Industrial Organization, Problem Set 3: Dynamics

Joe Wilske

December 2023

1. Firms' sequence problem:

$$\max_{i_t \in \{0,1\}} \left[\mathbb{E} \sum_{t=1}^{\infty} \beta^{t-1} \Pi(a_t, i_t, \varepsilon_{0t}, \varepsilon_{1t}, \theta) \right],$$

   where

$$\Pi(a_t, i_t, \varepsilon_{0t}, \varepsilon_{1t}, \theta) = i_t(R + \varepsilon_{1t}) + (1 - i_t)(\mu a + \varepsilon_{0t})$$

2. Value function as a Bellman Equation:

$$\begin{aligned} V(a_t, \theta) &= \max_{i_t \in \{0,1\}} \left[\Pi(a_t, i_t, \varepsilon_{0t}, \varepsilon_{1t}, \theta) + \beta \mathbb{E} V(a_t, i_t, \theta) \right] \\ &= \max_{i_t \in \{0,1\}} \left[\bar{V}_0(a_t, i_t = 0, \theta) + \varepsilon_{0t},\ \bar{V}_1(a_t, i_t = 1, \theta) + \varepsilon_{1t} \right] \end{aligned}$$

   where the choice-specific value functions $\bar{V}_0$ and $\bar{V}_1$ are

$$\begin{aligned} \bar{V}_0(a_t, i_t = 0, \theta) &= \mu a + \beta \mathbb{E} V(a_t, i_t = 0, \theta) \\ \bar{V}_1(a_t, i_t = 1, \theta) &= R + \beta \mathbb{E} V(a_t, i_t = 1, \theta) \end{aligned}$$

3. The function *contraction_map*(μ, R, β) in my code does the contraction mapping.

   (a) i. If $a_t = 2$, the firm is indifferent between replacing and not replacing its machine if

$$\varepsilon_{0t} - \varepsilon_{1t} = \bar{V}_1(a_t = 2, i_t, \theta) - \bar{V}_0(a_t = 2, i_t = 1, \theta).$$

   From my code, *contraction_map*$(-1, -3, 0.9)$ evaluates to the matrix

$$\begin{array}{cc} \bar{V}_0 & \bar{V}_1 \\ -15.364 & -16.5976 \\ -16.712 & -16.5976 \\ -17.8525 & -16.5976 \\ -18.9055 & -16.5976 \\ -19.9055 & -16.5976 \end{array}$$

The second row corresponds to $a_t = 2$, so the firm is indifferent if $\varepsilon_{0t} - \varepsilon_{1t} \approx 0.1145$.

ii. Then the probability that this firm will replace its machine at $a_t = 2$ is $\mathbb{P}[\varepsilon_{0t} - \varepsilon_{1t} < 0.1145] = F(0.1145)$, where F is the CDF of the Logistic$(0,1)$ distribution (don't ask me why). The probability of replacement when $a_t = 2$ comes out to $F(0.1145) \approx 0.5286$.

iii. The value of a firm when $(a_t, \varepsilon_{0t}, \varepsilon_{1t}) = (4, 1, 1.5)$ is

$$\begin{aligned}
V(a_t = 4, \theta) &= \max_{i_t \in \{0,1\}} \Big[\bar{V}_0(a_t = 4, i_t = 0, \theta) + \varepsilon_{0t},\ \bar{V}_1(a_t = 4, i_t = 1, \theta) + \varepsilon_{1t}\Big] \\
&= \max_{i_t \in \{0,1\}} \Big[\bar{V}_0(a_t = 4, i_t = 0, \theta) + 1,\ \bar{V}_1(a_t = 4, i_t = 1, \theta) + 1.5\Big] \\
&= \max\Big[-18.9055 + 1,\ -16.5976 + 1.5\Big] \\
&= -15.0976
\end{aligned}$$

So the firm chooses to replace.

4. In my code, *data* = *generate_data*(μ, R, β, T) simulates the data.

5. One set of estimates I generated was $\hat{\mu} = -1.0155$ and $R = -2.9918$, which seem pretty good to me.

6. (a) i. Conditional state transition matrices, where column numbers represent a_t, and row numbers represent a_{t+1} values.

$$F_0 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad F_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

ii. The estimated replacement probabilities are *replace_probs* = *get_replacement_probs*(data) in my code, and evaluate to

$$\Big[\mathbb{P}(i = 0),\ \mathbb{P}(i = 1)\Big] = \begin{bmatrix} 0.7733 & 0.2267 \\ 0.4768 & 0.5232 \\ 0.2388 & 0.7612 \\ 0.07497 & 0.9250 \\ 0.01667 & 0.9833 \end{bmatrix}$$

which implies the unconditional transition matrix is

$$\hat{F} = \begin{bmatrix} 0.2267 & 0.7733 & 0 & 0 & 0 \\ 0.5232 & 0 & 0.4768 & 0 & 0 \\ 0.7612 & 0 & 0 & 0.2388 & 0 \\ 0.9250 & 0 & 0 & 0 & 0.07497 \\ 0.9833 & 0 & 0 & 0 & 0.01667 \end{bmatrix}$$

iii. *values* $=$ *simulate_forward_values*$(\theta, \text{replace\_probs}, \text{periods}, \text{sims}, \beta)$ forward simulates the firm decisions in my code.

(b) One set of estimates that my code generated was $\hat{\mu} = -0.9726$ and $R = 2.9650$, which are quite good, but the *Rust* estimates tend to be just a little bit better. Also, the optimizer fails with certain initial values of θ, which doesn't happen with the Rust approach. For instance, $\theta = (0, 0)$ consistently converges to values near zero (for example, $(-0.0027, -0.0023)$. But the good estimate that I wrote above came from $(-5, -5)$, so it's not that my initial θ has to be right on top of the actual θ. Not sure what's going on here.

Given the econometric advantage of Rust over Hotz & Miller, as well as what appears to be a local minimum problem with Hotz & Miller, the Rust approach seems better. In addition, although I thought the contraction mapping algorithm was supposed to be very time-consuming, the Rust optimization converges to very good estimates in less than a second, while the Hotz & Miller optimization takes about 40 seconds (which is highly subject to the number of simulations). So given all that, NFP seems clearly better than CCP to me.