Optimal Replacement of GMC Bus Engines: An Empirical Model of Harold Zurcher

1. The Model

1.1 State Variable and Transition Probabilities

The unique state variable x_t is the accumulated mileage (since last replacement) on the bus engine at time t. The real mileage is discretized into 90 intervals of length 5000, i.e., $x_t \in \{1,2,3,\ldots,90\}$, where $x_t = j$ means that the period t mileage since last replacement lies in the interval $[5000*(j-1), 5000*j)$.

The transition probability of this state variable is characterized by parameter p_0, p_1, p_2 with $p_0+p_1+p_2=1$. To be specific, after each period, if the engine is not replaced i.e., $d_t=0$, then next period state variable x_{t+1} can only take three possible values $\{x_t, x_t+1, x_t+2\}$. The transition probability matrix is characterized by two parameters:

$$p_j = \Pr\{x_{t+1} = x_t + j \mid x_t, d_t = 0\}, j = 0,1$$

If at period t, the engine is replaced, i.e., $d_t = 1$, then the next period state variable can take values $1,2,3$, each with probability $p_j, j = 0,1,2$, respectively.

More specifically, the transition probability matrices are

$$P^{d=0} = \begin{Bmatrix} \pi_0 & \pi_1 & \pi_2 & 0 & 0 & \cdots & 0 & 0 \\ 0 & \pi_0 & \pi_1 & \pi_2 & 0 & \cdots & 0 & 0 \\ \vdots & & & & \ddots & & & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & \pi_0 & 1-\pi_0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 1 \end{Bmatrix}, \quad P^{d=1} = \begin{Bmatrix} \pi_0 & \pi_1 & \pi_2 & 0 & 0 & \cdots & 0 & 0 \\ \pi_0 & \pi_1 & \pi_2 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & & & & \ddots & & & \vdots \\ \pi_0 & \pi_1 & \pi_2 & 0 & 0 & \cdots & 0 & 0 \\ \pi_0 & \pi_1 & \pi_2 & 0 & 0 & \cdots & 0 & 0 \end{Bmatrix}$$

1.2 Per-Period Utility

Per-period utility function takes the form:

$$u(x_t, d_t, \theta) = \begin{cases} -RC - c(0,\theta) & \text{if } d_t = 1, \\ -c(x_t,\theta) & \text{if } d_t = 0. \end{cases}$$

where the replacement cost RC is a parameter to be estimated. Assume the maintenance cost function takes the linear form, with θ to be estimated,

$$c(x_t, \theta) = \theta * x_t.$$

Together, there are four parameters in this simple problem: RC, θ, p_0, p_1.

1.3 Problem

The corresponding Bellman Equation is:

$$V(x_t, \varepsilon_t) = \max_{i_t} \left\{ u(x_t, d_t, \theta) + \varepsilon_t(d_t) + \beta E\left[V\left(x_{t+1}, \varepsilon_{t+1}\right) \mid x_t, \varepsilon_t\right] \right\}$$

where

$$E\left[V\left(x_{t+1}, \varepsilon_{t+1}\right) \mid x_t, \varepsilon_t\right] \equiv \int_y \int_\eta V(y,\eta) p\left(dy, d\eta \mid x_t, \varepsilon_t, d_t\right).$$

We impose conditional independence assumption. And we further assume the distribution of the unobserved state variable $\varepsilon_t(i)$ follows an iid type I extreme value distribution. Altogether, we can get a closed solution to this dynamic programming problem.

Now we can describe the relationships between the ex ante (or integrated) value function $\overline{V}(x_t)$, the conditional value function $v(x_t, d_t)$ and conditional choice probabilities $p_t(d_t \mid x_t)$ in the following way (these terms are borrowed from Arcidiacono and Ellickson (2011) who provide a general representation of dynamic discrete choice problems):

$$\overline{V}(x_t) \equiv \int V(x_t, \epsilon_t) g(\epsilon_t) d\epsilon_t$$

$$v(x_t, d_t) \equiv u(x_t, d_t) + \beta \int \overline{V}(x_{t+1}) f(x_{t+1} \mid x_t, d_t) dx_{t+1}$$

$$p_t(d_t \mid x_t) = \frac{\exp[v_t(x_t, d_t)]}{\sum_{d_t' \in D} \exp[v_t(x_t, d_t')]} = \frac{1}{\sum_{d_t' \in D} \exp[v_t(x_t, d_t') - v_t(x_t, d_t)]}$$

In this specific binary choice case with extreme value distribution,

$$v(x_t, 1) = u(x_t, 1) + \beta * \left(\sum_{j=0,1,2} p_j * \overline{V}(j+1) \right), \qquad v(x_t, 0) = u(x_t, 0) + \beta * \left(\sum_{j=0,1,2} p_j * \overline{V}(x_t + j) \right); \qquad (\star^1)$$

$$\overline{V}(x_t) = \ln\left(e^{v(x_t,1)} + e^{v(x_t,0)} \right); \qquad (\star^2)$$

$$p(d_t = 1 \mid x_t) = \frac{\exp[v(x_t, 1)]}{\exp[v(x_t, 1)] + \exp[v(x_t, 0)]}, \qquad p(d_t = 0 \mid x_t) = \frac{\exp[v(x_t, 0)]}{\exp[v(x_t, 1)] + \exp[v(x_t, 0)]} \qquad (\Delta).$$

2. Full-Solution Estimation

2.1 Estimation Procedures: NXFP

1. We first nonparametrically estimate the parameters p_0, p_1.
2. Given the estimates $\hat{p_0}, \hat{p_1}$ and any possible parameter values for RC, θ, the ultimate goal is to get conditional choice probabilities (CCP) $P(d \mid x_t)$.
3. To calculate CCPs, we need to obtain the ex ante value function $\overline{V}(x_t)$ and the conditional value function $v(x_t, d)$ by functional iteration using Equations $(\star)$.
   The specific procedure is that given iteration i's guess of $\overline{V}^i(x_t)$, we first obtain iteration i's guess of $v^i(x_t, d_t)$ using equations $(\star^1)$; after that, we get iteration $(i+1)$'s guess of $\overline{V}^{i+1}(x_t)$ by plugging $v^i(x_t, d_t)$ into equation $(\star^2)$.
   We repeat the above process until the difference between $\overline{V}^N(x_t)$ and $\overline{V}^{N+1}(x_t)$ is indistinguishable.
4. After that, we can the conditional choice probabilities $p(d \mid x_t)$ using equations (Δ).
5. Finally, we go through all possible parameter values for RC, θ to maximize the log likelihood of the conditional choice.
   The recommended procedure proposed by Rust is BHHH algorithm. But here we just use the MATLAB original optimization function fmincon.

2.2 Notes for MATLAB Codes

1. Prepare the data by running step0_data_preparation.m.
2. Relevant functions used in step0_data_preparation.m:
   fun_data_input.m - Reshape the original data files into matrices. See Part 4 in file https://editorialexpress.com/jrust/nfxp.pdf for details.
   fun_data_preparation.m - Extract bus replacement information and discretize the state variable. The first 11 rows represent bus history, see part 1 in https://notes.quantecon.org/submission/6234fe0f96e1ce001b61fad8.
   fun_data_reshape.m - Reshape matrices into a $TN * 2$ matrix. The first column indicates the values of the state variable and the second column indexes engine replacement decision.
3. Estimate the parameters by running step1_estimation.m.

4. The first section in file step1_estimation.m is to estimate the transition probabilities using function fun_transition_prob.m.
   fun_transition_prob.m - Inputs: (possibly several) $T * N$ matrices (from different groups). Output: p_0, p_1, p_2, calculated by $\frac{n_i}{n_0+n_1+n_2}, i \in \{0, 1, 2\}$, where n_i denotes the number of observations for which the state variables change i in the next period. Note that we need to make sure the change in the state variable occurs in the same car!

5. The next step is to estimate parameters RC, θ.

6. To do this, we first need an algorithm to solve the inner fixed point problem:
   fun_iteration.m - Given an initial guess of $\overline{V}^i(x_t)$ (90 by 1 vector), the probability transition matrices (two 90 by 90 matrices, one for each decision), and utilities (two 90 by 1 vectors), use equations $(\star^1)$ and $(\star^2)$ to get the next iteration values (90 by 1 vector).
   fun_inner_algo.m - Return a “correct” ex ante value function, $\overline{V}(x_t)$ (90 by 1 vector).

7. Estimation the parameters by an outer maximization algorithm (second section in file step1_estimation.m):
   fun_loglike.m - Given the correct ex ante value function, $\overline{V}(x_t)$, compute the conditional choice probabilities (which gives the probability of engine replacement under current value of the state variable) and form (negative) log-likelihood of the sample.

IMPORTANT: Precision Problem
When we set the same parameters as those in the paper, the $\overline{V}$ can be large in absolute value, about -1400, but in MATLAB, $\log(e^{-1700}) = -\infty$!
Therefore, we need to carefully avoid such problem!
See line 22 in fun_iteration.m. Basically, we abstract the same value on the conditional value function to rescale it.
See https://mark-ponder.com/tutorials/discrete-choice-models/dynamic-discrete-choice-nested-fixed-point-algorithm/ for more details.

Hotz-Miller Methods

Intuition

1. Conditional choice probabilities can also be non-parametrically estimated from the data.
2. There is a one-to-one map from the conditional choice probabilities to the conditional value functions.
3. The renewal decision $d = 1$ means that the form of the mapping is quite easy.

For more explanation, see also Arcidiacono and Ellickson (2011).

Explanation of MATLAB Codes