EC8559, Problem Set 4: Single-Agent Dynamic Optimization

Julie Holland Mortimer

Due Date: December, 2023

In this problem set, we explore the computation and estimation of single-agent dynamic discrete-choice model of Rust (1987). Be sure to read Rust's 2000 Nested Fixed Point Algorithm Documentation Manual (Version 6). Sections 2.3 and 2.4 are especially relevant, as are Rust's comments about the inner loop.

Question 1:

(i) Explain the following two equations:

$$EV^1(x,i) = \int_y \log\left\{\sum_{j\in C(y)} \exp[u(y,j;\theta) + \beta EV^0(y,j)]\right\} p(dy|x,i) \tag{1}$$

$$P(i_t|x_t;\theta) = \frac{\exp(u(x_t,i_t;\theta) + \beta EV(x_t,i_t;\theta))}{\sum_{i=0,1}\exp(u(x_t,i;\theta) + \beta EV(x_t,i;\theta))}. \tag{2}$$

(ii) In order to compute equation (1) by iteration, you need to specify the transition matrix with parameters $\theta_3 = p, q$. Notes that Rust discretizes the observable state variable, mileage, into 90 bins. Assume that

$$\triangle x_t = \begin{cases} [0, 5000) & \text{w/prob } p \\ [5000, 10000) & \text{w/prob } q \\ [10000, \infty) & \text{w/prob } 1-p-q \end{cases}$$

So the relevant 'transition' is a move between these states. Using the same discretization as Rust (into 90 intervals), and using the parameter estimates from the top of Table IX in Rust's paper (i.e., $RC = 9.7558$, $c = 2.6275$, $\theta_{31} = 0.3489$, $\theta_{32} = 0.6394$ with $\beta = 0.9999$), graph $EV(x,i;\theta)$ for $i = 0$.

Tips for Question 1:

1. Build a transition matrix with special focus on the 89th and 90th rows. Don't try to replace values in a vector.
2. Recenter the Bellman equation after each iteration. (This was the main reason that students' results failed to converge in the past.)

Question 2:

The dataset "ps4_data.mat" contains the following information:

Column	1	2	3	4	5	6
Variable	bus id	group	year	month	mileage	Δmileage

Replicate the above parameters using Rust's nested fixed point algorithm.

Tips for Question 2:

1. For replicating p and q, your first instinct might be to use the Δ mileage variable and count the number of times this variable, conditional on non-replacement, takes values in $[0, 5000), [5000, 10000)$ and $[10,000, \infty)$, then divide each of these counts by the number of times the engine was not replaced. Instead, you should continue using the 90 discrete intervals and use the jumps in these intervals for each bus over time to calculate p and q.
2. In Matlab, fminunc tends to work better than fminsearch in this problem.