Application: Simple learning in a Cournot oligopoly

Murat Yıldızoğlu
Bordeaux University
http://yildizoglu.info

Our task is the development of a NetLogo program that represents the workings of a simple Cournot oligopoly where firms try to adapt their level of production to the strategies of their competitors.

1 Description of the main elements of the model

Our model concerns the market of a homogeneous good, for which the inverse demand function of the consumers is given by the following linear function:

$$p(Q) = a - bQ, \;\; a > 0, b > 0$$

For our computational experiments, we will adopt the following values for these parameters: $a = 256, b = 1$.

There are `nbFirms`$\in [2, 20]$ `(*)` firms in the oligopoly and the cost function of each of them is given by::

$$C(q) = F + cq + dq^2, F \geq 0, \;\; a > c > 0, d > 0$$

We adopt the following values: $F = 0, c = 56, d = 1$.

We can define three interesting symmetrical solutions for this oligopoly: Cournot equilibrium (C), Competitive equilibrium (W), and cooperative solution (CS, the cartel).

- **C :** $q^C = (a - c)/(2d + b(nbFirms + 1))$, $p^C = a - b \times nbFirms \times q^C$
- **W:** $q^W = (a - c)/(2d + b \times nbFirms)$, $p^W = a - b \times nbFirms \times q^W$
- **CS:** $q^{CS} = (a - c)/(2d + 2b \times nbFirms)$, $p^{CS} = a - b \times nbFirms \times q^{CS}$

We can also compute the corresponding profits for each one of these allocation, using the standard definition of the profit function: $\pi(q) = p \times q - C(q)$.

We desire to determine the solutions towards which the firms would converge under different learning schemes that they would adopt.

The quantity levels of the firms are initially drawn uniformly in the following interval $\left(q_{i,0} \in \left[0, 1.2q^W\right]\right)$. At the end of each period, each firm will adopt, following the learning scheme that the firms follow (the same for all firms in our case), a new production level that it will realize and sell in the next period.

We will consider two very simple alternative schemes for the learning process of the firms. The user will be able to select in the interface the learning scheme to use, before starting the simulations.

***Best-reply learning* :** In this case, each firm produces in period t its best reply to the total production of its competitors in period $t - 1$, assuming that they will continue to produce the same quantities also in t:

$$q_{i,t} = R_i\left(Q_{-i,t-1}\right), Q_{-i,t-1} = \sum_{j \neq i} q_{j,t-1}$$

which corresponds to the production level

$$q_{i,t} = R_i\left(Q_{-i,t-1}\right) = \frac{a - c - bQ_{-i,t-1}}{2(b + d)}, q_{i,t} \geq 0.$$

***Social learning* :** In this case the learning of the firm is based on the imitation of the behavior of their competitor who obtained the highest profit in the previous period: it just adopts in period t the quantity used by this competitor in period $t - 1$. If more than one competitors attained the same highest profit in $t - 1$, the firm randomly chooses the one to be imitated by it. This imitation takes place in each period for each firm, with a probability `probImitate`$\in [0, 0.3]$ `(*)`. The firm also does random experiments for production levels, by randomly drawing a new production level $qmut \in \left[0, 1.2q^W\right]$, from a Normal distribution $\mathcal{N}\left(\overline{q}_{t-1}, sdMut\right)$ where $\overline{q}_{t-1}$ is the average quantity produced by the firms in the previous period and `stdMut`$\in [0.01, 1]$ `(*)` is the standard deviation of the Normal law. This random experimenting takes places for each firm with a probability `probMutate`$\in [0, 0.1]$ `(*)` in each period.

You may want to consult Vallée & Yildizoglu (2009); Vallée & Yildizoglu (2013) for a more complete study on this question.

2 NetLogo program

The instructions of the program should follow the following flow:

1. The user choses in the interface the type of learning to study and the values of the parameters.
2. The program initializes other global variables, creates and initializes the agents (including the initial quantity of firms), and computes the equilibrium values of the variables as a function of the parameter values selected by the user (the **`setup`** procedure).
3. In each period (the **`go`** procedure),
   (a) the market takes place and the price is determined, as well as the profits of the firms
   (b) the time advances of one step
   (c) each firm decides the quantity for the next period, following the learning process selected at the start of the simulation

The GUI of the model should include:

- A World to which we should give the minimal size, in order to hide it as much as possible.
- Sliders allowing the selection of the values of the parameters in the acceptable ranges for each of them.
- A chooser that allows the selection of the learning type of the firms (either "*bestReply*", or "*socialLearning*").
- Monitors for observing the current equilibrium values of the prices and the quantities.
- A "Setup" button for launching the setup procedure.
- A "Go" button for repeatedly executing the go procedure until a second click by the user.
- A plot showing the evolution in time of the market price. This plot should also contain lines corresponding to different equilibrium prices $\left(p^C, p^W, p^{CS}\right)$.
- A plot showing the evolution in time of the average profit of the firms. It should also contain lines corresponding to different equilibrium profits.
- A plot showing the evolution in time of the average quantity of the firms, and the equilibrium quantities.

The code should consequently contain:

- The declaration of global variables (including the ones necessary for the definition of demand and cost functions, as well as the equilibrium values of different variables), and of the firm variables.
- A ***`setup`*** procedure that initializes the model, creates the firms and initializes their values.
- A ***`go`*** procedure that contains the instructions corresponding to all operations that take place in a period of the model.
- You may choose to dedicate specialized procedures to some of these operations, for the learning process, for example.

3 Experiments

These two models are very simple. You should conceive your experiments mainly for studying the impact of the learning process on the dynamics of firms' quantities and of the market price. An important parameter is the number of firms in the oligopoly.

`(*)`: variable to be created in the interface.
Instructions NetLogo to consider in the dictionary: `clear-all, reset-ticks, set, let, breed, turtles-own, create-turtles, ifelse, tick, max-one-of, random-float, random-normal, report.`

References

Vallée, Thomas, & Yildizoglu, Murat. 2009. Convergence in the Finite Cournot Oligopoly with Social and Individual Learning. *Journal of Economic Behaviour and Organization*, 670–690.

Vallée, Thomas, & Yildizoglu, Murat. 2013. Can they beat the Cournot equilibrium? Learning with memory and convergence to equilibria in a Cournot oligopoly. *Computational Economics*, **41**(4), 493–516.