ECON 899b: Problem Set 4

Katherine Kwok*

December 2021

Overview: For this assignment, the goal is to program a dynamic model of inventory control.

Program:

The attached code files "helper_functions.jl" contains implements this declares the necessary functions and structs. And then "main_program.jl" uses the defined functions to implement the following steps:

1. Solve for value function using the implicit equation for expected value (policy function iteration algorithm).
2. Solve fo value function using conditional choice probabilities generated by the simulated data.
3. Find λ that maximizes log-likelihood using BFGS algorithm, and using nested-fixed point algorithm.

Results:

First, we compare the value functions generated using two different methods. The first algorithm is policy function iteration. Since the error term ϵ follows a type-one extreme value distribution, we can write down the expectation of the ϵ term. Then, we can apply contraction mapping to the probability vector and recover the $\bar{V}$ function using the probability vector and the transition matrices (using the implicit equation). The second method is to use the simulated data is generate a vector of probabilities. Then, use conditional choice probability mapping to recover $\bar{V}$. I have summarized the resulting value functions in Table 1; the first three columns summarize the state variables, the fourth column contains the values from the first method, and the fifth column contains the values from the second method. We can see that the resulting values are very similar.

In the next part of this assignment, we find the λ value that maximizes the log-likelihood function, using both the BFGS and the nested-fixed point algorithm. The true $\lambda = -4$. We find that using the BFGS and nested-fixed point algorithm, the resulting λ values are around -4.0337 and -4.0244 respectively.

*I collaborated with Anya Tarascina and Claire Kim on this assignment. I also referred to the Ox code provided with this assignment to code up some of the algorithms.

Table 1: Value Functions

I	C	P	pol_func_iter	CCP_sim_data
0	0	4	61.13	60.91
1	0	4	65.01	64.8
2	0	4	68.48	68.27
3	0	4	71.67	71.44
4	0	4	74.63	74.38
5	0	4	77.39	77.1
6	0	4	79.96	79.59
7	0	4	82.26	81.74
8	0	4	84.07	83.38
0	1	4	58.49	58.27
1	1	4	63.13	62.91
2	1	4	67.01	66.8
3	1	4	70.48	70.27
4	1	4	73.67	73.43
5	1	4	76.63	76.38
6	1	4	79.39	79.1
7	1	4	81.96	81.59
8	1	4	84.26	83.64
0	0	1	63.24	63.0
1	0	1	66.89	66.68
2	0	1	70.2	69.98
3	0	1	73.26	73.01
4	0	1	76.11	75.84
5	0	1	78.76	78.44
6	0	1	81.2	80.74
7	0	1	83.28	82.68
8	0	1	84.27	83.31
0	1	1	61.02	60.81
1	1	1	65.24	65.03
2	1	1	68.89	68.68
3	1	1	72.2	71.98
4	1	1	75.26	75.02
5	1	1	78.11	77.84
6	1	1	80.76	80.4
7	1	1	83.2	82.77
8	1	1	85.28	84.68