47-805  Computational Methods for Economics       Mini-3 (Spring) 2015

Professor:     Dr. Yaroslav Kryukov
               kryukov@andrew.cmu.edu,
               (412) 268-9455
               GSIA 336
Office hours:  Mon., Wed., 10:30-12pm
Lectures:      Mon, Wed 1:30-3:20, Tepper 147

Objectives. This is a course in the basic tools of numerical analysis that can be used both to assess the quantitative implications of economic theory and to derive theoretical results of economic models without analytical solutions. While the most examples will come from macroeconomics, the generality with which the techniques will be presented in this course will make them applicable to a wide range of fields like econometrics, financial economics, marketing, game theory, and contract theory. To enable efficient application of numerical tools, this course endeavors to explain not only when and how to use various numerical algorithms but also how and why they work; in other words, the intention of the course is to open up some "black boxes" and provide the students with a versatile tool set.

The course will cover basic topics in numerical methods such as systems of equations, optimization problems, functional approximation and numeric integration. The course will then focus on solving dynamic problems using approaches like differential equations and dynamic programming. Subsequent advanced topics will include dynamic stochastic games or other methods if there is student interest.

Prerequisites. There are no formal prerequisites, but course assumes that students are familiar with real analysis and particularly the (constrained) optimization, linear algebra as well as the basics of differential equations and functional analysis. It is assumed that you have learned Dynamic Optimization theory (Bellman equations) from Macro I.

You also need to know Matlab or any other programming language.

Textbook. Main text is “Numerical Methods in Economics” by Ken Judd. Other readings might be assigned as necessary. Due to lack of time we will cover material in substantially lower level of detail than the textbook. Homework and exam will be based on material covered in class, unless you are explicitly asked to read a specific section of the book.
Software. Homework solutions will use MATLAB. You are free to use other programming languages (C, Fortran, Mathematica), but you will still be responsible for producing the same numerical results as well as a readable code. We will also learn to use the modeling language AMPL, and you will be required to use it; it is available via Web interface, so you do not need to worry about installing it. You are not allowed to use Excel, and it will be very hard to do so anyway.

Homework. Programming is a practical skill, so doing problem sets is the best way to understand the material. The lowest-scoring PS will not count towards your course grade (with the exception of the last one).

You can work in teams of up to three people. If you do, submit a single problem set but write down the names of all the team members. You can switch teams or choose to work alone.

When solving homework, you are allowed to use class materials and built-in Matlab help. You are not allowed to receive help from people outside of your team, or to use ready codes from the internet or previous-year solutions. Violations of this rule will reduce your learning and will be punished by zero scores.

Problem sets are due at the beginning of the class on the due date. Your solution should include a text document with results in numeric or graphical form, along with proper verbal discussion. This document should be uploaded to the Blackboard as a single document. The code should be also uploaded, as a single .zip file.

Communication. In addition to office hours, we will have occasional recitations or makeup classes on Friday; keep an eye on Blackboard announcements. Professor will respond to the e-mail questions within 24 hours, or 36 hours if the e-mail was sent on a weekend. Phone communication should be kept to a minimum.

Final exam will be a take-home exercise, with questions similar to homework ones. We will discuss the date in class. Unlike homework, it has to be done individually.

Grading. 60% of the grade will determined by the problem set scores (excluding the lowest one, as described above) and 40% - by the take-home final.

Copyright. You are not allowed to make audio or video recordings of the class without instructor’s permission.

Integrity. You are bound by the University’s academic integrity rules. Which means you cannot use work of others (including your classmates) without naming them. General rule is: when in doubt, cite your source.

Acknowledgements. Some of the course material is based on the similar courses taught by Ken Judd (University of Chicago & Hoover Institution), and Ulrich Doraszelski (Wharton @ UPenn).
Topics
The tentative list of topics and class dates is below. It is subject to change depending on pace of the class and students' interests. Order of the sections covering the applications will be determined later.

I. Numerical Analysis: General Considerations (Jan-12)
Chapters 1 and 2. General ideas of convergence rates, computational errors, error analysis.

II. Linear Equations (Jan-14)
Chapter 3. LU, QR, and Cholesky decomposition, condition numbers, Gauss-Jacobi and Gauss-Seidel methods.

No class Jan-19: Martin Luther King day

III. Nonlinear Equations (Jan-21)
Chapter 5. Gauss-Jacobi, Gauss-Seidel, Newton, continuation and homotopy methods. Application to general equilibrium.

IV. Optimization (Jan-26, 28)
Chapter 4. Search methods, bisection, Newton method, BFGS and DFP updates. Applications to consumer demand problems, incentive problems, and structural estimation.
Modeling languages (AMPL): Coding up the model, selecting solvers, submitting the model and processing the output.

V. Functional Approximation (Feb-2)
Chapter 6. Orthogonal polynomials, splines, interpolation.

VI. Numerical Integration and Monte Carlo simulation (Feb-4)
Chapters 7, and 8. Integration methods for single- and multidimensional integrals. Monte Carlo simulation methods. Applications to portfolio choice and Bayesian problems.

* Differential and Functional Equations

* Numerical Dynamic Programming
Chapter 12. Solutions to deterministic and stochastic dynamic programming problems. Applications to saving and consumption problems.

* Dynamic games (Feb-26)

If there is interest in discussing other advanced topics, let the professor know.