Mini Course on Structural Estimation of Static and Dynamic Games

Junichi Suzuki

University of Toronto

June 1st, 2009

About the Instructor

- Name: Junichi Suzuki
- Affiliation: Dept. of Economics, University of Toronto
- Major: Empirical Industrial Organization, Urban Economics
- Studied in Keio (BA, MA) and Univ. of Minnesota (PhD)

Today's Goal

- Present the very basics of estimating structural parameters of
  - static games (lecture 1)
  - single-agent dynamic optimization problem (lecture 2)
  - dynamic games (lecture 3)
- Designed for practitioners
- Focus on their implementation rather than proving their statistical properties etc

Outline of Each Section

- Construct a basic model
- Clarify the type of data
- Consider three approaches:
  - Conventional MLE
  - Nested-pseudo likelihood by Aguirregabiria and Mira (2007)
  - "BBL" by Bajari, Benkard and Levin (2007)

What Not Covered Today (But Typically Covered Graduate Empirical IO!)

- Demand: Berry, Levison and Pakes (1992), Nevo (2001), Petrin (2002)
- Productivity: Olley and Pakes (1996), Levinsohn and Petrin (2003)
- Auctions: Haile and Tamer (2003)

What is Structural Estimation?

- Consider a parametric model that characterizes agents' behavior and equilibrium
- The model should be consistent with economic theory
- Each parameter of the model represents agents' primary characteristics
  - Preference
  - Technology
- Structural estimation aims to identify these parameters from the data

Why Structural Estimation?

- Pros
  - Can present channels through which policy affects the resulting equilibrium
  - Can simulate policy impacts on welfare
  - Closely related to economic theory
  - Assumptions made are explicit
- Cons
  - High entry cost (theory, econometrics, numerical methods, data mining etc..)
  - Often require significant amount of computations

Computation

- Need to be familiar with some programming language
- For most cases, STATA is not enough
- One way is to use matrix-based languages (e.g., Matlab, Gauss)
  - Easy to write a program
  - Speed is slow
- Another option is to use primitive languages (e.g., Fortran, C)
  - Time consuming to write a program
  - Speed is faster

Part I: Estimation of Static Games

Motivation

- Many economic activities involve interaction between agents
  - Store opening of convenience stores
  - Adoption of technologies: VHS vs Beta, Blue-ray vs HD DVD
  - Product type choice: high-end service, low-end service
- Estimation should take into account potential interactions between agents
- Need game theoretic models

Model: Simple Simultaneous Static Game

- N players: $i \in \{1, \dots N\}$
- Each player's choice $a_i \in A = \{0, 1, \dots, K\}$
- Each player's payoff:
  $u_i\left(a_{i,} a_{-i}, s, \epsilon_i\right) = \pi_i\left(a_{i,} a_{-i}, s\right) + \epsilon_i\left(a_i\right)$
  - s: state variables
  - ϵ_i: choice-specific private shock: variables unobservable to econometricians, $\epsilon_i\left(0\right) = 0$

Examples

- Bresnahan and Reiss (1991): $A = \{\text{Entry, Not}\}$
- Mazzeo (2002):
  $A = \{\text{Not, Entry to low end, Entry to high end}\}$
- Seim (2006) :
  $A = \{\text{Not, Enter to Mkt 1}, \dots, \text{Enter to Mkt M}\}$
- Suzuki (2009):
  $A = \{\text{Not, Open 1 hotel}, \dots, \text{Open 7 hotels}\}$

Case 1: Game of Complete Information

- Each player observes not only its own ϵ_i but also its rivals' ϵ_{-i}
- ϵ_i can be firm-specific $(\epsilon_i \neq \epsilon_j)$ as well as market-specific $(\epsilon_i = \epsilon_j)$
- Players do not face uncertainty (but econometricians do!)
- A pure strategy Nash equilibrium of this game is a set of strategies $\left\{a_i^*\left(s,\epsilon\right)\right\}_{i=1}^N$ such that

$$\begin{aligned}\pi_i\left(a_i^*\left(s,\epsilon\right), a_{-i}^*\left(s,\epsilon\right), s\right) + \epsilon_i\left(a_i^*\left(s,\epsilon\right)\right) \\ \geq \pi_i\left(a_i\left(s,\epsilon\right), a_{-i}^*\left(s,\epsilon\right), s\right) + \epsilon_i\left(a_i\left(s,\epsilon\right)\right)\end{aligned}$$

for all $i \in \{1, \ldots N\}$ and $a_i \in A$

Case 2: Game of Incomplete Information

- Each player can observe only its own ϵ_i but not ϵ_{-i}
- Only the distribution of ϵ_{-i} is known
- Each player makes its decision based on its belief about the distribution of its rivals' decisions
- Need to employ a Bayesian Nash equilibrium as an equilibrium concept

Pure Strategy Bayesian Nash Equilibrium

1. a set of strategies $\left\{a_i^*\left(s, \epsilon_i, \sigma_{-i}\left(\cdot\right)\right)\right\}_{i=1}^N$ and
2. equilibrium beliefs $\left\{\sigma_i^*\left(a_i, s\right)\right\}_{i=1}^N$

such that

$$\begin{aligned} & \sum_{a_{-i}} \sigma_{-i}^*\left(a_{-i}, s\right)\left[\pi_i\left(a_i^*\left(s, \epsilon_i, \sigma_{-i}\left(\cdot\right)\right), a_{-i}, s\right)\right. \\ & \qquad \left. +\epsilon_i\left(a_i^*\left(s, \epsilon_i, \sigma_{-i}\left(\cdot\right)\right)\right)\right] \\ \geq\ & \sum \sigma_{-i}^{**}\left(a_{-i}, s\right)\left[\pi_i\left(a_i, a_{-i}, s\right)+\epsilon_i\left(a_i\right)\right] \end{aligned}$$

for all $i \in \{1, \ldots N\}$ and for all $a_i \in A$ and

$$\begin{aligned} \sigma_i^*\left(a, s\right) = & \int 1\left(a = \arg\max \sum_{a_{-i}} \sigma_{-i}^*\left(a_{-i}, s\right)\left[\pi_i\left(a_i^*\left(s, \epsilon_i, \sigma_{-i}^*\left(\cdot\right)\right)\right)\right.\right. \\ & \left.\left. +\epsilon_i\left(a_i^*\left(s, \epsilon_i, \sigma_{-i}^*\left(\cdot\right)\right)\right)\right]\right) dF\left(\epsilon_i\right) \end{aligned}$$

Estimation

- Want to recover the structural parameters of $\pi\left(a_i, a_{-i}, s\right)$ from the data
- Data should consist of firms' decisions $\left(\{a_i\}_{i=1}^{N}\right)$ and state variables s, coming from several markets
- Maximum likelihood is the most straightfoward way
- Stick with a simple entry model
- Start with a mere regression and examine why it is problematic

Example: Entry Model

- Consider the following entry model:

$$\begin{aligned} \pi\left(a_i, a_{-i}, s\right) &= a_i \left[\alpha_1 + \alpha_2 \ln Pop - \alpha_3 \left(\sum_j a_{j\neq i}\right) + \epsilon_i\right] \\ a_i &\in \{0, 1\} \end{aligned}$$

- One's profit depends on the number of rival firms and local market size
- Each firm has two options: "enter" $(a_i = 1)$ or "not enter" $(a_i = 0)$

Estimation: Reduced-Form Regression

- Consider the following reduced form regression:

$$y^* = \beta_1 + \beta_2 \ln Pop + \eta \qquad \text{where } y = \begin{cases} 1 & \text{if } y^* > 0 \\ 0 & \text{otherwise} \end{cases}$$

- Parameter estimates will be consistent
- β_2 does not reflect the direct impacts of population increase on profits ($\beta_2 \neq \alpha_2$)
- Rather, it also includes the impacts of its rivals' entry triggered by population increase

Estimation: Ignoring Interaction

- Next consider the following regression:

$$
\begin{aligned}
y^* &= \alpha_1 + \alpha_2 \ln Pop - \alpha_3 \left(\sum_j a_{j \neq i} \right) + \epsilon_i \\
\text{where } y &= \begin{cases} 1 & \text{if } y^* > 0 \\ 0 & \text{otherwise} \end{cases}
\end{aligned}
$$

- In a game of **complete** information, resulting estimates are inconsistent since α_j and ϵ_i are correlated
- When ϵ_i is high, its rivals are less likely to enter, causing underestimation of α_3
- In a game of **incomplete** information, resulting estimates are incorrect since player i does not know the value of a_j when it makes its own decision

Estimation: Taking Interaction Into Account

- Want to estimate the model by explicitly taking into account interaction between players
- Possible multiple equilibria are one of the main obstacles
- Games of complete information:
  - for a given error term $\{\epsilon_i\}_{i=1}^{N}$, more than one pair of entry decisions satisfy the conditions for N.E.
- Game of incomplete information
  - more than one belief and entry policy satisfy the conditions for B.N.E.

Dealing with Multiple Equilibria

- When a model has multiple equilibria, likelihood is not well-defined
- Several ways to deal with
  - Look at a variable that is unique to all equilibria (e.g., the total number of entrants)
  - Impose some arbitrary selection rule (e.g., pick the one that maximizes total profit)
  - Bound estimators

Computational Issue: A Game of Complete Information

- Assume that the model has the unique equilibrium
- A game of COMPLETE information often requires the calculation of highly complicated integrals
- To calculate the chance of certain events, need to find all combinations of $\{\epsilon_i\}_{i=1}^{N}$ that leads this event and calculate the integrals
- Often requires simulation to calculate the integral

Computational Issue: A Game of Incomplete Information

- Calculation of the likelihood in a game of INCOMPLETE information requires the calculation of equilibrium belief
- To evaluate the likelihood function for certain parameter values,
  - calculate the equilibrium belief as a fixed point of the best response function
  - calculate the probability that each player picks the choice
  - take log and summing them up
- This algorithm is called a nested fixed-point algorithm
- Note that finding the fixed point for every set of parameter can be computationally super costly!
- See Seim (2006) for its implementation

Example:

- Let's go back to the simple example:

$$\pi\left(a_i, a_{-i}, s\right) = a_i\left[\alpha_2 \ln Pop - \alpha_3\left(\sum_j a_{j\neq i}\right) + \epsilon_i\right]$$
$$a_i \in \{0,1\}$$

- Assume firms are symmetric and play the same equilibrium strategy and hence the same equilibrium belief σ^*

- Consider applying MLE

Nested Fixed Point Algorithms

- To evaluate the likelihood for a given $(\alpha_1, \alpha_2, \alpha_3)$, need to find equilibrium belief first

$$\sigma^*(\alpha) = \Phi\left(\alpha_2 \ln Pop - \alpha_3 \sum_{k=0}^{n-1}\left[\binom{n}{k}\sigma^{*k}(1-\sigma^*)^{n-k} k\right]\right)$$

- Note that you might find more than one $\sigma^*(\alpha)$ that satisfied this equation

- Next evaluate the resulting likelihood by calculating

$$\begin{aligned} L_i &= \sigma^*(\alpha)^{1(a_i=1)}(1-\sigma^*(\alpha))^{1(a_i=0)} \\ \ln L &= \sum\left[1(a_i=1)\ln\sigma^*(\alpha) + 1(a_i=0)\ln(1-\sigma^*(\alpha))\right] \end{aligned}$$

Difficulty in Nested Fixed Point Algorithms

- Calculating equilibrium belief for a given parameter requires solving all solutions for a system of nonlinear equations
- No algorithm guarantees to find all solutions
- Need to rely on generic methods such as homotopy method
- When the model has multiple equilibria, likelihoods are not well-defined

Two-Step Methods

- Nested fixed point algorithm is not practical when games involve many players and large choice sets
- Two step methods avoid this computation problem at the expense of efficiency (but not consistency!)
- You can apply similar idea to the estimation of single-agent dynamic optimization problem as well as dynamic games

Step 1: Estimate Reduced-Form Policy Functions

- Estimate each agent's choice probabilities conditional on state variables in a flexible way
- In practice, people use logit/probit by adding state variables and their interaction terms
- Can use more flexible semiparametric method as well. See Bajari et al.
- This policy function should represent their equilibrium strategy
- Implicitly assume that players always pick the same equilibrium even under multiple equilibria

Step 2: Estimate Structural Parameters

- Assume its rivals follow the policy function estimated in the first step
- For each possible choice, we can calculate choice-specific expected payoff
- That transforms the model into the one of single-agent discrete choice model
- Estimation only involves multinomial probit/logit
- No need to find the fixed point anymore

Step 1: Estimating Policy Functions

- Consider the following a flexible logit/probit:

$$\begin{aligned} y^* &= \beta_1 + \beta_2 \ln Pop + \beta_3 (\ln Pop)^2 + \eta \\ \text{where } y &= \begin{cases} 1 & \text{if } y^* > 0 \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

- Assuming symmetry, can calculate the probability of entry conditional on population

$$\begin{aligned} \hat{p}(Pop) &= \Pr\left(\beta_1 + \beta_2 \ln Pop + \beta_3 (\ln Pop)^2 + \eta > 0\right) \\ &= 1 - \Phi\left(-\beta_1 - \beta_2 \ln Pop - \beta_3 (\ln Pop)^2\right) \end{aligned}$$

- Can calculate the distribution of its rivals' entry decisions

$$\widehat{\Pr}\left(\sum_j a_{j \neq i} = k\right) = \begin{pmatrix} n \\ k \end{pmatrix} \hat{p}(Pop)^k (1 - \hat{p}(Pop))^{n-k}$$

Step 2: Estimating Structural Parameters

- Now we can estimate structural parameters
- Estimate the following binomial discrete choice model

$$y^* = \alpha_1 + \alpha_2 \ln Pop - \alpha_3 \left[\sum_{k=0}^{N-1} \widehat{\Pr} \left(\sum_j a_{j \neq i} = k \right) k \right] + \epsilon_i$$

- Note that we transformed a model with interactions between players into single-agent discrete choice model
- We are going to use the same trick again and again

Nested Pseudo Likelihood Approach

- Aguiregabiria and Mira (2007) suggests iterating this two-step method
- Iteration does not help to increase asymptotic efficiency
- In finite sample, iteration might help to improve efficiency

Implementing NPL

- Using this updated-policy function, maximize the (pseudo) likelihood and obtain new updated parameter estimates
- Using parameter estimates and policy function as given, calculate each player's best response
- Check if updated policy functions are close enough to the previous policy function
- Iterate this process until you get convergence

Summary

- Study very basics of estimation of static games
- As games become complicated, brute-force estimation becomes impractical
- Two step method works at the expense of efficiency