Quantitative Macroeconomics Final Project

Sebastian A. Roy

7 January 2020

1 Simple Variant of Krusell-Smith Algorithm

1.1 Proof of the Proposition 3

Statement

Under assumptions of logarithmic utility and particular shock distribution (positive support, unity-mean and independence), equilibrium dynamics are given by the following formulas:

$$k_{t+1} = \frac{1}{(1+g)(1+\lambda)} s(\tau)(1-\tau)(1-\alpha)\zeta_t k_t^\alpha \tag{1}$$

$$s(\tau) \equiv \frac{\beta\Phi(\tau)}{1+\beta\Phi(\tau)} \leq \frac{\beta}{1+\beta} \tag{2}$$

With $\Phi(\tau)$ defined as:

$$\Phi(\tau) \equiv \mathbb{E}_t \Big(\frac{1}{1 + \frac{1-\alpha}{\alpha(1+\lambda)\rho_{t+1}}(\lambda\eta_{i,2,t+1} + \tau(1+\lambda(1-\eta_{i,2,t+1})))} \Big) \leq 1 \tag{3}$$

Proof

We start from saying that tomorrow assets of the old are equal to today youngs' savings from the labour income after taxation. Since *ex ante* all agents are homogenous, we can disregard idiosyncratic income shocks, thus having:

$$K_{t+1} = a_{2,t+1} = s(1-\tau)(1-\alpha)\mathcal{Y}_t\zeta_t k_t^\alpha \tag{4}$$

Hence capital per unit of efficient labour is given by:

$$k_{t+1} = \frac{1}{(1+g)(1+\lambda)} s(\tau)(1-\tau)(1-\alpha)\zeta_t k_t^\alpha \tag{5}$$

Capital path given by 5 is the one we have been proving (1). Next, we can note that consumption of the old cohort is equal to return from the savings made when young and risky labour income. Including the shock structure this gives:

$$c_{i,2,t+1} = s(1-\tau)(1-\alpha)\mathcal{Y}_t\zeta_t k_t^{\alpha}\alpha\zeta_{t+1}\rho_{t+1}k_{t+1}^{\alpha-1} + \\ + (1-\alpha)\mathcal{Y}_{t+1}\zeta_{t+1}k_{t+1}^{\alpha}(\lambda\eta_{i,2,t+1} + \tau(1+\lambda(1-\eta_{i,2,t+1}))) \quad (6)$$

Using formula 5 for capital path and substituting to the equation above yields:

$$c_{i,2,t+1} = \Big(\alpha\rho_{t+1}(1+\lambda) + (1-\alpha)\big(\lambda\eta_{i,2,t+1} + \tau(1+\lambda(1-\eta_{i,2,t+1}))\big)\Big)\cdot \\ \cdot \mathcal{Y}_{t+1}\zeta_{t+1}k_{t+1}^{\alpha} \quad (7)$$

Analogically, young cohort consumption is given as the non-saved fraction of the deterministic labour income when young, $c_{1,t} = (1-s)(1-\tau)(1-\alpha)\mathcal{Y}_t\zeta_t k_t^{\alpha}$. Substituting both consumption formula to the Euler equation (being a first-order optimality condition) gives:

$$1 = \beta\mathbb{E}_t\Big(\frac{c_{1,t}(1+r_{t+1})}{c_{i,2,t+1}}\Big) = \frac{\beta(1-s)}{s}\Phi \quad (8)$$

Simple algebra leads to the conclusion that the formula above is equivalent to (2), which concludes the proof.

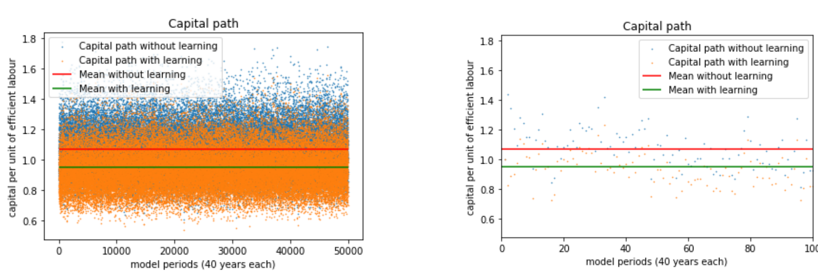
1.2 Capital Path Simulation

Discussion of the method

For code, see `1.2_code.py`.

First, simulation is carried out **without shock discretisation**, i.e., with continuous, log-normal shocks. The results can be seen in Figure 1.

One issue worth discussing here is the problem of expectations. The model altogether includes three shocks, η, ρ, ζ. One of them, ζ, appears directly in the capital path formula, but two other shocks make influence only indirectly, as



(a) Capital path for all 50,000 periods (b) Capital path for the first 100 periods

Figure 1: Capital path simulation with log-normal shocks

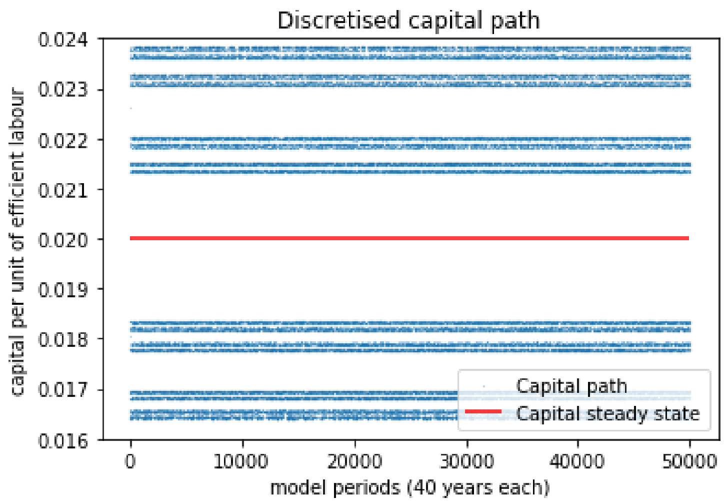


Figure 2: Discretised capital path

they appear in the Φ formula and, more importantly, in expectations. Since all shocks are *iid*, we are allowed to assume that

$$\mathbb{E}_t(\eta_{t+1}) = \mathbb{E}_t(\rho_{t+1}) = 1$$

The problem is, however, that such an approach kills those shocks and makes Θ constant.

Our approach to deal with that is to employ less orthodox assumptions and allow for learning instead of rational expectations. In that framework agents update their learnt expectation of the shocks, which they treat as a sample mean. Hence perceived expectation of some shock β is given by:

$$\mathbb{E}_t^{\mathcal{P}}(\beta_{t+1}) = \frac{1}{t}\sum_{i=0}^{i=t}\beta_i$$

Then, we proceed **with discretised simulation**. Shocks ρ, ζ are approximated with a binomial discrete distribution, while η comes from a Gaussian Quadrature. The results are presented in Figure 2.

Result interpretation

Continuous shock simulation

- Similarly to the logic presented in Harenberg and Ludwig (2015) we interpret each model period as 40 actual years, which allows to use *iid* shock distributions.

- As we can see in Figure 1, capital does not converge to some deterministic steady-state. This is perfectly understandable, as in every model period the economy is hit by multiple shocks.

Discrete shock simulation

- Under discrete shock simulation one can observe 'clusters' of capital, which comes from the fact that shocks no longer are continuous. Hence their cumulated impact is also clustered.
- Due to unchanged assumption of *iid* distribution, capital moves freely between those 'clusters' - regardless of the state today, any state tomorrow is equally likely. This might be considered a strong case of Markovian principle with equal transition probabilities.

1.3 Simple Implementation of Krusell-Smith Algorithm

a) Calculation of $\psi_i(z)$ parameters

Theoretical values of $\psi_{0,1}$ are presented in Table 1.

ψ_0	ψ_1
-2.74	0.3

Table 1: Coefficients $\psi_{0,1}$

b) Krusell-Smith Algorithm Implementation

Note: This section does not deliver quantitative results, because we have encountered some trouble with Python's solver (root-finding routine). However, as we believe our proposed algorithm is pretty accurate, it will be discussed below.

For code, see `1.3_code.py`.

As suggested in the project instructions, we try to solve the HH problem using the Euler equation, which needs to be linearised first so that linear solver might be used. Denoting gross interest rate by R_t our starting point is:

$$1 = \beta \mathbb{E}_t \Big(\frac{c_{1,t} R_{t+1}}{c_{2,t+1}} \Big) \tag{9}$$

After transformation we arrive at the linear equivalent:

$$0 = \beta c_{1,t} - \mathbb{E}_t \frac{c_{2,t+1}}{R_{t+1}} \tag{10}$$

Knowing that we propose a following algorithm:

- Start a 5×2 matrix for states (5 for capital, 2 for shocks)

- Define functions for gross interest rate, wages, old cohort labour income, and old cohort pension income
- Using those definitions, define linearised Euler equation (10) as a function of states (capital and shocks) and assets, and optimise

2 Complex Variant of Krusell-Smith Algorithm

In this case, the starting-point code is extremely complex (1,200 lines of code in Matlab!). Most unfortunately, we have not succeeded in upgrading that code. **However, several *ex ante* remarks might be drawn**.

Definitely, this problem runs into the dimensionality problem (that is why we are not expected to run 50 000 periods), as we add an additional shock. Furthermore, some approximation of Markov process (e.g. Tauchen method) is to be employed in order to generate a technology shock matrix.