

*What I cannot create, I do not understand.*

– Richard Feynman

# Software Engineering for Economists

## Agenda

- ▶ Class
  - ▶ Content
  - ▶ Main Tools
  - ▶ Group project
- ▶ Initiatives
- ▶ Getting Started

- ▶ We acquire basic software engineering skills to tackle computer intensive economic research projects. During lectures and a group project, we explore the following set of topics:

- ▶ Version Control
- ▶ Unit Testing
- ▶ Debugging and Profiling
- ▶ Code Documentation
- ▶ Design Patterns
- ▶ Data Management
- ▶ Code Optimization
- ▶ Cloud Computing

## Why?

- ▶ Research
  - ▶ ... expand the set of possible economic questions to explore
  - ▶ ... improve the quality of answers
  - ▶ ... learn about what is driving the *New Economy*
- ▶ Private Sector
  - ▶ .. acquire skills in high demand (e.g. Finance, Technology)

## *Main Tools*

- ▶ The ***Python*** programming language is accessible to novice programmers seeking to develop software engineering skills, and powerful enough for serious computation. *Python* is used by computer programmers and scientists alike. Thus, it provides the tools used in software engineering as well as numerous libraries for scientific computing. In addition, *Python* is an open source project easily linked with other languages such as *Fortran* and *C*.

- ▶ ***Ubuntu*** is a Linux-based operating system. It is based on the principle of open-source development and users are encouraged to use free software, study how it works, improve upon it, and distribute it. Using *Ubuntu* serves as a preparation for the use of high performance computing clusters, who mostly rely on Linux-based operating system.



- ▶ ***Microsoft Azure*** is a cloud computing platform which provides the power and scalability required for collaboration, high performance computation, and data-intensive processing. Class participants receive access to the cloud using *Microsoft Azure Academic Passes* for six months.

**Website:** <http://research.microsoft.com/azure>

- ▶ The ***structToolbox*** is a computer program for the simulation and estimation of a dynamic model of female labor supply decisions (Keane et al., 2011). It is designed as a teaching tool to show how, by acquiring software engineering skills, structural econometricians can more readily absorb research from computational science, improve the transparency of implementations, and ensure recomputability of results.

**Website:** <http://policy-lab.org/structToolbox>

- ▶ The ***Quantitative Economics*** website provides a series of lectures on quantitative economic modelling using *Python*. Topics include economic theory and empirics, mathematical and statistical concepts related to quantitative economics, algorithms and numerical methods for studying economic problems, and coding skills.

**Website:** <http://quant-econ.net>

## *Group Project*

- ▶ The group projects consist of a contribution to the *Quantitative Economics* project, an open source library for quantitative economic modeling. Each group prepares a high-quality implementation of a quantitative economic model of their choice. We will build on existing material provided as code supplements to several economic textbooks (see course website for selected references). As the course moves along, we use our newly acquired skills to iteratively improve our workflow and the quality of our implementations. Groups report on their progress and receive feedback in a series of presentations throughout the semester.

# Initiatives

- ▶ Courses

- ▶ Software Engineering Bootcamp
- ▶ Software Engineering for Economists
- ▶ Numerical Methods in Economics

- ▶ Seminar

- ▶ Computational Economics Colloquium

- ▶ Infrastructure

- ▶ Social Sciences Computing Services
- ▶ Research Computing Center
- ▶ Computation Institute

*Courses*



## ► Software Engineering Bootcamp

The *Bootcamp* introduces students to the basics of the *Python* programming language and the *Ubuntu* operating system. It is self-contained but also intended to prepare students for the regular class *Software Engineering for Economists*.

**Website:**

<http://www.policy-lab.org/teaching/softecon-bootcamp>

## ► **Software Engineering for Economists**

We acquire basic software engineering skills to tackle computer-intensive economic research projects. These basic techniques allow us to leverage tools from computational science, increase the transparency of our implementations, and ensure the re-computability of results. Thus, they expand the set of possible economic questions that we can address and improve the quality of our answers.

## ► Numerical Methods for Economists

The objective of this course is to introduce graduate students to computational approaches for solving economic models. We will formulate economic problems in computationally tractable form and use techniques from numerical analysis to solve them. We will study examples of computational techniques in the current economics literature as well as discuss areas where these techniques may be useful in future research of economic problems.

*Seminars*

## ► Computational Economics Colloquium

The *Colloquium* offers students a unique opportunity to learn from leading scholars who teach economic modeling, numerical methods, and software engineering. Top researchers in this field will share their innovative approaches throughout the academic year.

**Speakers:** Micheal Keane, Victor Zhorin, Andrew Shepard

*Infrastructure*

► **Social Sciences Computing Services**

*Social Sciences Computing Services* is made up of several distinct support teams that provide several services to the Social Sciences Division such as Desktop and Server Support.

**Website:** <https://sscs.uchicago.edu>

## ► Research Computing Center

The *Research Computing Center* is dedicated to providing the *University of Chicago* community a full-service high-performance computing (HPC) center, including visualization resources, access to software, workshops, one-on-one consulting with domain experts, and complete data-management strategies to researchers across all departments and divisions.

**Website:** <https://rcc.uchicago.edu>



## ► Computation Institute

The *Computation Institute* is a joint initiative between the *University of Chicago* and *Argonne National Laboratory* to advance science through innovative computational approaches. It is a resource center for those collecting and analyzing large quantities of data and detailed numerical simulations of complex phenomena.

**Website:** <https://www.ci.uchicago.edu>

# Getting Started

Please visit the class website:

<http://www.policy-lab.org/teaching/softecon>

- ▶ Sign up for the class mailing list.
- ▶ Download the shared class folder.
- ▶ Check out the resources for learning *Python* and *Ubuntu*.
- ▶ Prepare your computer for the use of Virtual Machine.

*What is Next?*

## **Week 1**

Monday	Welcome and First Steps
--------	-------------------------

Wednesday	Basic Explorations into Ubuntu and Python
-----------	---

Friday	Office Hours
--------	--------------

## **Week 2**

Monday	Introductory Applications using Python
--------	--

Wednesday	Introductory Applications using Python
-----------	--

Thursday	Computational Economics Colloquium
----------	------------------------------------

# Appendix

## *References*



Keane, M. P., Todd, P. E., and Wolpin, K. I. (2011). The Structural Estimation of Behavioral Models: Discrete Choice Dynamic Programming Methods and Applications. In Ashenfelter, O. and Card, D., editors, *Handbook of Labor Economics*, volume 4A, pages 331–461. Elsevier Science.