Econ 8185 (002): Quant PS3

Bipul Verma

December 30, 2021

Abstract

This document calculates the Policy rules for Aiyagari Model using Endogenous grid method.

1 Basic Aiyagari Model

We'll start with the basic Aiyagari model first with no endogenous labor, taxes, transfers or government. The basic Aiyagari model has the households solving the following optimization problem:

$$\max_{c_{it}} \mathbb{E}_0 \sum_{t=0}^{\infty} \beta^t \frac{c_{it}^{1-\gamma}}{1-\gamma}$$
$$\text{s. t. } c_t + a_{t+1} = w_t \epsilon_t + (1+r_t) a_t$$
$$a_{t+1} \geq 0.$$

The household optimization problem can be summarized in the following Euler condition:

$$u_c(c(a,\epsilon)) \geq \beta(1+r) \sum \Pi(\epsilon'|\epsilon) u_c(c(a'(a,\epsilon),\epsilon')$$
$$(= \quad \text{if} \quad a'(a,\epsilon) > 0).$$

For the purpose of present exercise we'll also assume that the shocks follow a Markov process with transition matrix Π.

1.1 Computation Algorithm: Policy Function

1. We need to compute the consumption policy function $c(a,\epsilon)$. We can always use the budget equation to back out the asset policy function.
2. Construct grid for assets (a) and labour supply shocks (ϵ).
3. Guess the policy function as $c^{(m)}[a_i, \epsilon_j] = ra_i + \epsilon_j$.
4. We'll now calculate the current current consumption and asset levels for a given target level of assets tomorrow.
5. Compute $c^*[a'_i, \epsilon_j] = Uc^{-1}(\beta(1+r)\sum \Pi(\epsilon'|\epsilon_j) u_c(c^{(m)}(a'_i, \epsilon')).$
6. Compute $a^*[a'_i, \epsilon_j] = (c^*[a'_i, \epsilon_j] + a'_i - w\epsilon_j)/(1+r)$.
7. We thus have the policy function c^* defined on endogenous grid of assets a^*. We'll interpolate the policy function such that it is defined on the original asset grid.
8. Update as follows:

$$c^{(m+1)}[a_i, \epsilon_j] = \begin{cases} (1+r)a_i + w\epsilon_j - a_0 & ; a_i < a^*[a_0, \epsilon_j] \\ \text{LinearInterpolate}(c^*[a_k, \epsilon_j], c^*[a_{k+1}, \epsilon_j]) & ; a^*[a_k, \epsilon_j] < a_i < a^*[a_{k+1}, \epsilon_j] \end{cases}$$

9. Repeat till convergence.
10. *Check at this point of time that the asset policy function for the lowest shock is below the 45 degree line.*

1.2 Computation Algorithm: Stationary Distribution

Note that the distribution in the present case is defined over the continuum (a, ϵ). We can discretize the space and keep track of distribution by assigning densities to those discrete points. These densities will then evolve according to transition matrix based on the asset policy function. Note that one needs to be careful on what transition probabilities each entry of transition matrix Q represents. We'll take a short detour to make the matter more concrete. Suppose that we have a first order markov process s which can take value $\{H, L\}$ with the associated transition matrix T given by:

$$\begin{bmatrix} \pi(L|L) & \pi(H|L) \\ \pi(L|H) & \pi(H|H) \end{bmatrix}$$

where $\pi(s'|s)$ represents the probability of moving from state s to s'. In this case the probability over states evolves according to:

$$\lambda(s') = T'\lambda(s).$$

The following code constructs a transition matrix and looks for the stationary distribution. Note that in our case matrix Q is like T' in the previous case.

Construction of Q'

The idea of construction of Q' is the following. For agent at each state today (a, ϵ), we first find out the asset level that the agent will end up next tomorrow based on the policy function. We then combine this with all possible realization of income shock that the agent can receive in the next period, based on the markov process for ϵ. Below is the algorithm which implements this.

- For each ϵ_j;
- $s_j = (j-1) * N_a$
  - for each a_i;
  - find k s.t $a_k \leq a'(a_i, \epsilon_j) < a_{k+1}$,
  - construct the histogram weights as $w_k = \frac{a'(a_i,\epsilon_j)-a_k}{a_{k+1}-a_k}$;
    * for each ϵ_m;
    * $t_m = (m-1) * N_a$,
    *

$$Q'[i + s_j, k + t_m] = (1 - w_k)\pi(\epsilon_m|\epsilon_j)$$
$$Q'[i + s_j, k + 1 + t_m] = w_k\pi(\epsilon_m|\epsilon_j).$$

Construction of Q

McKay(2018) outlines the construction of Q directly. The idea is simply to interchange the rows and columns in the last step, i.e

$$Q[k + t_m, i + s_j] = (1 - w_k)\pi(\epsilon_m|\epsilon_j)$$
$$Q[k + 1 + t_m, i + s_j] = w_k\pi(\epsilon_m|\epsilon_j).$$

1.3 Equilibrium r and w

Once we have the stationary distribution of individuals over the states, we can calculate the equilibrium interest rate which clears the market. Let $\bar{\mu}$ denote the stationary distribution over (a, ϵ). Then the aggregates are given by:

$$
\begin{aligned}
C &= \int c(a,\epsilon)d\bar{\mu}(a,\epsilon) \\
K &= \int a'(a,\epsilon)d\bar{\mu}(a,\epsilon) \\
N &= \int \epsilon d\bar{\mu}(a,\epsilon)
\end{aligned}
$$

Note that at the stationary distribution we'll have $K' = \int a'(a,\epsilon)d\bar{\mu}(a,\epsilon) = K = \int a d\bar{\mu}(a,\epsilon)$. Hence the aggregate resource constraint will be given by:

$$C + \delta K = K^{\theta} N^{1-\theta}$$

In order to compute the equilibrium r, we find r such that $K^d = K^s$, where:

$$
\begin{aligned}
K^d(r) &= N(\frac{r+\delta}{\theta})^{\frac{-1}{1-\theta}} \\
K^s(r) &= \int a'(a,\epsilon;r)d\bar{\mu}(a,\epsilon;r)
\end{aligned}
$$

Finally, to get the distribution of assets across the agents, we make use of the stationary distribution of agents across states to compute the proportion of agents at various assets levels.

1.4 Results from Basic Aiyagari

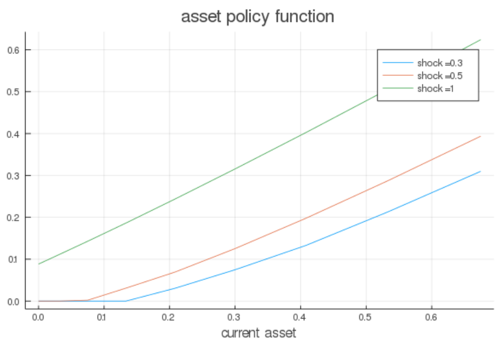

Figure 1: Asset Policy Function

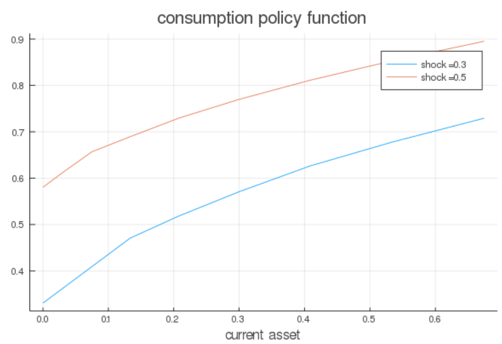

Figure 2: Consumption Policy function

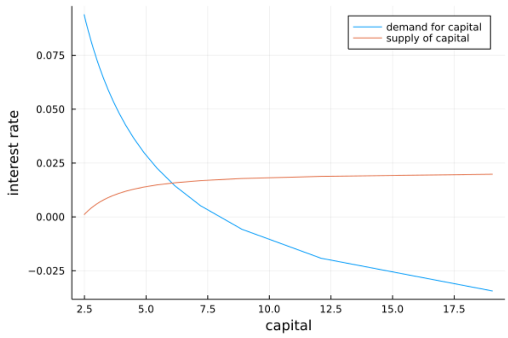

Figure 3: Asset Market Clearing

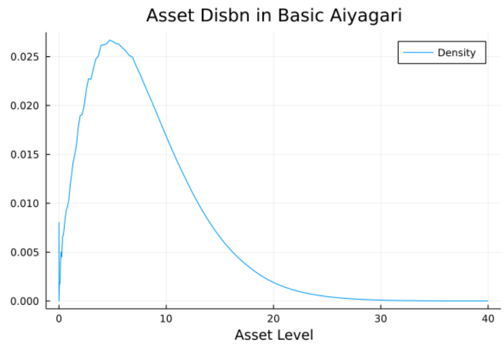

Figure 4: Stationary Disbn of Assets

2 Aiyagari with Endogenous Labor

The Aiyagari model with endogenous labor is can be written as:

$$\begin{aligned} & \max \mathbb{E}_0 \sum \beta^t U(c_t, l_t) \\ \text{s.t.} \quad & c_t + a_{t+1} = (1+r)a_t + w_t \epsilon_t (1 - l_t) \\ & a_{t+1} \geq 0 \end{aligned}$$

The consumers optimization is summarized by the following conditions:

$$\begin{aligned} U_c(c,l) &\geq \beta(1+r)\mathbb{E}U_c(c',l') \\ &(=)\text{if } a' > 0 \\ \frac{U_l(c,l)}{U_c(c,l)} &= w\epsilon \end{aligned}$$

Constraint on labor/leisure

Before we jump on to the actual discussion, we will sidetrack a bit to discuss the appropriate constraint on l_t which turn out to be important while computing equilibrium with endogenous labour supply. Let l_t denote leisure and n_t denote labour. The usual assumption is that $0 \leq l_t \leq 1$, and $n_t + l_t = 1$.

2.1 With separable utility

In this case the utility function is assumed to be of the following form:

$$U(c,l) = \frac{c^{1-\mu} - 1}{1-\mu} - \psi \frac{(1-l)^{1+\frac{1}{\gamma}}}{1+\frac{1}{\gamma}}.$$

For this utility function, the expression for marginal utility are:

$$\begin{aligned} U_c(c,l) &= c^{-\mu} \\ U_l(c,l) &= \psi(1-l)^{\frac{1}{\gamma}} \end{aligned}$$

The consumers consumption leisure choice tradeoff at optimum can then be summarized as follows:

$$\frac{\psi(1-l)^{\frac{1}{\gamma}}}{c^{-\mu}} = w\epsilon$$

We can thus solve for labor choice in terms of consumption choice at the optimum:

$$l(c) = 1 - [\frac{w\epsilon c^{-\mu}}{\psi}]^{\gamma}.$$

In this case the with can use the EGM to solve for consumption policy function and solve labor as residual given consumption using the MRS. The only additional modification will be in the update equation for consumption policy where we need to use root finding to solve for implicit c^{j+1}, when the asset constraint binds. In this case the update equation is given as follows:

$$c^{(m+1)}[a_i, \epsilon_j] = \begin{cases} (1+r)a_i + w\epsilon_j(1 - l(c^{(m+1)}[a_i, \epsilon_j])) - a_0 & ; a_i < a^*[a_0, \epsilon_j] \\ \text{LinearInterpolate}(c^*[a_k, \epsilon_j], c^*[a_{k+1}, \epsilon_j]) & ; a^*[a_k, \epsilon_j] < a_i < a^*[a_{k+1}, \epsilon_j] \end{cases}$$

Note: When the asset constraint is binding, we need to use a root solver to find c^*. A trick to improve the speed of EGM iteration comes from noticing the fact that when the asset constraint in binding, we are solving the same set of equations. Hence all these roots can be evaluated outside the iteration loop, rather than each time during the iteration.

2.2 Without separable utility

Consider the utility function as given below:

$$U(c, l) = \frac{(c^{\eta} l^{1-\eta})^{1-\mu}}{1-\mu}.$$

For this utility function, the expression for marginal utility are:

$$U_c(c, l) = (c^{\eta} l^{1-\eta})^{-\mu} \eta (\frac{c}{l})^{\eta - 1}$$
$$U_l(c, l) = (c^{\eta} l^{1-\eta})^{-\mu} (1-\eta)(\frac{c}{l})^{\eta}$$

The consumers consumption leisure choice tradeoff at optimum can then be summarized as follows:

$$\frac{1-\eta}{\eta} \frac{c}{l} = w\epsilon$$

We can thus solve for labor choice in terms of consumption choice at the optimum:

$$l(c) = \min\{\frac{1-\eta}{\eta} \frac{c}{w\epsilon}, 1\}$$

Note: The min condition is to ensure that the constraint $l_t \leq 1$ holds.
We can thus limit out search to optimal policy function for consumption and then derive labor policy as residual using the previous equation.

2.3 Computation Algorithm

1. Guess policy function for consumption $c^j(a_i, \epsilon_j) = ra_i + w\epsilon_j$.

2. For this guess of c^j solve the c^* such that the following euler holds on future assets:

$$U_c(c^*(a_i',\epsilon_j),l(c^*(a_i',\epsilon_j))) = \beta(1+r)\sum \Pi(\epsilon'|\epsilon_j)U_c(c^{(j)}(a_i',\epsilon'),l(c^{(j)}(a_i',\epsilon')))$$

Note that given the form of utility function the expression for $U_c(c,l(c))$ can be simplified as follows:

When $l(c) < 1$

$$\begin{aligned} U_c(c,l(c)) &= (c^\eta l(c)^{1-\eta})^{-\mu}\eta(\frac{c}{l(c)})^{\eta-1} \\ &= (c^\eta(\frac{1-\eta}{\eta}\frac{c}{w\epsilon})^{1-\eta})^{-\mu}\eta(\frac{c}{(\frac{1-\eta}{\eta}\frac{c}{w\epsilon})})^{\eta-1} \\ &= c^{-\mu}(\frac{1-\eta}{\eta}\frac{1}{w\epsilon})^{(\eta-1)\mu}\eta(\frac{1-\eta}{\eta}\frac{1}{w\epsilon})^{1-\eta} \\ &= \eta c^{-\mu}(\frac{1-\eta}{\eta}\frac{1}{w\epsilon})^{(1-\eta)(1-\mu)} \\ &= c^{-\mu}\Phi(\epsilon) \quad \text{ where } \Phi(\epsilon) = \eta(\frac{1-\eta}{\eta}\frac{1}{w\epsilon})^{(1-\eta)(1-\mu)}; \end{aligned}$$

When $l(c) = 1$

$$U_c(c,l(c)) = \eta c^{\eta(1-\mu)-1}.$$

Note that $U(c,l(c))$ is continuous and monotonically decreasing in c. Hence $U_c^{-1}(.)$ is well defined. We need to solve for c^* which solves the following:

$$U_c(c^*(a_i',\epsilon_j),l(c^*(a_i',\epsilon_j))) = \beta(1+r)\sum \Pi(\epsilon'|\epsilon_j)U_c(c^{(j)}(a_i',\epsilon'),l(c^{(j)}(a_i',\epsilon'))).$$

One way to solve for c^* is to simply use a root finding algorithm. However, root finding can be time intensive. We can use the following analytical approach to get c^*. Compute c_1^*, c_2^* from the analytical expression for two pieces of U_c. Compute $l(c_1^*), l(c_2^*)$. If $l(c_1^*) < 1$, then $c^* = c_1^*$, else $c^* = c_2^*$. Note that we are able to solve for c^* analytically which is computationally fast. We can then obtain $l^* = l(c^*)$ using the MRS as outlined earlier.

3. Given $c^*(a_i',\epsilon_j)$ solve for the endogenous asset levels as follows:

$$a^*(a_i',\epsilon_j) = \frac{c^*(a_i',\epsilon_j) + a_i' - w\epsilon_j(1-l^*(a_i',\epsilon_j))}{(1+r)}$$

4. Update the guess for the policy function as follows:

$$c^{(m+1)}[a_i,\epsilon_j] = \begin{cases} (1+r)a_i + w\epsilon_j(1-l(c^{(m+1)}[a_i,\epsilon_j])) - a_0 & ; a_i < a^*[a_0,\epsilon_0] \\ \text{LinearInterpolate}(c^*[a_k,\epsilon_j], c^*[a_{k+1},\epsilon_j]) & ; a^*[a_k,\epsilon_j] < a_i < a^*[a_{k+1},\epsilon_j] \end{cases}$$

Note that the first sub-case in the update equation can be simplified as follows:

$$\begin{aligned}
c^{(m+1)}[a_i, \epsilon_j] &= (1+r)a_i + w\epsilon_j(1 - l(c^{(m+1)}[a_i, \epsilon_j])) - a_0 \\
&= (1+r)a_i + w\epsilon_j(1 - \frac{1-\eta}{\eta}\frac{c^{(m+1)}[a_i, \epsilon_j]}{w\epsilon_j} - a_0 \\
&= (1+r)a_i + w\epsilon_j - \frac{1-\eta}{\eta}c^{(m+1)}[a_i, \epsilon_j] - a_0 \\
\implies (1 + \frac{1-\eta}{\eta})c^{(m+1)}[a_i, \epsilon_j] &= (1+r)a_i + w\epsilon_j - a_0 \\
\implies c^{(m+1)}[a_i, \epsilon_j] &= \eta((1+r)a_i + w\epsilon_j - a_0).
\end{aligned}$$

Hence the update equation can be simplified as follows:

$$c^{(m+1)}[a_i, \epsilon_j] = \begin{cases} \eta((1+r)a_i + w\epsilon_j - a_0) & ; a_i < a^*[a_0, \epsilon_0] \\ \text{LinearInterpolate}(c^*[a_k, \epsilon_j], c^*[a_{k+1}, \epsilon_j]) & ; a^*[a_k, \epsilon_j] < a_i < a^*[a_{k+1}, \epsilon_j] \end{cases}$$

5. Iterate till convergence.

Once we have the policy functions, the estimation of stationary distribution and equilibrium r and w remains the same with the only modification being that the aggregate labour supply is given as $N = \int \epsilon(1 - l(a, \epsilon))d\bar{\mu}(a, \epsilon)$

3 Aiyagari with Taxes and Transfers

We'll solve this for the one with no endogenous labor supply first. Define:

$$\begin{aligned}
\tilde{r}_t &= (1 - \tau_y)r_t \\
\tilde{w}_t &= (1 - \tau_y)w_t.
\end{aligned}$$

The the consumers problem in this case can be described as:

$$\begin{aligned}
&\max \mathbb{E}_0 \sum \beta^t U(c_t, l_t) \\
&\text{s.t.} \quad c_t + a_{t+1} = (1 + \tilde{r}_t)a_t + \tilde{w}_t\epsilon_t(1 - l_t) + T_t \\
&a_{t+1} \geq 0.
\end{aligned}$$

The Euler condition and MRS from the consumers optimization remains the same with the transformed variables $\tilde{r}, \tilde{w}$. The only additional change is in the consumers budget constraint with inclusion of government transfers T_t. We can proceed by solving for the policy function for a given $\tilde{r}$ and T, using the EGM. We need to set r and T such that the Asset market and Government Budget constraint simultaneously clear at the stationary equilibrium. This means that the following equations holds with equality.

$$\begin{aligned}
A_t &= K_t + B_t \\
G_t + r_tB_t + T_t &= \tau_y(w_tN_t + r_tA_t).
\end{aligned}$$

We outline the computation algorithm for the case when $G_t = \phi_g Y_t$, $B_t = \phi_b Y_t$.

Computation Algorithm

1. For the given r, T get the following:

$$A^s(r,T) = \int a'(a,\epsilon) d\bar{\mu}(a,\epsilon)$$
$$N(r,T) = \int n(a,\epsilon)\epsilon d\bar{\mu}(a,\epsilon)$$
$$K^d(r,T) = N(r,T)(\frac{r+\delta}{\theta})^{\frac{1}{\theta-1}}$$
$$Y^d(r,T) = (K^d)^\theta (N)^{1-\theta}$$
$$B(r,T) = \phi_b Y^d(r,T)$$
$$G(r,T) = \phi_g Y^d(r,T)$$

2. Simultaneously solve for r, T such that the following holds:

$$A^s(r,T) = K^d(r,T) + B(r,T)$$
$$G(r,T) + rB(r,T) + T = \tau_y(rA^s(r,T) + wN(r,T)).$$

3.1 Results with preferences separable in labour

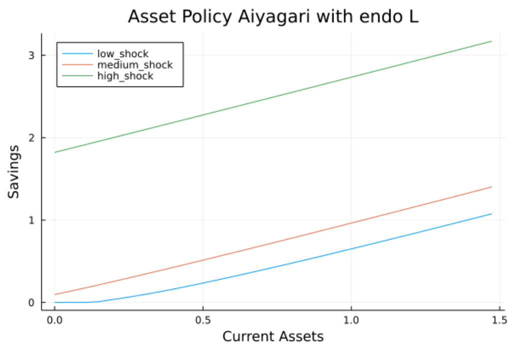

Figure 5: Asset Policy Function

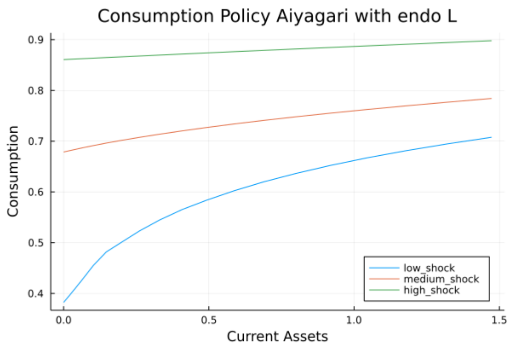

Figure 6: Consumption Policy Function

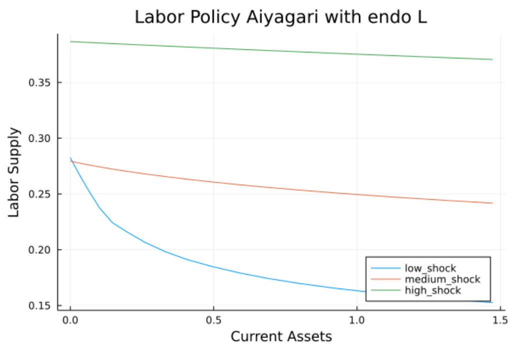

Figure 7: Labor Policy Function

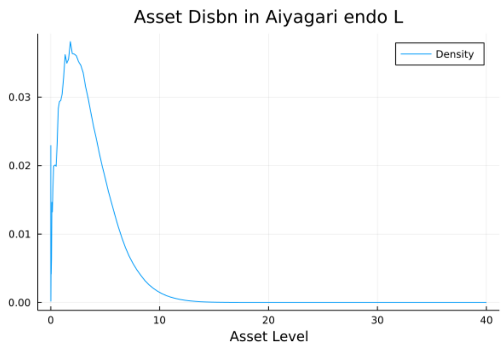

Figure 8: Asset Distribution