Lecture 15: Mathematical Methods II
Recursive Methods and Introduction to Continuous Time Models

Adam Hal Spencer

The University of Nottingham

Advanced Monetary Economics 2018

Roadmap

Motivation

- All the way back in L1 we took a refresher on Lagrangian calculus for constrained optimisation.
- We studied discrete time dynamic models and optimisation.
- This lecture will introduce two more techniques:
  - **(i)** Dynamic programming (also known as recursive methods),
  - **(ii)** Continuous time optimisation.

Roadmap

Dynamic Programming

- Recall the social planner's problem we studied for the RBC model in L2.

$$\max_{\{c_t, n_t, k_{t+1}\}_{t=0}^{\infty}} \mathbb{E}_0 \sum_{t=0}^{\infty} \beta^t \left[\frac{c_t^{1-\sigma}}{1-\sigma} - \frac{n_t^{1+\psi}}{1+\psi} \right]$$

subject to their budget constraints and law of motion for capital

$$c_t + k_{t+1} - (1-\delta)k_t = \underbrace{ak^{\alpha}n^{1-\alpha}}_{\text{Output}}$$

$$\log(a_t) = \rho_a \log(a_{t-1}) + \epsilon_{a,t}, \;\; \epsilon_{a,t} \sim N(0,1)$$

$$k_{t+1} \geq 0 \;\; \forall t$$

$$k_0, a_0 \;\; \text{given}$$

Dynamic Programming

- We solved this problem using a Lagrangian and taking the derivatives.
- The above problem is known as a sequence problem.
- An alternative approach is to study what's known as the problem's recursive formulation.
- Also known as dynamic programming.

Dynamic Programming

- The recursive formulation of an optimisation problem relies on the fact that the sequence problem is infinite.
- That is: it involves choosing an infinite sequence of consumption, capital and labour.
- Instead of finding the sequence of infinite choices, we can solve for a function of the current state variables that applies for all time periods.

Recursive Formulation

- The recursive formulation for the social planner's problem above is given as

$$V(a,k) = \max_{c,k',n} \left[\frac{c^{1-\sigma}}{1-\sigma} - \frac{n^{1+\psi}}{1+\psi} \right] + \beta \mathbb{E}[V(a',k')] \tag{1}$$

subject to

$$c + k' - (1-\delta)k = ak^{\alpha} n^{1-\alpha}$$
$$\log(a) = \rho_a \log(a_{-1}) + \epsilon_a, \;\; \epsilon_a \sim N(0,1)$$

where the object $V(a,k)$ is referred to as the value function corresponding to state (a,k).

Value Function

- The value function gives us the value of the objective at the optimal solution to the problem, (for the given state).
- That is

$$V(a_0, k_0) = \sum_{t=0}^{\infty} \beta^t \left[\frac{(c_t^*)^{1-\sigma}}{1-\sigma} - \frac{(n_t^*)^{1+\psi}}{1+\psi} \right]$$

where $\{c_t^*, k_{t+1}^*, n_t^*\}_{t=0}^{\infty}$ solves the sequence problem.

- Where does this come from?

Recursive Formulation

- Heuristically

$$
\begin{aligned}
V(a_0, k_0) &= \max_{\{c_t, n_t, k_{t+1}\}_{t=0}^{\infty}} \ \mathbb{E}_0 \sum_{t=0}^{\infty} \beta^t \left[\frac{c_t^{1-\sigma}}{1-\sigma} - \frac{n_t^{1+\psi}}{1+\psi} \right] \\
&= \max_{\{c_0, k_1, n_0\}} \ \frac{c_0^{1-\sigma}}{1-\sigma} - \frac{n_0^{1+\psi}}{1+\psi} + \max_{\{c_t, k_{t+1}, n_t\}_{t=1}^{\infty}} \mathbb{E}_0 \sum_{t=1}^{\infty} \beta^t \left[\frac{c_t^{1-\sigma}}{1-\sigma} - \frac{n_t^{1+\psi}}{1+\psi} \right] \\
&= \max_{\{c_0, k_1, n_0\}} \ \frac{c_0^{1-\sigma}}{1-\sigma} - \frac{n_0^{1+\psi}}{1+\psi} + \beta \mathbb{E}_0 [V(a_1, k_1)]
\end{aligned}
$$

where the β comes out the front since the value function at $t = 1$ doesn't have the period utility discounted.

Recursive Formulation

- Notice that there are no time subscripts on any of the variables in the recursive formulation.
- Instead previous period varaibles are denoted by subscript $_{-1}$ and forward variables are with $'$ superscripts.
- Why?...

Recursive Formulation

- ...because this problem is time invariant.
- The value function and policy functions for the controls will be the same each period for each corresponding state vector.
- This is because the problem spans an infinite time horizon.
- In principle, the solution to the recursive formulation will be functions $V(a,k), k'(a,k), n(a,k)$ and $c(a,k)$.

Solution

- How do we solve this thing?
- Usual: substitute-in the resource constraint and take the FOCs.

$$\frac{\partial V(a,k)}{\partial k'} = 0 \Rightarrow (-1)(c)^{-\sigma} + \beta \mathbb{E}\left[\frac{\partial V(a',k')}{\partial k'}\right] = 0$$
$$\frac{\partial V(a,k)}{\partial n} = 0 \Rightarrow -n^{\psi} + a(1-\alpha)k^{\alpha}n^{-\alpha} = 0$$

where notice that only the FOC for k' has implications for next period's value function.

Solution

- We're done with the labour solution.
- What about capital though? What is $\frac{\partial V(a',k')}{\partial k'}$?
- We don't know what the value function is explicitly!

Envelope Theorem

- The Envelope Theorem says that

$$\frac{\partial V(a,k)}{\partial k} = \frac{\partial}{\partial k}\left\{\left[\frac{c^{1-\sigma}}{1-\sigma} - \frac{n^{1+\psi}}{1+\psi}\right] + \beta\mathbb{E}[V(a',k')]\right\}$$
$$= \frac{\partial}{\partial k}\left\{\left[\frac{[ak^{\alpha}n^{1-\alpha} + (1-\delta)k - k']^{1-\sigma}}{1-\sigma} - \frac{n^{1+\psi}}{1+\psi}\right] + \beta\mathbb{E}[V(a',k')]\right\}$$
$$= c^{-\sigma}[\alpha a k^{\alpha-1}n^{1-\alpha} + (1-\delta)]$$

that is — just look for the places where k features and take the derivative: no need to worry about functions of k.

Envelope Theorem

- We can then iterate forwards by one period

$$\frac{\partial V(a', k')}{\partial k'} = (c')^{-\sigma}[\alpha a'(k')^{\alpha-1}(n')^{1-\alpha} + (1-\delta)]$$

Euler Equation

- Combine the updated envelope condition with the FOC for capital to get

$$c^{-\sigma} = \beta \mathbb{E} \left\{ (c')^{-\sigma} [\alpha a' (k')^{\alpha-1} (n')^{1-\alpha} + (1-\delta)] \right\}$$

which is our standard Euler equation!

Roadmap

Motivation

- So far, we've only studied problems in discrete time $t \in \{0, 1, 2, ...\}$.
- Another tool we use is continuous time models where $t \in [0, \infty)$.
- It can make things easier to solve analytically.

Example

- Let's think about a simple motivating example.
- Say you are a consumer with a single gold coin.
- There is some probability that you will bump into someone who wants to trade you a good for that coin.
- There will be some value associated with having a coin, denote this by V_1 (one for has a coin).
- Once you give your coin up, you'll be a person without a coin, who has future value V_0 (zero for no coin).
- How can we characterise V_1?

Example

- You can take the limit of a discrete time problem.
- Assume that the length of a time period is given by $\Delta > 0$.
- We'll write-down the discrete time problem and then take the limit $\lim_{\Delta \to 0}$ of all the variables.
- Denote the probability you find someone who'll trade you a good for the coin by $p\Delta$ (i.e. the probability is proportional to the length of the time period).
- You receive utility $u > 0$ if you trade.
- Utilise continuous discounting of the form $e^{-r\Delta}$ where $r > 0$ is a discount rate.

Example

- Write the discrete time Bellman equation as

$$
\begin{aligned}
V_1 &= (p\Delta)[u + e^{-r\Delta}V_0] + (1 - p\Delta)e^{-r\Delta}V_1 \\
\Rightarrow 0 &= (p\Delta)[u + e^{-r\Delta}V_0] + (1 - p\Delta)e^{-r\Delta}V_1 - V_1 \\
\Rightarrow 0 &= (p\Delta)u + (p\Delta)e^{-r\Delta}(V_0 - V_1) + [e^{-r\Delta} - 1]V_1.
\end{aligned}
$$

Now divide-through by Δ to get

$$0 = pu + pe^{-r\Delta}(V_0 - V_1) + \frac{[e^{-r\Delta} - 1]}{\Delta}V_1.$$

Remember L'Hopital's rule? See that

$$\lim_{\Delta \to 0} \frac{[e^{-r\Delta} - 1]}{\Delta}V_1 = -rV_1.$$

Example

- Notice also that $\lim_{\Delta \to 0} e^{-r\Delta} = 1$.
- Follows then when we take the limit that

$$0 = pu + p(V_0 - V_1) - rV_1$$
$$\Rightarrow rV_1 = p[u + V_0 - V_1].$$

- Left-side is referred to as the flow value to having the coin.
- Obviously, we'd need to find V_0 to fully solve this system.
- We'll talk about this in the next lecture.
- For now just understand that this is how we'll approach continuous time models: taking limits of discrete time equations.

Roadmap

Takeaways

- Recursive formulations (using value functions) are a convenient way of representing optimisation problems.
- Useful for computing models as well, (e.g. see numerical methods class).
- For our purposes, we'll use recursive methods as it's a neater way of characterising problems than sequence problems.