PSC 585 Dynamic and Computational Modeling

Problem Set 4
May 3, 2011

1. Consider a game between the government g of a country and its k provinces (or the population therein). The (geographical or transportation) distance between any two provinces is summarized by a $k \times k$ symmetric distance matrix D, so that $d_{ij} = d(i, j)$ is the distance between province i and province j. The government of the country has one army unit at its disposal and it can choose to locate it at any one of the k provinces in each period. Also, provinces can choose to revolt or not, so that the state of the country in any period can be described by a $1 \times (k + 1)$ vector $(\ell, s) \in \{1, \ldots, k\} \times \{0, 1\}^k$. The first coordinate ℓ records the location of the army unit, while the remaining k coordinates in s record whether each of the k provinces $i = 1, \ldots, k$ experienced a successful revolt in the previous period ($s_i = 1$) or not ($s_i = 0$).

   Let the state in any period be (ℓ, s). The government observes (ℓ, s) and then decides to send the army at some province
   $$a_g \in A_g = \{1, \ldots, k\},$$
   at a cost equal to the distance between the current army location and a_g, $d(\ell, a_g)$. Similarly, each province observes (ℓ, s) and decides whether to revolt ($a_i = 1$) or not ($a_i = 0$). A province i that initiates a revolt when $s_i = 0$ incurs a cost c_r. Each province i that does not experience a revolt yields a tax revenue r_i to the government. On the other hand a province that revolts gets to keep that revenue. If a province i revolts in period t ($a_i = 1$) and the government sends the army to i in that period ($a_g = i$), then the revolt is suppressed in that province in that period (so that in the next period $s_i = 0$) and the province incurs a war cost w_i and the government a war cost w_{gi}. Thus, the government's payoff in a period with state (ℓ, s) and action profile $a = (a_g, a_1, \ldots, a_k)$ is given by
   $$u_g(\ell, s, a) = \sum_{i=1}^{k} (1 - a_i) r_i - d(\ell, a_g) - a_{a_g} w_{a_g} + \epsilon_g(a_g),$$
   while the payoff of province i is given by
   $$u_i(\ell, s, a) = a_i(r_i - (1 - s_i)c_r - I_{\{i\}}(a_g) w_i) + \epsilon_i(a_i).$$
   The terms $\epsilon_g(a_g), \epsilon_i(a_i)$ are action specific random payoff drawn from the logit version of the GEV model. All players discount the future by δ.

   A revolt in province i is successful in a period with action profile a if $a_i = 1$ and $a_g \neq i$. Thus, the probability of transition from state (ℓ, s) to state (ℓ', s') when action profile a is chosen is:
   $$p(\ell', s' \mid \ell, s, a) = \begin{cases} 1 & if \quad s'_i = a_i, i \neq a_g, s'_{a_g} = 0, \ell' = a_g, \\ \\ 0 & \qquad\qquad \text{otherwise.} \end{cases}$$
   Note that the number of states is $n = k * s^k$.

   (a) *Parameters:* Assume that the discount factor is known and fixed at $\delta = 0.8$, that $w_g = 0.4$ and that the model is parameterized by five parameters $\theta_i = 1, \ldots, 5$ so that
   $$\begin{aligned} r_i &= \theta_1 + \theta_2 x_i \\ c_r &= \theta_3 \\ w_i &= \theta_4 + \theta_5 y_i \end{aligned}$$

Here x_i is a known variable that correlates with the wealth generated by the i-th province (e.g., population, wealth, etc.), and y_i is a variable that correlates with the cost of war (e.g., terrain, etc.). The goal in what follows is to develop a procedure to recover the parameters θ from observed data.

(b) *Mapping* Φ*:* Let P be a vector of conditional choice probabilities for all players. Assuming players play according to P, then we will define the valuation operator Φ as the solution (in observable state value functions, V^P) of the following system of equations (for all ℓ, s and all i):

$$\begin{aligned}
V_g^P(\ell,s) &= \sum_{a_g=1}^{k} P[a_g|\ell,s](\tilde{u}_g^P(a_g,\ell,s;\theta)+e_g^P(a_g,\ell,s))+\delta\sum_{\ell',s'}\tilde{p}^P(\ell',s'|\ell,s)V_g^P(\ell',s') &(1)\\
V_i^P(\ell,s) &= \sum_{a_i=0}^{1} P[a_i|\ell,s](\tilde{u}_i^P(a_i,\ell,s;\theta)+e_i^P(a_i,\ell,s))+\delta\sum_{\ell',s'}\tilde{p}^P(\ell',s'|\ell,s)V_i^P(\ell',s'). &(2)
\end{aligned}$$

where

$$\begin{aligned}
e_g^P(a_g,\ell,s) &= -log(P_g[a_g|\ell,s]),\\
e_i^P(a_i,\ell,s) &= -log(P_i[a_i|\ell,s]),\\
\tilde{u}_g^P(a_g,\ell,s;\theta) &= \frac{1}{\sigma_g}\left(\sum_{i=1}^{k} P_i[0\mid \ell,s](\theta_1+\theta_2 x_i)-d(\ell,a_g)-w_g\right),\\
\tilde{u}_i^P(a_i,\ell,s;\theta) &= \begin{cases}\theta_1+\theta_2x_i-(1-s_i)\theta_3-P_g[i\mid\ell,s](\theta_4+\theta_5y_i) & if \quad a_i=1\\ 0 & if \quad a_i=0,\end{cases}\\
\tilde{p}^P(\ell',s'|\ell,s) &= \begin{cases}P_g[\ell'\mid\ell,s]\prod_{i\neq\ell'}P_i[s_i\mid\ell,s] & if \quad s'_{\ell'}=0,\\ 0 & \text{otherwise.}\end{cases}
\end{aligned}$$

The parameter σ_g is exogenously given. The solution to the above system can be expressed as follows. First, for each province $i=1,\dots,k$ and $a_i\in\{0,1\}$ we define the $n\times 5$ matrix $Z_i^P(a_i)$ whose row that corresponds to state ℓ, s is given by:

$$Z_i^P(a_i,\ell,s)=\begin{cases}\begin{pmatrix}1 & x_i & -(1-s_i) & -P_g[i|\ell,s] & -P_g[i|\ell,s]y_i\end{pmatrix} & \text{if} \quad a_i=1\\ \begin{pmatrix}0&0&0&0&0\end{pmatrix} & \text{if} \quad a_i=0\end{cases}$$

We also define the $n\times 1$ vector $E_i^P(a_i)$ whose entry that corresponds to state ℓ, s is given by:

$$E_i^P(a_i,\ell,s)=-\log(P_i[a_i|\ell,s]).$$

We similarly define corresponding matrices for the government:

$$\begin{aligned}
Z_g^P(a_g,\ell,s) &= \frac{1}{\sigma_g}\begin{pmatrix}1 & \sum_{i=1}^k P_i[0|\ell,s]x_i & -d(\ell,a_g) & -w_g & 0\end{pmatrix}\\
E_g^P(a_g,\ell,s) &= -log(P_g[a_g|\ell,s]).
\end{aligned}$$

Finally, define the conditional choice probability averages $Z_i^P, E_i^P, Z_g^P, E_g^P$ of these matrices for each player (the ℓ, s row is):

$$
\begin{aligned}
Z_i^P(\ell,s) &= (P_i[0|\ell,s]Z_i^P(0,\ell,s) + P_i[1|\ell,s]Z_i^P(1,\ell,s)),\\
E_i^P(\ell,s) &= (P_i[0|\ell,s]E_i^P(0,\ell,s) + P_i[1|\ell,s]E_i^P(1,\ell,s)),\\
Z_g^P(\ell,s) &= \sum_{a_g=1}^{k} P_g[a_g|\ell,s]Z_g^P(a_g,\ell,s),\\
E_g^P(\ell,s) &= \sum_{a_g=1}^{k} P_g[a_g|\ell,s]E_g^P(a_g,\ell,s).
\end{aligned}
$$

Then, the valuation mapping $\Phi_i(P)$ can be obtained as:

$$
\begin{aligned}
\Phi_g(P;\theta) &= (I-\delta\tilde{P}^P)^{-1}(Z_g^P \begin{pmatrix}\theta_1 & \theta_2 & 1 & 1 & 0\end{pmatrix}' + E_g^P)\\
\Phi_i(P;\theta) &= (I-\delta\tilde{P}^P)^{-1}(Z_i^P \begin{pmatrix}\theta_1 & \theta_2 & \theta_3 & \theta_4 & \theta_5\end{pmatrix}' + E_i^P), i=1,\ldots,k,
\end{aligned}
$$

where $\tilde{P}^P$ is the $n\times n$ transition matrix with entries $\tilde{p}^P(\ell',s'|\ell,s)$. Note that setting $V_i^P = \Phi_i(P;\theta)$ and $V_g^P = \Phi_g(P;\theta)$ solves the equations (1) and (2).

(c) *Mapping Ψ:* Still assuming players play according to P we will define the operator Ψ that maps from the space of conditional choice probabilities into itself. Suppose $v_i^P(a_i,\ell,s;\theta)$ is the action specific value function of province i from using action a_i at state ℓ, s, given that future play is according to P and parameters are given by θ and similarly, $v_g^P(a_g,\ell,s;\theta)$, for the government, i.e.,

$$
\begin{aligned}
v_i^P(a_i,\ell,s;\theta) &= \tilde{u}_i^P(a_i,\ell,s;\theta) + \delta\sum_{\ell',s'}\tilde{p}_i^P(\ell',s'|\ell,s,a_i)\Phi_i(P;\theta)(\ell',s')\\
v_g^P(a_g,\ell,s;\theta) &= \tilde{u}_g^P(a_g,\ell,s;\theta) + \delta\sum_{\ell',s'}\tilde{p}_g^P(\ell',s'|\ell,s,a_g)\Phi_g(P;\theta)(\ell',s'),
\end{aligned}
$$

where $\tilde{p}_i^P(\ell',s'|\ell,s,a_i)$ is the transition probability from state ℓ,s to state ℓ',s' when i chooses a_i and all other players play according to P (and similarly for $\tilde{p}_g^P(\ell',s'|\ell,s,a_g)$). Then

$$
\begin{aligned}
\Psi_g(a_g|\ell,s,P;\theta) &= \frac{\exp\{v_g^P(a_g,\ell,s;\theta)\}}{\sum_{a_g'=1}^{k}\exp\{v_g^P(a_g',\ell,s;\theta)\}}\\
\Psi_i(a_i|\ell,s,P;\theta) &= \frac{\exp\{v_i^P(a_g,\ell,s;\theta)\}}{\exp\{v_i^P(0,\ell,s;\theta)\}+\exp\{v_i^P(1,\ell,s;\theta)\}}.
\end{aligned}
$$

(d) *NPL estimator:* Suppose we have data

$$\{(\ell^t, s^t, a_1^t, a_2^t, \ldots, a_k^t, a_g^t)\}_{t=1}^T.$$

The Pseudo Likelihood function given some conditional choice probabilities P is given as a function of parameters θ as follows:

$$Q(\theta,P) = \sum_{t=1}^{T}\left(\sum_{i=1}^{k}\log(\Psi_i(a_i^t|\ell^t,s^t,P;\theta) + \log(\Psi_g(a_g^t|\ell^t,s^t,P;\theta))\right).$$

The Nested Pseudo-Likelihood (NPL) estimator of Aguiregabiria and Mira (2007) proposes the following iterative scheme:

i. Initialize with some P^0 and at the j-th iteration:

ii. Set

$$\theta^j = \arg\max_{\theta}\{Q(\theta, P^{j-1})\}.$$

iii. Set

$$P^j = \Psi(P^{j-1}; \theta^j).$$

iv. Upon convergence in iteration q, let $\hat{\theta}_{NPL} = \theta^q$.

Write code that implements a version of the NPL estimator above. Step (iii) is straightforward (actually, I provide code for it) and is described in parts (a) to (c) above, for any given θ^j. In part (e) below I describe a method to partially execute the maximization in step (ii).

(e) *Maximization of Pseudo-Likelihood:* Instead of maximizing the pseudo-likelihood, which fuses a series of binary logits with a mulinomial logit objective, we will ignore the data on the choice of the government a_g^t and instead maximize the *partial pseudo-likelihood*:[1]

$$\tilde{Q}(\theta, P) = \sum_{t=1}^{T}\sum_{i=1}^{k} \log(\Psi_i(a_i^t|\ell^t, s^t, P; \theta).$$

As is explained below, this can be done by essentially estimating a single logit model (for which canned routines can be applied). First, note that we can rewrite the conditional probability that province i revolts generated from the mapping Ψ in standard logit form

$$\Psi_i(1|\ell, s, P; \theta) \quad = \quad \frac{\exp\{v_i^P(1, \ell, s; \theta) - v_i^P(0, \ell, s; \theta)\}}{1 + \exp\{v_i^P(1, \ell, s; \theta) - v_i^P(0, \ell, s; \theta)\}}. \tag{3}$$

Furthermore, if we let $\tilde{P}_i^P(a_i)$ be the transition matrix with transition probabilities $\tilde{p}_i^P(\ell', s'|\ell, s, a_i)$, we can write the difference in province i's action-specific value functions more compactly (in $n \times 1$ vector form) as:

$$v_i^P(1;\theta) - v_i^P(0;\theta) = W_i^P \begin{pmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \end{pmatrix} + C_i^P = [W_i^P C_i^P] \begin{pmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \\ 1 \end{pmatrix},$$

where W_i^P is a $n \times 5$ matrix

$$W_i^P = (Z_i^P(1) + \delta(\tilde{P}_i^P(1) - \tilde{P}_i^P(0))(I - \delta\tilde{P}^P)^{-1} Z_i^P)$$

and C_i^P is a $n \times 1$ vector

$$C_i^P = \delta(\tilde{P}_i^P(1) - \tilde{P}_i^P(0))(I - \delta\tilde{P}^P)^{-1} E_i^P.$$

It follows that the choice of the province to revolt follows a standard logit model with explanatory variables given in the matrix $[W_i^P C_i^P]$ and such that the coefficient vector has the last entry constrained to equal 1. So, in order to execute (the modified) step (ii) of the NPL algorithm:

i. Compute the matrices $W_i^{P^{j-1}}, C_i^{P^{j-1}}$ for each i.

ii. Create data matrices W_d (which is a $k * T \times 5$), C_d, and Y_d (both $k * T \times 1$ vectors) from the above by, for each i and each t, setting:

$$\begin{aligned} W_d((i-1)*T+t, :) &= W_i^{P^{j-1}}(\ell^t, s^t), \\ C_d((i-1)*T+t, 1) &= C_i^{P^{j-1}}(\ell^t, s^t), \\ Y_d((i-1)*T+t, 1) &= a_i^t. \end{aligned}$$

[1]This is, of course, inefficient, but the estimator maintains other properties of $\hat{\theta}_{NPL}$.

iii. obtain estimates of $\theta_1, \ldots, \theta_5$ by running the MATLAB command:

$$\text{glmfit}(W_d, Y_d, \text{'binomial'}, \text{'constant'}, \text{'off'}, \text{'offset'}, C_d).$$

(f) *Supporting material:* Material uploaded on blackboard for this problem consists of the following:

- The data is provided in file *FinalData.mat* which contains a $T \times (k+2)$ matrix, with $T = 2,000$. The first column records the state ℓ, s enumerated according to the order of the coordinates (so that the first 2^k coordinates correspond to $\ell = 1$, then $\ell = 2$, etc. The last column contains the action of the government.; k in between columns contain the actions of the provinces.
- Function *Ptilde.m* takes conditional choice probabilities P as an input and returns transition matrix $\tilde{P}$. You can use this function in order to compute $\tilde{P}_i(a_i)$ by appropriately adjusting choice probabilities in the input P. Here and in the rest of the code, the conditional choice probabilities P are stored in two matrices Pg and Pp. The former is of size $n \times k$ and has the choice probabilities of the government (state in row, province a_g in column). Pp is $n \times 2 * k$ and each pair of successive columns has the probability of choosing $a_i = 0$ and $a_i = 1$, respectively for $i = 1, \ldots, k$.
- Function *Phigprov.m* takes conditional choice probabilities P and θ as an input and returns values V^P. This is the mapping Φ of part (b).
- Function *NewP.m* takes conditional choice probabilities P and θ as an input and returns new conditional choice probabilities. This is the mapping Ψ of part (c).
- All three of the above functions take an extra argument: *model*. This is provided in file *FinalModel.mat* which contains a data structure *model*. E.g., $model.k = k = 7$, $model.m = 2^k = 128$, $model.n = n = 896$, $model.delta = \delta = 0.8$, $model.D = D$, etc.