# Persistent Unobservables

C.Conlon - Adapted from M. Shum

October 15, 2019

Grad IO

# Persistent Unobserved Heterogeneity

Suppose we think about a model with a friction such as a switching cost.

- If $y_{it} \neq y_{i,t-1}$ you pay a switching cost $F_i$.
- How do we use data to tell apart large switching costs $F_i \gg 0$ from persistent tastes $Cov(\epsilon_{i,t}, \epsilon_{i,t-1}) > 0$ ?
- The conditional independence assumption tells us it has to be the switching cost not the autocorrelated unobservables.
- This is probably why people don't like this assumption.

# Discrete Unobserved Types

- Up until now we consider models satisfying Rust's conditional independence assumption on the $\varepsilon$'s. This rules out persistence in unobservables which are economically meaningful.
- Suppose there are two types of buses good $(s_i = g)$ and bad $(s_i = b)$.
- Assume that this is known to HZ but not the econometrician.
- Single period utility now depends on $s_i$ so $u(x_{it}, s_i, d_{it}; \theta)$ unobserved state variable.
- In case of the nested fixed point algorithm, this unobserved persistent heterogeneity is not a big problem as we can solve for the value function (and expected policy functions) given the state variables and integrate it out in the likelihood

# Unobserved State Variables: What happened?

$$Pr(d_{i1}, \ldots, d_{iT} | x_{i1}, \ldots, x_{iT}) = \sum_{s} \prod_{t=1}^{T} Pr(d_{it} | x_{it}) p(s_i)$$

- Conditional on $s_i$ replacement decisions are independent across $t$ given $x_{it}$.
- The resulting likelihood is just a finite mixture model.
- These can be hard to solve when both $s_i$ and its distribution $p(s_i)$ are unknown.
- Arcidiacono and Miller (2011) provide theoretical results for these types of problems.

# A much earlier application

**Pakes (1986): Patents as Options**

How much are patents worth? Valuable for optimal patent length and design? Sufficient incentive for innovation?

- $Q_A$: value of patent at age $A$
- Goal of paper is to estimate $Q_A$ using data on their renewal. $Q_A$ is inferred from patent renewal process via revealed preference for patent renewal behavior.
- Treat renewal systems as exogenous (in Europe)

Timing

- For $a = 1, \ldots, L$ a patent can be renewed by paying the fee $c_a$.
- At age $a = 1$ patent holder gets $r_1$ from patent
- Decide whether or not to renew (pay $c_1$ and go to $a_2$).
- At age $a = 2$ get $r_2$ from patent
- and so on...

# Pakes (1986)

Gives us the value function

$$V \equiv \max_{t \in [a,L]} \sum_{a'=1}^{L-a} \beta^{a'} R(a+a')$$

$$R(a) = \begin{cases} r_a - c_a, & \text{if } t \geq a \text{ when you hold patent} \\ 0 & \text{if } t < a \text{ after patent expires} \end{cases}$$

- $t$ above denotes the age which allows the patent to expire and is the choice variable. Another optimal stopping problem.
- $R(a)$ are the profits from year $a$. This is a controlled stochastic process. It is random but affected by the actions of the agent.

# Pakes (1986)

- The maximum age $L$ is finite so it is finite-horizon DP.
- The single period revenue $r_a$ is the state variable.
- We can solve the problem with *backward recursion*.

$$V_a(r_a) = \max\{0, Q_a \equiv r_a + \beta E[V_{a+1}(r_{a+1})|\Omega_a] - c_a\}$$

- Renew iff $Q_a - c_a > 0$.
- $\Omega_a$: history up to age $a = \{r_1, r_2, \ldots, r_a\}$.
- Expectation is over $r_{a+1}|\Omega_a$. The sequence of conditional distributions $G_a \equiv F(r_{a+1}|\Omega_a)$, $a = 1, 2, \ldots$ is an important component of model specification.

$$r_{a+1} = \begin{cases} 0 & \text{w. prob } \exp(-\theta r_a) \\ \max(\delta r_a, z) & \text{w. prob } 1 - \exp(-\theta r_a) \end{cases}$$

# Pakes (1986)

Model has the following parameters

- density of $z$ $q_a = \frac{1}{\sigma_a} \exp[-(\gamma + z)/\sigma_a]$ and $\sigma_a = \phi^{a-1}\sigma$, for $a = 1, \ldots, L-1$.
- $(\delta, \theta, \gamma, \phi, \sigma)$ are the structural parameters of the model
- Break down the model period by period and decide whether or not to renew if $Q_a = r_a +$ "option value".
- Option value is about keeping the patent alive in case it pays off in the future.

Implications

- Drop out at age $a$ if $c_a > Q_a$
- Optimal decision is characterized by cutoff points $Q_a > c_a \Leftrightarrow r_a > \overline{r}_a$ (Key assumptions is $Q_a$ increasing /single crossing )
- Cutoff points are increasing sequence $\overline{r}_a < \overline{r}_{a+1} < \ldots < \overline{r}_{L-1}$.

# Estimation

Instead of using Pakes' notation $r_t$ for the patent revenue. We will use the generic Rust notation of $\epsilon_t$ the unobserved state variable, and $i_t$ to denote the choice (renewal).

- For a single patent $\tilde{T}$ denotes the age at which it is allowed to expire. Let $T = \min(L-1, \tilde{T})$ denote the period sins which the agent makes a renewal decision where we model the agent's choice.
- $\epsilon$ follows a first-order Markov process $F(\epsilon'|\epsilon)$
- Age-specific policy function by $i_t^*(\epsilon)$.

Likelihood function is

$$l(i_1, \ldots, i_T|\epsilon_0, i_0, \theta) = \prod_{t=1}^{T} Prob(i_t|i_0, \ldots, x_{t-1}, i_{t-1}; \epsilon_0, \theta)$$

Serial correlation in $\epsilon$ means there is dependence among $i_t, i_{t-2}$ even after conditioning on $x_{t-1}, i_{t-1}$.

# Simulation

- It might seem like we were stuck since it no longer has a closed form. However, we can simulate the "outer loop" of the nested fixed point routine given a guess of $i_t^*(\epsilon, \theta)$.
- Because $\epsilon$ is serially correlated we need to start with an initial $\epsilon_0$ (or distribution) and assume that it is known. This is the initial conditions problem of finite MDPs.
- Note that simulation is part of the "outer loop" of nested fixed point estimation routine. So at the point when we simulate, we already know the policy functions $i_t^*(\epsilon, \theta)$ (How would you compute this?)

# Naive Frequency Simulator (Don't do this...)

Go back to the full likelihood function (condition on initial $\epsilon_0$ for serial correlation):

$$l(i_1, \ldots, i_T | i_0, \epsilon_0, \theta) = Pr(i_t^*(\epsilon_t, \theta) = i_t, \forall t = 1, \ldots, T)$$

Need to take probability over distribution of $(\epsilon_1, \ldots, \epsilon_T | \epsilon_0)$. Let $F(\epsilon_{t+1} | \epsilon_t, \theta)$ then the above probability can be expressed as the integral:

$$\int \cdots \int \prod_t \mathbf{1}(i_t^*(\epsilon_t, \theta) = i_t) \prod_t dF(\epsilon_t | \epsilon_{t-1}; \theta)$$

Simulate by drawing sequences of $(\epsilon_t)$.

# Naive Frequency Simulator (Don't do this...)

Simulate by drawing sequences of $(\epsilon_t)$ and for each draw $s = 1, \ldots, S$ we take as initial values $(x_0, i_0, \epsilon_0)$ then

- Generate $(\epsilon_1^s, i_1^s)$
  1. Generate $\epsilon_1^s \sim F(\epsilon_1|\epsilon_0)$
  2. Compute $i_1^s = i_1^*(\epsilon_1^s; \theta)$
- Generate $(\epsilon_2^s, i_2^s)$
  1. Generate $\epsilon_2^s \sim F(\epsilon_2|\epsilon_1^s)$
  2. Subsequently compute $i_2^s = i_2^*(\epsilon_2^s; \theta)$
- And so on, up to $(\epsilon_T^s, i_T^s)$.

# Naive Frequency Simulator (Don't do this...)

And for the case where $(i, x)$ are both discrete (Rust) we can approximate:

$$l(i_t, \ldots, i_T | \epsilon_0, i_0; \theta) \approx \frac{1}{S} \sum_s \prod_{t=1}^{T} \mathbf{1}(i_t^s = i_t)$$

Frequency of simulated sequences which match observed sequence. $T$ long or $S$ small you're in trouble (non-smooth).

# Importance Sampling: Particle Filtering

- We can use importance sampling to simulate the likelihood function.
- This is not straightforward given time dependence in $(i_t, \epsilon_t)$
- Consider particle filtering approach from Fernandez-Villaverde and Rubio-Ramirez (2007) or Flury and Shehard (2008) (non-Gaussian Kalman filtering).
- A more up to date take: Blevins (2016) : Sequential Monte Carlo Methods for Estimating Dynamic Microeconomic Models

# Importance Sampling: Particle Filtering

- Evolution of utility shocks $\epsilon_t|\epsilon_{t-1} \sim f(\epsilon'|\epsilon)$. Ignore dependence of distribution of $\epsilon$ on age $t$ for convenience.
- As before, the policy function is $i_t = i^*(\epsilon_t)$
- Let $\epsilon^t \equiv \{\epsilon_1, \ldots, \epsilon_t\}$.
- The initial values of $y_0$ and $\epsilon_0$ are known

Go back to the factorized likelihood

$$\begin{aligned} l(y^T|y_0, \epsilon_0) &= \prod_{t=1}^{T} l(y_t|y^{t-1}, y_0, \epsilon_0) = \prod_{t=1} \int l(y_t|\epsilon^t, y^{t-1}) p(\epsilon^t|y^{t-1}) d\epsilon^t \\ &\approx \frac{1}{S} \sum_s l(y_t|\epsilon^{t|t-1,s}, y^{t-1}) \end{aligned}$$

We omit conditioning on $(\epsilon_0, y_0)$ for convenience, and $\epsilon^{t|t-1,s}$ is a simulated draw of $\epsilon^t \sim p(\epsilon^t|y^{t-1})$.

# Importance Sampling: Particle Filtering

Let's look more closely at the last line:

- first term: $l(y_t, |\epsilon^t, y^{t-1})$ we can calculate for a value of $\epsilon_t$

$$l(y_t|\epsilon^t, y^{t-1}) = p(i_t|\epsilon^t, y^{t-1}) = p(i_t|\epsilon_t) = \mathbf{1}(i(\epsilon_t) = i_t)$$

- the second term $p(\epsilon^t|y^{t-1})$ is generally not obtainable in closed form. So numerical integration is not feasible. Particle filtering let's us draw $\epsilon^t$ from this distribution for every period $t$.

Particle filtering proposes a recursive approach to draw sequences $p(\epsilon^t|y^{t-1})$ for every $t$

# Particle Filtering Algorithm

**First period:** $t = 1$ In order to simulate the integral corresponding to the first period we need to draw from $p(\epsilon^1|y^0, \epsilon_0)$ (easy).

- We draw $\{\epsilon^{1|0,s}\}_{s=1}^{S}$ according to $f(\epsilon'|\epsilon_0)$.
- The notation $\epsilon^{1|0,s}$ makes it explicit that the $\epsilon$ is a draw from $p(\epsilon^1|y^0, \epsilon_0)$
- Use the $S$ draws we can evaluate the period $t = 1$ likelihood.

**Second period:** $t = 2$. We need to draw from $p(\epsilon^2|y^1)$ factorize as:

$$p(\epsilon^2|y^1) = p(\epsilon^1|y^1) \cdot p(\epsilon_2|\epsilon^1) \text{ recall } \epsilon^2 \equiv \{\epsilon_1, \epsilon_2\}$$

# Filtering Step

Getting a draw from $p(\epsilon^1|y^1)$, given that we already have draws $\{\epsilon^{1|0,s}\}$ from $p(\epsilon^1|y_0)$, from the previous period $t = 1$, is the heart of particle filtering. We use the principle of importance sampling: by Bayes' Rule

$$p(\epsilon^1|y^1) \propto p(y_1|\epsilon^1, y^0) \cdot p(\epsilon^1|y^0)$$

Hence, if our desired sampling density is $p(\epsilon^1|y^1)$, but we actually have draws $\{\epsilon^{1|0,s}\}$ from $p(\epsilon^1|y^0)$, then the importance sampling weight for the draw $\epsilon^{1|0,s}$ is proportional to

$$\tau_1^s \equiv p(y_1|\epsilon^{1|0,s}, y^0)$$

Note that this coincides with the likelihood contribution for period 1, evaluated at the shock $\epsilon^{1|0,s}$.The SIR algorithm in Rubin (1988) proposes that making $S$ draws with replacement from samples $\{\epsilon^{1|0,s}\}_{s=1}^S$, using weights proportional $\tau_1^s$ yields draws from the desired density $p(\epsilon^1|y^1)$ which we denote $\{\epsilon^{1|0,s}\}_{s=1}^S$.

# Prediction Step

For the second term in the equation: we simply draw one $\epsilon_2^s$ from $f(\epsilon'|\epsilon^{1,s})$, for each draw $\epsilon^{1,s}$ from the filtering step. This is the **prediction** step.

By combining the draws from these two terms, we have $\{\epsilon^{2|1,s}\}_{s=1}^{S}$.which is $S$ drawn sequences from $p(\epsilon^2|y^1)$. Using these $S$ draws, we can evaluate the simulated likelihood for period 2

# Prediction Step (Continued)

**Third period,** $t = 3$: start again by factoring

$$p(\epsilon^3|y^2) = p(\epsilon^2|y^2) \cdot p(\epsilon^3|\epsilon^2)$$

As above, drawing from requires filtering the draws $\{\epsilon^{2|1,s}\}_{s=1}^{S}$, from the previous period $t = 2$, to obtain draws $\{\epsilon^{2,s}\}_{s=1}^{S}$. Given these draws, draw $\epsilon_3^s \sim f(\epsilon'|\epsilon^{2,s})$ for each $s$.

And so on. By the last period $t = T$, you have

$$\left\{\{\epsilon^{t|t-1,s}\}_{s=1}^{S}\right\}_{t=1}^{T}$$

# Prediction Step (continued)

Hence the factorized likelihood can be approximated by simulation as:

$$\prod_t \frac{1}{S} \sum_s l(y_t | \epsilon^{t|t-1,s}, y^{t-1})$$

As noted above, the likelihood term $l(y_t | \epsilon^{t|t-1,s}, y^{t-1})$ coincides with the simulation weight $\tau_t^s$. Hence the simulated likelihood can also be constructed as:

$$\log l(y^T | y_0, \epsilon_0) = \sum_t \log \left\{ \frac{1}{S} \sum_s \tau_t^s \right\}$$

# Particle Filtering (Summary)

- Start by drawing $\{\epsilon^{1|0,s}\}_{s=1}^{S}$ from $p(\epsilon^1|y^0, \epsilon_0)$.
- In period $t$, we start with $\{\epsilon^{t-1|t-2,s}\}_{s=1}^{S}$ draws from $p(\epsilon^{t-1}|y^{t-2}, \epsilon_0)$.
  1. **Filter step:** Calculate proportion weights $\tau_{t-1}^{s} \equiv p(y_{t-1}|\epsilon^{t-1|t-2,s}, y^{t-2})$ using $p(i_t|\epsilon_t)$. Draw$\{\epsilon^{t-1|t-1,s}\}_{s=1}^{S}$ by resampling from $\{\epsilon^{t-1|t-2,s}\}_{s=1}^{S}$ with weights $\tau_{t-1}^{s}$.
  2. **Prediction step:** Draw $\epsilon_t^s$ from $p(\epsilon_t|\epsilon^{t-1|t-1,s})$ , for $s = 1, \ldots, S$. Combine to get $\{\epsilon^{t|t-1,s}\}_{s=1}^{S}$.
- Set $t = t + 1$ and go back to step 2. Stop when $t = T + 1$.

The difference is that the crude simulator draws $S$ sequences and puts zero weight on those which don't match the observed sequence. In each period $t$ we just keep sequences where predicted choices match observed choice of *that period*. This is more accurate likelihood as long as $S$ is large enough that we don't have all the weight on a single sequence in period $t$.

# References


- Fernandez-Villaverde, J., and J. Rubio-Ramirez (2007): "Estimating Macroeconomic Models: A Likelihood Approach," Review of Economic Studies, 74, 1059-1087.
- Flury, T., and N. Shephard (2008): "Bayesian inference based only on simulated likelihood: particle filter analysis of dynamic economic models," manuscript, Oxford University
- Pakes, A. (1986): "Patents as Options: Some Estimates of the Value of Holding European Patent Stocks," Econometrica, 54(4), 755-84.
- Rubin, D. (1988): "Using the SIR Algorithm to Simulate Posterior Distributions," in Bayesian Statistics 3, ed. by J. Bernardo, M. DeGroot, D. Lindley, and A. Smith. Oxford University Press.