Macro, Money and (International) Finance – Problem Set 3

September 27, 2022

Prepared by Andrey Alexandrov (alexandrov@princeton.edu). Please let me know if any tasks are unclear or you find mistakes in the problem descriptions.

Please submit to `alexandrov@princeton.edu` by 1pm on Monday, October 3 (Princeton time).

1 Numerical Methods

1. Let's construct the M matrix for the model from Lecture 5. Recall that (arithmetic) drift and volatility of η are given by:

$$\mu_\eta(\eta) = \mu_t^\eta \eta = \eta(1-\eta_t)\left[(\varsigma_t^e - \sigma - \sigma_t^q)(\sigma_t^\eta + \sigma + \sigma_t^q) - (\varsigma_t^h - \sigma - \sigma_t^q)\left(-\frac{\eta_t \sigma_t^\eta}{1-\eta_t} + \sigma + \sigma_t^q\right) - \left(\frac{C_t^e}{N_t^e} - \frac{C_t^h}{N_t^h}\right)\right]$$

$$\sigma_\eta(\eta) = \sigma_t^\eta \eta = (\chi_t - \eta_t)(\sigma + \sigma_t^q)$$

Use parameter values from Problesm Set 2 and construct the M matrix on the $(0,1)$ grid with $\Delta\eta = 0.001$.[1]. You may use the provided function `buildM.m` or adapt it to your programming language.

2. Solve for the stationary distribution of η in three ways:

   (a) Iterating the distribution forward. Guess an initial distribution,[2] iterate forward using implicit method: $g^{n+1} = (I - \Delta t M')^{-1} g^n$ until convergence.

   (b) Solving given some value of g_1:

$$0 = M'g \Longleftrightarrow \begin{bmatrix} M'_{11} & M'_{1,2:N} \\ M'_{2:N,1} & M'_{2:N,2:N} \end{bmatrix} \begin{bmatrix} g_1 \\ g_{2:N} \end{bmatrix} = 0 \Longleftrightarrow g_{2:N} = -(M'_{2:N,2:N})^{-1} M'_{2:N,1} g_1$$

[1] To stay somewhat more general (and reuse this code later in the problem set), you can create additional variables for ς_t^i and $\frac{C_t^i}{N_t^i}$, assign them values appropriate for log utility, and refer to these variables in the expression for drift.

[2] E.g. uniform: `g = ones(N,1); g = g/(N*deta);`. The iteration algorithm preserves the mass of $g(\cdot)$, so it is convenient to have the initial density integrate to one.

(c) Solving for the null space (kernel) of the matrix M' using some built-in method (in Matlab the function is `null`).

Whenever you invert a matrix, use the 'backslash' operator (`A \ B` instead of `inv(A) * B` in Matlab) to speed up your code.

3. Confirm analytically that if $\Delta x \to 0$, then Δt must be of order $(\Delta x)^2$ to ensure monotonicity when using explicit method. You might find Yuliy's video "6. Explicit Scheme" helpful.

4. Consider a simple valuation equation:

$$v(x,t) = E\left[\int_t^T e^{-\rho s} u(x_s) ds \;\middle|\; x_t = x\right]$$

Let x_t evolve on the $[-1,1]$ interval in the following way:

$$dx_t = -x_t dt + \sigma(1 - x_t^2) dZ_t$$

(a) Write down the HJB equation for $v(x,t)$

(b) Set the grid for $x = [-1,1]$ with $\Delta x = 0.1$ and time grid $[0,T]$ with $\Delta t = 0.01$ and $T = 1$. Construct the M matrix and solve backward in time (starting at $t = T$) for the value function $v(x,t)$ using explicit and implicit methods. Set $\rho = 0.1$, $\sigma = 1$, $u(x) = x$ and note that $v(x,T) = 0$.

(c) What happens if you set $\Delta x = 0.01$ keeping the rest of the parameters same?

2 Two Sector Model with CRRA Utility

In this exercise you will program the model from Lecture 6 under the assumption of CRRA utility. Set $\gamma = 2$ and the rest of parameters as in Problem Set 2. You should make use of the code you wrote for log utility, or take the code from the google drive folder.

1. First, adjust the 'inner loop' of the problem, so that it can solve for $q, \kappa, \chi, \iota, \sigma^q$ given value functions v^h and v^e. Turn your code from Problem Set 2 into a function that takes model parameters and value functions v^h and v^e as inputs, and returns $q, \kappa, \chi, \iota, \sigma^q$ as output. Alternatively, you may start with the `inner_loop.m` function from the google drive folder.

2. Second, write a function that updates value function v given a vector of its values at time t, discount rate ρ, vector of 'payoff flows'[3] u, drift of η ($\eta\mu^\eta$) and volatility of η ($\eta\sigma^\eta$):

$$\rho v_{t-\Delta t} = \frac{v_t - v_{t-\Delta t}}{\Delta t} + u + M v_{t-\Delta t}$$
$$v_{t-\Delta t} = \left((1+\rho\Delta t)I_N - \Delta t M\right)^{-1}(\Delta t u + v_t)$$

3. Finally, write an outer loop that (i) uses the inner loop to compute $q, \kappa, \chi, \iota, \sigma^q$, (ii) computes $\mu^\eta, \sigma^\eta, u^i$, (iii) updates value functions. For the initial guess of value functions, you may use $v^i = (a^e)^{-\gamma}(\eta^i)^{1-\gamma}$.

[3] The vector corresponding to $u(v,\eta)$ from Lecture 8. You would have to rearrange terms slightly, just like we did in the lecture.

3 Two Sector Model with Epstein-Zin Utility

Repeat the exercise, now with Epstein-Zin preferences (under IES = 1 and $\gamma = 2$). Reuse the code you wrote in Problem 2, note that it requires only a couple adjustments:

1. In the 'inner loop': the goods market clearing condition is simplified since $\frac{C_t^i}{N_t^i} = \rho^i$ now.
2. In the main file: the 'payoff flow' term u needs to be adjusted. To do that, use the expression for $\frac{\partial f}{\partial U}$ on Slide 9 of Lecture 6, plug in $U = V_i = \frac{1}{\rho^i}\frac{(\omega^i n^i)^{1-\gamma}}{1-\gamma}$ and rewrite such that $\frac{\partial f}{\partial U}$ is expressed in terms of $\gamma, \rho^i, v^i, q, \eta^i$.

Compare the model with log utility ($\gamma = \psi = 1$), CRRA utility ($\gamma = 2$, $\psi = 1/2$) and Epstein-Zin utility ($\gamma = 2$, $\psi = 1$). Plot graphs with variables that you find most informative for the comparison and dicsuss the differences.