Introduction to Programming
Syllabus

Master in Economics, Sciences Po
Spring, 2017-2018

Hugo Lhuillier*

1   Course description

The main purpose of this course is to introduce you to computer programming in a practical way. By the end of the course, you should be able to solve problems by writing computer code from scratch. To achieve this goal, the course will sometimes be very applied – how to write a function that counts the prime factors of x, and sometimes more theoretical – how can your computer even possibly understand this function.

Although this course is mainly targeted to beginners, people with little or no previous experience with computer programming, the level of complexity will steadily rise throughout the semester, such that even people with intermediate knowledge might find some interest in the class.

We will focus on two programming languages: `R` and `Julia`. The first part of the course will be based on `R` due to its user-friendliness. Moreover, `R` is fantastic for people working with data. Towards the end of the semester, we will switch to `Julia` to cover more advanced topics in programming. The two languages are relatively similar, so going from one to another should not cause any trouble.

Note that this course is not on numerical methods, nor on applied econometrics, even though we might deal with econometrics example.

2   Prerequisites

Little to no programming experience is required for this course. However, you will need to come to class with your laptop, and some softwares installed on it:

- R (v.3 or later) and RStudio

- Julia (v0.6)

- Git

---

*Do not hesitate to contact me at hugo.lhuillier@sciencespo.fr

- For the Windows lovers: a Unix emulator (like Gow or Cygwin)

Please do not install those for now, unless you absolutely know what you are doing. I will get back to you before the semester starts with precise instructions on how to install these softwares. Additionally, I will be holding some office hours to help you with those installs.

3   Grading

This course will be graded on a pass / fail basis, even though you will not receive any credit for it. The grading will be based on

- weekly problem sets, that should be completed as much as possible alone,

- a four-hour hackathon.

Copying is very easy in this type of class. However, note that

- knowing how to program efficiently is key in nowadays research, and this can be learn only by practicing frequently,

- professors in future classes will expect you to master the content of this course.

4   Expected schedule

The course will be made of three sections of unequal lengths

1. **Introduction**. How a computer "thinks"? How to interact with it? How to work collaboratively on programming projects?

2. **Programming: the basics in R**. Introduction to data types and structures, logical operators, functions, data analysis, plots, input/output, etc.

3. **Advanced programming in Julia**. Introduction to type systems, package management, code profiling and optimization, etc.

The hackathon will take place during the last two lectures.