

Macro II

Professor Griffy

UAlbany

Spring 2022

Introduction

- ▶ Today: Solution techniques I
- ▶ Some solution techniques:
 - ▶ Value function iteration (you've basically done this);
 - ▶ guess and verify (method of undetermined coefficients).
- ▶ Test on Thursday.
- ▶ Homework due tonight! (So I can release answer key before exam...)

Solution Methods: Introduction

- ▶ Today we will explore solutions to the Neoclassical Growth model.
- ▶ The first two directly solve the problem at hand:
 1. guess and verify: guess the functional form for the value/policy function
 2. (by hand) value function iteration: iterate on value function until you see a pattern.
- ▶ First: recall definition of recursive competitive equilibrium.

Recursive Competitive Equilibrium

Definition. Let K denote the state of the aggregate economy and a denote the personal state of an agent. Then a *Recursive Competitive Equilibrium (RCE)* is a set of functions that describe

Quantities: $K' = G(K)$, $a' = g(a, K)$ (agg. and ind. policy fns)

Lifetime Utility: $V(a, K)$ (the value function)

Prices: $r(K) = f_K(K, N) + 1 - \delta$, $w(K) = f_N(K, N)$ (competitive prices)

such that:

1. Prices are complete and given
2. $V(a, K)$ and $g(a, K)$ solve the consumer's problem
3. Consistency: $G(K) = g(K, K)$

That is, households must know prices so they can make utility maximizing decisions and if we gave on person all of the capital, their choice would coincide with the aggregation of choices.

Guess and Verify

- ▶ This method is also known as the *Method of Undetermined Coefficients*
- ▶ It involves *guessing* the form of the value/policy function, and then *verifying* that the guess is consistent with the optimization problem
- ▶ The idea is that, if the guess is correct, then when “operated on” it should recover that same form and we can back-out the coefficients previously left *undetermined*
 - ▶ Recall our solution is a fixed point: $Tv^* = v^*$
- ▶ This solution technique has several requirements:
 - ▶ Unique Solution
 - ▶ “Correct Guess”

Guess and Verify: Policy Function

Neoclassical growth model. Households solve the following:

$$U = \max_{\{c_t\}_{t=0}^{\infty}} \sum_{t=0}^{\infty} \beta^t \ln(c_t) \quad \text{s.t.} \quad c_t + k_{t+1} = k_t^\alpha$$

The recursive problem is given by the following

$$V(k) = \max_{k'} \{ \ln(k^\alpha - k') + \beta V(k') \}$$

Steps:

1. Guess that $k' = \eta k^\alpha$.
2. Solve for the undetermined coefficient, η , and
3. find the policy function for capital and consumption

Policy G & V

First take the F.O.C. of the optimization problem.

$$\frac{1}{k^\alpha - k'} = \beta \frac{dV(k')}{dk'}$$

Note that there is a k' on the LHS.

Optimal policy fn must satisfy the F.O.C \rightarrow we plug in our guess.

$$\frac{1}{k^\alpha - (\eta k^\alpha)} = \beta \frac{dV(k')}{dk'} \quad \implies \quad \frac{1}{(1 - \eta)k^\alpha} = \beta \frac{dV(k')}{dk'}$$

Envelope theorem for $dV(k')/dk'$:

$$\frac{dV(k)}{dk} = \frac{1}{k^\alpha - k'} (\alpha k^{\alpha-1}) = \underbrace{\frac{1}{(1 - \eta)k^\alpha} (\alpha k^{\alpha-1})}_{\text{plug in the guess}} = \frac{\alpha}{(1 - \eta)k}$$

Policy G & V II

“Pushing forward,” we have

$$\frac{dV(k')}{dk'} = \frac{\alpha}{(1-\eta)k'} = \underbrace{\frac{\alpha}{(1-\eta)\eta k^\alpha}}_{\text{plug in the guess}}.$$

Plug into the F.O.C. \rightarrow can now solve for undetermined coeff. η .

$$\frac{1}{(1-\eta)k^\alpha} = \frac{\alpha\beta}{(1-\eta)\eta k^\alpha} \quad \implies \quad \eta = \alpha\beta,$$

which we can verify is a constant. Policy fns:

$$k' = \alpha\beta k^\alpha \quad \text{and} \quad c = (1 - \alpha\beta)k^\alpha.$$

Guess and Verify: Value Function

Now let's try our hand at guessing the value function for the same problem:

$$V(k) = \max_{k'} \{ \ln(k^\alpha - k') + \beta V(k') \} .$$

1. Guess that $V(k) = A + B \ln(k)$, where A and B are the undetermined coefficients.
2. Note: guess should always mirror utility function + constant.

Value G & V

First solve the RHS maximization problem, with guess for $V(k')$.

$$\frac{dRHS}{dk'} = \frac{d}{dk'} \{ \ln(k^\alpha - k') + \beta [A + B \ln(k')] \} = 0$$

$$\Rightarrow k' = \frac{\beta B k^\alpha}{1 + \beta B}$$

Now, to evaluate the RHS at the optimum we plug in k' .

$$\begin{aligned} RHS(k'^*) &= \ln \left(k^\alpha - \frac{\beta B k^\alpha}{1 + \beta B} \right) + \beta \left[A + B \ln \left(\frac{\beta B k^\alpha}{1 + \beta B} \right) \right] \\ &= \ln \left(\frac{k^\alpha}{1 + \beta B} \right) + \beta A + \beta B \ln \left(\frac{\beta B k^\alpha}{1 + \beta B} \right) \end{aligned}$$

Verifying

Now, group the constants together and the k terms separately.

$$\begin{aligned} RHS(k'^*) &= \underbrace{-\ln(1 + \beta B) + \beta A + \beta B \ln\left(\frac{\beta B}{1 + \beta B}\right)}_{\text{constant}} \\ &\quad + \underbrace{\alpha(1 + \beta B)}_{\ln(k)\text{-term coeff.}} \ln(k) \end{aligned}$$

The V on the LHS should also have the form of our guess:

$$A = \text{constant} = -\ln(1 + \beta B) + \beta A + \beta B \ln\left(\frac{\beta B}{1 + \beta B}\right)$$

$$B = \ln(k)\text{-term coeff.} = \alpha(1 + \beta B)$$

Because it can be written as $V(k) = A + B \ln(k)$, guess is verified.

Value G & V III

Now solve for B using the second equation, and then A from the first.

$$B = \frac{\alpha}{1 - \alpha\beta} \quad \text{and} \quad A = \frac{1}{1 - \beta} \left[\ln(1 - \alpha\beta) + \frac{\alpha\beta}{1 - \alpha\beta} \ln(\alpha\beta) \right]$$

We might also want to get the policy function. Recall that we actually solved for it earlier. All we need to do is plug in for B .

$$k' = \frac{\beta B k^\alpha}{1 + \beta B} \quad \implies \quad k' = \alpha\beta k^\alpha$$

Value Function Iteration

- ▶ This method relies on the ideas of a contraction / fixed point
- ▶ Recall: (*paraphrasing*) "starting from any possible V_0 in the space of potential solutions, when iterating on V_0 we will get closer and closer to the true V "
- ▶ We are going to make an initial guess at V (generally $V_0 = 0$), and then iterate.
- ▶ Look for a pattern.

Value Function Iteration

- ▶ The above is true if we are assuming an infinite horizon; what if there is some terminal period T (for example, a lifecycle model)?
- ▶ Then you can think of this process like you would backwards induction...
 - ▶ V_0 is the value at $T + 1$ ($= 0$)
 - ▶ V_1 is the value you get in T when optimizing knowing that $V_0 = 0$
 - ▶ etc.
- ▶ You would continue this process until you found the value function for the first date through the last date:
 V_T, \dots, V_1, V_0 , and your answer would effectively be the sequence of value (or policy) functions

First Iteration

Let's try our hand at an easy example; consider the problem we've been working with and initialize the iterations with $V_0(k') = 0$

$$V_1(k) = \max_{k'} \left\{ \ln(k^\alpha - k') + \beta \underbrace{0}_{V_0(k')} \right\}$$

Optimization requires that $k' = g_1(k) = 0$ (calculus doesn't work here, we are at a "corner"). Knowing this, we can plug in this optimal policy to find $V_1(k)$.

$$V_1(k) = \ln(k^\alpha - 0) = \alpha \ln(k)$$

Now for a second round . . .

Second Iteration

$$V_2(k) = \max_{k'} \{ \ln(k^\alpha - k') + \beta \underbrace{\alpha \ln(k')}_{V_1(k')} \}$$

Let's optimize.

$$\frac{dV_2(k)}{dk'} = 0 \quad \implies \quad k' = g_2(k) = \frac{\alpha\beta}{1 + \alpha\beta} k^\alpha$$

Knowing this, we can plug in this optimal policy to find $V_2(k)$.

$$\begin{aligned} V_2(k) &= \ln\left(\frac{k^\alpha}{1 + \alpha\beta}\right) + \alpha\beta \ln\left(\frac{\alpha\beta}{1 + \alpha\beta} k^\alpha\right) \\ &= \ln\left(\frac{1}{1 + \alpha\beta}\right) + \alpha\beta \ln\left(\frac{\alpha\beta}{1 + \alpha\beta}\right) + \alpha(1 + \alpha\beta) \ln(k) \end{aligned}$$

Let's go for a third, round, paying attention to an emerging pattern.

Third Iteration

$$V_3(k) = \max_{k'} \left\{ \ln(k^\alpha - k') \right. \\ \left. + \beta \underbrace{\left[\ln\left(\frac{1}{1 + \alpha\beta}\right) + \alpha\beta \ln\left(\frac{\alpha\beta}{1 + \alpha\beta}\right) + \alpha(1 + \alpha\beta)\ln(k') \right]}_{V_2(k')} \right\}$$

Optimize.

$$\frac{dV_3(k)}{dk'} = 0 \quad \implies \quad k' = g_3(k) = \frac{\alpha\beta + (\alpha\beta)^2}{1 + \alpha\beta + (\alpha\beta)^2} k^\alpha$$

Plug the optimal policy back in to find $V_3(k)$.

$$V_3(k) = \beta \ln\left(\frac{1}{1 + \alpha\beta}\right) + \alpha\beta^2 \ln\left(\frac{\alpha\beta}{1 + \alpha\beta}\right) + (\alpha\beta + (\alpha\beta)^2) \ln\left(\frac{\alpha\beta + (\alpha\beta)^2}{1 + \alpha\beta + (\alpha\beta)^2}\right) \\ + \alpha(1 + \alpha\beta + (\alpha\beta)^2) \ln(k)$$

Pattern

We could continue (or make our computer do it), but for this problem we can see a pattern emerge. We can see that as we let the iteration $s \rightarrow \infty$, we'll have

$$g^*(k) = \lim_{s \rightarrow \infty} g_s(k) = \alpha\beta k^\alpha.$$

See guess earlier for how to “derive” the above. The Value function itself can be shown to converge to

$$V^*(k) = \lim_{s \rightarrow \infty} V_s(k) = \frac{1}{1 - \beta} \left[\ln(1 - \alpha\beta) + \frac{\alpha\beta}{1 - \alpha\beta} \ln(\alpha\beta) \right] \\ + \frac{\alpha}{1 - \alpha\beta} \ln(k),$$

which is what we found earlier with *Guess and Verify*.

Functional Euler Equation

- ▶ This last “method” doesn’t actually solve for the value / policy functions directly (though you can back them out)
- ▶ Sometimes you might see this called the Euler-Lagrange Equation
- ▶ It involves the construction of a “Lagrangian” using the Bellman Operator; here, though, we are mapping functions to functions (hence “functional equation”)
- ▶ The procedure should feel very familiar to you, as you’ve sort of seen it in previous sections; set up the “Lagrangian” and take F.O.C.s, find the EE and use constraints / conditions (e.g. market clearing) to solve for whatever you desire

Functional Euler Equation

Just to get an idea, let's try it out on our running example:

$$\mathcal{L} = \ln(c) + \beta V(k') + \lambda [k^\alpha - c - k']$$

$$\frac{\partial \mathcal{L}}{\partial c} = 0 : \quad \lambda = \frac{1}{c} \quad (1)$$

$$\frac{\partial \mathcal{L}}{\partial k'} = 0 : \quad \lambda = \beta \frac{dV(k')}{dk'} \quad (2)$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = 0 : \quad c + k' = k^\alpha \quad (3)$$

Combine (1) and (2).

$$\frac{1}{c} = \beta \frac{dV(k')}{dk'} \quad (4)$$

Functional Euler Equation

Now let's use the envelope theorem. Evaluating \mathcal{L} at the same value as V on the RHS, the two are equal. Further we know that $V \leq \mathcal{L}$. Thus

$$\frac{\partial \mathcal{L}}{\partial k} = \frac{dV(k)}{dk} \implies \frac{dV(k)}{dk} = \lambda[\alpha k^{\alpha-1}]$$

Pusing this forward, and plugging in for $\lambda' = 1/c' \dots$

$$\frac{dV(k')}{dk'} = \frac{1}{c'}[\alpha k'^{\alpha-1}], \quad (5)$$

noting that $\alpha k^{\alpha-1} = 1 + r$ when $\delta = 1$ (which is what we have in this example).

$$\rightarrow \frac{1}{c} = \beta[\alpha k'^{\alpha-1}] \frac{1}{c'} \quad (\text{EE})$$

Functional Euler Equation

- ▶ From here we can do many things. . .
- ▶ You might want to find steady state values: plug in to budget constraints / market clearing conditions, do comparative statics, etc.
- ▶ Alternatively, you may want to recover the policy function: one way would be to iterate on the EE like we did in section 2 (you would find $k' = \alpha\beta k^\alpha$)
- ▶ This technique may seem more roundabout than last time (recall I just plugged the constraint right into the objective), but just note that this method is slightly more general insofar as it handles situations where you can't easily substitute in all constraints

Conclusion

- ▶ Next time: midterm!
- ▶ Homework due Tonight!