Econ 8210: Computational Economics
Homework II

Haosi Shen

Fall 2024

Model

Consider the stochastic neoclassical growth model, where there is a representative household with preferences over consumption c_t and labor l_t:

$$\mathbb{E}_0 \sum_{t=0}^{\infty} \beta^t \Big(\log c_t - \frac{l_t^2}{2} \Big)$$

where $\beta = 0.97$.
The household consumes, saves, and works with a budget constraint:

$$c_t + i_t = w_t l_t + r_t k_t$$

The production function is Cobb-Douglas with capital share $\alpha = 0.33$:

$$c_t + i_t = e^{z_t} k_t^{\alpha} l_t^{1-\alpha}$$

with the law of motion of capital:

$$k_{t+1} = (1 - \delta) k_t + i_t$$

where depreciation $\delta = 0.1$, and technology level z_t that follows an AR(1) process:

$$z_t = \rho z_{t-1} + \sigma \varepsilon_t$$

where $\rho = 0.95$, $\sigma = 0.007$, and $\varepsilon_t \sim N(0, 1)$.

Social Planner's Problem

Putting together the resource constraints, the social planner's problem in recursive formulation is

$$
\begin{aligned}
V(k,z) = \max_{c,l,k'} & \left\{ \log c - \frac{l^2}{2} + \beta \mathbb{E}\big[V(k',z') \mid z\big] \right\} \\
\text{s.t.} \quad & c + k' = e^z k^\alpha l^{1-\alpha} + (1-\delta)k \\
& z' = \rho z + \sigma \varepsilon', \quad \varepsilon \sim \mathcal{N}(0,1)
\end{aligned}
$$

Equilibrium Characterization

Taking the FOCs and invoking the envelope theorem, we obtain the optimality conditions that characterize the equilibrium. Namely, we have the resource constraint, intertemporal Euler equation, and the intratemporal condition:

$$k_{t+1} = (1-\delta)k_t + e^{z_t} k_t^\alpha l_t^{1-\alpha} - c_t \tag{1}$$

$$\frac{1}{c_t} = \beta\, \mathbb{E}_t\left[\frac{1}{c_{t+1}}\Big(\alpha e^{z_{t+1}} k_{t+1}^{\alpha-1} l_{t+1}^{1-\alpha} + (1-\delta)\Big)\right] \tag{2}$$

$$c_t = \frac{e^{z_t} k_t^\alpha (1-\alpha)}{l_t^{\alpha+1}} \tag{3}$$

Assume that at the steady state, productivity shock $z_{ss} = 0$. The steady state equilibrium is characterized by:

$$c_{ss} = k_{ss}^\alpha l_{ss}^{1-\alpha} - \delta k_{ss} \tag{4}$$

$$1/\beta = \alpha k_{ss}^{\alpha-1} l_{ss}^{1-\alpha} + (1-\delta) \tag{5}$$

$$c_{ss} = \frac{k_{ss}^\alpha (1-\alpha)}{l_{ss}^{\alpha+1}} \tag{6}$$

To compute the steady state, I proceed by eliminating c_{ss} and solving from the capital-labor ratio, which is independent of l_{ss}, and finally backing up c_{ss} and k_{ss}. Table 1 displays the levels of consumption, labor, and capital at the deterministic steady state.

Table 1: Steady-State

Variable	**Steady-State Value**
Consumption (c_{ss})	1.1161
Labor (l_{ss})	0.9465
Capital (k_{ss})	3.7612
Output (y_{ss})	1.4923

1 Spectral Method: Chebyshev Polynomials

Compute the solution to the model when you use six Chebyshev polynomials on capital and a three-point finite approximation z_t using Tauchen's method.

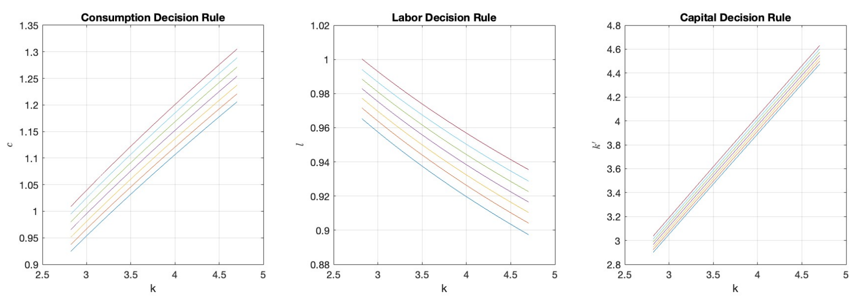

Figure 1: Policy Functions using Chebyshev Polynomials

I discretized the productivity shocks using Tauchen's method ($m = 3,\ N = 7$). The collocation is based on 6th order Chebyshev polynomials. Figure 1 and 2 illustrate the policy functions and Euler equation residuals respectively. Figure 3 shows the distribution of simulated capital.

2 Finite Elements

Compute the solution to the model when you use eight finite elements on capital and a three-point finite approximation to z_t using Tauchen's method. Use a Galerkin weighting scheme.

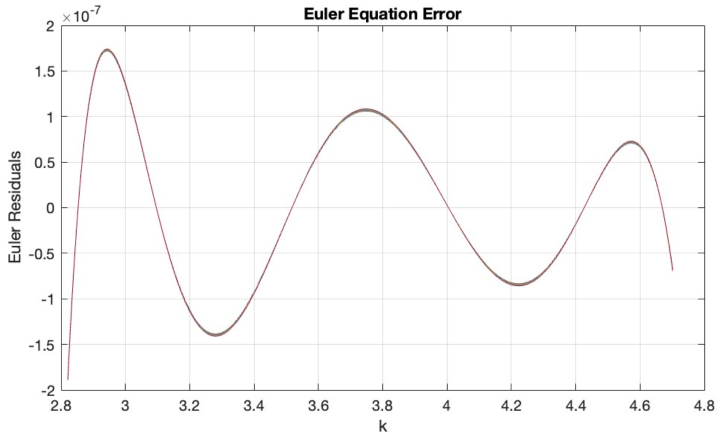

Figure 2: Euler Equation Errors (Spectral Method)

Similar to Chebyshev, the finite elements method also relies on projection with respect to a basis. I use tents as the functional basis. The main difference with Chebyshev is that, instead of evaluating the Euler equation at the collocation nodes, we evaluate it at a finer capital grid and integrate over that grid using the finite elements. In particular, we compute consumption with the respective tents, evaluate the residuals over that fine grid, weight those residuals by the basis using the Galerkin scheme, and finally integrate over the subintervals.

3 Perturbation

Find a third-order perturbation solution to the model.

My approach is to reconstruct the policy functions from the output of the model solved in Dynare, by substituting in the coefficient matrices to the Taylor approximation.

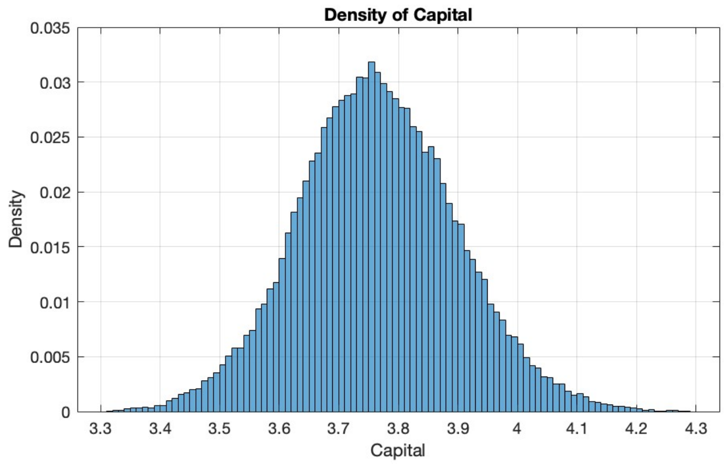

Figure 3: Distribution of Capital (Spectral Method)

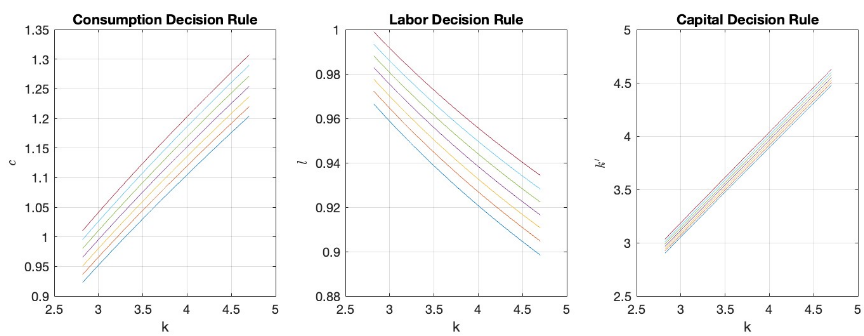

Figure 4: Policy Functions using Finite Elements

4 Deep Learning

Compute the solution to the model when you use a neural network with three layers and 25 nodes per layer on capital and a three-point finite approximation to z_t using Tauchen's method.

- I constructed a neural network with three layers: an input layer featured by (k, z); two hidden layers, each with 25 nodes, using ReLU activation; and an output layer predicting next-period capital.
- The loss is characterized by the Euler equation error. I am using the

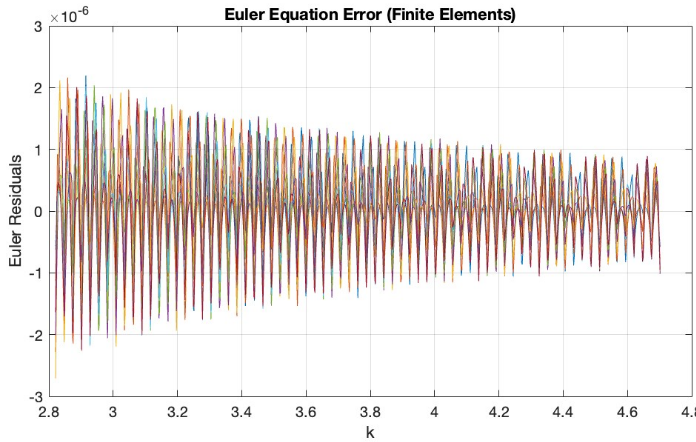

Figure 5: Euler Equation Residuals (Finite Elements)

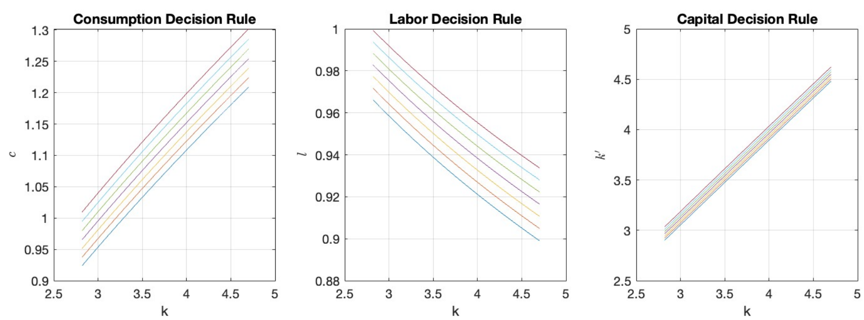

Figure 6: Policy Functions using Perturbations

Adam optimizer with a learning rate of 0.001, chosen for its stability and faster convergence. While minimizing the Euler equation residual directly, I use the Mean Squared Error approach to penalize deviations from intertemporal optimality.

5 Comparison

- *Chebyshev Polynomials (Spectral Method)*: The policy functions are smooth and capture nonlinearities well. The approximation is fairly well, as the Euler equation residuals are relatively small and evenly

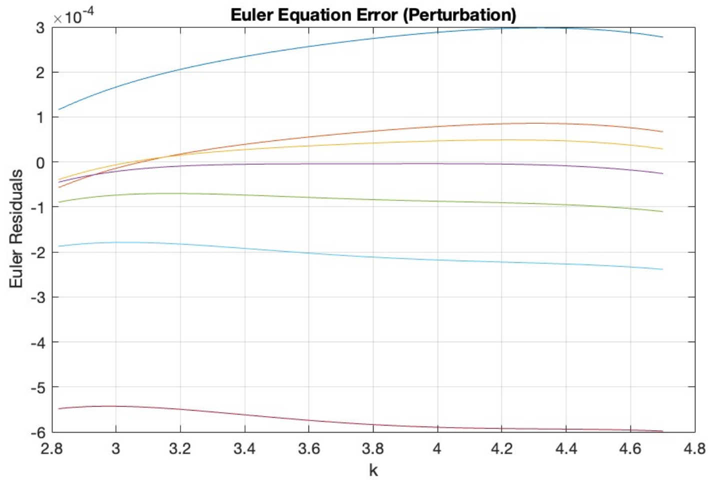

Figure 7: Euler Equation Residuals (Perturbation)

distributed across the state space. Note that the approximations may be less accurate near the boundaries of the state space due to the use of fixed collocation nodes.

- *Finite Elements*: The Euler equation residuals appear well-controlled and more evenly distributed than in the spectral method, particularly for finer grids. The finite elements method tends to be more robust for capturing potential irregularities due to its local basis functions.

- *Perturbation*: This works better for approximating solutions near the steady state due to its use of the analytical Taylor expansion. The policy functions are smooth and capture nonlinearities well. Overall, this method is more sensitive to stochastic shocks compared to the other three.

- *Neural Network*: Deep learning works better over broader state spaces and higher dimensions, since it does not rely on predefined basis functions. The nonlinear dynamics are well captured. However, my solution might be subject to potential overfitting in some regions. Revising the network configuration or training parameters may help improve the results. In terms of computational costs, this is the most demanding among the four due to its iterative training and back-propagation processes.

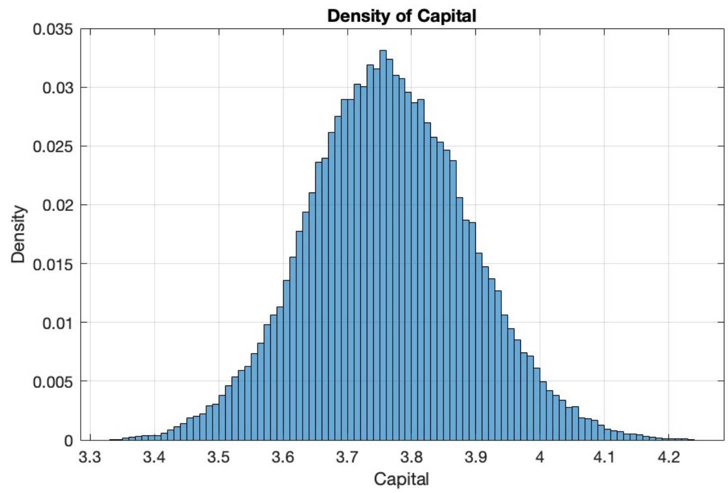

Figure 8: Distribution of Capital (Perturbation)

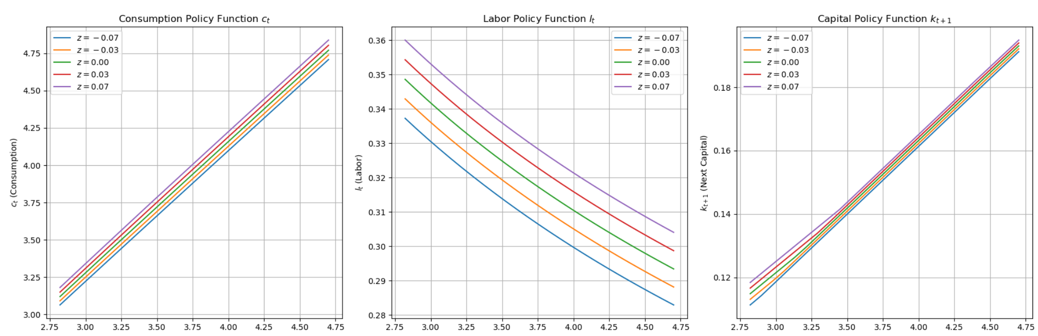

Figure 9: Policy Functions using Neural Network