INTRO TO DYNAMIC PROGRAMMING: EXAMPLES

QUANTITATIVE ECONOMICS 2024

Piotr Żoch

December 17, 2024

INTRODUCTION

- Shortest Path
- Tree-cutting
- Resource Extraction
- Neoclassical Growth Model

SHORTEST PATH

SHORTEST PATH

- We have a set of nodes, S: $s_1, s_2 \ldots, s_N$.
- Let F_s be a set of nodes that can be reached from the node s in one step.
- $c(s, s')$ is the cost of moving from node s to node $s' \in F_s$.
- The goal is to find the shortest path from some starting node to some destination d.

SHORTEST PATH

- Let $J(s)$ be the length of the shortest path from the node s to the destination. Suppose we know $J(s)$ for all $s \in N$.
- Suppose the starting node is s. We can find the shortest path by choosing a node s' such that

$$s' = \underset{s' \in F_s}{\operatorname{argmin}}\, c(s, s') + J(s').$$

- It must also be the case that

$$J(s) = \min_{s' \in F_s} c(s, s') + J(s')$$

- The above is the Bellman equation for the shortest path problem.

SHORTEST PATH

- If we know $J(s)$ for all $s \in S$, we can find the shortest path easily. But we don't know it!
- How to find J? We only know that $J(d) = 0$.
- We will start with a guess $J_0(s)$ for all $s \in N$:
  - $J_0(s) = M$ for all $s \neq d$;
  - $J_0(s) = 0$, where M is a very large number.

SHORTEST PATH

- We will use the following algorithm:
    1. Set $k = 0$.
    2. Set $J_{k+1}(s) = \min_{s' \in F_s} c(s, s') + J_k(s')$ for all $s \in N$.
    3. If $J_{k+1} = J_k$, stop. Otherwise, set $k = k + 1$ and go to the previous step.

TREE-CUTTING

TREE-CUTTING

- There is a tree of size s.
- You can either cut down the tree now and sell the wood or wait until next period:
  - If you cut down the tree now, you get $f(s)$ dollars. You do not have the tree anymore.
  - If you wait until the next period, the tree grows to size $s+h$, unless $s=\bar{S}$, which is the maximum size of the tree. If $s=\bar{S}$ the tree stays the same size.
- Your objective function is to maximize the present discounted value of money you get from cutting down the tree.
- The discount factor is $\frac{1}{1+r}$, where $r>0$ is the interest rate.

TREE-CUTTING

- Let $v(s)$ be the value function. State: the size of the tree, s.
- Bellman equation:

$$v(s) = \max\left\{ \underbrace{f(s)}_{\text{cut down}}, \underbrace{\frac{1}{1+r} v\left(\min\{\bar{S}, s+h\}\right)}_{\text{wait}} \right\}.$$

- We can use value function iteration to solve this problem.

TREE-CUTTING

1. Suppose that cutting down the tree costs $c > 0$ dollars.
2. Suppose that if you wait until the next period, the tree might not grow at all - it happens with probability p.
3. Suppose that if you wait until the next period, the tree might die which means that you have to cut it down and get only 1/2 of the value of the tree. It happens with probability p.
4. Suppose that if you wait until the next period, the tree might become sick which means that it will stop growing. It happens with probability p. With probability q a sick tree can recover.

- How to modify the Bellman equation?

TREE-CUTTING

- Suppose that cutting down the tree costs $c > 0$ dollars.
- Bellman equation:

$$v(s) = \max \left\{ \underbrace{f(s) - c}_{\text{cut down}}, \underbrace{\frac{1}{1+r} v\left(\min\{\bar{S}, s + h\}\right)}_{\text{wait}} \right\}.$$

TREE-CUTTING

- Suppose that if you wait until the next period, the tree might not grow at all - it happens with probability p.
- Bellman equation:

$$v(s) = \max \left\{ \underbrace{f(s)}_{\text{cut down}}, \underbrace{\frac{1}{1+r}\left[(1-p)\cdot v(\min\{\bar{S}, s+h\}) + p \cdot v(s)\right]}_{\text{wait}} \right\}.$$

TREE-CUTTING

- Suppose that if you wait until the next period, the tree might die which means that you have to cut it down and get only 1/2 of the value of the tree. It happens with probability p.
- Bellman equation:

$$v(s) = \max \left\{ \underbrace{f(s)}_{\text{cut down}}, \underbrace{\frac{1}{1+r}\left[(1-p)\cdot v(\min\{\bar{S}, s+h\}) + p\cdot \frac{1}{2} f(s)\right]}_{\text{wait}} \right\}.$$

TREE-CUTTING

- Suppose that if you wait until the next period, the tree might become sick which means that it will stop growing. It happens with probability p. With probability q a sick tree can recover.
- Additional state variable - health of the tree.
- Bellman equation (H is for healthy, S is for sick):

$$v^H(s) = \max\left\{ \underbrace{f(s)}_{\text{cut down}}, \underbrace{\frac{1}{1+r}\left[(1-p)\cdot v^H(\min\{\bar{S}, s+h\}) + p\cdot\cdot v^S(s)\right]}_{\text{wait}} \right\}$$

$$v^S(s) = \max\left\{ \underbrace{f(s)}_{\text{cut down}}, \underbrace{\frac{1}{1+r}\left[(1-q)\cdot v^S(s) + q\cdot v^H(\min\{\bar{S}, s+h\})\right]}_{\text{wait}} \right\}$$

RESOURCE EXTRACTION

RESOURCE EXTRACTION

- You have a resource of size s that you can extract. s is an integer.
- The current price of the resource is $p \overset{iid}{\sim} \phi$ with N possible values: $p_1, p_2, \ldots, p_N$.
- The cost of extracting x units of the resource is $c(x)$. Extracting x units of the resource reduces the size of the resource to $s - x$ units next period. It is never possible to extract more than s units of the resource.
- The objective is to maximize the expected present discounted value of the profit from extracting the resource:

$$\mathbb{E}_0 \sum_{t=0}^{\infty} \beta^t \left(p_t x_t - c(x_t)\right)$$

RESOURCE EXTRACTION

- Bellman equation:

$$v(s,p) = \max_{x \in \{0,1,\ldots,s\}} \left\{ px - c(x) + \beta \sum_{p' \in \mathrm{P}} v\big(s - x, p'\big)\phi\big(p'\big) \right\}.$$

NEOCLASSICAL GROWTH MODEL

NEOCLASSICAL GROWTH MODEL

- A representative agent's problem

$$\sum_{t=0}^{\infty} \beta^t u(c_t), \quad \text{s.t. } k_{t+1} = f(k_t) + (1-\delta)\, k_t - c_t, \quad k_0 \text{ given.}$$

- $\beta \in (0,1)$ is the discount factor, $c_t \geq 0$ is the consumption, $k_t \geq 0$ is the stock of capital.
- $u(\cdot)$ is the utility function, $f(\cdot)$ is the production function, $0 < \delta \leq 1$ is the depreciation rate.
- We will assume $u(c) = \frac{c^{1-\gamma}}{1-\gamma}$, $f(k) = k^{\alpha}$, with $\gamma > 0$, $\alpha \in (0,1)$.
-

NEOCLASSICAL GROWTH MODEL

- Bellman equation:

$$v(k) = \max_{c \in [0, f(k) + (1-\delta)k]} \left\{ u(c) + \beta v(f(k) + (1-\delta)k - c) \right\}.$$

- Alternatively, we can write it as:

$$v(k) = \max_{k' \in [0, f(k) + (1-\delta)k]} \left\{ u\left(f(k) + (1-\delta)k - k'\right) + \beta v\left(k'\right) \right\}.$$

NEOCLASSICAL GROWTH MODEL

- Note: in the problem itself there is nothing that says there is only a finite number of k.
- We will discretize the state space and create a finite grid of k: $k_1, k_2, \ldots, k_N$.
- We need to decide on the grid size and the grid points. These are properties of the solution method, not the problem itself!
- What we will find is only an approximation to the true value and policy function.

NEOCLASSICAL GROWTH MODEL

- Let k^* be a level of capital that satisfies

$$1 = \beta \left[\alpha {k^*}^{\alpha-1} + 1 - \delta \right].$$

- In this model we call k^* the steady state level of capital. If $k_t = k^*$, then $k_{t+1} = k^*$.
- We also know that if we start from $k_0 > 0$, then $k_t \to k^*$ as $t \to \infty$.
- It makes sense to have a grid that includes k^*.