UNIVERSITY OF EXETER

Course Program
Research Methods II
"Computation for Economics"
MRes. in Economics
Term 2, 2020

1 General Information

Lecturer	Damian Clarke
Contact Email	damian.clarke@protonmail.com
Website	`http://business-school.exeter.ac.uk/module/index.php?mod_code=BEEM143`
Github Page	https://github.com/damiancclarke/compEcon
Course Code	BEEM143
Pre-requisites	BEEM136
Meeting Time	Fridays, 15:00-17:00

2 Course Description

This course aims to give an overview of key themes in computation for economists. Advances in computational hardware and software mean that problems faced in many economic research applications can be addressed with a personal computer, and relatively simple-to-use high-level programming languages. Recent advances have also opened up avenues of research which previously required calculations which were too complex to compute, and now can be resolved on powerful servers or computer clusters. In economics there is a growing use of techniques from Machine Learning and Artificial Intelligence, based on algorithms which have been widely implemented in programming languages used by many economists.

This course aims to provide a 'language-agnostic' overview of key aspects of computation in economics. This includes areas such as general work flow and programming tools, numeric optimization, data management, data visualisation, and web-scraping to generate novel datasets. The course is not aimed to provide an overview of a single language or syntax, but rather make you feel comfortable with the underlying ideas in programming and computation so that you can work in the languages which are most suitable for your research needs. Nonetheless, during the course it is envisioned that we will work in at least three languages, namely MATLAB, Stata and Python. We will also discuss a number of other languages and tools frequently used by economists, as well as their key benefits. These include R, Perl, Julia, FORTRAN, Octave, LaTeX, git and Regular Expressions. We will undertake a brief discussion of the importance of overarching operating systems.

A key goal of this course is that upon completion, you should feel comfortable working in a range of languages, and feel that you have the tools necessary to resolve issues as they arise, and work in different languages if these offer considerable benefits for a particular problem or research agenda. This course will include a large number of applied examples in which you will resolve models and problems commonly encountered in microeconomics, macroeconomics and econometrics. We will examine this together in classes (which will be implemented via zoom and videod for posterity), and you will be asked to resolve a number of problem sets, potentially tailored to your own research interests in economics.

All classes will consist of slides (which will be made available on Exeter's ELE page). The majority of classes will also include code and/or data. This will also be made available on the ELE page, as well as a github repository, which you can follow and fork if you would like, available at: `https://github.com/damiancclarke/compEcon`. This will be updated during the term.

3 Evaluation

The evaluation in this course will consist of a series of applied problems (or 'problem sets'), as well as a end of semester take-home exam. Each of these two components will account for 50% of the final mark for this course. The applied problems will be made available every 2 weeks during term (generally no later than Thursday of each week, and should be handed in via email by the following Thursday at 11:59PM. Submissions should consist of a series of responses in a pdf file, and the full set of code used, which should be fully replicable. More details on each of these applied problems will be given when they are assigned during term.

The end of term take-home exam will consist of a series of applied problems that can be resolved on a personal computer. In all cases, you can use any programming language which allows you to resolve the problems assigned. Our work in class may influence this choice (for example by providing basic code excerpts in a particular language), however you should feel free to use the tools which are most comfortable and useful for you.

A general description of the assessment materials and the proposed calendar is provided below. In the case of Applied Problems, the dates indicate the day that the problems will be assigned, not the due date. This calendar may be modified during term if necessary.

Evaluation	**Weight**	**Date (Approximate)**
Applied Problems 1	12.5%	Jan 30, 2020
Applied Problems 2	12.5%	Feb 13, 2020
Applied Problems 3	12.5%	Feb 27, 2020
Applied Problems 4	12.5%	Mar 12, 2020
Final Exam	50%	March

4 Contents

This course consists of 7 units which will be addressed in approximately 10 2 hour classes. A broad description of the contents of each unit is provided on the following page. Topics 2, 3, and 4 are (respectively) applications of computational problems frequently found in microeconomics, econometrics and macroeconomics. Units 1, 5, 6, and 7 are broadly relevant any economist using computational tools.

Class	**Contents/Discussion**
(1) Introduction to General Tools and a Matrix-based Programming Language	(a) Discussion of General Tools - High-level vs. low-level programming languages - Work-flow, git - Operating Systems - How to get help (b) Discussion of a Matrix-based Language
(2) Optimization I	(a) 'Behavioural' optimization in MATLAB – **microeconomic application** (b) "Programming" (c) Optimization tools
(3) Optimization II	(a) Optimization for Estimation – **econometric application** (b) Maximizing a likelihood function (c) 'Solving' just- and over-identified systems of equations
(4) Dynamic Programming	(a) Finite and Infinite horizon – **macroeconomic application** (b) Bellman equations (c) Estimating dynamic models
(5) Visualization and Data Management	(a) Working with data (b) Mapping (c) An Overview of Graphing tools
(6) Web-scraping	(a) Python work-flow (b) Web-scraping libraries (c) Regular expressions
(7) Discussion	(a) 1:1 discussions (b) Where to from here?

5 Bibliography (Potentially Incomplete)

Below I list a number of resources which we may rely on in this course. Many of these are not text-books or papers, but rather websites and slide-decks. We will discuss the particular relevance of these (and other) references in our class discussions.

Adams, A., Clarke, D., and Quinn, S. (2016) Microeconometrics and MATLAB: An Introduction, Oxford University Press.

S. Boraan Aruoba and Jesús Fernández-Villaverde, (2015) A comparison of programming languages in macroeconomics, Journal of Economic Dynamics and Control, Volume 58, pp. 265–273.

Jesús Fernández-Villaverde "Computational Methods for Economists": `https://www.sas.upenn.edu/~jesusfv/teaching.html`

Matthew Gentzkow and Jesse M. Shapiro "Code and Data for the Social Sciences: A Practitioner's Guide": `https://web.stanford.edu/~gentzkow/research/CodeAndData.pdf`
Kenneth L. Judd (1998) Numerical Methods in Economics, MIT Press.

Jesse Perla, Thomas J. Sargent and John Stachurski, (2019) Quantitative Economics with Julia, `https://julia.quantecon.org`

Frank Pinter, "Git for economists" (2019), `https://www.frankpinter.com/git/`

Ljubica "LJ" Ristovska (2019) Coding for Economists: A Language-Agnostic Guide to Programming for Economists, `https://www.dropbox.com/s/wqefknwfb91kop8/Coding_For_Econs_20190221.pdf?dl=0`

Thomas J. Sargent and John Stachurski (2020) Quantitative Economics with Python, `https://python.quantecon.org`