‘MATH+ECON+CODE’ MASTERCLASS ON COMPETITIVE EQUILIBRIUM: WALRASIAN EQUILIBRIUM WITH SUBSTITUTES

Alfred Galichon (NYU)

Spring 2018
Day 1, May 21, 2018: the smooth case
Block 3. Code: pricing algorithms

- Coding assignments: organization
- Optimization: comparing gradient descent, Newton descent and coordinate descent
- Pricing: comparing Jacobi and Gauss-Seidel
- Introducing SOR methods

Section 1

Coding assignments: organization

Github directory for assignments

- Those wishing to take the class for credit can be evaluated (at their option) through participation in the coding assignments or a short final paper, to be discussed with instructor.
- The coding assignments will be uploaded on the following Github directory: mec_equil_codeassignments.
- In order to do this, you need to Github directory
- PLEASE NOTE: This is a Github directory, so your answers will be publicly available for anyone to see. If you have a problem with this, come and talk to us.

- Each submission is made of two files:
  - One pdf file for findings and comments
  - The code, in a single file.
- The name of the files will be 'D-code-LASTNAME' and 'D-results-LASTNAME', where D=1,2,...,6 is the day, and LASTNAME is your name.
- Participants should post no later than midnight of each days their coding assignment in the relevant subdirectory of the given day.

Section 2

Coding assignment for day 1

- Today we will work on a very simple model of surge pricing. Assume that $\mathcal{X}$ is passenger location, $\mathcal{Z}$ is the location of rides pickup, and $\mathcal{Y}$ is car location. These sets are all set to be a grid of 100 points in a square city $[0,1]^2$. That is, $\mathcal{X} = \mathcal{Y} = \mathcal{Z} = \{1/10, 2/10, 3/10, ..., 9/10, 1\} \times \{1/10, 2/10, 3/10, ..., 9/10, 1\}$.
- Assume that there is a mass one $n_x = 1$ of passengers at each $x \in \mathcal{X}$.
- Assume that if a car $y \in \mathcal{Y}$ has coordinates $\left(y^1, y^2\right)$ there is a mass $m_y = y^1 \times y^2$ of passengers at y.

- Let p_z be the price of the ride starting at z, which is what we are looking for.
- Assume that the demand for rides pickup at z is given by

$$D_z(p) = \sum_{x \in \mathcal{X}} n_x \frac{\exp(u_{xz} - p_z)}{1 + \sum_{z' \in \mathcal{Z}} \exp(u_{xz'} - p_{z'})}$$

  where $u_{xz} = -10 \times 1\{d(x,z) \geq 0.1\}$, where $d(x,z)$ is the Euclidian distance between x and z, given by $d(x,z) = \sqrt{(x^1 - z^1)^2 + (x^2 - z^2)^2}$.
- Assume that the supply for z is given by

$$S_z(p) = \sum_{y \in \mathcal{Y}} m_y \frac{\exp(p_z - c_{yz})}{1 + \sum_{z' \in \mathcal{Z}} \exp(p_{z'} - c_{yz'})}$$

  where $c_{yz} = d(y,z)^2$ is the squared Euclidian distance between y and z.

- The goal of this assignment is to determine the price of each ride z, and the distribution of the number of rides l_z. More precisely, the goal is to determine a price vector (p_z) such that for all z,

$$precision := \max_{z \in \mathcal{Z}} \left\{ \frac{|S_z(p) - D_z(p)|}{S_z(p) + D_z(p)} \right\} \leq 10^{-5}$$

- In your answer, you should report:

(i) the precision number as defined above
(ii) the total number of rides $\sum_{z \in \mathcal{Z}} l_z$
(iii) the average price of a ride $\sum_{z \in \mathcal{Z}} l_z p_z$.
(iv) the number of iteration – for Jacobi and Gauss-Seidel, an iteration corresponds to a complete update of all the coordinates
(v) the time taken on your machine

Report the informations above:

1. using gradient descent in the optimization formulation
2. using Newton descent
3. using coordinate updates – Jacobi version
4. using coordinate updates – Gauss Seidel version
5. repeat questions 3 and 4 above when the supply is replaced by

$$S_z(p) = \sum_{y \in \mathcal{Y}} m_y \frac{\exp\left(F\left(p_z\right) - c_{yz}\right)}{1 + \sum_{z' \in \mathcal{Z}} \exp\left(F\left(p_{z'}\right) - c_{yz'}\right)}$$

with $F\left(p_z\right) = p_z + \log\left(1 + e^{-p_z}\right)$.