‘math+econ+code’ masterclass on equilibrium transport and matching models in economics

Alfred Galichon (NYU)

Day 1: competitive equilibrium with gross substitutes
a. Getting started

Preliminary stuff

- Schedule: June 8-12, 2020. Classes meet 2pm-6pm Paris time / 8am-noon New York time. In addition, supplementary material (approximately 2 hours in length) is made available each day to complement the main lectures.
- Format: Zoom webinar.
- Course webpage: http://alfredgalichon.com/mec_equil
- Course repository: https://github.com/math-econ-code/mec_equil

© 2020 by Alfred Galichon. Support from NSF DMS-1716489 and ERC CoG-866274 EQUIPRICE grants, and contributions by Keith O'Hara, Lucas Vernet, Octavia Ghelfi and James Nesbit are acknowledged.

Introductions

- Introducing me: A. Galichon, professor of economics and of mathematics at NYU (ag133@nyu.edu)
- Introducing you

Remarks on this course

- Targeting Math and Econ students. Self-contained for both audiences
- Teaching format: 5 “days”; each day = four hours, 1/3 economics modeling, 1/3 mathematical toolbox and 1/3 Python coding Plus 2 hours of complementary material (to study depending on your time zone).
- Programming: our demos will be done Python and the support will be in these languages only, but you are welcome to use the language of your choice e.g. R, Matlab, C++, Julia...
- This course will be recorded, and the recording may be publicly broadcasted in part or in full. If you are participating in the Webinar, you are implicitely consenting that your voice and image should be broadcasted. You have the option to turn your camera and your microphone off.
- Questions?

This course

- is focused on the computation of competitive equilibrium, which is at the core of surge pricing engines and allocation mechanisms. It will investigate diverse applications such as network congestion, surge pricing, and matching platforms. It provides a bridge between theory, empirics and computation and will introduce tools from economics, mathematical and computer science.
- will provide the conceptual basis of competitive equilibrium with gross substitutes, along with various computational techniques (optimization problems, equilibrium problems). It will apply parallel computation is adapted to the computation of equilibrium. Applications to hedonic equilibrium, multinomial choice with peer effects, and congested traffic equilibrium on networks.
- will introduce tools from economic theory, mathematics, econometrics and computing, on a needs basis, without any particular prerequisite other than the equivalent of a first year graduate sequence in econ or in applied math.
- is complementary to the m+e+c_optim masterclass (taught in Paris June 2019 and in NYC in Jan 2020), but can be taken without it as a prerequisite.

Outline

- Day 1: competitive equilibrium with gross substitutes
- Day 2: matching models with fully transferable utility
- Day 3: matching models with imperfectly transferable utility
- Day 4: matching models with nontransferable utility
- Day 5: network equilibrium

Software installation

- See the setup.md document in the github repository:
  https://github.com/math-econ-code/mec_equil/blob/master/Setup.md

Knowing you

Please fill out:
https://goo.gl/forms/5WLfkpc3GNCecd6A3

Assessment

- Students taking this class for credit have a choice between:
  - Either a take-home exam (24 hours), to be taken over a 24 hour period, June 20, 10am NY time - June 21, 10am NY time.
  - Or a short paper (12 pages or more), to be discussed with the instructor. The paper will bear some connections, in a broad sense, with the topics of the course. Many papers are considered acceptable: original research paper, survey paper, report on numerical experiments, replication of existing empirical results... are all acceptable. The requirement is to be innovative on a theoretical, empirical, or computational level. This work should be submitted before December 31, 2020.
- Email me to indicate which of these you are opting for.

Setup

https://github.com/math-econ-code/mec_equil/blob/master/Setup.md

Anaconda (including Python)

- Anaconda is a powerful platform that simplifies the management of Python (and R) packages. It comes seamlessly with Python and the Jupyter notebook environment.
- Follow the installation instructions on the Anaconda website, and download the Python 3.7 version.
- For commands in conda Mac users can just use terminal, Windows users should open Anaconda Prompt.
- Open Jupyter notebook using Anaconda Navigator or by typing
  ```
  jupyter notebook
  ```
  in conda and open an Python 3 notebook.
- It can be convenient to also install the program **nbopen** to open Jupyter notebook .ipynb files by double clicking them. (Google for installation instructions).

Git / Github

- To install git
  - Mac users: open terminal and run git –version. If you don't have git installed, you will be prompted to do so.
  - Windows users: go to the Git for Windows download page and install. I recommend also installing Visual Studio Code for a text editor, unless you already use SublimeText or vim. Alternatively GitHub Desktop is a nice GUI for Git.
- Open git and navigate to your home directory
  Mac: `cd /Users/user/my_project`
  Windows: `cd /c/user/my_project`
- Clone the mec_equil repo

  ```
  git clone https://github.com/math-econ-code/mec_equil
  ```
- Whenever the repository is updated

  ```
  git pull origin master
  ```
- Much more can be done with Git and GitHub, you can easily find good tutorials online.

Installing Docker (optional)

- Docker (and the world of containers) are fascinating and extremely useful if you intend to run serious computation (on the Cloud or HPC clusters). However, the following can be a bit more involved and time-consuming, so not all of you will find the need for it.
- Install Docker: Mac and Windows users can install Docker Desktop from `https://www.docker.com/products`
- `/docker-desktop`
  - If your Windows is the Pro edition, you should be fine.
  - If your Windows edition is the Home edition, you may run into problems. Try to install Windows subystem for Linux (https://docs.microsoft.com/en-us/windows/wsl/install-win10). This may take you to upgrade Windows.

Setting up the Docker container (optional)

- Download the m-e-c.Dockerfile from https://github.com/math-econ-code/mec_equil/tree/master/Docker.
- To build the container:, open the shell, and cd to the directory where the dockerfile is, and run
  ```
  docker build --tag=m-e-c:latest --tag=m-e-c:stable -f
  m-e-c.Dockerfile .
  ```
  (be patient, it will take a while). Note that the m-e-c container contains more tools that we need for just this course, as it will also serve the mec_optim masterclass.
- To run the container: create a local folder, and run
  ```
  docker run -it -p 8888:8888 -v
  <your/local/folder>:/home/mec m-e-c
  ```
  you are now running a bona fide Linux machine on your computer!
- To launch the Jupyter notebook, prompt `cd ../home/mec`
  ```
  jupyter lab --ip=0.0.0.0 --allow-root
  ```
  then open a browser and go to the URL as displayed.

Setting up HPC / GCP

- Server access on NYU's HPC cluster: coming up.
- On Google Colab:
  https://colab.research.google.com/