Applied Computational Economics - Winter 2020

ECON 647

Professor Barczyk

Office: LEA 321b
E-mail: daniel.barczyk@mcgill.ca
Office Phone: 514-398-6981

Class Location: LEA 424
Class Time: T/R 10:05am - 11:25am
Office Hours: by appointment

Course description

This is a graduate course open to both MA and PhD students. The prerequisites for the course are linear algebra, multivariable calculus, a good understanding of probability, and at least a B+ in ECON620 I or equivalent. If you are a PhD student there is no grade requirement. The goal of the course is to first familiarize you with basic numerical methods and then to combine these methods into larger algorithms that can be used to numerically solve more complicated economies for which closed-form solutions are rarely available. The first part of the course focuses on basic numerical methods and the second part on solving dynamic models in both discrete and continuous time. Throughout, the emphasis is on applications. In order to perform computations I will be using Matlab. I do not assume that you have any previous experience with Matlab or any other computing software.

Textbook

In the first part of the course we will closely follow the book *Applied Computational Economics and Finance* by Mario Miranda and Paul Fackler. A wider coverage of numerical methods may be found in *Dynamic General Equilibrium Modeling* by Burkhard Heer and Alfred Maussner and *Numerical Methods in Economics* by Kenneth Judd. A good introduction to dynamic programming with numerical techniques is the book *Dynamic Economics: Quantitative Methods and Applications* by Jerome Adda and Russell Cooper. In the second part of the course I will make notes available.

Grading

PhD students

Problem sets 30%, term paper 60%, participation 10%.

MA students

Problem sets 30%, term paper 40%, class presentations 20%, participation 10%.

Problem sets

For the problem sets you are encouraged to work in groups of two or three. Your write-up should be typed and include:

1. a brief summary of the methods used;

2. tables and/or graphs to describe your results;
3. your computer programs in an appendix.

Class presentations
You present the solutions to the problem sets in a structured, seminar-style way. You should discuss the theory and the solution method underlying the given problem and show how you have solved the problem using Matlab.

Term paper
You have considerable flexibility for what you do for your term paper. However, you should come up with your own research question and utilize computational methods from the course. A research question that is a derivative of an existing paper can also be acceptable. There is no need to have a long paper, e.g., avoid a long-winded introduction or literature review, instead, concentrate your efforts on finding an interesting topic, on the computations, and the interpretation of your results.

April 1: Paper proposal is due

May 1: Paper is due (by email)

The paper proposal can be short (1-2 pages) and should explain your research question and the computational methods you expect to utilize. Your paper should be accompanied by a "code toolbox" which is structured in a way that enables me to replicate your results easily. What this means is that there is a main program that calls all the other functions and returns the results in tables and/or graphs. Also, document your code along the way with short comments.

Course Outline

Part I: Tools for larger models

1. Solving linear and non-linear equations
2. Numerical integration, differentiation, and optimization
3. Interpolation techniques
4. Solving functional equations

Part II: Dynamic programming in discrete and continuous time

1. Discrete time, *discrete* state dynamic (Markov) models. Economic examples. Numerical solution algorithms: Backward recursion, function iteration, and policy iteration.
2. Discrete time, *continuous* state dynamic (Markov) models. Theory and examples of dynamic optimization and dynamic equilibrium models such as American option pricing, economic growth, dynamic games, and asset pricing models.

3. Continuous-time dynamic programming. The Hamilton-Jacobi-Bellman equation and the Kolmogorov forward equation. Solving a heterogeneous-agents economy in continuous time using Barczyk and Kredler (2014, *Altruistically Motivated Transfers Under Uncertainty, Quantitative Economics*).

Academic Integrity

McGill University values academic integrity. Therefore all students must understand the meaning and consequences of cheating, plagiarism and other academic offenses under the code of student conduct and disciplinary procedures (see http://www.mcgill.ca/integrity/ for more information).

L'université McGill attache une haute importance à l'honnêteté académique. Il incombe par conséquent à tous les étudiants de comprendre ce que l'on entend par tricherie, plagiat et autres infractions académiques, ainsi que les conséquences que peuvent avoir de telles actions, selon le Code de conduite de l'étudiant et des procédures disciplinaires (pour de plus amples renseignements, veuillez consulter le site http://www.mcgill.ca/integrity).