Programming for Economists: Syllabus

Tyler Abbot

July 1, 2016

1 Course Description

This class will aim to provide first year graduate students with the tools necessary to perform basic computation. Given the broad range of available tools, this course will seek to provide in depth instruction in *programming* instead of a programming language. That being said, we will work mainly in R and introduce Julia if time permits.

By the end of the course students should know how to design pseudo-code to outline a problem solution, read and write basic programs in R, know how to run and debug basic programs in R, should have R and Julia installed on their computers, and most importantly should feel comfortable tackling a computational problem from start to finish, from algorithm design to choosing the most effective tools to sharing the results.

Teaching Objectives:

- Introduce students to the basic workings of a computer: logic, loops, memory, processing, etc.
- Provide elementary practice in the programming process:
    1. Problem definition.
    2. Algorithm design.
    3. Pseudo-code generation.
    4. Language selection.
    5. Coding.
    6. Optimization.
    7. Results presentation.
- Help students to gain a basic knowledge of the use of R and Julia.

2 Course Requirements

Little to no prior programming knowledge is required. All texts and resources will be readily available online. Students should come to class with a laptop capable of running open source software and with plenty (at least 5 gigabytes) free diskspace before installing any of the required packages. If you do not have a laptop, you can always work using internet based resources (e.g. PythonAnywhere) on Sciences Po computers, looking on with others in class.

Suggested Reading: There is no required text, but the following are useful resources:

- Chambers, John. "Software for Data Analysis: Programming with R".
- Braun, W. John and Murdoch, Duncan J. "A First Course in Statistical Programming with R".
- Dalgaard, Peter. "Introductory Statistics with R".
- Press, William H., et. al. "Numerical Recipes: The Art of Scientific Computing".

3 Grading

The course will be graded on a pass/fail basis.

Weekly problem sets will be given on a completion basis. Copying is easy, but your instructors next semester will expect you to know these topics, so it is in your interest to at least try.

There will be a semester project to be completed in groups or alone.

There will be a take home final exam to be returned after the final course.

4 Proposed Schedule

NOTE: This schedule is preliminary with lots of extra time at the end of the semester to accomodate topics of interest to the students.

- **Week 1: Introduction to programming.** How a computer "thinks", how you communicate with a computer (interpretted versus compiled), introduction to open source and GitHub, introduction to R.
- **Week 2: Basic problem solving in R.** Example based and exercise based practice in native R, commands, functions, function objects, objects, object vs. function oriented programming, debugging.

- **Week 3: Linear Algebra in R.** Syntax for linear algebra in R, vectorization, advantages/disadvantages of interpreted languages.
- **Week 4: R Packages.** The most important packages for economists and their applications to applied econometrics, how to create a package, why you would create a package.
- **Week 5: Dealing with data in R.** Data I/O, data set merge, data set manipulation, descriptive statistics, (possibly data visualization).
- **Week 6: Optimization in R.** Smooth optimzaton in R, different algortithms for smooth optimization, optimizing your optimization.
- **Week 7: Introduction to Julia.** What is Julia for?, How to get Julia?, How to code 'properly' in Julia?, multiple dispatch and Julia's strengths/weaknesses.
- **Week 8: Data Visualization in All Languages.** Data visualization as a study in itself.
- **Week 9: Advanced topics in code optimization.** How to choose the correct tool, multiprocessing, multithreading, vectorization, jit, etc.
- **Week1 10: Review/workshop.** In class workshop going back over the course and working on the final projejct.
- **Week 11: Project presentations.**
- **Week 12: Project presentation or other.**