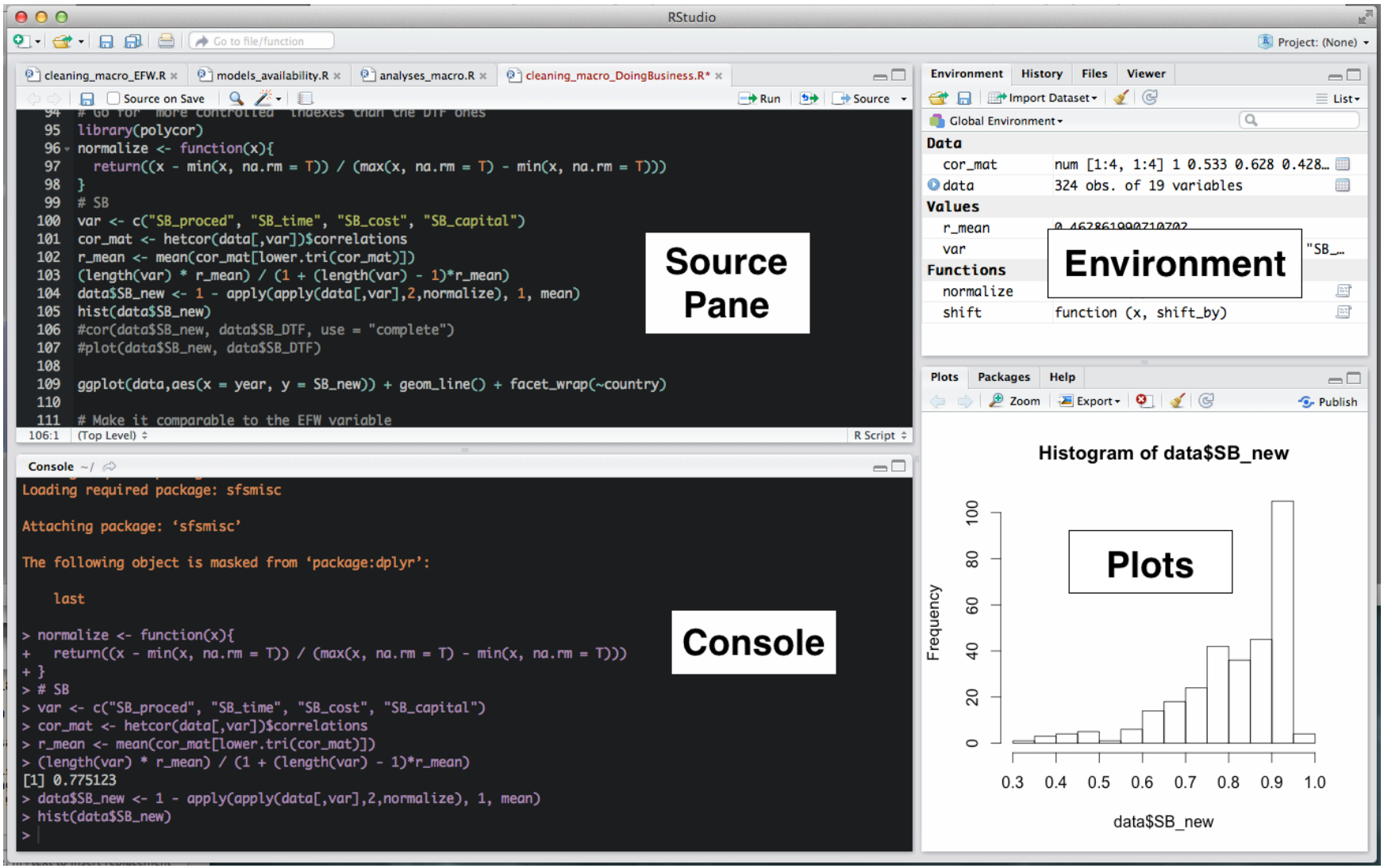
The site also provides a number of manuals and help facilities:
`https://cran.r-project.org/manuals.html`

Installing RStudio on your computer

`https://www.rstudio.com/products/rstudio/download/`

- You need to download and install R separately
- For the current version of RStudio to run, you will need R version `2.11.1` (or higher)

RStudio interface



R file types

- `.r`: a syntax file
- `.RData`: the "workspace", i.e. all the data, functions and other objects that you have created
- `.Rhistory`: the "history", i.e. all you have been doing during the current session

Learn by doing!

- The rest of the class will be practical
- We will use again data from the Comparative Manifesto Project (CMP)
- We will replicate many things that we have done with Stata, and some more
- Our main goal is to get a basic understanding of how R works, and be able to perform basic tasks

Variable types in R

R can handle several different data type. The ones that you will encounter more frequently are:

- **numeric**
- **characters**: strings of characters identified using quotation marks: ‘ ‘ ’ ’.
- **factors**: nominal values, that define categories rather than actual numbers
- **logical**: boolean values, i.e. `TRUE` and `FALSE`