Applications

1 How to Compute Solutions

- We face the sequence problem

\[
\sup_{\{x_t\}_{t=0}^{\infty}} \sum_{t=0}^{\infty} \beta^t F(x_t, x_{t+1}) \quad \text{s.t.} \\
x_{t+1} \in \Gamma(x_t) \quad \forall t \\
x_0 \text{ given (1)}
\]

- We want to find: what is the optimal \(x_{t+1}\) for a given \(x_t\)

- We set up the problem recursively:

\[
V(x) = \sup_{y \in \Gamma(x)} F(x, y) + \beta V(y) \quad (2)
\]

- Suppose the following conditions hold:

1. \(\Gamma\) nonempty, compact and continuous
2. \(F\) is bounded
   - \(F\) bounded has three consequences:
     (a) it ensures that the limit of the infinite sum exists (and is finite)
     (b) it ensures that any solution \(V\) to the Bellman equation is bounded, so

\[
\lim_{T \to \infty} \beta^T V(x_T) = 0
\]

so that theorem SLP 4.3 (which ensures that the solution to the Bellman equation is the value function of the sequence problem) holds

(c) it allows us to use Blackwell’s sufficient conditions to show that \(T\) is a contraction
- The Contraction Mapping Theorem gives us a recipe for finding $V$:

1. Define the operator $T$ the way we did last class:

$$T(f)(x) = \max_{y \in \Gamma(x)} F(x,y) + \beta f(y)$$

2. Guess any $V_0 : X \to \mathbb{R}$

3. Define $V_{n+1}$ by

$$V_{n+1}(x) = T(V_n)(x)$$

- To completely define this function, it needs to be defined for every $x \in X$.
- Since $X$ is a compact subspace of $\mathbb{R}^n$ (we have assumed this to prove theorems), then it has infinite elements to the maximization on the RHS of (2) needs to be done an infinite amount of times.
- For computational purposes, we discretize: define a finite grid of points

$$G = \{x_1, x_2, \ldots, x_N\} \subset X$$

- Two separate steps of discretization:
  
  (a) Solve RHS (2) only for points on the grid. This gives us $V_{n+1}(x)$ for every $x \in G$
  
  (b) Consider only values of $y$ on the grid when we solve (2), i.e. replace the RHS of (2) with

$$\max_{\substack{y \in \Gamma(x) \\ y \in G}} F(x,y) + \beta V(y)$$

4. The hope is that if $G$ has enough points, solving the discretized problem is a good approximation to solving the real problem

- Step $a$ is unavoidable
- Step $b$ can be replaced by a different sort of approximation:
  
  - Suppose that we have defined $V_n$ for every point in $x$, not just in $G$
  - Then we can solve the RHS of (2) letting $y$ take any value, not just those in $G$
  - This will give us values for $V_{n+1}(x)$ for all $x \in G$
  - In order to be able to do the same thing in the next step, we need to extend $V_{n+1}$ to all points in $X$
  
  Various forms of interpolation: linear interpolation, cubic splines
• If you doing both steps of discretization, a useful computational trick is, before you start, compute the values of $F(x,y)$ for all $x \in G$, $y \in G$ and store them in a matrix
  - This makes the maximization part of solving (3) a matter of looking up values on a matrix rather than evaluating function $F$ each time

4. Compute the distance between $V_{n+1}$ and $V_n$ (using sup distance)

• If you used both steps of discretization, this is

$$\rho_{n+1} = \max_{x \in G} |V_{n+1}(x) - V_n(x)|$$

• If you have interpolated $V$ to all $X$ you can instead compute

$$\rho_{n+1} = \max_{x \in X} |V_{n+1}(x) - V_n(x)|$$

(the two versions should be very similar)

5. If $\rho_{n+1} < \varepsilon$ for some prespecified tolerance $\varepsilon$, then declare that $V_n \approx V$. Otherwise, repeat steps 3 and 4 until the inequality holds

• This is justified by:

$$\rho(T(V_{n+1}), T(V_n)) \leq \beta \rho(V_{n+1}, V_n)$$
$$\rho(V_{n+2}, V_{n+1}) \leq \beta \rho(V_{n+1}, V_n)$$
$$\rho(V_{n+2}, V_n) \leq (1 + \beta) \rho(V_{n+1}, V_n)$$
$$\ldots$$

$$\rho(V, V_n) \leq \sum_{k=0}^{\infty} \beta^{k-1} \rho(V_{n+1}, V_n)$$

$$\leq \frac{\varepsilon}{1 - \beta}$$

6. Find the policy function by solving the RHS of either (2) or (3) for every $x \in G$, using $V_n$

### 1.1 Policy function iteration (sometime known as Howard’s improvement algorithm)

• Maximization is computationally expensive

• Iteration is not
Consider the following operator
\[ T_g : S \to S \]
\[ T_g(f)(x) = F(x, g(x)) + \beta f(x) \]

This is just like our operator \( T \), except that there is no maximization: we just fix a policy function \( g \).

Easy to check that \( T_g \) is a contraction mapping.

The fixed point of \( T_g \) is \( V_g \), which satisfies
\[ V_g(x) = F(x, g(x)) + V_g(g(x)) \] (4)

Economic interpretation: value of just following policy \( g \).

Two ways to compute \( V_g \):

1. Iterate, just as above, except without maximizing
2. Manipulate matrices: (this only works if you did both steps of discretization)
   (a) Let \( v_g \) be a \( N \times 1 \) vector such that \( v_{gi} = V_g(x_i) \) for \( i = 1, \ldots, N \).
      - \( v_g \) is what we want to find out
   (b) Let \( J \) be a \( N \times N \) matrix of zeros and ones where
      \[ J_{ij} = \begin{cases} 1 & \text{if } g(x_i) = x_j \\ 0 & \text{otherwise} \end{cases} \]
      - Each row of \( J \) has exactly one 1 and \( N - 1 \) zeros
      - \( J \) summarizes the discretized policy function \( g \)
      - For each \( x_i \in G \) it indicates what element of \( G \) satisfies \( x_j = g(x_i) \)
   (c) Let \( F \) be a \( N \times 1 \) vector such that \( F_i = F(x_i, g(x_i)) \)
   (d) The discretized version of (4) can be written as
      \[ (I - \beta J) v_g = F \]
      \[ N \times N \quad N \times N \quad N \times 1 \quad N \times 1 \]
      or
      \[ v_g = (I - \beta J)^{-1} F \] (5)
      so you can just compute \( v_g \) by inverting the matrix \( (I - \beta J) \)
• What’s the use of this?

1. A good way to come up with an initial guess for $V$

   • Propose a simple policy, e.g. $g(x_i) = x_i$
   • Find $V_g$ by either iterating without maximization or by using (5)
   • For the simple policy $g(x_i) = x_i$, then $J$ is an identity matrix, so

   $$v_g = (1 - \beta)^{-1} F$$

   which means

   $$v(x) = \frac{F(x,x)}{1 - \beta}$$

   (unsurprisingly)

2. A way to save on computation time. Modify the algorithm as follows:

   (a) Guess a policy function $g_0$. The same logic can be applied whether you interpolate and allow $g_0(x_i) \notin G$ or you stick to the grid and require $g_0(x_i) \in G$.
   (b) Find $V_g$ for policy function $g_0$ by either iterating without maximization or by using (5)
   (c) For each $x_i \in G$, find $y$ that solves:

   $$\max_{y \in \Gamma(x_i)} F(x_i, y) + \beta V_g(y)$$

   and maybe $y \in G$
   (d) Find $V_g$ for policy function $g_1$
   (e) Iterate until the successive rounds of $V_g$ are sufficiently close

• Note: In step (b) if you are doing by iteration rather than matrix inversion, there is no need to wait until convergence, you can just do a fixed number of iterations

2 More on the Neoclassical Growth Model

• Recursive version of the planner’s problem:

   $$V(k) = \max_{k' \in [0, f(k) + (1 - \delta)k, (1 - \delta)k]} \left[ u(f(k) + (1 - \delta)k - k') + \beta V(k') \right]$$
• Define $\bar{k}$ by

$$f(\bar{k}) + (1 - \delta) \bar{k} = \bar{k}$$

• Interpretation: maximum capital level such that $k' \geq k$ is feasible

• Let the set $X$ be $[0, \max\{\bar{k}, k_0\}]$

Assumption 1. $u(0)$ is finite

• Under this assumption, $u$ is bounded and we can use our results

• This rules out using $u(c) = \log(c)$. To extend our results to that case is a bit more work

Proposition 1.

1. The value function for the planner’s sequence problem is the unique bounded solution to the Bellman equation

2. $V$ is continuous, strictly increasing, strictly concave and differentiable

3. The policy function $g(k)$ is continuous

2.1 Is the policy function increasing?

Proposition 2. $g(k)$ is nondecreasing

Proof. Proof by Topkis.

• The function

$$R(k, k') = u(f(k) + (1 - \delta)k - k') + \beta V(k')$$

has increasing differences in $k, k'$. Why?

$$\frac{\partial R}{\partial k} = u'(f(k) + (1 - \delta)k - k') [f'(k) + (1 - \delta)]$$

which is increasing in $k'$

• Topkis Theorem: If $R$ has increasing differences, then

$$\arg \max_{k' \in \Gamma} R(k, k')$$

is nondecreasing in $k$ ($\Gamma$ here is a fixed set)
• **Therefore**

\[
\arg \max_{k' \in [0, f(k) + (1-\delta)k]} R(k, k')
\]

is also increasing in \(k\)

• Note that we used concavity of \(V\) to establish that there is a unique \(\arg \max\) (otherwise Topkis is in terms of lattices)

• Note that we didn’t use that \(V\) is differentiable

**Proof.** Proof by first order conditions

• Since \(V\) is differentiable, and concave, the FOC holds (as long as \(k'\) is interior):

\[
-u'(f(k) + (1-\delta)k - k') + \beta V'(k') = 0
\]

• The result follows from \(u\) and \(V\) being concave

• Note: the proof by FOCs assumes that \(k'\) is interior.

• Exercise: rule out corners

• Also, the proof by FOCs proves \(g(k)\) is strictly increasing

**Proposition 3.** Consumption \(c(k) = f(k) + (1-\delta)k - k'\) is increasing in \(k\)

**Proof.** Rewrite the FOC as

\[
u'(c(k)) = \beta V'(g(k))
\]

The result follows because both \(u\) and \(V\) are concave and \(g\) is increasing.

2.2 Do we converge to the steady state?

• We have already proved that there is a unique steady state where \(g(k) = k\) and \(k\) is positive.

  – If \(f(0) = 0\), then \(k = 0\) is another trivial steady state

• We didn’t quite prove that \(k_t\) would necessarily converge to the steady state.

**Proposition 4.** If \(k < k_{ss}\) then \(g(k) \in (k, k_{ss})\)

**Proof.**
1. Since $g$ is strictly increasing

$$g(k) < g(k_{ss}) = k_{ss}$$

2. Assume $g(k) < k$. Then

$$u'(f(k) + (1 - \delta)k - g(k)) = \beta V'(g(k))$$

$$> \beta V'(k)$$

$$= \beta u'(f(k) + (1 - \delta)k - g(k)) [f'(k) + (1 - \delta)]$$

$$> u'(f(k) + (1 - \delta)k - g(k))$$

- The first step is the FOC
- The second uses the assumption that $g(k) < k$
- The next is the envelope theorem
- The final one uses that for $k_{ss}$

$$\beta [f'(k_{ss}) + (1 - \delta)] = 1$$

and $k < k_{ss}$

- The same proof going the other way establishes that if $k > k_{ss}$ then $g(k) < k$

**Corollary 1.** The economy will converge monotonically to the steady state.

### 3 Investment with adjustment costs

- Profit function $\pi(k)$

- This could be the result of something like

$$\pi(k) = \max_L f(k, L) - wL$$

- Cost of investment (over and beyond the output you give up)

$$c(k, i)$$

- Dynamics of $k$:

$$k' = (1 - \delta)k + i$$
• Cash flow:

\[ F(k, k') = \pi(k) - i - c(k, i) \]
\[ = \pi(k) - k' + (1 - \delta)k - c(k, k' + (1 - \delta)k) \]

• No outside investment \(\rightarrow\) cash flow nonnegative

• Examples:
  - \( c = 0 \)
  - \( c = \left(\frac{i}{k}\right)^2k \)
  - \( c = \max\{-i, 0\} \)

• Assume:
  - \( \pi \) strictly increasing, concave & differentiable
  - \( c \) convex, differentiable and nonnegative
  - \( \frac{\partial^2 c}{\partial i \partial k} < 0 \) (i.e. a given level of investment is less costly if you have more capital)

• Sequence problem:

\[
\max_{\{k_{t+1}\}_{t=0}^{\infty}} \sum_{t=0}^{\infty} \beta^t F(k_t, k_{t+1})
\]

s.t.

\[ k_{t+1} \in [0, \pi(k_t) + (1 - \delta)k_t - c(k_t, k_{t+1} + (1 - \delta)k_t)] \]

• Recursive problem

\[ V(k) = \max_{k' \in [0,\pi(k) + (1-\delta)k - c(k, k' + (1-\delta)k)]} F(k, k') + \beta V(k') \]

• If we restrict the domain as is the Neoclassical model, then \( F \) is bounded

• By the results from last class (easy to check assumptions hold), we know \( V \) is bounded, strictly increasing, strictly concave and differentiable

• FOC:

\[
\frac{\partial F(k, k')}{\partial k'} + \beta V'(k') = 0
\]
\[ -1 - \frac{\partial c}{\partial k'} + \beta V''(k') = 0 \]
• Is the policy function increasing?
  
  − Suppose I increase $k$: $\downarrow \frac{\partial c}{\partial k'}$
  − Restore equality by raising $k'$ $\Rightarrow \uparrow \frac{\partial c}{\partial k'}$ AND $\downarrow V'(k')$

• Euler equation:

$$F_2(k, g(k)) + \beta F_1(g(k), g(g(k))) = 0$$

$$-1 - \frac{\partial c(k, g(k))}{\partial k'} + \beta \left[ \pi'(g(k)) + (1 - \delta) - \frac{\partial c(g(k), g(g(k)))}{\partial k} \right] = 0$$

• Interpretation

• What if there were no adjustment costs?

$$-1 + \beta [\pi'(g(k)) + (1 - \delta)] = 0$$

$$\pi'(g(k)) + (1 - \delta) = \frac{1}{\beta}$$

  − Equate gross return on capital to interest rate $\frac{1}{\beta}$
  − $g(k)$ does not depend on $k$: the problem is essentially static