Syllabus

ECON 21410: Computational Methods in Economics

University of Chicago, Spring 2019

Administrative Details

- **Class:** Tuesdays and Thursdays, 3:30 - 4:50 PM in SHFE 103
- **TA Session:** Wednesday 5:30-6:20 PM in SHFE 203
- **Lecturer:** Jeremy Bejarano, jbejarano@uchicago.edu
- **Office Hours:** Time TBD, in the graduate commons (SHFE 201)
- **Teaching Assistant:** Ari Boyarsky, aboyarsky@uchicago.edu
- **TA Office Hours:** TBD
- **Website:** Canvas will be used for grades. Homeworks and notes will be posted on the course GitHub repo: https://github.com/jmbejara/comp-econ-sp19

About the Course

Course Description

This course introduces the empirical and computational techniques necessary for numerically solving and estimating economic models. The course covers topics in numerical methods, such as optimization, function approximation, and Monte Carlo techniques, as well as topics in data exploration, visualization, and estimation. Emphasis will be placed on developing effective programming and research practices. The course is structured through a series of applications in such topics as macroeconomic fluctuations, industrial organization, and asset pricing. The course will be taught primarily in Python. Though helpful, no previous experience with computer programming is necessary for this class.

Prerequisite(s): ECON 20100 and ECON 21020 or ECON 21030
Recommended: ECON 20200

Overview

This course aims to prepare students to begin computational and empirical research, to be effective research assistants, and to be more prepared for graduate school. Many of the skills taught in this class will also apply to those who plan on doing empirical work in the finance or consulting industries. Aiming to complement the theoretical training offered to top undergraduates at the University of Chicago, this class will focus on operationalizing economic models and econometric methods. The course is designed to equip students with the skills to

1. take theoretical models and translate them into useful economic simulations and empirical tools,
2. apply numerical methods to solve economic problems,
3. become familiar with common sources of economic data,
4. develop, implement, and manage an empirical project,
5. and produce professional output to clearly convey your results.

While these five goals are quite broad, they are the five foundational skills necessary for effective empirical research. The course will focus on Python (an open-source, general purpose programming language that is also relatively easy to learn). You will be required to complete most of your exercises in Python. The course will also provide a short introduction to the R language and 1 or 2 assignments will be in R. Learning how to program in Python will help you learn any other language you may prefer later (R, Matlab, STATA, Julia, Scala, C, C++, etc.). Note that, beyond the course prerequisites, this course should be relatively self-contained and no prior experience in computer programming is strictly necessary.

As far as computer programming goes, by the end of this quarter the goal is to equip you with the skills to

- Read and understand Python code,
- Write clear, documented, and reusable Python code,
- Use packaged functions and software libraries (like SciPy or Pandas) to do work for you,
- Keep track of your work progress and learn to code in teams using the version control system Git and the development platform GitHub,
- Summarize work using the markup language LaTeX.

To conclude, note that although this course will teach topics related more generally to computational science and data science, this is primarily a course about economics. However, the skills taught in this course are designed to provide an important foundation for those interested in learning more about these subjects later on.

Material

- **Textbook:** The course will draw on material from the online text Lectures in Quantitative Economics, by Thomas J. Sargent and John Stachurski and from Python Data Science Handbook, by Jake VanderPlas. These texts are free and are available in an online format that can be viewed on the web or downloaded as a PDF. In addition to these texts, the class will be supplemented by various notes that will be distributed in class.
- **Laptop Computer:** Students are required to have access to a laptop computer that they have administrative privileges on and that they can bring to class. It is strongly recommended that each student have their own laptop computer. This course will hold live coding sessions in which students can practice writing code in class. The exams

include coding exercises to be completed in class. Either a Mac, PC, or Linux computer will suffice.

- **Reference Material:** The following books are nice references for some of the material we will be covering in this class. They are not required. Most of them go well beyond the scope of this class. However, they're listed here for your convenience and for those interested in pursuing some of these topics further.
    - For those interested in numerical methods and computational economics, the books Numerical Methods in Economics, by Kenneth L. Judd and Applied Computational Economics and Finance, by Miranda and Fackler are great references.
    - For learning the basics of Python, the free O'Reilly book A Whirlwind Tour of Python is recommended. For more in-depth Python references, see Python Data Science Handbook, Python Cookbook, The Hitchhiker's Guide to Python, and Python for Data Analysis.

Schedule Overview

For your convenience, I've reproduced here the Spring 2019 Academic calendar. Note that this gives us about 19 regular class sessions and about 9 TA sessions before the reading period.

Date	**Event/Deadline**
Monday, April 1	Spring Quarter Begins
Monday, May 27	Memorial Day
Thursday–Friday, June 6–7	College Reading Period
Saturday, June 15	Convocation
Saturday, June 15	Spring Quarter Ends

Content Outline

Below is an rough outline of the course content.

- Introduction to the basic course tools
    - Install and go over basics of Python, Anaconda Distribution, Project Jupyter
    - Get started with Git and GitHub
- Python Essentials
- Numpy, SciPy, and Matplotlib
- Pandas (Python for Data Analysis)
- Data Visualization
- Review of Linear Regression using Python

- Panel data and the Fixed-Effects Model (with an application to estimating firm productivity)
- Monte Carlo Simulation
- An example project using microeconomics data to answer a macroeconomic question about wages during the Great Recession (an introduction to the IPUMS data project)
- Maximum Likelihood Estimation (MLE)
- Replicating a classic paper in empirical industrial organization (an application of MLE to a structural IO model)
- Dynamic Programming (with an application to the study of unemployment)
- Discrete-time Finite State Markov Chains
- Solving macro models via log-linear approximation
- Basic Time Series Analysis
- Other economic models to be covered throughout: Schelling's segregation model, McCall search model, a stochastic growth model of the macroeconomy, the permanent income model of consumption, the lucas asset pricing model, the capital asset pricing model

Exams

- **Midterm Exam:** Thursday, April 25. The exam will take place during class, 3:30-4:50 PM in SHFE 103.
- **Final Exam:** Thursday, June 13th, 4-6pm (See the registrar's site for more details.)

Grading

Final grades will be determined as follows:

- Homework – 45%
- Midterms – 20%
- Final – 35%

Class participation will be a factor for students on the margin between grades.

Policies

- **Academic Honesty:** We take cheating and plagiarism seriously. Every class you have ever take probably states that cheating will not be tolerated, but we mean it. Cheating or plagiarism will result in a 0 on that assignment with possible additional repercussions. Please take precautions to avoid putting the TA or myself in a situation where we are forced to decide if two documents are "too similar". As future researchers, consultants, bankers, etc, learning to do honest work in a timely manner is more important than getting everything correct.
- **Homework Groups:** You are welcome to work together in groups up to 4, but you are required to submit your own write-up and your own code. You are required to write the

names of your other group members on your assignment. If you discuss the homework in detail with people not in your group, please note this in a footnote as well.

- **Using Outside Material:** You are allowed to use any online resources available. Any use of previous homework solution constitutes cheating and is strictly forbidden.
- **Late Work:** Homeworks submitted late, but prior to the TA session will receive half credit. We will not accept homework assignments submitted after the TA session where answers are discussed. If you have a conflict, please plan in advance. We will consider granting extensions requested more than 5 days in advance, as well as documented family emergencies.