Dynamic discrete choice models: full solution methods

Olivier De Groote

TSE

MRes EEE, 2021-2022

Outline

Dynamic optimization I

- A crucial assumption in static models is that agents choose the option with the highest utility $u_{ij} + \epsilon_{ij}$

- Now let's add a time dimension to describe utility in period t

$$u_{ijt} + \epsilon_{ijt}$$

- We could still assume agents choose the option with the highest utility (in each period)

- But what if their choice today impacts their future utility too?

Dynamic optimization II

- Example: go to college (c) or work (0) in period t, work in period $t+1$
  - If they dislike studying, $(u_{ict} + \epsilon_{ict}) - (u_{i0t} + \epsilon_{i0t})$ will be small
  - However, if college graduates earn higher wages, $E\left[u_{i0t+1} + \epsilon_{i0t+1} | d_{it} = c\right] - E\left[u_{i0t+1} + \epsilon_{i0t+1} | d_{it} = 0\right]$ will be large
- Therefore assuming agents choose the option with the highest $u_{ijt} + \epsilon_{ijt}$ is unrealistic
- Solution: define the conditional value function

$$v_{ijt} = u_{ijt} + \sum_{\tau=t+1}^{T} \beta^{\tau-t} E\left[u^*_{i\tau} + \epsilon^*_{i\tau} | d_{it} = j\right]$$

  with $u^*_{i\tau} + \epsilon^*_{i\tau}$ the utility of the chosen alternative of i in period τ
- let agents choose the option with the highest value of $v_{ijt} + \epsilon_{ijt}$

Dynamic optimization III

- Adds some complexity
  - Even with some functional form assumption on u_{ijt} we need to know more before we can write our (logit) probabilities as in a static model
  - β -> new parameter appears, can we identify it from data?
  - T -> what's the time horizon?
  - $E[]$
    - -> how do we integrate over future unobserved heterogeneity (ϵ_{ijt})?
    - -> what if impact of choice today is stochastic?
- A crucial contribution is Rust (1987)
  - Rust proposes some assumptions that look restrictive but help a lot to make a dynamic model tractable
  - Recent papers have relaxed the restrictive assumptions but in a way that we can still benefit a lot from Rust's insights

Rust bus engine model (simplified) I

- Paper: Rust, J. 1987. “Optimal Replacement of GMC Bus Engines: An Empirical Model of Harold Zurcher”. *Econometrica* 55 (5): 999-1033.
- I use the Rust model to explain how to estimate a dynamic model but I don’t show the exact model Rust estimates
  - Different notation to be consistent with rest of course (and most of the recent papers)
  - Different functional form assumptions on how variables transition over time
  - See paper if you’re interested in the application (and the details)
- Harold Zurcher, superintendent of maintenance at the Madison Metropolitan Bus Company has to decide when to replace a bus engine

Rust bus engine model (simplified) II

- Let the (flow) utility of regular maintenance ($j = 0$) be given by

$$u_{i0t} = u_0(x_{it}) = \alpha_x x_{it}$$

- The utility of replacing the engine instead ($j = 1$) is

$$u_{i1t} = u_1(x_{it}) = \alpha_{RC}$$

- x_{it} is the mileage of a bus engine, $x_{it} = 0$ for a new engine and we expect $\alpha_x < 0$ and $\alpha_{RC} < 0$

Rust bus engine model (simplified) III

- x_{it} is like an individual characteristic in the static model but now it can change over time, in the dynamic context we usually call it an "observed state variable"
- Sometimes ϵ_{ijt} is called the "unobserved state variable" but I call it taste shock to avoid confusion with another type of unobserved state variable that directly enters u (we will cover this later)
- Assume Harold Zurcher maximizes lifetime utility and therefore chooses the option with the highest $v_{ijt} + \epsilon_{ijt}$
- As we did in a static model, we can assume ϵ_{ijt} is extreme value type 1 and use the logit probabilities in the likelihood function:

$$Pr(d_{ijt} = 1 | x_{it}) = \frac{exp(v_{ijt})}{\sum_{j'} exp(v_{ij't})}$$

Rust bus engine model (simplified) IV

- So all we need to know are the conditional value functions:

$$v_{i0t} = \alpha_x x_{it} + \sum_{\tau=t+1}^{T} \beta^{\tau-t} E\left[u^*_{i\tau} + \epsilon^*_{i\tau} | d_{it} = 0\right]$$
$$v_{i1t} = \alpha_{RC} + \sum_{\tau=t+1}^{T} \beta^{\tau-t} E\left[u^*_{i\tau} + \epsilon^*_{i\tau} | d_{it} = 1\right]$$

- Note that with $\beta = 0$ this is simply our static model
- With $\beta \in (0, 1)$ things get more complicated
- An important simplification results from Rust's (1987) conditional independence assumption:

$$p(x_{it+1}, \epsilon_{it+1} | x_{it}, \epsilon_{it}, d_{it}) = g(\epsilon_{it+1} | x_{it}) f(x_{it+1} | x_{it}, d_{it})$$

Rust bus engine model (simplified) V

- $g()$ part: in practice, we often simplify further and assume iid ϵ_{it+1}, its expected value at time t is then given by the ex ante value function $\bar{V}_{t+1}(x_{it+1}) = \int V_{t+1}(x_{it+1}, \epsilon_{it+1}) g(\epsilon_{it+1}) d\epsilon_{it+1}$ with $V_{t+1}(x_{it+1}, \epsilon_{it+1})$ the value of behaving optimally in $t+1$
- $f()$ part: implies unobservables do not influence the way mileage changes (except through the observed choice)
- We can then write the conditional value functions as simple Bellman equations without ϵ

$$\begin{aligned} v_{ijt} &= u_j(x_{it}) + \beta E \bar{V}_{t+1}(x_{it+1}) \\ &= u_j(x_{it}) + \beta \int \bar{V}_{t+1}(x_{it+1}) f(x_{it+1} | x_{it}, d_{it}) dx_{it+1} \end{aligned}$$

Rust bus engine model (simplified) VI

- Now we still need to specify $f(x_{it+1}|x_{it}, d_{it})$ and we need to calculate $\bar{V}_{t+1}(x_{it+1})$
- To keep things simple, let's assume
  - if $d_{it} = 1$: $x_{it+1} = 0$
  - if$d_{it} = 0$: $x_{it+1} = x_{it} + 1$ with prob of 50% and $x_{it+1} = x_{it}$ with prob 50%
- We then get:

$$v_{i0t} = \alpha_x x_{it} + \beta\left(0.5\bar{V}_{t+1}(x_{it+1} = x_{it}) + 0.5\bar{V}_{t+1}(x_{it+1} = x_{it} + 1)\right)$$
$$v_{i1t} = \alpha_{RC} + \beta\bar{V}_{t+1}(x_{it+1} = 0)$$

Rust bus engine model (simplified) VII

- Now we only need to know what $\bar{V}_{t+1}(x_{it+1})$ looks like, therefore we need to know T
- Rust assumes $T = \infty$ but let's first look at the two-period case: $T = 2$.
- If we assume taste shocks are distributed extreme value type 1, we can use the familiar logsum formula:

$$\begin{aligned}\bar{V}_T(x_{iT}) &= \gamma + ln \sum_{j'} exp(v_{ij'T}) \\ &= \gamma + ln \sum_{j'} exp(u_{ij'T})\end{aligned}$$

- with γ Euler's constant
- where the last equality follows from the fact that T is the final period (i.e. the model stops being dynamic)

Rust bus engine model (simplified) VIII

- substituting this back into our value functions at $t = T - 1 = 1$:

$$
\begin{aligned}
v_{i0t} &= \alpha_x x_{it} + \beta[0.5\,(\gamma + ln\,(exp(u_0(x_{it+1} = x_{it})) + exp(u_1(x_{it+1} = x_{it}))))\\
&\quad + 0.5\,(\gamma + ln\,(exp(u_0(x_{it+1} = x_{it} + 1)) + exp(u_1(x_{it+1} = x_{it} + 1))))]\\
&= \alpha_x x_{it} + \beta[0.5\,(\gamma + ln\,(exp(\alpha_x x_{it}) + exp(\alpha_{RC})))\\
&\quad + 0.5\,(\gamma + ln\,(exp(\alpha_x(x_{it} + 1)) + exp(\alpha_{RC})))]\\
v_{i1t} &= \alpha_{RC} + \beta\,(\gamma + ln\,(exp(u_0(x_{it+1} = 0)) + exp(u_1(x_{it+1} = 0))))\\
&= \alpha_{RC} + \beta\,(\gamma + ln\,(exp(0) + exp(\alpha_{RC})))
\end{aligned}
$$

- we did it! can now fill in our logit probabilities as we only see parameters to estimate and observables
- what if $T > 2$? solve backwards from T to $t = 1$

Rust bus engine model (simplified) IX

- what if $T = \infty$? look for a fixed point
  - need to assume stationarity: $\bar{V}_t(x_{it}) = \bar{V}(x_{it})$
  - now apply the logsum formula not only to $t+1$ but also to t

$$\begin{aligned}\bar{V}(x_{it}) &= \gamma + ln\left(exp(v_{i0t}) + exp(v_{i1t})\right)\\ &= \gamma + ln[exp\left(\alpha_x x_{it} + \beta\left(0.5\bar{V}(x_{it+1} = x_{it}) + 0.5\bar{V}(x_{it+1} = x_{it}+1)\right)\right)\\ &+ exp\left(\alpha_{RC} + \beta\bar{V}(x_{it+1} = 0)\right)]\end{aligned}$$

  - Fixed point iteration to get $\bar{V}$ for each possible value of x_{it}
  - Note also how great it is that we got rid of the ϵ before doing this!
  - But can still be hard if x_{it} takes a lot of values or is continuous... (also with finite T, see approximation method in Keane and Wolpin (1994))

Rust bus engine model (simplified) X

- Also don't forget that we don't know the parameters of the model yet, so for each candidate of the parameters in our ML procedure (outer loop), we need to search for the fixed point (inner loop)...

Rust bus engine model (simplified) XI

- what if we do not know the stochastic process of x_{it}?
  - Estimate it.
  - E.g. instead of 0.5 we could write $\frac{exp(\rho x_{it})}{1+exp(\rho x_{it})}$ and $1-\frac{exp(\rho x_{it})}{1+exp(\rho x_{it})}$ with ρ an additional parameter to estimate
  - The joint loglikelihood then becomes

$$
\begin{aligned}
lnL &= ln\prod_{it} L_{1it}(\alpha,\rho)L_{2it}(\rho) \\
&= \sum_{it}(lnL_{1it}(\alpha,\rho)+lnL_{2it}(\rho)) \\
&= \sum_{it} lnL_{1it}(\alpha,\rho)+\sum_{it} lnL_{2it}(\rho)
\end{aligned}
$$

  - with $L_{1it}(\alpha,\rho)$ the likelihood contribution for the choices and $L_{2it}(\rho)$ the likelihood contribution of the process of x

Rust bus engine model (simplified) XII

- Because $f(x_{it+1}|d_{it}, x_{it}; \rho)$ (and therefore $L_{2it}(\rho)$) does not depend on α (or ϵ), we can consistently estimate ρ in a first stage by only using the second part of the loglikelihood function

References I

Michael P. Keane and Kenneth I. Wolpin.
The Solution and Estimation of Discrete Choice Dynamic Programming Models by Simulation and Interpolation: Monte Carlo Evidence.
The Review of Economics and Statistics, 76(4):648, November 1994.

John Rust.
Optimal Replacement of GMC Bus Engines: An Empirical Model of Harold Zurcher.
Econometrica, 55(5):999–1033, September 1987.