Quant Macro Final Project

Szymon Wieczorek

January 2020

1 Simple variant of Krusell-Smith Algorythm

In these exercise we were supposed to introduce simple variant of Krusell-Smith algorythm step by step. Each of 3 subpoints was constructed to help us in this case.

1.1 Restated proof of the Proposition 3

Statement

Under assumption of logarythmic utility and shock distribution such that they have positive support, mean of 1 and are independent, the equilibrium dynamics are given by the following formulas:

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

$s(\tau) = \frac{\beta\Phi(\tau)}{1+\beta\Phi(\tau)} <= \frac{\beta}{1+\beta}$

Where $\Phi(\tau)$ is defined as:

$\Phi(\tau) = E_t[\frac{1}{1+\frac{1-\alpha}{\alpha(1+\lambda)\rho_{t+1}}(\lambda\eta_{i,2,t+1}+\tau(1+\lambda(1-\eta_{i,2,t+1})))}] <= 1$

Proof

Capital is made of assets created by savings of young households that come from the labor income. Since utility functions for all household is exactly the same (no individual heterogeneity), aggregate capital equation can be written in the form:

$K_{t+1} = a_{2,t+1} = s(\tau)(1-\tau)w_t = s(\tau)(1-\tau)(1-\alpha)\Upsilon_t\zeta_t k_t^\alpha$

Capital per unit of effective labor dynamics is:

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

So the capital path has been proven. Now we can focus on consumption. After several transformations in original formula and implementation of formula for capital we get that:

$c_{2,t+1} = \Upsilon_{t+1}\zeta_{t+1}k_{t+1}^\alpha(\alpha(1+\lambda)\rho_{t+1}+(1-\alpha)(\lambda\eta_{i,2,t+1}+\tau(1+\lambda(1-\eta_{i,2,t+1}))))$

And since all non-saved income is consumed,

$c_{1,t} = (1-s(\tau))(1-\tau)(1-\alpha)\Upsilon_t k_t^\alpha\zeta_t$.

Now, we can use the Euler equation to relate consumption in both periods:

$1 = \beta E_t[\frac{c_{1,t}(1+r_{t+1})}{c_{i,2,t+1}}] = \beta E_t[\frac{(1-s(\tau))(1-\tau)(1-\alpha)k_t^\alpha \zeta_t k_{t+1}^{-1}\rho_{t+1}}{(1+g)(\alpha(1+\lambda)\rho_{t+1}+(1-\alpha)(\lambda\eta_{i,2,t+1}+\tau(1+\lambda(1-\eta_{i,2,t+1}))))}]$
Which after transformations becomes:
$1 = \frac{\beta(1-s(\tau))}{s(\tau)}\Phi(\tau)$, where $\Phi(\tau)$ has the same value as described in Harenberg and Ludwig (2015).
And after small transformation we get that:
$s(\tau) = \frac{\beta\Phi(\tau)}{1+\beta\Phi(\tau)}$,
which is exactly what we looked for.

1.2 FOD simulation

In this point we were supposed to simulate the capital path of the first-order equation of capital in logs:

$ln(k_{t+1}) = ln(\frac{1-\alpha}{1-\lambda}) + ln(s(\tau)) + ln(1-\tau) + ln(\zeta_t) + \alpha ln(k_t)$,
using pre-defined values of parameters: g = 0, $\alpha = 0.3$, $\lambda = 0.5$, $\beta = 0.99^{40}$ and $\tau = 0$.

Additionally, we assume that z^r are represented by such values of ζ_t, ρ_t, $\eta_{i,2,t}$ such that $k_t < k_{ss}$. Analogically, z^b are represented by such values of ζ_t, ρ_t, $\eta_{i,2,t}$ such that $k_t > k_{ss}$.

As it can be seen on the plot, the steady state value of capital in log is about -1.37.

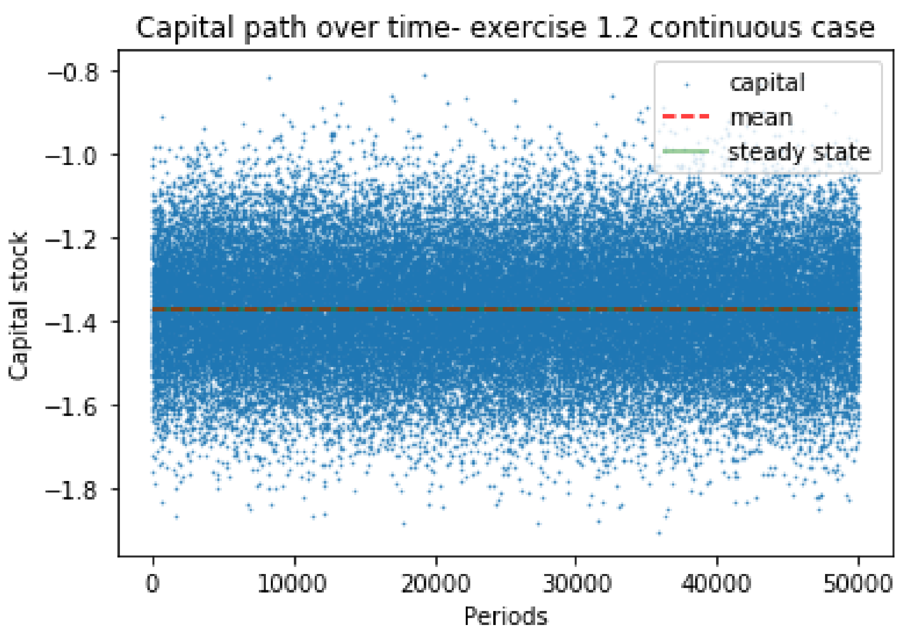

Figure 1: Simulation of path of capital with g = 0 in 50k periods

The value of capital stock between periods varies between -2 and -0.8 in log values, depending on the values of ζ_t.

The next step was to implement Gaussian quadrature method to calculate the capital path. Gaussian quadrature is a different method of estimating path of capital. This time, we generate different values of shocks and attach probabilities to them. Since each of the 3 shocks can take 2 values and their probabilities are symmetric, probability of drawing each of the possible states of nature is p = 0.125. Additionally, since we draw Φ as a probability adjusted sum of recession state case and boom state case, this coefficient takes 2 values, giving 16 possible paths of capital. A scatter plot below shows all possible capital paths over time.

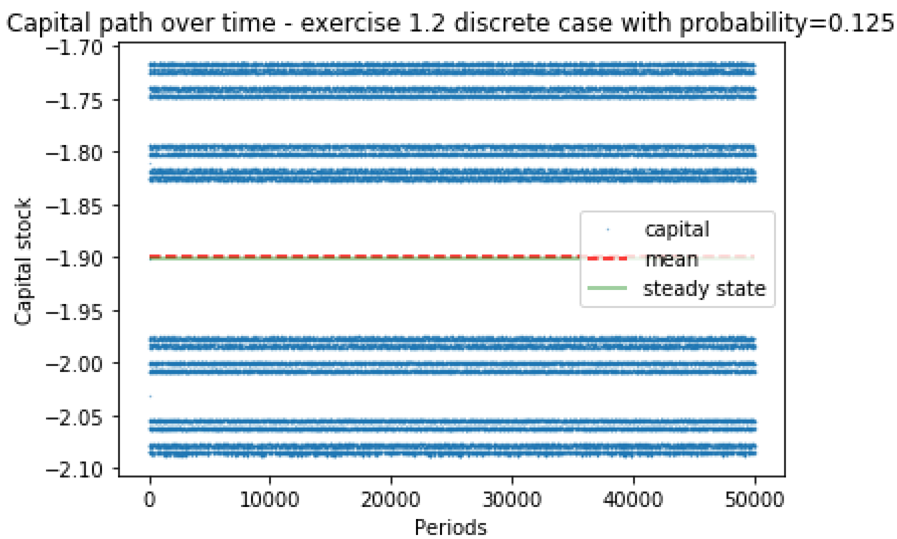

Figure 2: Simulation of path of capital with g = 0 in 50k periods according to Gaussian quadrature method

The algorythm generates 16 different paths of capital, as it was predicted above. Various paths converge very fast to certain values of k_t and they stay at the same level for all the set of observations. Mean and steady state are close to -1.9 in log values, but all of the generated paths stay in certain distance to that value.

1.3 Implementation of Krusell-Smith Algorythm

1.3.1 Theoretical values of coefficients of capital equation

According to the equations in paper written by Harenberg and Ludwig (2015), the theoretical values of coefficients ψ_i were calculated, according to the equation below:

$ln(k_{t+1}) = \psi_0(z) + \psi_1(z) ln(k_t)$

And the equations in **1.2** as well as equations concerning savings from Harenberg and Ludwig(2015) imply that:

$ln(k_{t+1}) = ln(\frac{1-\alpha}{1-\lambda}) + ln(s(\tau)) + ln(1-\tau) + ln(\zeta_t) + \alpha ln(k_t)$, where
$s(\tau) = \frac{\beta\Phi(\tau)}{1+\beta\Phi(\tau)}$, and
$\Phi(\tau) = E_t[\frac{1}{1+\frac{0.7}{0.3\lambda\rho_{t+1}}(\lambda\eta_{i,2,t+1}+\tau(1+\lambda(1-\eta_{i,2,t+1})))}]$

After using the fact that $\tau = 0$, $\lambda = 0.5$, $\alpha = 0.3$ and $\beta = 0.99^{40}$:
$\Phi(0) = E_t[\frac{1}{1+\frac{0.7}{0.15\rho_{t+1}}(0.5\eta_{i,2,t+1})}]$,
$s(0) = \frac{0.99^{40}\Phi(0)}{1+0.99^{40}\Phi(0)}$,
$ln(k_{t+1}) = ln(\frac{0.7}{0.5}) + ln(s(0)) + ln(1) + ln(\zeta_t) + 0.3ln(k_t)$

Inserting $\Phi(0)$ and $s(0)$ into the capital function as well as using the properties of logarythmic function yields:

$$ln(k_{t+1}) = ln(1.4\zeta_t \frac{0.99^{40}E_t[\frac{1}{1+\frac{0.7}{0.15\rho_{t+1}}(0.5\eta_{i,2,t+1})}]}{1+0.99^{40}E_t[\frac{1}{1+\frac{0.7}{0.15\rho_{t+1}}(0.5\eta_{i,2,t+1})}]}) + 0.3ln(k_t)$$

(isn't it 0.3 instead of 0.15?)
And since expected value of all the shock terms is 1:

$$ln(k_{t+1}) = ln(\frac{0.99^{40}*0.15/0.5*1.4\zeta_t}{0.99^{40}*0.15/0.5+1}) + 0.3ln(k_t) = ln(\frac{0.99^{40}*0.3*1.4\zeta_t}{0.99^{40}*0.3+1}) + 0.3ln(k_t)$$

Therefore:

$\psi_0 = ln(\frac{0.99^{40}*0.15*1.4\zeta_t}{0.99^{40}*0.15+0.5}) = ln(\zeta_t) - 1.452$
$\psi_1 = 0.3$

1.3.2 Carrying out the algorythm

In this part we implement the algorythm using our Python code.
PART I
Here we calculate the vector of savings without using explicit formula from paper. After defining parameters, we define shock values, which take mean+1 st.dev. value in case of boom and mean-1 st.dev. in case of recession. Probability of recession value is equal to the probability of the boom value.

After declaring shocks, we use them to estimate the values of Φ and s, because s depends on Φ. Then we draw the shocks and use them to calculate values of ψ^i.

Having steady-state path of capital calculated, we use primitives of the model to solve the households' problem. We simplify and implement the formulas used in the paper. Having also defined nodes of capital (representing booms and recessions for capital), we use root finding process to find the solution to the Euler (FOC for consumption in period 1 and period 2) equation. ...

...

PART II

After simulating the path of savings we use it to simulate the economy ... (simulation of economy) ...

PART III

... (self-updating algorythm) ...

1.3.3 Comparison of numerical and analytical solutions

('steady' state of capital) calculated earlier vs ...

1.3.4 Changed τ and calculated expected utility

...

2 Complex variant of Krusell-Smith Algorythm

This exercise ... However, we were unable to implement the necessary steps in a time given to solve this homework.