Dynamic Programming

VOLUME I: FOUNDATIONS

QUANTECON BOOK II

Thomas J. Sargent and John Stachurski

June 3, 2022

Contents

Preface

To be completed. Note that "a preface or foreword deals with the genesis, purpose, limitations, and scope of the book and may include acknowledgments of indebtedness."

We work within an abstract setting that builds on the framework in Bertsekas (2018). This setting includes standard dynamic programming problems as discussed in, say, Lucas and Stokey (1989), Rust (1996), or Puterman (2005), as well as the various recursive preference models, robust control methods and other more sophisticated preference features adopted within economics and finance in recent years.

All code presented in the textbook is written in Julia. We chose Julia because it is elegant, readable, open source, and powerful. Other great options exist. For example, at the time of writing, Python's has a large range of sophisticated and well-tested numerical libraries. A Python version of our source code is on the to-do list and all help is appreciated!

Common Symbols

$\mathbb{1}\{P\}$	indicator, equal to 1 if statement P is true and 0 otherwise
$\alpha := 1$	α is defined as equal to 1
$f \equiv 1$	function f is everywhere equal to 1
$\wp(A)$	the power set of A; that is, the collection of all subsets of given set A
$[n]$	$\{1, \ldots, n\}$
$\mathbb{N}$, $\mathbb{Z}$ and $\mathbb{R}$	the natural numbers, integers and real numbers respectively
$\mathscr{D}(\mathsf{X})$	the set of distributions on X
$\mathbb{Z}_+$, $\mathbb{R}_+$, etc.	the nonnegative elements of $\mathbb{Z}$, $\mathbb{R}$, etc.
$\|x\|$, $\|B\|$	the absolute value of $x \in \mathbb{R}$, the cardinality of set B
$\mathbb{R}^n$	all n-tuples of real numbers
$x \leqslant y \quad (x, y \in \mathbb{R}^n)$	$x_i \leqslant y_i$ for $i = 1, \ldots n$ (pointwise partial order)
$x \ll y \quad (x, y \in \mathbb{R}^n)$	$x_i < y_i$ for $i = 1, \ldots n$
$\mathscr{D}(F)$	the set of distributions (or probability mass functions) on F
e_n	the n-th canonical basis vector
$\mathbb{R}^M$	all functions from M to $\mathbb{R}$
$\langle a, b \rangle$	the inner product of a and b
$b\mathsf{X}$	the set of bounded, real-valued functions on X
$b\mathscr{E}$	the set of real-valued $\mathscr{E}$ measurable functions on $(\mathsf{E}, \mathscr{E})$
$bc\mathsf{X}$	the set of continuous functions in $b\mathsf{X}$
$ib\mathsf{X}$	the set of increasing functions in $b\mathsf{X}$
$\mathscr{B}$	the Borel measurable subsets of X
IID	independent and identically distributed
$X \stackrel{d}{=} Y$	X and Y have the same distribution
$X \sim F$	X has distribution F
$F \preceq_{\mathrm{F}} G$	F first order stochastically dominates G

Chapter 1

Introduction

Dynamic programming is a technique for solving optimization problems in dynamic settings. Typically, for these problems, the system evolves as follows:

---

an initial state X_0 is given
$t \leftarrow 0$
while $t < T$ **do**
  the controller of the system observes the current state X_t
  the controller responds by choosing an action A_t
  the controller receives a reward R_t based on the current state and action
  the state updates to X_{t+1}
  $t \leftarrow t + 1$
end

Figure 1.1 illustrates the first three rounds. If $T < \infty$ then the problem is called a **finite horizon** problem. Otherwise it is called an **infinite horizon** problem. The state update depends on the current state and action, in the sense that X_t and A_t typically affect X_{t+1}. The update rule can also depend on shocks and other random elements.

For decision makers facing systems such as the one described above, dynamic programming provides a way to maximize expected *lifetime* rewards, which aggregate the reward sequence (R_t) received at each time t.

Example 1.0.1. Consider a retailer who sets prices and manages inventories in order to maximize profits. In this setting, we might take X_t to be a vector that quantifies the current business environment, the size of the inventories, prices set by competitors and other factors relevant to management. The action A_t is a vector that specifies current prices and orders of new stock. If current reward R_t is current profit, then a typical choice of lifetime reward is the expected net present value of this profit flow.

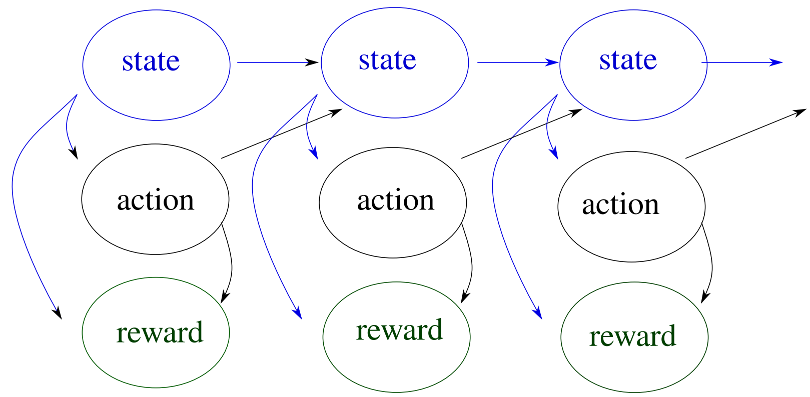


Figure 1.1: A dynamic program

Dynamic programming has a *vast* array of applications, ranging from robotics and artificial intelligence to the sequencing of DNA. Dynamic programming is used around the world every day to control aircraft, route shipping, test products, recommend information on media platforms and solve major research problems. Some companies now produce specialized computer chips that are designed for specific dynamic programming applications.

Within economics and finance, dynamic programming is applied to topics including unemployment, monetary policy, fiscal policy, asset pricing, firm investment, wealth dynamics, inventory control, commodity pricing, sovereign default, the division of labor, natural resource extraction, human capital accumulation, retirement decisions, portfolio choice, and dynamic pricing. We discuss many of these applications in the chapters below.

The theory of dynamic programming is elegant and seemingly simple. But for realistic problems, dynamic programming is often computationally demanding. Much of the modern theory of dynamic programming deals with managing this complexity.

Example 1.0.2. Continuing on with Example 1.0.1, suppose that the store in question is a book store, and, for each book, the retailer chooses to hold between 0 and 10 copies. If there are 100 books to choose from, then the number of possible combinations for her inventories is 11^{100}, which is around 20 orders of magnitude larger than the number of atoms in the known universe. In reality there are probably many more books to choose from, as well as other factors in the business environment that affect the choices of the retailer.

In this book we discuss fundamental theory, traditional economic applications and modern applications with large state spaces and computationally demanding environments. We also cover recent trends towards more sophisticated specifications of lifetime rewards, often called recursive preferences. Throughout the text, theory and computation are combined, since, for interesting problems, brute-force computation is futile, while theory alone provides limited insight. The interplay between interesting applications, fundamental theory, computational methods and evolving hardware capability makes dynamic programming a fascinating and exciting field.

1.1 Getting Started

Dynamic programs imply nonlinear equations that restrict optimal policies. This chapter reviews techniques for solving such equations, via a branch of mathematics called *fixed point theory*. Fixed point theory contains many beautiful results and has applications throughout economics and finance.

However, before we dive into fixed point theory, we introduce a finite-horizon dynamic program, where such techniques are not required. Our aim is to introduce the recursive structure of dynamic programming in a simple setting. After solving a finite-horizon model, we briefly introduce an infinite-horizon version and explain how the problem produces a system of nonlinear equations. Then we turn to fixed point theory.

1.1.1 Finite-Horizon Job Search

We begin with a celebrated model of job search created by McCall (1970). McCall modeled the decision problem of an unemployed worker in terms of current and likely future wage offers, impatience, and the availability of unemployment compensation. To solve the decision problem he used dynamic programming. While the McCall model has been extended in many directions, here we study a plain vanilla version in which essential ideas of dynamic programming are laid bare.

1.1.1.1 A Two Period Problem

Consider someones who begins her working life at time $t = 1$ without employment. While unemployed, she receives a new job offer paying wage w_t at each date t. She has two choices: accept the offer and work permanently at w_t or reject the offer, receive unemployment compensation c, and reconsider next period. We assume that the wage offer sequence $\{w_t\}$ is IID and nonnegative, with distribution φ. In particular,

- $\mathsf{W} \subset \mathbb{R}_+$ is a finite set of possible wage outcomes and
- $\varphi \colon \mathsf{W} \to [0,]$ is a probability distribution on W, assigning a likelihood $\varphi(w')$ to each wage outcome w'.

(We are assuming here that W is finite because it simplifies the mathematics and computer code. We drop the finite assumption later in the text and confront resulting complications.)

The person cares about the future but is impatient. Impatience is parameterized by a time discount factor $\beta \in (0, 1)$. This means that the present value to the agent of a next-period payoff of y dollars is βy. Since $\beta < 1$, indicating some impatience, the agent will be tempted to accept reasonable offers, rather than waiting for a better one. The key question is how long to wait.

Suppose as a first step that the working life of the agent is just two periods. To solve our problem we will work backwards, starting at the final date $t = 2$, when w_2 is observed. If she is already employed, the agent has no decision to make: she continues working at her current wage. If she is unemployed, then she should take the largest of c and w_2.

One of the essential techniques in dynamic programming is the use of "value functions," which keep track of maximal rewards from a given state at a given time. In order to familiarize ourseves with value functions, we define $v_2(w_2) = \max\{c, w_2\}$. The function v_2 is called the **time 2 value function** and is shown at the time 2 decision node in Figure 1.2. Here it represents the maximum value obtained in the final stage as a function of the time 2 wage offer.

Now we step back to $t = 1$, which is the first decision node in Figure 1.2. At this time, having received offer w_1, the unemployed worker's options are (a) accept this offer w_1 and receive it in both periods or (b) reject it, receive unemployment compensation c, and then, in the second period, choose the maximum of w_2 and c. Which one is better.

Let's assume that the agent seeks to maximize expected present value. The expected present value of option (a) is $w_1 + \beta w_1$, which is sometimes called the **stopping value**. The present expected value of option (b) is

$$h_1 := c + \beta \sum_{w' \in \mathsf{W}} v_2(w') \varphi(w'), \tag{1.1}$$

which is called the **continuation value**. The sum in (1.1) computes the expectation of $\max\{c, w_2\}$. We are working with expected values because, at $t = 1$, the wage offer w_2 is, as yet, unknown.

The optimal choice at $t = 1$ is now clear:

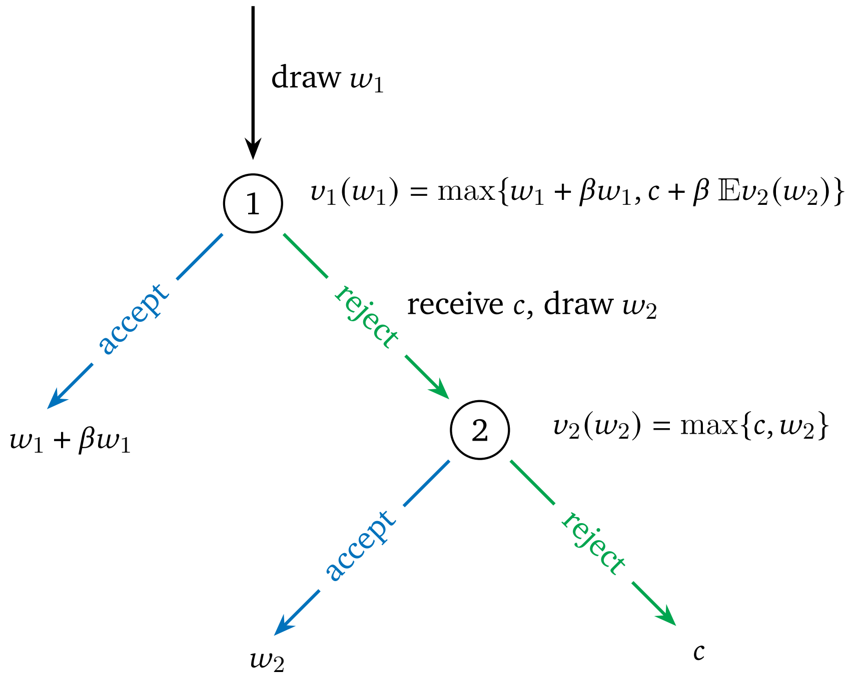


Figure 1.2: Decision tree for the two period problem

(i) If $w_1 + \beta w_1 \geqslant h_1$, then accept the job offer.

(ii) If not, then reject and wait for the next offer.

The **time 1 value function** v_1 is obtained by maximizing over the two options:

$$v_1(w_1) := \max \left\{ w_1 + \beta w_1,\ c + \beta \sum_{w' \in \mathsf{W}} v_2(w')\varphi(w') \right\}. \tag{1.2}$$

It represents the present value of expected lifetime income accruing to the agent, once the first offer w_1 has been received, *if* she chooses optimally in both periods.

The value function is shown in Figure 1.3 as the pointwise maximum of the stopping value, as a function of w_1, and the continuation value. Figure 1.3 also shows

$$w_1^* := \frac{h_1}{1+\beta}, \tag{1.3}$$

the **reservation wage**, which is the w that solves

$$w + \beta w = c + \beta \sum_{w' \in \mathsf{W}} v_2(w')\varphi(w'),$$

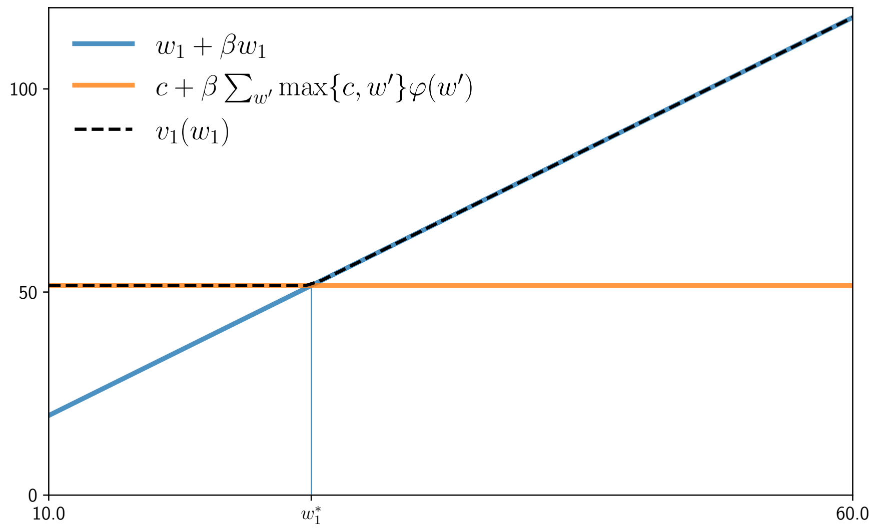


Figure 1.3: The value function v_1 and the reservation wage

equalizing the value of stopping and the value of continuing. For an offer w_1 above the reservation wage, the stopping value exceeds the continuation value. For an offer below the reservation wage, the reverse is true. Hence, the optimal choice for the agent at $t = 1$ is determined entirely by the reservation wage.

The parameters and functions used to create the figure are shown in Listing 1.

Studying (1.3) is already instructive: we can see that higher unemployment compensation shifts up the continuation value and increases the reservation wage, so the agent will, on average, spend more time unemployed when unemployment compensation is higher.

1.1.1.2 Three Periods

Now let's suppose that the agent works in period $t = 0$ as well as $t = 1, 2$. Figure 1.4 shows the decision tree for the three periods. Below we analyze the decision sequence and pin down the optimal actions as a function of the primitives.

At $t = 0$, the value of accepting the current offer w_0 is $w_0 + \beta w_0 + \beta^2 w_0$, while maximal value of rejecting and waiting, is c plus, after discounting by β, the maximim value that can be obtained by behaving optimally from $t = 1$. Fortunately, this value has already been calculated, for every possible value of w_1: it is just $v_1(w_1)$, as given in (1.2)!

```julia
using Distributions

"Creates an instance of the job search model, stored as a NamedTuple."
function create_job_search_model(;
        n=50,         # wage grid size
        w_min=10.0,   # lowest wage
        w_max=60.0,   # highest wage
        a=200,        # wage distribution parameter
        b=100,        # wage distribution parameter
        β=0.96,       # discount factor
        c=10.0        # unemployment compensation
    )
    w_vals = collect(LinRange(w_min, w_max, n+1))
    ϕ = pdf(BetaBinomial(n, a, b))
    return (; n, w_vals, ϕ, β, c)
end

" Computes lifetime value at t=1 given current wage w_1 = w. "
function v_1(w, model)
    (; n, w_vals, ϕ, β, c) = model
    return max(w + β * w, c + β * max.(c, w_vals)'ϕ)
end

" Computes reservation wage at t=1. "
function res_wage(model)
    (; n, w_vals, ϕ, β, c) = model
    return (c + β * max.(c, w_vals)'ϕ) / (1 + β)
end
```

Listing 1: Computing v_1 and w_1^* (two_period_job_search.jl)

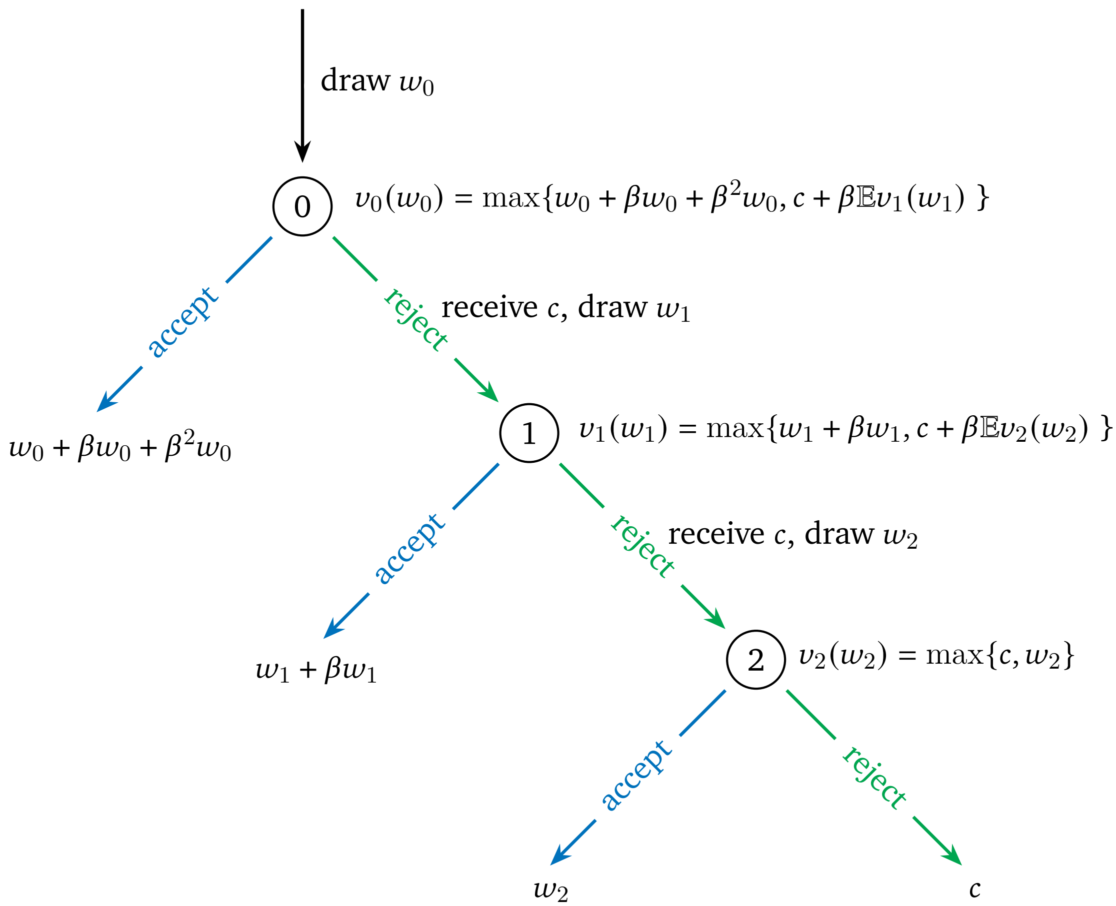


Figure 1.4: Decision tree for the job seeker

Since total value $v_0(w_0)$ is the maximum of the value of these two options, we can now write

$$v_0(w_0) = \max\left\{ w_0 + \beta\, w_0 + \beta^2\, w_0,\ c + \beta \sum_{w' \in Wsf} v_1(w')\varphi(w') \right\}. \tag{1.4}$$

By plugging v_1 from (1.2) into this expression, we can determine v_0, as well as the optimal action, which is the one that achieves the largest value in the max term in (1.4).

The solution method described above uses **backwards induction**: solving the last period problem first, using it to solve the previous period problem, and so on. This process is clear from inspecting Figure 1.4. The last period value function v_2 is trivial to obtain. With v_2 in hand we can compute v_1. With v_1 in hand we can compute v_0. Once all the value functions are available, we can calculate whether to accept or reject at each point in time.

Exercise 1.1.1. The optimal action at time $t = 0$ is determined by a time zero reservation wage w_0^*, where the agent should accept the time zero wage offer if and only if w_0 exceeds w_0^*. Calculate w_0^* for this problem, by analogy with w_1^* in (1.3).

Notice how we broke the three period problem down into a pair of two period problems, given by the two equations (1.2) and (1.4). Breaking many-period problems down into a sequence of two period problems is the essence of dynamic programming. The recursive relationships between v_0 and v_1 in (1.4), as well as between v_1 and v_2 in (1.2), are examples of what are called **Bellman equations**. We will see many other examples shortly.

Exercise 1.1.2. Extend the above arguments to T time periods, where T can be any finite number. Using Julia or any other suitable programming language, write a function that takes T as an argument and returns $(w_0^*, \ldots, w_T^*)$, the sequence of reservation wages for each period.

1.1.2 Infinite Horizons: A First Look

Next we consider an infinite horizon, which is in some ways more challenging and someways simpler and cleaner. On one hand, the lack of a terminal period means that we cannot do backwards induction and, as a result, we have to use fixed point theory—details are explained below. On the other hand, the infinite horizon means that the agent always faces an infinite future, so the current decision is not time dependent—and hence more straightforward. This will become clearer as the section unfolds.[1]

With the above discussion in mind, let us consider a worker who aims to maximize

$$\mathbb{E} \sum_{t=0}^{\infty} \beta^t Y_t, \tag{1.5}$$

where $Y_t \in \{c, W_t\}$ is earnings at time t. As before, jobs are permanent, so accepting a job at a given wage means earing that wage in every subsequent period.

Let's clarify our assumptions:

Assumption 1.1.1. The wage process satisfies $\{W_t\} \overset{\text{IID}}{\sim} \varphi$ where $\varphi \in \mathscr{D}(\mathsf{W})$ and $\mathsf{W} \subset \mathbb{R}_+$ with $|\mathsf{W}| < \infty$. The parameters c and β are positive and $\beta \in (0, 1)$.

Note 1.1.1. Regarding notation,

- We are now using capitals for random variables.
- Here and below, for any finite or countable set F, the symbol $\mathscr{D}(F)$ indicates the set of distributions (or probability mass functions) on F.

1.1.2.1 Intuition

As with the finite state case, applying dynamic programming involves a two step procedure that first assigns values to states and then deduces optimal actions given those values. We begin with an intuitive discussion and then formalize the main ideas.

To trade off current and future rewards optimally, we need to compare current payoffs we get from our two choices with the states that those choices lead to and the maximum value that can be extracted from those states. But how do we calculate the maximum value that can be extracted from each state when lifetime is infinite?

Consider first the present expected lifetime value of being employed with wage $w \in \mathsf{W}$. This case is easy because, under the current assumptions, workers who accept a job are employed forever and has no remaining choices to exercise. Lifetime payoff is

$$w + \beta w + \beta^2 w + \cdots = \frac{w}{1 - \beta}. \tag{1.6}$$

[1] Incidentally, imposing an infinite horizon is not the same as assuming humans live forever. Rather, it corresponds to the idea that humans have no specific "termination" date. More generally, we can understand an infinite horizon as a reasonable approximation to a finite horizon when observations are recorded at relatively high frequency and no clear termination date exists.

How about maximum present expected lifetime value attainable when entering the current period unemployed with wage offer w in hand? Denote this (as yet unknown) value by $v^*(w)$. We call v^* the **value function**. While v^* is not trivial to pin down, the task is not impossible. Our first step in the right direction is to observe that it satisfies the **Bellman equation**

$$v^*(w) = \max \left\{ \frac{w}{1-\beta}, \, c + \beta \sum_{w' \in \mathsf{W}} v^*(w') \varphi(w') \right\} \tag{1.7}$$

at every $w \in \mathsf{W}$. (Here w' is the offer next period.)

Our reasoning is as follows: The first term inside the max operation is the **stopping value**, or lifetime payoff from accepting current offer w. The second term inside the max operation is the **continuation value**, or current expected value of rejecting and behaving optimally thereafter. Maximal value is obtained by selecting is the largest of these two alternatives.

At this point, you should note the similarity between (1.30) and our finite horizon Bellman equations (1.2) and (1.4). The only real difference is that the value function is no longer time-dependent. To repeat, this is because the worker always looks forward toward an infinite horizon, regardless of the current date.

Mathematically, (1.30) is viewed as an equation to be solved for a function $v^* \in \mathbb{R}^{\mathsf{W}}$, assuming this is possible. Once we have solved for v^*, optimal choices can be made by observing current w and then choosing the largest of the two alternatives on the right hand side of (1.30).

How, then, should we solve for v^*? For this problem we use fixed point theory. To this end, let's now spend some time on fixed point theory. In §1.4, we return to the job search problem and apply this theory to solving for v^*.

1.2 Fixed Points

This section contains an introduction to fixed point theory, focusing on the finite-dimensional setting. (Later we study fixed points in more general settings.) We analyze both linear and nonlinear equations.

1.2.1 Linear Equations

Fixed point theory is used to solve equations, so let's begin by discussing equations and then circle back to fixed points. About the easiest equation to understand is the one-dimensional linear equation $x = ax + b$. If $|a| < 1$, then we can solve this equation for x, obtaining

$$x^* = \frac{b}{1-a} = \sum_{k \geqslant 0} a^k b.$$

This scalar result extends naturally to vectors. In particular, if x and b are column vector in $\mathbb{R}^n$, A is an $n \times n$ matrix,

$$r(A) := \max\{|\lambda| : \lambda \text{ is an eigenvalue of } A\} \tag{1.8}$$

and I is the $n \times n$ identity matrix, then we can state the following:

Theorem 1.2.1 (Neumann Series Lemma)**.** *If $r(A) < 1$, then $I - A$ is nonsingular, the sum $\sum_{k \geqslant 0} A^k$ converges, and, moreover these two matrices are equal. In particular, the vector equation $x = Ax + b$ has the unique solution*

$$x^* = (I - A)^{-1} b = \sum_{k \geqslant 0} A^k b.$$

The value $r(A)$ in (1.8) is called the **spectral radius** of A. The expression $|\lambda|$ indicates the modulus of the complex number λ. The code in Listing 2 shows how to compute the spectral radius of an arbitrary matrix A in Julia. The print statement produces `0.5828`, so, for this matrix, $r(A) < 1$.

```
using LinearAlgebra
r(A) = maximum(abs(λ) for λ in eigvals(A))  # Spectral radius
A = [0.4 0.1;                               # Test with arbitrary A
     0.7 0.2]
print(r(A))
```

Listing 2: Computing the spectral radius (`compute_spec_rad.jl`)

Exercise 1.2.1. Prove that $r(\alpha B) = |\alpha| r(B)$ for all $\alpha \in \mathbb{R}$.

An intuitive proof of the Neumann series lemma runs as follows. If $S := \sum_{k \geqslant 0} A^k$, then

$$I + AS = I + A \sum_{k \geqslant 0} A^k = I + A + A^2 + \cdots = S.$$

Rearranging $I + AS = S$ gives $S = (I - A)^{-1}$. Since $x = Ax + b$ is equivalent to $(I - A)x = b$, we have $x = (I - A)^{-1}b = Sb$, which matches the claim in the Neumann series lemma.

This argument lacks rigor, however. To complete it, we need to prove that (a) the sum $\sum_{k \geqslant 0} A^k$ converges and (b) the matrix $I - A$ is invertible.

To resolve these issues, we introduce the **matrix norm**

$$\|B\|_\infty := \max_{i,j} |b_{ij}|.$$

Lemma 1.2.2. *If B is any square matrix, then*

$$r(B)^k \leqslant \|B^k\|_\infty \text{ for all } k \in \mathbb{N} \qquad \text{and} \qquad \|B^k\|_\infty^{1/k} \to r(B) \text{ as } k \to \infty.$$

The second result in Lemma 1.2.2 is a version of **Gelfand's formula**.

Exercise 1.2.2. Using Lemma 1.2.2, show that

(i) $r(B) < 1$ implies $\|B^k\|_\infty \to 0$ as $k \to \infty$.

(ii) $r(B) > 1$ implies $\|B^k\|_\infty \to \infty$ as $k \to \infty$.

Exercise 1.2.3. Prove: $r(A) < 1$ implies that the series $\sum_{k \geqslant 0} A^k$ converges, in the sense that every element of the matrix $S_K := \sum_{k=0}^{K} A^k$ converges as $K \to \infty$.

From this last result, one can show that $(I - A)^{-1}$ exists:

Exercise 1.2.4. Prove this claim by showing that, when $\sum_{k \geqslant 0} A^k$ exists, the inverse of $I - A$ exists and indeed $(I - A)^{-1} = \sum_{k \geqslant 0} A^k$.[2]

Listing 3 helps illustrate the result in Exercise 1.2.4, although we truncate the infinite sum $\sum_{k \geqslant 0} A^k$ at 50.

The output is `5.621e-12`, which is essentially zero.

Remark 1.2.1. Some authors automatically identify vectors with column vectors, which can be transposed to obtain row vectors. In contrast, we follow the mathematical convention that a vector in $\mathbb{R}^n$ is just an n-tuple of real values. This coincides with the viewpoint of Julia: vectors are, by default, “flat” arrays. At the same time, if we use vectors in matrix algebra, they can be understood as column vectors unless we state otherwise.

[2] Hint: To prove that A is invertible and $B = A^{-1}$, it suffices to show that $AB = I$. See, for example, Sargent and Stachurski (2022).

```
A = [0.4 0.1;
     0.7 0.2]
b = [1.0; 2.0]

# Method one: direct inverse
B_inverse = inv(I - A)

# Method two: power series
B_sum = zeros((2, 2))
A_power = I
for k in 1:50
    B_sum += A_power
    A_power = A_power * A
end

# Print maximal error
print(maximum(B_inverse - B_sum))
```

Listing 3: Matrix inversion vs power series (`power_series.jl`)

1.2.2 More General Equations

All the equations discussed above have been linear (actually, *affine*, but most authors call them linear). For nonlinear equations the situation is more complex. We will have to think harder about how to solve our equations—or if solutions even exist.

One systematic way to look at the problem of solving equations is through the lense of fixed point theory. To recall the basic definitions, we will say that T is a **self-map** on an arbitrary set S if T is a function from S into itself. For a self-map T on S, a point $x^* \in S$ is called a **fixed point** of T in S if $Tx^* = x^*$.

Remark 1.2.2. In fixed point theory, it is common to write Tx for the image of x under the function T, rather than $T(x)$. In addition, T is often called an **operator** rather than a function. One reason is that, in the applications that follow, x can itself be a function. In such settings, confusion can be avoided by calling T an operator.

Example 1.2.1. Let $S = \mathbb{R}^n$ and let T be defined by $Tx = Ax + b$, where A and b are as in §1.2.1. Since x is a fixed point of T if and only if $x = Ax + b$, solving the equation $x = Ax + b$ is the same as searching for the fixed point of T.

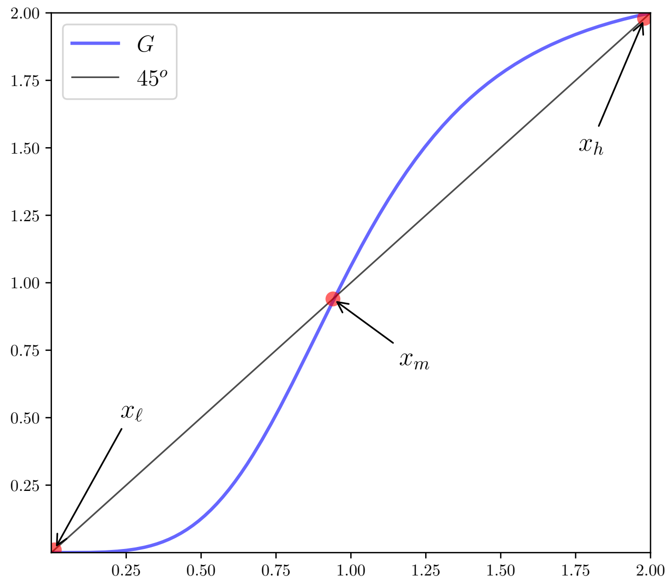


Figure 1.5: Graph and fixed points of $G\colon x \mapsto 2.125/(1 + x^{-4})$

Example 1.2.2. Every x in set S is fixed under the identity map $I\colon x \mapsto x$.

Example 1.2.3. If $S = \mathbb{N}$ and $Gx = x + 1$, then G has no fixed point.

Figure 1.5 shows another example, for a self-map G on $S = [0, 2]$. Fixed points are numbers $x \in [0, 2]$ where G meets the 45 degree line. In this case there are three.

Exercise 1.2.5. Let S be any set and let T be a self-map on S. Suppose there exists an $\bar{x} \in S$ and an $m \in \mathbb{N}$ such that $T^k x = \bar{x}$ for all $x \in S$ and $k \geqslant m$. Prove that, under this condition, $\bar{x}$ is the unique fixed point of T in S.

Exercise 1.2.6. Let T be a self-map on $S \subset \mathbb{R}^d$. Prove the following: If $T^m u \to u^*$ as $m \to \infty$ for some pair $u, u^* \in S$ and, in addition, T is continuous at u^*, then u^* is a fixed point of T.

It turns out that the most natural way to write down general theorems about solving scalar equations, vector equations and more abstract equations is in terms of fixed points. Indeed, an abstract representation of a system of equations is $x = Tx$, where x takes values in an abstract set S and T is a self-map on S. By definition, solutions to this system coincide with fixed points of the mapping T.

When considering fixed points, given a self-map T on S, we typically seek conditions on T and S under which the following properties hold:

- T has at least one fixed point on S (existence)
- T has at most one fixed point on S (uniqueness)
- T has a fixed point on S and the fixed point can be computed using some suitable numerical scheme.

Example 1.2.4. If $S = \mathbb{R}^n$ and T is defined by $Tx = Ax + b$, then, by the Neumann series lemma, T has a unique fixed point $x^* \in \mathbb{R}^n$ whenever $r(A) < 1$. Moreover, that fixed point can be computed, at least approximately, by using either $x^* = (I - A)^{-1}b$ or $x^* = \sum_{k \geqslant 0} A^k b$.

1.2.3 Algorithms

As indicated above, we are interested not only in existence and uniqueness of fixed points, but also in how to compute them. In studying these issues, we consider a self-map T on a set S, where S is a nonempty subset of $\mathbb{R}^n$. We seek algorithms that compute fixed points of T, assuming they exist.

1.2.3.1 Successive Approximation

Self-map T is called **globally stable** on S if T has a unique fixed point x^* in S and, moreover, $T^k x \to x^*$ as $k \to \infty$ for all $x \in S$. Here T^k indicates k compositions of T with itself. Global stability is a very desirable property in the setting of dynamic programming and a number of our results rely on it.

Exercise 1.2.7. As in Example 1.2.4, let $S = \mathbb{R}^n$ and let T be defined by $Tx = Ax+b$. Using induction, prove that

$$T^k x = A^k x + A^{k-1} b + A^{k-2} b + \cdots + Ab + b$$

for all $x \in S$ and $k \in \mathbb{N}$. Next, show that T is globally stable on S whenever $r(A) < 1$.

If T is globally stable on S, then a natural algorithm for approximating the unique fixed point x^* of T in S is to pick any $x \in S$ and iterate with T for some finite number of steps:

---

fix x_0 and $k = 0$
while *some stopping condition fails* **do**
  $x_{k+1} \leftarrow Tx_k$
  $k \leftarrow k + 1$
end

By the definition of global stability, this sequence converges to x^*. The algorithm just described is called **successive approximation**. As a stopping condition for the successive approximation algorithm, it is common to iterate until the distance between successive iterates falls below some tolerance. Listing 4 provides a function that implements this procedure. Listing 5 applies it to the map $Tx = Ax + b$.

As a stopping condition for the successive approximation algorithm, it is common to iterate until the distance between successive iterates falls below some tolerance. Listing 5 provides a function that implements this procedure and applies it to the map $Tx = Ax + b$.

Figure 1.6 shows the sequence of iterates generated by four runs of the successive approximation algorithm, each with a different starting condition x_0. The map and parameters are the same as in Listing 5. It is clear from the figure that a good choice of initial condition (i.e., close to the fixed point) accelerates convergence.

Let T be a self-map on $S \subset \mathbb{R}^n$. We call T **invariant** on $C \subset S$ and call C an **invariant set** if T is also a self-map on C; that is, if $u \in C$ implies $Tu \in C$.

EXERCISE 1.2.8. Let T be a globally stable self-map on $S \subset \mathbb{R}^n$, with fixed point u^*. Prove the following: If C is closed and T is invariant on C, then $u^* \in C$.

1.2.3.2 Nonlinear Maps

Of course for the linear map $Tx = Ax+b$, there is no need to use successive approximation to compute the fixed point (unless, say, the matrix $I-A$ is too large to be inverted). However, for nonlinear and globally stable maps, successive approximation is a reliable and routinely used method for computing fixed points. This is certainly true in the case of dynamic programming, as we soon discuss.

To illustrate successive approximations in a nonlinear setting, we now present an extended example related to the the Solow–Swan growth model, which is a typical starting point for analysis of economic growth in undergraduate studies. For the version we present, the fixed point can be computed with pencil and paper, so successive

```
"""
Computes the an approximate fixed point of T via successive approximation.

"""
function successive_approx(T,                          # Operator (callable)
                           x_0;                        # Initial condition
                           tolerance=1e-6,             # Error tolerance
                           max_iter=10_000,            # Max iteration bound
                           print_step=25)              # Print at multiples
    x = x_0
    error = Inf
    k = 1
    while (error > tolerance) & (k <= max_iter)
        x_new = T(x)
        error = maximum(abs.(x_new - x))
        if error % print_step == 0
            println("Completed iteration $k with error $error.")
        end
        x = x_new
        k += 1
    end
    if k < max_iter
        println("Terminated successfully in $k iterations.")
    else
        println("Warning: Iteration hit max_iter bound $max_iter.")
    end
    return x
end
```

Listing 4: Successive approximation (`s_approx.jl`)

```
include("s_approx.jl")
using LinearAlgebra

# Compute the fixed point of Tx = Ax + b via linear algebra
A, b = [0.4 0.1; 0.7 0.2], [1.0; 2.0]
x_star = (I - A) \ b  # compute (I - A)^{-1} * b

# Compute the fixed point via successive approximation
T(x) = A * x + b
x_0 = [1.0; 1.0]
x_star_approx = successive_approx(T, x_0)

# Test for approximate equality (prints "true")
print(isapprox(x_star, x_star_approx, rtol=1e-5))
```

Listing 5: Using successive approximations to compute x^* (`linear_iter.jl`)

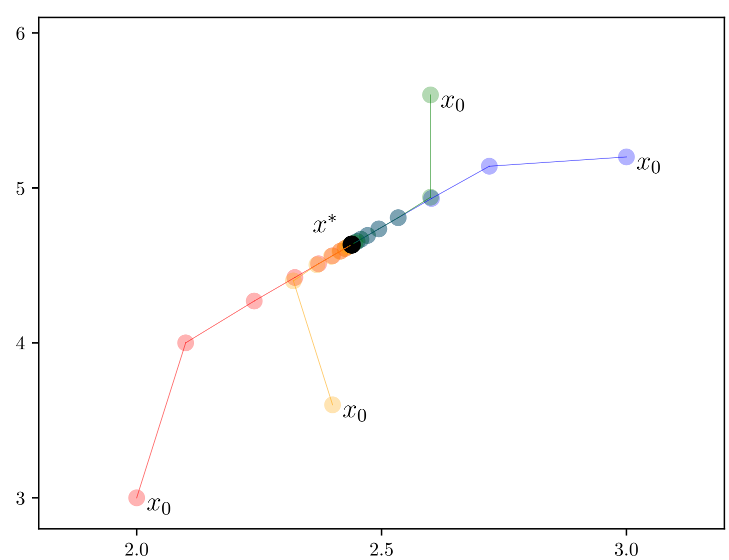


Figure 1.6: Successive approximation from different initial conditions

approximation can be avoided. However, building understanding and intuition in this simple setting is valuable, as we will soon meet systems where numerical methods like successive approximation are essential.

A simple version of the Solow–Swan growth dynamics is

$$k_{t+1} = sf(k_t) + (1 - \delta)k_t, \qquad t = 0, 1, \ldots, \tag{1.9}$$

where k_t is capital stock per worker, $f\colon (0, \infty) \to (0, \infty)$ is a production function, $s > 0$ is a savings rate and $\delta \in (0, 1)$ is a rate of depreciation. If we set $g(k) := sf(k)+(1-\delta)k$, then iterating with g from some starting point k_0 (i.e., setting $k_{t+1} = g(k_t)$ for all $t \geqslant 0$) generates the sequence in (1.9). At the same time, we can understand this process as using successive approximation to compute the fixed point of g.

Exercise 1.2.9. Show that the unique fixed point of g in $S = (0, \infty)$ is

$$k^* := \left(\frac{sA}{\delta}\right)^{1/(1-\alpha)}$$

Prove that, for $k \in S$,

(i) $k \leqslant k^*$ implies $k \leqslant g(k) \leqslant k^*$ and

(ii) $k^* \leqslant k$ implies $k^* \leqslant g(k) \leqslant k$.

Conclude that g is globally stable on S. (Why?)

Figure 1.7 illustrates the dynamics in a 45 degree diagram when $f(k) = Ak^\alpha$. In the top subfigure, $A = 2.0$, $\alpha = 0.3$, $s = 0.3$ and $\delta = 0.4$. The function g is plotted alongside the 45 degree line. Readers will recall that, when $g(k_t)$ lies strictly above the 45 degree line, then $k_{t+1} = g(k_t) > k_t$ and so capital per worker rises. If $g(k_t) < k_t$ then it falls. One trajectory $\{k_t\}_{t \geqslant 0}$, produced by starting from a particular choice of k_0, is traced out in the figure.

The bottom subfigure is similar, with parameters adjusted to $A = 3.0$, $\alpha = 0.05$, $s = 0.4$ and $\delta = 0.6$.

The figure helps illustrate the fact that k^* is the unique fixed point of g in S and all sequences converge to it. The second statement can be rephrased as: successive approximation successfully computes the fixed point of g by stepping through the time path of capital.

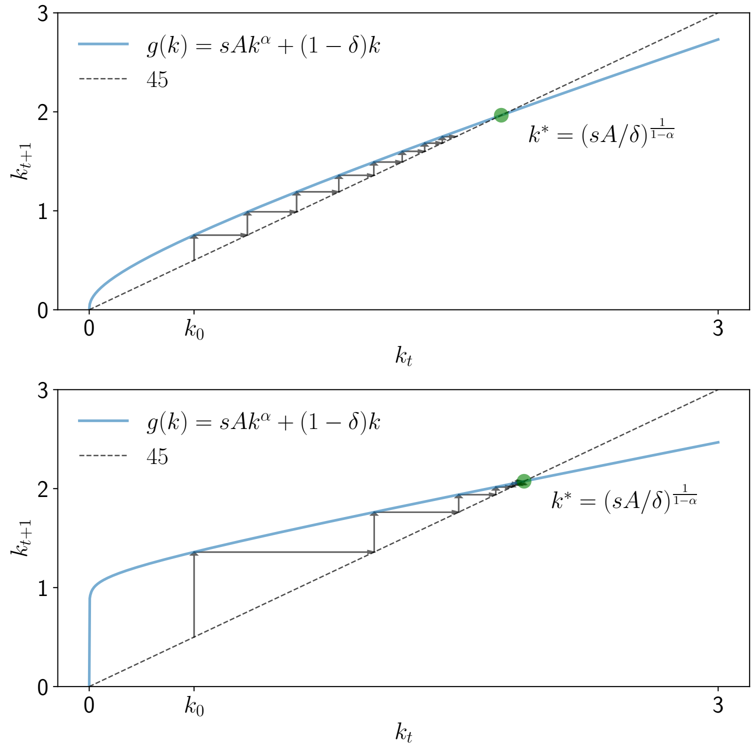


Figure 1.7: Successive approximation for the Solow–Swan model

1.2.3.3 Speed of Convergence

Notice that the speed of convergence is faster in the bottom subfigure of Figure 1.7. The change in parameter values implies that successive approximation achieves the same level of accuracy in few steps. Intuitively, in the top subfigure, g is close to the 45 degree line and hence convergence is slower. Conversely, faster convergence occurs in the second parameterization because the function g is "flatter" in the neighborhood of the fixed point.

The idea of the function g being relatively "flat" is meaningful in one dimension but not in $\mathbb{R}^n$. Another way to think about g being flat that does generalize to higher dimensions is to say that g is more "contractive" near the fixed point in the second parameterization. By this we mean that, for any k, k' near k^*, the distance $|g(k)-g(k')|$ is much less than the distance $|k-k'|$. In section 1.2.4 below we discuss contraction maps in more detail, and connect the degree of contractivity with the rate of convergence in successive approximation.

1.2.3.4 Newton's Method

Successive approximation is not always the best algorithm to compute fixed points, even when global stability holds. In many cases, faster algorithms are available, with speed gains achieved using extra information such as function gradients. One particularly useful gradient-based technique is **Newton's method**.

To illustrate Newton's method in the univariate case, suppose first that h is a differentiable real-valued function on $(a, b) \subset \mathbb{R}$, and that our aim is to find a **root** of h, which is an x^* such that $h(x^*) = 0$. Our plan is to start with a guess x_0 of x^* and then update it. To do this we use the approximation $h(x_1) \approx h(x_0) + h'(x_0)(x_1 - x_0)$. Setting the right-hand side equal to zero (seeking an approximate root) and solving for x_1 gives $x_1 = x_0 - h(x_0)/h'(x_0)$. This intuition leads us to consider the sequences of guesses

$$x_{k+1} = q(x_k) \quad \text{where} \quad q(x) := x - \frac{h(x)}{h'(x)}, \qquad k = 0, 1, \ldots \tag{1.10}$$

This sequence corresponds to **Newtons' method**. Notice that we do not need to write a new solver, since the successive approximation function in Listing 4 can be applied to q.

Newton's method can easily be adapted to solve for fixed points instead of roots. For example, in the Solow–Swan case, where we seek the fixed point of g, we can instead search for the root of h defined by $h(k) = g(k) - k$.

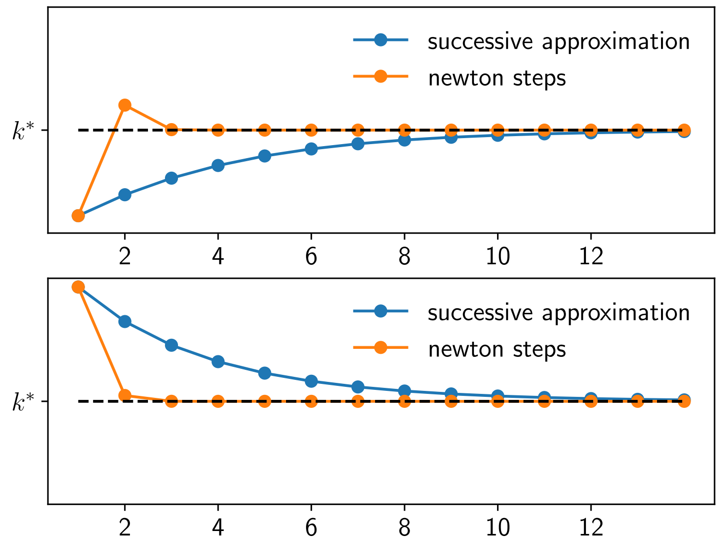


Figure 1.8: Newton's method applied to the Solow–Swan update rule

Figure 1.8 shows both the Newton approximation sequence and the successive approximation sequence for two different initial conditions (top and bottom subfigures). Notice how the Newton sequence approaches the fixed point much faster.

1.2.3.5 Speed vs Robustness

Within numerical methods, there is typically a trade-off between speed and robustness. One way to think about this is that fast methods need more structure and tend to make more assumptions than slower methods. These additional requirements are more easily violated, which negatively impacts the robustness of fast methods.

Relative to other algorithms, successive approximation tends to be robust but slow. We saw one illustration of the relatively slow rate of convergence in Figure 1.8. But we can also see its relatively strong robustness properties via the same example, by inspecting Figure 1.9, which compares the update rule of successive approximation (the function g) with the update rule for Newton's method (the function q in (1.10)). Also plotted is the dashed 45 degree line.

The parameterization for the model is the same as the top subfigure in Figure 1.7. As previously discussed, the shape of g implies global convergence of successive approximation. However, the same is not true of q. What we can see is that q is very

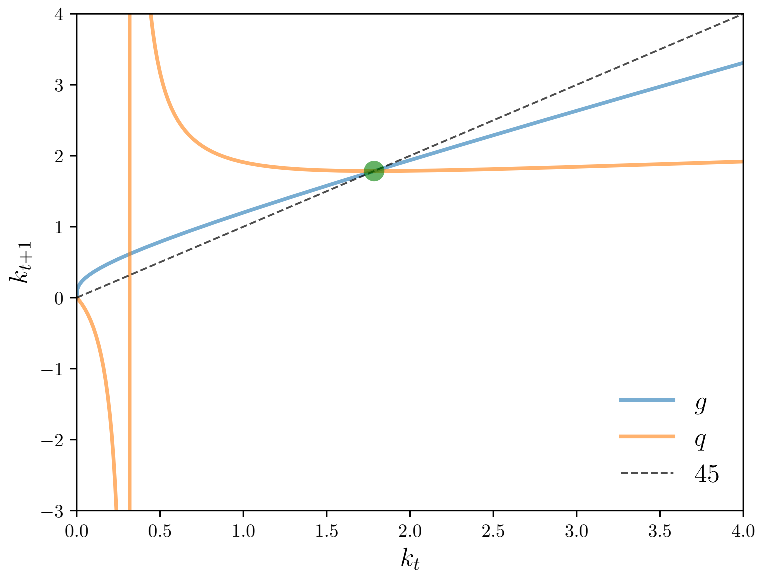


Figure 1.9: Robustness of successive approximation vis-a-via Newton's method

well behaved near the fixed point (i.e., very flat and hence very contractive), but also very badly behaved away from the fixed point. Hence Newton's method is slow but less robust.

For these reasons, successive approximation is often used as a starting point, to reliably find a reasonable approximation to the fixed point. From there, we can apply a faster technique, such as Newton's method.

1.2.3.6 Higher Dimensions

Newton's method extends naturally to multiple dimensions. When h is a map from $S \subset \mathbb{R}^n$ to itself, the term $1/h'(x)$ is replaced by $J_h(x)^{-1}$, where $J_h(x)$ is the Jacobian matrix of h at x. While inverting the Jacobian in high dimensions can be computationally expensive, many of the operations can be successfully parallelized in multithreaded computing environments. This multithreading is often carried out automatically by modern software libraries.

If we compare Newton's method to successive approximation in high dimensions, Newton's method typically involves fewer steps, but each one is more computationally expensive (due to the need to compute and invert the Jacobian). This is often beneficial, since the parallelization discussed above can be used at each individual step, to accelerate execution of that step. However, for very large systems, even storing the Jacobian in memory becomes problematic, and some form of successive approximation might be the only option.

We discuss high-dimensional implementations of numerical methods in the context of dynamic programming later in the text.

1.2.4 Banach's Theorem in Finite Dimensions

Before finishing our discussion of fixed points and numerical methods, we present one fixed point theorem for nonlinear operators that among the most important and widely used results in applied analysis: the Banach fixed point theorem.

1.2.4.1 Norms on Finite Vector Space

Prior to introducing Banach's theorem, we briefly cover alternative norms on $\mathbb{R}^n$. These alternatives are important for applications of Banach's theorem because they provide more flexibility when checking its conditions.

A function $\| \cdot \| \colon \mathbb{R}^n \to \mathbb{R}$ is called a **norm** on $\mathbb{R}^n$ if, for any $\alpha \in \mathbb{R}$ and $u, v \in \mathbb{R}^n$,

(a) $\|u\| \geqslant 0$ (nonnegativity)

(b) $\|u\| = 0 \iff u = 0$ (positive definiteness)

(c) $\|\alpha u\| = |\alpha| \|u\|$ and (positive homogeneity)

(d) $\|u + v\| \leqslant \|u\| + \|v\|$ (triangle inequality)

The **Euclidean norm**

$$\|u\| := \sqrt{\langle u, u \rangle} \qquad (u \in \mathbb{R}^n)$$

is a norm on $\mathbb{R}^n$, as suggested by its name. (Here $\langle u, v \rangle$ stands for the **inner product** of vectors u and v, which is the sum $\sum_{i=1}^n u_i v_i$.) The Euclidean norm satisfies the **Cauchy–Schwarz inequality**

$$| \langle u, v \rangle | \leqslant \|u\| \cdot \|v\| \quad \text{for all } u, v \in \mathbb{R}^n.$$

This inequality can be used to prove that the triangle inequality is valid for the Euclidean norm (see, e.g., Kreyszig (1978)).

Example 1.2.5. The **ℓ_1 norm** of a vector $u \in \mathbb{R}^n$ is defined by

$$u = (u_1, \ldots, u_n) \mapsto \|u\|_1 := \sum_{i=1}^{n} |u_i|. \tag{1.11}$$

In machine learning applications, $\|\cdot\|_1$ is sometimes called the "Manhattan norm," and $d_1(u, v) := \|u - v\|_1$ is called the "Manhattan distance" or "taxicab distance" between vectors u and v. We will refer to it more simply as the **ℓ_1 distance** or **ℓ_1 deviation**.

Exercise 1.2.10. Verify that the ℓ_1 norm on $\mathbb{R}^n$ satisfies (a)–(d) above.

The ℓ_1 norm and the Euclidean norm are special cases of the so-called **ℓ_p norm**, which is defined for $p \geqslant 1$ by

$$u = (u_1, \ldots, u_n) \mapsto \|u\|_p := \left(\sum_{i=1}^{n} |u_i|^p \right)^{1/p}. \tag{1.12}$$

It can be shown that $u \mapsto \|u\|_p$ is a norm for all $p \geqslant 1$, as suggested by the name (see, e.g., Kreyszig (1978)). For this norm, the subadditivity in (d) is called **Minkowski's inequality**.

Since the Euclidean case is obtained by setting $p = 2$, the Euclidean norm is also called the ℓ_2 norm, and we write $\|\cdot\|_2$ rather than $\|\cdot\|$ when extra clarity is required.

Exercise 1.2.11. Prove that the **supremum norm**, defined by $\|u\|_\infty := \max_{i=1}^{n} |u_i|$, is also a norm on $\mathbb{R}^n$.

(The symbol $\|u\|_\infty$ is used because, for all $u \in \mathbb{R}^n$, we have $\|u\|_p \to \|u\|_\infty$ as $p \to \infty$.)

For the next exercise, we recall that the **indicator function** of logical statement P, denoted here by $\mathbb{1}\{P\}$, takes value 1 (resp., 0) if P is true (resp., false). For example, if $x, y \in \mathbb{R}$, then

$$\mathbb{1}\{x \leqslant y\} = \begin{cases} 1 & \text{if } x \leqslant y \\ 0 & \text{otherwise.} \end{cases}$$

If $A \subset S$, where S is any set, then $\mathbb{1}_A(x) := \mathbb{1}\{x \in A\}$ for all $x \in S$.

Exercise 1.2.12. The so-called ℓ_0 "norm" $\|u\|_0 := \sum_{i=1}^{n} \mathbb{1}\{u_i \neq 0\}$, routinely used in data science applications, is *not* in fact a norm on $\mathbb{R}^n$. Prove this.

1.2.4.2 Equivalence of Vector Norms

When u and $(u_m) := (u_m)_{m\in\mathbb{N}}$ are all elements of $\mathbb{R}^n$, we say that (u_m) **converges** to u and write $u_n \to u$ if

$$\|u_m - u\| \to 0 \text{ as } m \to \infty \text{ for some norm } \|\cdot\| \text{ on } \mathbb{R}^n.$$

It might seem that this definition is imprecise. Don't we need to clarify that the convergence is with respect to a particular norm?

In fact we do not. This is because any two norms $\|\cdot\|_a$ and $\|\cdot\|_b$ on $\mathbb{R}^n$ are **equivalent**, in the sense that there exist finite constants M, N such that

$$M\|u\|_a \leqslant \|u\|_b \leqslant N\|u\|_a \quad \text{for all } u \in \mathbb{R}^n. \tag{1.13}$$

Exercise 1.2.13. Let us write $\|\cdot\|_a \sim \|\cdot\|_b$ if there exist finite M, N such that (1.13) holds. Prove that $\sim$ is an equivalence relation on the set of norms on $\mathbb{R}^n$.

Exercise 1.2.14. Let $\|\cdot\|_a$ and $\|\cdot\|_b$ be any two norms on $\mathbb{R}^n$. Given a point u in $\mathbb{R}^n$ and a sequence (u_m) in $\mathbb{R}^n$, use (1.13) to confirm that $\|u_m - u\|_a \to 0$ implies $\|u_m - u\|_b \to 0$ as $m \to \infty$.

Recall that a set $C \subset \mathbb{R}^n$ is called **closed** in $\mathbb{R}^n$ if, for all $u \in \mathbb{R}^n$ and sequences $\{u_m\} \subset \mathbb{R}^n$ with $u_m \in C$ for all m, we also have $u \in C$. It follows directly from Exercise 1.2.14 that if C is closed for $\|\cdot\|_a$ then it is also closed for $\|\cdot\|_b$. Since G is **open** if and only if G^c is closed, then the same is true for open sets. Similar logic carries over to compact sets. Hence, *topological properties of $\mathbb{R}^n$ and its subsets are the same for any norm*.

1.2.4.3 Banach's Fixed Point Theorem

Let U be a nonempty subset of $\mathbb{R}^n$. A self-map T on U is called a **contraction of modulus λ** on U (or, more simply, a **contraction**) if there exists a $\lambda < 1$ and a norm $\|\cdot\|$ on $\mathbb{R}^n$ such that

$$\|Tu - Tv\| \leqslant \lambda\|u - v\| \quad \text{for all} \quad u, v \in U. \tag{1.14}$$

Exercise 1.2.15. Let T be a contraction on U. Recall that T is called **continuous at $u \in U$** if $Tu_m \to Tu$ for any $\{u_m\} \subset U$ with $u_m \to u$; and **continuous** if T is continuous at every $u \in U$. Show that, under the contraction assumption, T is continuous on U and has at most one fixed point on U.

Let $\|\cdot\|$ be any norm on $\mathbb{R}^n$. The **operator norm** of an $n \times m$ matrix A is defined as

$$\|A\|_o := \sup\left\{\frac{\|Ax\|}{\|x\|} \;:\; x \in \mathbb{R}^m,\, x \neq 0\right\}. \tag{1.15}$$

Exercise 1.2.16. Prove that $\|A\|_o = \|A\|$ when $m = 1$ (i.e., A is just a vector).

Exercise 1.2.17. Let $U = \mathbb{R}^n$ and let $\|\cdot\|$ be any norm on $\mathbb{R}^n$. Let $Tx = Ax + b$, where A is $n \times n$ and b is $n \times 1$. Prove that T is a contraction of modulus $\|A\|_o$ on U whenever $\|A\|_o < 1$.

Exercise 1.2.18. The Solow-Swan map $g(k) = sk^{\alpha} + (1 - \delta)k$ from §1.2.3.2 sends $U := (0, \infty)$ into itself. Here $s > 0$ and α and δ are in $(0, 1)$. Prove that this map is *not* a contraction on U. [Hint: use the definition of the derivative of g as a limit and consider the derivative $g'(k)$ for k close to zero.]

The fundamental importance of contractions stems from the following theorem.

Theorem 1.2.3 (Banach's contraction mapping theorem)**.** *If U is closed in $\mathbb{R}^n$ and T is a contraction of modulus λ on U with respect to some norm $\|\cdot\|$ on $\mathbb{R}^n$, then T has a unique fixed point u^* in U and*

$$\|T^n u - u^*\| \leqslant \lambda^n \|u - u^*\| \quad \textit{for all } n \in \mathbb{N} \textit{ and } u \in U. \tag{1.16}$$

In particular, T is globally stable on U.

We complete a proof of Theorem 1.2.3 in stages.

Exercise 1.2.19. Let U and T have the properties stated in Theorem 1.2.3. Fix $u_0 \in U$ and let $u_m := T^m u_0$. Show that

$$\|u_m - u_k\| \leqslant \sum_{i=m}^{k-1} \lambda^i \|u_0 - u_1\|$$

holds for all $m, k \in \mathbb{N}$ with $m < k$.

EXERCISE 1.2.20. Using the results in Exercise 1.2.19, prove that (u_m) is a Cauchy sequence in $\mathbb{R}^n$.

EXERCISE 1.2.21. Using Exercise 1.2.20, argue that (u_m) hence has a limit $u^* \in \mathbb{R}^n$. Prove that $u^* \in U$.

Proof of Theorem 1.2.3. In the exercises we proved existence of a point $u^* \in U$ such that $T^m u \to u^*$. The fact that u^* is a fixed point of T now follows from Exercise 1.2.6 and Exercise 1.2.15. Uniqueness is implied by Exercise 1.2.15. The bound (1.16) follows from iteration on the contraction inequality (1.20) while setting $v = u^*$. □

1.2.5 Finite-Dimensional Function Space

In this section we clarify notation concerning functions and discuss how sets of real-valued functions are similar to sets of vectors.

1.2.5.1 Real-Valued Functions

If M is any set and f maps M to $\mathbb{R}$, then we call f a **real-valued function** on M and write $f: M \to \mathbb{R}$. Let $\mathbb{R}^M$ be the set of all real-valued functions on M.

In general, if $f, g \in \mathbb{R}^M$ and $\alpha, \beta \in \mathbb{R}$, then the expressions $\alpha f + \beta g$, fg, etc., are also elements of $\mathbb{R}^M$, defined at $x \in M$ by

$$(\alpha f + \beta g)(x) = \alpha f(x) + \beta g(x), \quad (\alpha f)(x) = \alpha f(x), \quad \text{etc.} \tag{1.17}$$

Similarly, $f \vee g$ and $f \wedge g$ are real-valued functions on M defined by

$$(f \vee g)(x) = f(x) \vee g(x) \quad \text{and} \quad (f \wedge g)(x) = f(x) \wedge g(x). \tag{1.18}$$

Figure 1.10 illustrates.

We note for future reference that if $f: A \to B$ and $g: B \to C$, then $g \circ f$ is called the **composition** of f and g. It is the function mapping $a \in A$ to $g(f(a)) \in C$.

1.2.5.2 Functions vs Vectors

Let's now clarify an almost trivial issue that can nonetheless case some degree of confusion. Let M be any finite set. As stated above, $\mathbb{R}^M$ is the set of all real-valued

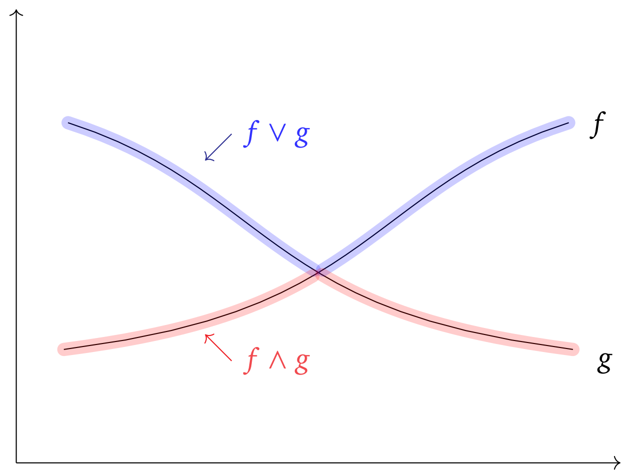


Figure 1.10: Functions $f \vee g$ and $f \wedge g$ when defined on a subset of $\mathbb{R}$

functions on set M. If $|M| = n$ (i.e., M has n elements), then $\mathbb{R}^M$ is, in essence, the vector space $\mathbb{R}^n$ expressed in different notation. The next lemma clarifies.

Lemma 1.2.4. *If $|M| = n$, then*

$$\mathbb{R}^M \ni f \quad \longleftrightarrow \quad (f(x_1), \ldots, f(x_n)) \in \mathbb{R}^n \tag{1.19}$$

is a one-to-one correspondence between $\mathbb{R}^M$ and the vector space $\mathbb{R}^n$.

The lemma just states that a function f can be identified by the set of values that it takes on M, which is an n-tuple of real numbers. Throughout the text, whenever the supporting set M is finite, we freely use the identification in (1.19), adopting whichever notation is most convenient for the application in question.

If $\|\cdot\|$ is any norm on $\mathbb{R}^n$, then we extend $\|\cdot\|$ to $\mathbb{R}^M$ with $|M| = n$ via the identification in (1.19). That is, for $f \in \mathbb{R}^M$, the value $\|f\|$ is given by the norm of the vector $(f(x_1), \ldots, f(x_n))$. We say that a subset of $\mathbb{R}^M$ is closed if the corresponding subset of $\mathbb{R}^n$ is closed, and so on.

For an illustration, observe that Banach's contraction mapping theorem extends directly to operators on $\mathbb{R}^M$ when $|M| = n$. Indeed, if C is closed in $\mathbb{R}^M$ under some norm $\|\cdot\|$ on $\mathbb{R}^M$, and T is a contraction on $C \subset \mathbb{R}^M$, in the sense that, for some $\lambda < 1$,

$$\|Tf - Tg\| \leqslant \lambda \|f - g\| \quad \text{for all} \quad f, g \in C \tag{1.20}$$

then T has a unique fixed point f^* in C and

$$\|T^n f - f^*\| \leqslant \lambda^n \|f - f^*\| \quad \text{for all } n \in \mathbb{N} \text{ and } f \in M.$$

There is no need to supply a new proof: we just need to identify functions in $\mathbb{R}^M$ with vectors in $\mathbb{R}^n$ under the correspondence (1.19).

1.3 Order

As discussed above, fixed point theory plays an important role in dynamic programming, due to the need to solve nonlinear equations. But fixed point theory alone is not sufficient, since dynamic programming also involves optimality. To handle optimality we need one more branch of mathematics, called *order theory*. In fact order theory and fixed point theory intersect in significant ways, as we shall see below.

1.3.1 Partial Orders

Order theory starts with abstract definitions of order over sets. For us it suffices to start with the concept of a partial order, which will already be familiar for most readers. To recall, a **partial order** on a nonempty set P is a relation $\preceq$ on $P \times P$ satisfying, for any p, q, r in P,

$$
\begin{aligned}
& p \preceq p, && \text{(reflexivity)} \\
& p \preceq q \text{ and } q \preceq p \text{ implies } p = q \text{ and} && \text{(antisymmetry)} \\
& p \preceq q \text{ and } q \preceq r \text{ implies } p \preceq r && \text{(transitivity)}
\end{aligned}
$$

When paired with a partial order $\preceq$, the set P (or the pair $(P, \preceq)$) is called a **partially ordered set**.

Example 1.3.1. The usual order $\leqslant$ on $\mathbb{R}$ is a partial order on $\mathbb{R}$.

Exercise 1.3.1. Let P be any set and consider the relation induced by equality, so that $p \preceq q$ if and only if $p = q$. Show that this relation is a partial order on P.

Exercise 1.3.2. Let M be any set. Show that $\subset$ is a partial order on $\wp(M)$, the set of all subsets of M.

A partial order $\preceq$ on P is called a **total order** if either $p \preceq q$ or $q \preceq p$ for all $p, q \in P$.

Example 1.3.2. The usual order $\leqslant$ on $\mathbb{R}$ is a total order, as is the same order on $\mathbb{N}$.

EXERCISE 1.3.3. Is the partial order defined in Exercise 1.3.2 a total order? Either prove or provide a counterexample.

1.3.1.1 Pointwise Orders

Most of the partial orders we care about in this text are pointwise orders. All of these pointwise orders are special cases of the following example.

Example 1.3.3 (Pointwise order over functions)**.** Let M be any set. For f, g in $\mathbb{R}^M$, we write

$$f \leqslant g \text{ if } f(x) \leqslant g(x) \text{ for all } x \in M.$$

This relation $\leqslant$ on $\mathbb{R}^M$ is a partial order called the **pointwise order** on $\mathbb{R}^M$.

A subset B of a partially ordered set $(P, \preceq)$ is called

- **increasing** if $x \in B$ and $x \preceq y$ implies $y \in B$.
- **decreasing** if $x \in B$ and $y \preceq x$ implies $y \in B$.

EXERCISE 1.3.4. Describe the set of increasing sets in $(\mathbb{R}, \leqslant)$.

Example 1.3.4 (Pointwise order over vectors)**.** For vectors $x = (x_1, \ldots, x_n)$ and $y = (y_1, \ldots, y_n)$ in $\mathbb{R}^n$, we write

- $x \leqslant y$ if $x_i \leqslant y_i$ for all $i \in [n]$ and
- $x \ll y$ if $x_i < y_i$ for all $i \in [n]$.

The statements $x \geqslant y$ and $x \gg y$ are defined analogously.[3] The relation $\leqslant$ is a partial order on $\mathbb{R}^n$, also called the **pointwise order**. (In fact, the present example is a special case of Example 1.3.3 under the identification in Lemma 1.2.4 (page 30).) On the other hand, $\ll$ is not a partial order on $\mathbb{R}^n$. (Which axiom fails?)

EXERCISE 1.3.5. Limits in $\mathbb{R}$ preserve weak inequalities. Use this fact to prove that the same is true in $\mathbb{R}^n$. In particular, show that, for vectors $a, b \in \mathbb{R}^n$ and sequence (x_k) in $\mathbb{R}^n$, we have $a \leqslant x_k \leqslant b$ for all $k \in \mathbb{N}$ and $x_k \to x$ implies $a \leqslant x \leqslant b$.

[3] The notation $x \leqslant y$ over vectors is standard, while $x \ll y$ is less so. In some fields, $n \ll k$ is used as an abbreviation for "n is much smaller than k." Our usage lines up with most of the literature on partially ordered vector spaces. See, e.g., Zhang (2012).

Example 1.3.5 (Pointwise order over matrices)**.** Analogous to vectors, for $n \times k$ matrices $A = (a_{ij})$ and $B = (b_{ij})$, we write

- $A \leqslant B$ if $a_{ij} \leqslant b_{ij}$ for all i, j.
- $A \ll B$ if $a_{ij} < b_{ij}$ for all i, j.

The relation $\leqslant$ is a partial order on $\mathbb{M}^{n \times k}$, the set of real-valued $n \times k$ matrices. As for vectors, we call this the **pointwise order**.

1.3.1.2 Pointwise Operations on Vectors

In this text, operations on real numbers such as $|\cdot|$ and $\vee$ are applied to vectors pointwise. For example, for vectors $a = (a_i)$ and $b = (b_i)$ in $\mathbb{R}^n$, we set

$$|a| = (|a_i|), \quad a \wedge b = (a_i \wedge b_i)_{i=1}^n \quad \text{and} \quad a \vee b = (a_i \vee b_i)_{i=1}^n$$

(The last two are special cases of (1.18).)

Lemma 1.3.1. *For all $a, b, c \in \mathbb{R}^n$, the following statements are true:*

- $|a + b| \leqslant |a| + |b|$.
- $(a \wedge b) + c = (a + c) \wedge (b + c)$ *and* $(a \vee b) + c = (a + c) \vee (b + c)$.
- $(a \vee b) \wedge c = (a \wedge c) \vee (b \wedge c)$ *and* $(a \wedge b) \vee c = (a \vee c) \wedge (b \vee c)$.
- $|a \wedge c - b \wedge c| \leqslant |a - b|$.
- $|a \vee c - b \vee c| \leqslant |a - b|$.

The first item is called the **triangle inequality**. A proof of lemma 1.3.1 can be found in Theorem 30.1 of Aliprantis and Burkinshaw (1998).

It is also true that, if $a, b \in \mathbb{R}^n_+$ and $c \in \mathbb{R}^n$, then

$$(a + b) \wedge c \leqslant (a \wedge c) + (b \wedge c). \tag{1.21}$$

Exercise 1.3.6. Prove: If $a, b \in \mathbb{R}_+$ and $c \in \mathbb{R}$, then $|a \wedge c - b \wedge c| \leqslant |a - b| \wedge c$.

Exercise 1.3.7. Prove: If B is $m \times k$ and $B \geqslant 0$, then $|Bx| \leqslant B|x|$ for all $k \times 1$ column vectors x.

In dynamic programming, we often deal with maxima and suprema in the context of contraction maps. In these settings, the following lemma will be helpful.

Lemma 1.3.2. *Let D be any set. If f and g are bounded functions in $\mathbb{R}^D$, then*

$$|\sup_{z \in D} f(z) - \sup_{z \in D} g(z)| \leqslant \sup_{z \in D} |f(z) - g(z)|. \tag{1.22}$$

EXERCISE 1.3.8. Prove Lemma 1.3.2. (If you are unfamiliar with suprema, you can assume that D is finite and prove the claim in Lemma 1.3.2 after replacing sup with max. If you are familiar with suprema, then confirm that, if the maxima exist, then we can replace sup with max in Lemma 1.3.2 and the statement is still true.)

EXERCISE 1.3.9. Let U be a closed subset of $\mathbb{R}^n$ with the property that $u, v \in U$ implies $u \vee v \in U$. Let T_1 and T_2 be contraction maps on U under the supremum norm $\| \cdot \|_\infty$. Prove that the self-map $T: U \to U$ defined by $Tu := (T_1 u) \vee (T_2 u)$ is also contraction on U under the supremum norm.

1.3.2 Order-Preserving Maps

Given two partially ordered sets $(P, \preceq)$ and $(Q, \trianglelefteq)$, a map T from P to Q is called **order-preserving** if

$$p, p' \in P \text{ and } p \preceq p' \quad \Longrightarrow \quad Tp \trianglelefteq Tp'. \tag{1.23}$$

In the case where $Q = \mathbb{R}$ and $\trianglelefteq$ is the standard order $\leqslant$, it is common to call T "increasing" instead of order-preserving. We conform to this terminology. In particular, given partially ordered set $(P, \preceq)$, we call $h \in \mathbb{R}^P$

- **increasing** if $p \preceq p'$ implies $h(p) \leqslant h(p')$ and
- **decreasing** if $p \preceq p'$ implies $h(p) \geqslant h(p')$.

We frequently use the symbol $i\mathbb{R}^P$ for the set of increasing functions in $\mathbb{R}^P$.

Example 1.3.6. Let $\leqslant$ denote the pointwise partial order over vectors and matrices. If A is $n \times n$ with $A \geqslant 0$, then $T: \mathbb{R}^n \to \mathbb{R}^n$ defined by $Tx = Ax + b$ is order preserving on $\mathbb{R}^n$, since $x \leqslant y$ implies $y - x \geqslant 0$, and hence $A(y - x) \geqslant 0$. But then $Ax \leqslant Ay$ and hence $Tx \leqslant Ty$.

Example 1.3.7. Let $\mathscr{C}$ be all continuous functions from $M := [a, b]$ to $\mathbb{R}$ and let $\leqslant$ be the pointwise partial order on $\mathscr{C}$. Integration can be understood as a mapping I from $\mathscr{C}$ to $\mathbb{R}$ such that

$$I(f) := \int_a^b f(x) dx \qquad (f \in \mathscr{C}).$$

Since $f \leqslant g$ implies $\int_a^b f(x)dx \leqslant \int_a^b g(x)dx$, the map I is order-preserving on $\mathscr{C}$.

EXERCISE 1.3.10. Let X be a random variable mapping Ω to finite M. Define $\ell\colon \mathbb{R}^M \to \mathbb{R}$ by $\ell h = \mathbb{E}h(X)$. Show that ℓ is increasing when $\mathbb{R}^M$ has the pointwise order.

As usual, if $h\colon P \to Q$ and $P, Q \subset \mathbb{R}$, then we will call h

- **strictly increasing** if $x < y$ implies $h(x) < h(y)$, and
- **strictly decreasing** if $x < y$ implies $h(x) > h(y)$.

1.3.3 Parametric Monotonicity

A major concern in mathematical modeling is whether or not a change in a parameter shifts an endogenous outcome (e.g., solution or equilibrium) up or down. For example, the parameter in question might enter into a central bank decision rule for pegging a particular interest rate, and the aim is to know whether increasing that parameter will increase or decrease steady state inflation. By providing sufficient conditions for monotone shifts in fixed points, results in this section can help tackle such questions.

Let $(P, \preceq)$ be a partially ordered set. Given two self-maps S and T on a set P, we write $S \preceq T$ if $Su \preceq Tu$ for every $u \in P$ and say that T **dominates** S on P.

Example 1.3.8. Let $P = \mathbb{R}^n_+$ with the pointwise order on vectors, let $Sx = Ax + b$ and $Tx = Bx + b$, where $b \in \mathbb{R}^n$ and A and B are $n \times n$. If $A \leqslant B$, then, for any $x \in \mathbb{R}^n_+$, we have $Ax \leqslant Bx$. Hence $Sx \leqslant Tx$ and T dominates S on $\mathbb{R}_+$.

EXERCISE 1.3.11. Let $(P, \preceq)$ be a partially ordered set, let $\mathscr{S}$ be the set of all self-maps on P and write $S \preceq T$ if T dominates S on P, as above. Show that $\preceq$ is a partial order on $\mathscr{S}$.

One might assume that, in a setting where T dominates S, the fixed points of T will be larger. This can hold, as in Figure 1.11, but it can also fail, as in Figure 1.12. One difference between these two scenarios is that, in the case of Figure 1.11, the map T is globally stable. This leads us to our next result.

Proposition 1.3.3. *Let S and T be self-maps on $M \subset \mathbb{R}^n$ and let $\leqslant$ be the pointwise partial order. If T dominates S on M and, in addition, T is order-preserving and globally stable on M, then its unique fixed point dominates any fixed point of S.*

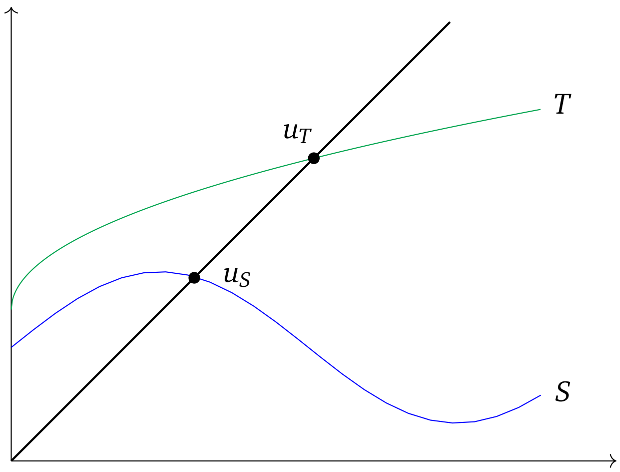


Figure 1.11: Ordered fixed points when global stability holds

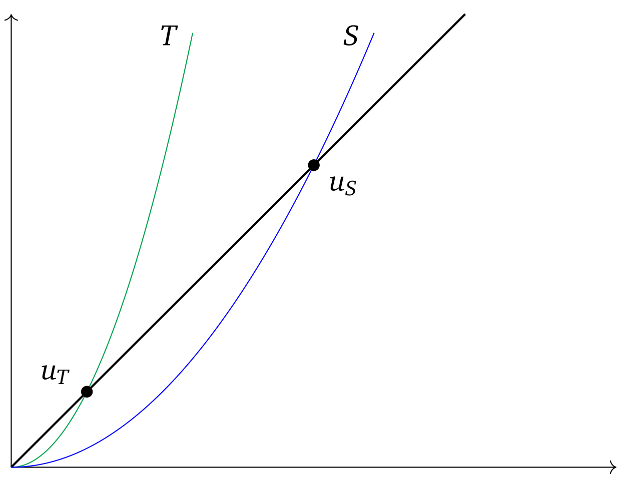


Figure 1.12: Reverse-ordered fixed points when global stability fails

Proof of Proposition 1.3.3. Assume the conditions of the proposition and let u_T be the unique fixed point of T. Let u_S be any fixed point of S. Since $S \leqslant T$, we have $u_S = Su_S \leqslant Tu_S$. Applying T to both sides of this inequality and using the order-preserving property of T and transitivity of $\leqslant$ gives $u_S \leqslant T^2 u_S$. Continuing in this fashion yields $u_S \leqslant T^k u_S$ for all $k \in \mathbb{N}$. Taking the limit in k and using the fact that $\leqslant$ is closed under limits gives $u_S \leqslant u_T$. □

Proposition 1.3.3 will be applied many times in the remainder of the notes.

As an application of Proposition 1.3.3, consider again the Solow–Swan growth model $k_{t+1} = g(k_t) := sf(k_t) + (1 - \delta)k_t$. We saw in §1.2.3.2 that if $f(k) = Ak^\alpha$ where $A > 0$ and $\alpha \in (0, 1)$, then g is is globally stable on $M := (0, \infty)$. Clearly $k \mapsto g(k)$ is order-preserving on M. If we now increase, say, the savings rate s, then g will be shifted up everywhere, implying, via Proposition 1.3.3, that the fixed point will also rise. Exercise 1.3.12 asks you to step through the details.

Exercise 1.3.12. Let $g(k) = sAk^\alpha + (1 - \delta)k$ where all parameters are strictly positive, $\alpha \in (0, 1)$ and $\delta \leqslant 1$. Let $k^*(s, A, \alpha, \delta)$ be the unique fixed point of g in M. Without using the expression we derived for k^* previously, show that

(i) $k^*(s, A, \alpha, \delta)$ is increasing in s and A.

(ii) $k^*(s, A, \alpha, \delta)$ is decreasing in δ.

Figure 1.13 helps illustrate the results of Exercise 1.3.12. The top left sub-figure shows the default parameterization, with $A = 2.0$, $s = \alpha = 0.3$ and $\delta = 0.4$. The other sub-figures show how the steady state changes as parameters shift from that default.

1.4 Infinite-Horizon Job Search

Now we are armed with useful fixed point methods, let's return to the job search problem first discussed in §1.1.2 and solve for optimal choices more carefully.

1.4.1 Values and Policies

In this section we solve for the value function of the infinite horizon job search problem and use it to make optimal choices.

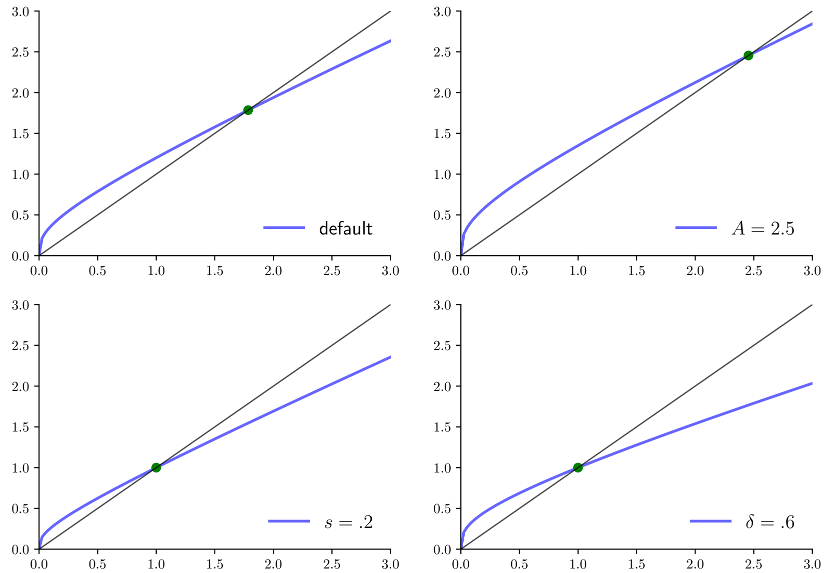


Figure 1.13: Parametric monotonicity for the Solow-Swan model

1.4.1.1 Optimal Choices

In §1.1.2.1 we proposed a strategy for solving the infinite-horizon job search problem, which required computing the value function v^*. You will recall that v^* solves the Bellman equation, which is

$$v^*(w) = \max\left\{\frac{w}{1-\beta},\ c + \beta \sum_{w' \in \mathsf{W}} v^*(w')\varphi(w')\right\} \qquad (w \in \mathsf{W}). \tag{1.24}$$

Suppose for the moment that we can compute v^*, and let

$$h^* := c + \beta \sum_{w'} v^*(w')\varphi(w') \tag{1.25}$$

be the infinite-horizon **continuation value**. The optimal decision at any given time, facing current wage draw $w \in \mathsf{W}$, is as follows:

(i) If $w/(1-\beta) \geqslant h^*$, then accept the job offer.

(ii) If not, then reject and wait for the next offer.

1.4.1.2 The Bellman Operator

The methodology proposed above requires that we solve for v^*. To do so, we introduce an operator T, called the **Bellman operator**, such that any fixed point of T solves the Bellman equation and vice versa. This is true by construction for T defined at $v \in \mathbb{R}^{\mathsf{W}}$ by

$$(Tv)(w) = \max\left\{\frac{w}{1-\beta}, \, c + \beta \sum_{w' \in \mathsf{W}} v(w')\varphi(w')\right\} \qquad (w \in \mathsf{W}) \tag{1.26}$$

Let $\mathscr{V} := \mathbb{R}_+^{\mathsf{W}}$ and let $\|\cdot\|_\infty$ be the supremum norm on $\mathscr{V}$. The distance between two elements f, g of $\mathscr{V}$ is measured by $\|f - g\| = \max_{w \in \mathsf{W}} |f(w) - g(w)|$. Under this norm distance, we have the following result.

Proposition 1.4.1. *The Bellman operator T is a contraction of modulus β on $\mathscr{V}$.*

The proof of Proposition 1.4.1 is given below. One key implication of the proposition is that $T^k v \to v^*$ as $k \to \infty$ for any $v \in \mathscr{V}$. In other words, we can compute v^* to any required degree of accuracy by successive approximation.

For the proof of Proposition 1.4.1, we will use the elementary bound

$$|\alpha \vee x - \alpha \vee y| \leqslant |x - y| \qquad (\alpha, x, y \in \mathbb{R}) \tag{1.27}$$

(Here $a \vee b = \max\{a, b\}$. You can check (1.27) by sketching it on a line.)

Proof of Proposition 1.4.1. Take any f, g in $\mathscr{V}$ and fix any $w \in \mathsf{W}$. The bound in (1.27) gives

$$\begin{aligned} |(Tf)(w) - (Tg)(w)| &\leqslant \left| c + \beta \sum_{w'} f(w')\varphi(w') - \left(c + \beta \sum_{w'} g(w')\varphi(w')\right)\right| \\ &= \beta \left| \sum_{w'} [f(w') - g(w')]\varphi(w') \right|. \end{aligned}$$

Applying the triangle inequality, we obtain

$$|(Tf)(w) - (Tg)(w)| \leqslant \beta \sum_{w'} |f(w') - g(w')|\varphi(w') \leqslant \beta \|f - g\|_\infty.$$

Taking the supremum over all w on the left hand side of this expression leads to

$$\|Tf - Tg\|_\infty \leqslant \beta \|f - g\|_\infty.$$

Since f, g were arbitrary elements of $\mathscr{V}$, the contraction claim is verified. □

1.4.1.3 Optimal Policies

The entire field of dynamic programming centers around the problem of finding optimal policies. In order to prepare ourselves for this perspective, we briefly introduce the notion of polices and related them to the job search application.

In general, for a dynamic program, choices by the controller aim at maximizing lifetime rewards and consist of a sequence $\{a_t\}_{t \geqslant 0}$ specifying how the agent acts at each point in time. Since agents are not clairvoyant, it is natural to assume that a_t can depend on present and past events but not future ones. In other words, a_t is a function of past state-action pairs (a_{t-i}, x_{t-i}) for $i \geqslant 1$ and the current state x_t.

One of the key ideas of dynamic programming is that the state should be designed such that the current state x_t is sufficient to determine the optimal current action.

Example 1.4.1. In Example 1.0.1, the retailer must choose stock orders and prices in each period. Every quantity relevant to this decision should be included in the current state, contingent on keeping the problem tractable. Thus, the current state might contain not just current inventories and business conditions, but also information such as the rate at which inventories have changed over each of the past six months.

If the current state x_t determines the current action, then we must be able to write $a_t = \sigma_t(x_t)$ for some function σ_t. In the language of dynamic programming, σ_t is called a **policy function**, or a policy. In general, policies are maps from states to actions.

In the case of the job search model, the state is the current wage offer, while the possible actions are accept or reject the current offer. With 0 interpreted as reject and 1 understood as accept, the action space is $\{0, 1\}$, so a **policy** is a map σ from W to $\{0, 1\}$. Let Σ be the set of all such maps.

You should understand a policy as an "instruction manual" for the agent: for an agent following $\sigma \in \Sigma$, if current wage offer is w, the agent always responds with $\sigma(w) \in \{0, 1\}$. In particular, the policy dictates whether the agent accepts or rejects at any given wage.

Notice that, as we have written it, the policy σ does *not* depend on t. This is because the agent always looks forward toward an infinite horizon. Hence, if choice $\sigma(w)$ is optimal at time t, given wage offer w, then $\sigma(w)$ is also the right choice at date $t + k$, for any $k \in \mathbb{N}$.

Remark 1.4.1. The above statement is really an assumption: we have not *proved* that the agent can obtain maximal value by fixing a policy and following it forever, although we did provide some intuition as to why that should be so. Later, we will verify that this assumption costs us nothing: the agent cannot do better by allowing the policy to change across time.

For each $v \in \mathscr{V}$, let us define a **v-greedy policy** to be a $\sigma \in \Sigma$ satisfying

$$\sigma(w) = \mathbb{1}\left\{\frac{w}{1-\beta} \geqslant c + \beta \sum_{w' \in \mathsf{W}} v(w')\varphi(w')\right\} \quad \text{for all } w \in \mathsf{W}. \tag{1.28}$$

A v-greedy policy uses v to compute a continuation value and then chooses to accept or reject based on the action that generates a expected higher payoff. Our discussion of optimal choices in §1.4.1.1 can now be summarized as follows:

The agent should adopt a v^*-greedy policy.

The statement above is sometimes called **Bellman's principle of optimality**. We will formalize all of these ideas in the remainder of the text.

Inserting v^* into (1.28) and rearranging, we can express a v^*-greedy policy via

$$\sigma^*(w) = \mathbb{1}\{w \geqslant w^*\} \quad \text{where } \ w^* := (1-\beta)h^*. \tag{1.29}$$

The term w^* in (1.29) is called the **reservation wage**, and parallels the reservation wage that we introduced for the finite-horizon problem. Equation (1.29) states that value maximization requires accepting an offer if and only if it exceeds the reservation wage. Thus, w^* provides a scalar summary of the solution to the problem.

1.4.2 Computation

Now we have a method for solving for the optimal policy, let's turn to computation. In §1.4.2.1, we apply a standard dynamic programming method, called value function iteration. Below, in §1.4.2.2, we apply a more specialized method, which uses the structure of the job search problem to speed up computation.

1.4.2.1 Value Function Iteration

Recall that, by Proposition 1.4.1, we can compute an approximate optimal policy by applying successive approximation via the Bellman operator. In the language of dynamic programming, this is called **value function iteration**. The standard procedure is given in Algorithm 1.

While $T^k v$ never exactly attains v^* in most cases, we can obtain a close approximation by monitoring the distance between successive iterates, waiting until they become

Algorithm 1: Value function iteration for job search

input $v_0 \in \mathscr{V}$, an initial guess of v^*
input τ, a tolerance level for error
$\varepsilon \leftarrow \tau + 1$
$k \leftarrow 0$
while $\varepsilon > \tau$ **do**
  for $w \in \mathsf{W}$ **do**
    $v_{k+1}(w) \leftarrow (Tv_k)(w)$
  end
  $\varepsilon \leftarrow \|v_k - v_{k+1}\|_\infty$
  $k \leftarrow k + 1$
end
Compute a v_k-greedy policy σ
return σ

small. Later we will quantify this distance in terms of k, the number of iterations, as well as the parameters.

Listing 6 implements value function iteration for the infinite-horizon job search model, using the function for successive approximation from Listing 4.

Figure 1.14 shows a sequence of iterates $\{T^k v\}$ when $v \equiv 0$ and parameters are as given in Listing 1 (page 7). Iterates $0, 1$ and 2 are shown, in addition to a limiting function (iterate 1000). If you experiment with different initial conditions, you will see that the converges to the same limit.

Figure 1.15 shows an approximation of v^* computed using the code in Listing 6, along with the stopping reward $w/(1-\beta)$ and the corresponding continuation value (1.25). As expected, the value function is the pointwise supremum of the stopping reward and the continuation value. The agent chooses to accept an offer only when that offer exceeds some value close to 43.5.

1.4.2.2 Computing the Continuation Value Directly

The technique we employed to solve the job search model in §1.4.1 follows a standard approach to dynamic programming. In fact, for this particular problem, there is an easier way to compute the optimal policy that sidesteps calculating the value function. This section explains how.

```julia
include("two_period_job_search.jl")
include("s_approx.jl")

" The Bellman operator. "
function T(v, model)
    (; n, w_vals, ϕ, β, c) = model
    return [max(w / (1 - β), c + β * v'ϕ) for w in w_vals]
end

" Get a v-greedy policy. "
function get_greedy(v, model)
    (; n, w_vals, ϕ, β, c) = model
    σ = w_vals ./ (1 - β) .>= c .+ β * v'ϕ  # Boolean policy vector
    return σ
end

" Solve the infinite-horizon IID job search model by VFI. "
function vfi(model=default_model)
    (; n, w_vals, ϕ, β, c) = model
    v_init = zero(model.w_vals)
    v_star = successive_approx(v -> T(v, model), v_init)
    σ_star = get_greedy(v_star, model)
    return v_star, σ_star
end
```

Listing 6: Value function iteration (`iid_job_search.jl`)

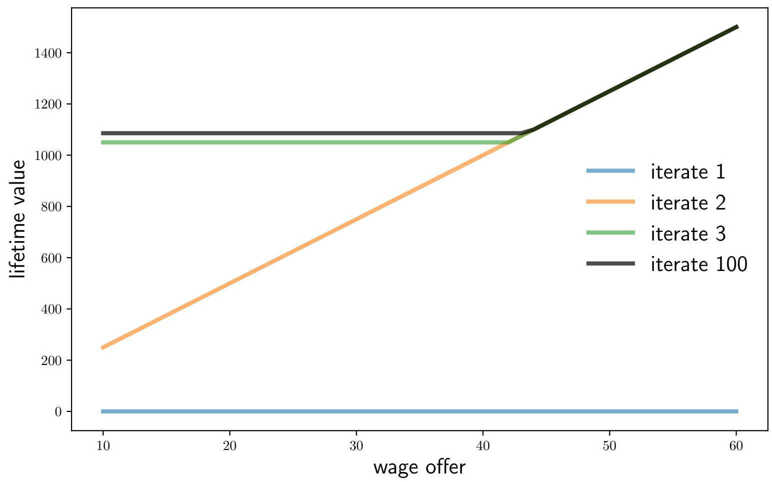


Figure 1.14: A sequence of iterates of the Bellman operator

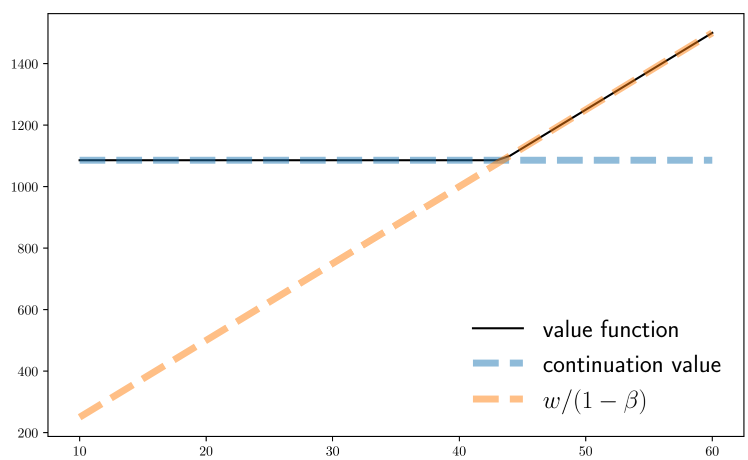


Figure 1.15: The approximate value function for job search

Recall that the value function satisfies the Bellman equation

$$v^*(w) = \max\left\{\frac{w}{1-\beta},\ c + \beta \sum_{w'} v^*(w')\varphi(w')\right\} \qquad (w \in \mathsf{W}), \tag{1.30}$$

and that the continuation value is given by (1.25). We can use h^* to eliminate v^* from (1.30). First we insert h^* on the right hand side of (1.30) and then we replace w with w', which gives $v^*(w') = \max\{w'/(1-\beta),\ h^*\}$. Now we take expectations of both sides, multiply by β and add c to obtain

$$h^* = c + \beta \sum_{w'} \max\left\{\frac{w'}{1-\beta},\ h^*\right\}\varphi(w'). \tag{1.31}$$

To obtain the unknown value h^*, we introduce the mapping $g\colon \mathbb{R}_+ \to \mathbb{R}_+$ defined by

$$g(h) = c + \beta \sum_{w'} \max\left\{\frac{w'}{1-\beta},\ h\right\}\varphi(w'). \tag{1.32}$$

By construction, h^* solves (1.31) if and only if h^* is a fixed point of g.

Exercise 1.4.1. Show that g is a contraction map on $\mathbb{R}_+$ under the usual Euclidean distance. Conclude that h^* is the unique fixed point of g in $\mathbb{R}_+$.

Solving for the fixed point h^* is much easier that value function iteration, since the fixed point problem is in $\mathbb{R}_+$ rather than $\mathbb{R}^n_+$. Figure 1.16 visualizes this fixed point problem.

Once we obtain h^*, or a close approximation, we have essentially solved the dynamic programming problem, since a policy σ^* is v^*-greedy if and only if it satisfies

$$\sigma^*(w) = \mathbb{1}\left\{\frac{w}{1-\beta} \geqslant h^*\right\} \qquad (w \in \mathbb{R}_+). \tag{1.33}$$

Figure **??** shows the function g using the discrete wage offer distribution and parameters as adopted previously. The unique fixed point is h^*. In view of the results in Exercise 1.4.1, this value can be computed by iterating with g on any initial condition in $\mathbb{R}_+$. Doing so produces a value of around 1086. The reservation wage w^* is then calculated as $w^* = (1-\beta)h^* \approx 43.4$.

Exercise 1.4.2. As a computational exercise, compare the value function v^* computed via

$$v^*(w) = \max\left\{\frac{w}{1-\beta},\ h^*\right\}$$

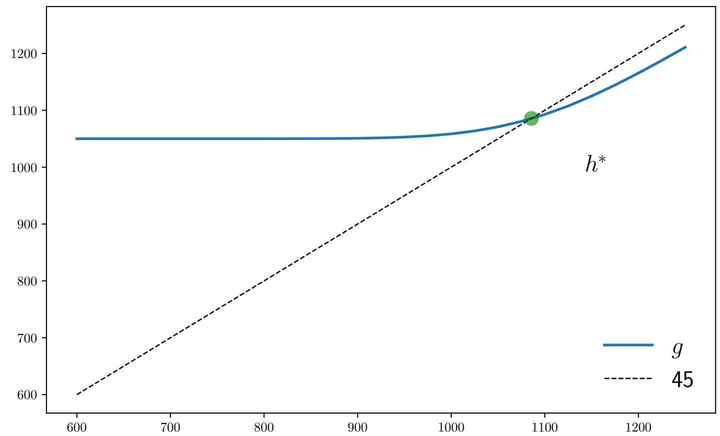


Figure 1.16: Computing the continuation value as the fixed point of g

with our previous result, shown in Figure 1.15. You should find them essentially identical.

EXERCISE 1.4.3. Prove that the optimal continuation value h^* is increasing in β. [Hint: Use Proposition 1.3.3.]

1.5 Chapter Notes

To be added.

Chapter 2

Valuation

Over the next few chapters, we build a detailed understanding of a class of dynamic programs called finite Markov decision processes (Finite MDPs). Finite MDPs are both a workhorse and a fundamental building block. When we define them in Chapter 4, you will see that the essence of the optimization problem is to choose a Markov chain to maximize a discounted sum of rewards. In this chapter, we lay the groundwork for that analysis by introducing Markov chains and showing how to compute discounted reward sums in a range of applications.

2.1 Finite Markov Chains

Markov processes are an essential workhorse for countless models in economics and finance. In fact almost every kind of stochastic process studied in these fields can be represented as a Markov process after suitable choice of the state space. Markov processes with discrete state spaces are typically called Markov chains. In this section, we study the foundations of Markov chains, with a eye towards understanding valuation in a Markov setting.

2.1.1 Matrices and Dynamics

As a first step, we review nonnegative matrices and their properties, as well as describing how they connect to transition probabilities for Markov chains.

2.1.1.1 Nonnegative Matrices

In what follows, we call a matrix A **nonnegative** and write $A \geqslant 0$ if all the elements of A are nonnegative. In other words, A dominates the zero matrix in the pointwise partial order over the set of real $n \times k$ matrices introduced on page 33.

A nonnegative $n \times k$ matrix A is called **positive**, and we write $A \gg 0$, if every element of A is strictly positive. A nonnegative square matrix A is called **irreducible** if $\sum_{k \in \mathbb{N}} A^k \gg 0$. This is obviously stronger than nonnegativity but weaker than positivity. An interpretation in terms of connected networks is given in Chapter 1 of Sargent and Stachurski (2022).

EXERCISE 2.1.1. Let A be $n \times k$. Show that the map $x \mapsto Ax$ is order-preserving on $\mathbb{R}^k$, under the usual pointwise order, whenever $A \geqslant 0$.

EXERCISE 2.1.2. Let A and B be $n \times n$ with $0 \leqslant A \leqslant B$. Prove that $A^k \leqslant B^k$ for all $k \in \mathbb{N}$ and, in addition, that $r(A) \leqslant r(B)$.

Let A be $n \times n$. It is not always true that the spectral radius $r(A)$ is an eigenvalue.[1] However, when $A \geqslant 0$, we have the following:

Theorem 2.1.1 (Perron–Frobenius). *If $A \geqslant 0$, then $r(A)$ is an eigenvalue of A with nonnegative, real-valued right and left eigenvectors. If, in addition A is irreducible, then these eigenvalues are everywhere positive and unique.*

The theorem tells us that, given $A \geqslant 0$, we can find a nonnegative, nonzero column vector e and a nonnegative, nonzero row vector ε such that

$$Ae = r(A)e \quad \text{and} \quad \varepsilon A = r(A)\varepsilon. \tag{2.1}$$

It also tells us that if A is irreducible, then $e \gg 0$, $\varepsilon \gg 0$, and these vectors are unique. Uniqueness is up to a positive multiple, so if $\hat{e}$ is a nonnegative nonzero vector satisfying $A\hat{e} = r(A)\hat{e}$, then $\hat{e} = \alpha e$ for some $\alpha > 0$. A similar statement holds for the left eigenvalue ε.

(In fact the Perron–Frobenius theorem contains other useful results beyond those stated here. A complete statement and full proof of the theorem can be found in Meyer (2000).)

Using the Perron–Frobenius theorem, we can provide useful bounds on the spectral radius of a nonnegative matrix. In what follows, fix $n \times n$ matrix $A = (a_{ij})$ and set

[1] For example, if $A = \mathrm{diag}(-1, 0)$ then the eigenvalues of A are $\{-1, 0\}$. Hence $r(A) = |-1| = 1$, which is not an eigenvalue of A.

- $\mathrm{rs}_i(A) := \sum_j a_{ij}$ = the i-th row sum of A and
- $\mathrm{cs}_j(A) := \sum_i a_{ij}$ = the j-th column sum of A.

Lemma 2.1.2. *If $A \geqslant 0$, then*

(i) $\min_i \mathrm{rs}_i(A) \leqslant r(A) \leqslant \max_i \mathrm{rs}_i(A)$ *and*

(ii) $\min_j \mathrm{cs}_j(A) \leqslant r(A) \leqslant \max_j \mathrm{cs}_j(A)$.

EXERCISE 2.1.3. Prove Lemma 2.1.2. (Hint: Since e and ε are nonnegative and nonzero, and since eigenvectors are defined only up to nonzero multiples, you can assume that both of these vectors sum to 1.)

The next result is called a "local" spectral radius theorem. While it is similar to Gelfand's formula (page 13), it replaces the matrix norm in that result with an arbitrary vector norms. This can be more convenient, as we will see below.

Lemma 2.1.3. *Let $\|\cdot\|$ be any norm on $\mathbb{R}^n$. If A is $n \times n$, $A \geqslant 0$ and $h \gg 0$, then*

$$\|A^k h\|^{1/k} \to r(A) \qquad (k \to \infty). \tag{2.2}$$

Lemma 2.1.3 tells us that, eventually, for any positive h, the norm of the vector $A^k h$ grows at rate $r(A)$. A proof can be found in Krasnoselskii (1964) or Theorem B1 of Borovička and Stachurski (2020).

2.1.1.2 Stochastic Matrices

An $n \times n$ matrix $P = (P_{ij})$ is called a **stochastic matrix** if

$$P \geqslant 0 \quad \text{and} \quad P\mathbb{1} = \mathbb{1}, \text{ where } \mathbb{1} \in \mathbb{R}^n \text{ is a column vector of ones.}$$

In other words, P is nonnegative and has unit row sums.

EXERCISE 2.1.4. Let P, Q be $n \times n$ stochastic matrices. Prove the following facts.

(i) PQ is also stochastic.

(ii) $r(P) = 1$.

(iii) There exists a row vector $\psi \in \mathbb{R}^n_+$ such that $\psi\mathbb{1} = 1$ and $\psi P = \psi$.

(iv) If P is irreducible, then the vector ψ in (iii) is everwhere positive and unique, in the sense that no other vector $\psi \in \mathbb{R}^n_+$ satisfies $\psi\mathbb{1} = 1$ and $\psi P = \psi$.

The vector ψ in part (iii) of Exercise 2.1.4 is called a **stationary distribution** for P. Such distributions play an important role in the theory of Markov chains and we discuss their interpretation and significance in §2.1.1.6.

2.1.1.3 Markov Chains

Let X be a finite set with elements $x_1, \ldots, x_n$. We will consider random processes $(X_t)_{t \geqslant 0}$ taking values in X and, in this setting, X is called the **state space** of the process. Our particular interest is Markov chains, each one of which will be generated by some stochastic matrix P. In particular, the element P_{ij} gives the probability of the chain moving from the i-th element of X to the j-th.

In what follows, the ideas are clearer if we write $P(x, x')$ for the probability of moving from state x to state x'. To formalizes this notation, we note that each stochastic $n \times n$ matrix $P = (P_{ij})$ can be identified with a function P on $\mathsf{X} \times \mathsf{X}$ via $P(x_i, x_j) := P_{ij}$. This map is obviously one-to-one, and the resulting function P on $\mathsf{X} \times \mathsf{X}$ obeys

$$P \geqslant 0 \quad \text{and} \quad \sum_{x' \in \mathsf{X}} P(x, x') = 1 \text{ for all } x \in \mathsf{X}. \tag{2.3}$$

In view of the one-to-one correspondence, we will freely call any $P \in \mathbb{R}^{\mathsf{X} \times \mathsf{X}}$ satisfying (2.3) a **stochastic matrix**. The spectral radius of such a function is defined as the spectral radius of the corresponding matrix, and so on.

Consistent with previous notation, $\mathscr{D}(\mathsf{X})$ denotes all $\varphi \in \mathbb{R}^{\mathsf{X}}_+$ with $\sum_{x \in \mathsf{X}} \varphi(x) = 1$ and is called the set of **distributions** on X. Note that, with this notation, (2.3) can also be written as

$$P(x, \cdot) \in \mathscr{D}(\mathsf{X}) \quad \text{for all } \; x \in \mathsf{X}.$$

Since we can identify any $f \in \mathbb{R}^{\mathsf{X}}$ with a corresponding vector in $\mathbb{R}^n$ (see page 30), the set $\mathscr{D}(\mathsf{X})$ can also be thought of as a subset of $\mathbb{R}^n$. This set of vectors (i.e., the nonnegative vectors that sum to unity) is sometimes called the **unit simplex**. In matrix expressions, we view distributions *as row vectors*. This convention will simplify notation in what follows.

Let $(X_t)_{t \geqslant 0}$ be a sequence of random variables taking values in X. We say that (X_t) is a **Markov chain** on X if there exists a stochastic matrix P on X such that

$$\mathbb{P}\{X_{t+1} = y \mid X_0, X_1, \ldots, X_t\} = P(X_t, x') \quad \text{for all} \quad t \geqslant 0, \; y \in \mathsf{X}. \tag{2.4}$$

In this context, P is called the **transition matrix** of the Markov chain.

To simplify terminology, we also call an X-valued random process $(X_t)_{t \geq 0}$ **P-Markov** when it satisfies (2.4). We call either X_0 or its distribution ψ_0 the **initial condition** of (X_t) depending on context.

The definition of a Markov chain says two things:

(i) When updating to X_{t+1} from X_t, earlier states are not required.

(ii) The matrix P encodes all of the information required to perform the update, given the current state X_t.

One way to think about Markov chains is algorithmically: Let P be a stochastic matrix and let ψ_0 be an element of $\mathscr{D}(\mathsf{X})$. Now generate (X_t) via Algorithm 2. The resulting sequence is P-Markov with initial condition ψ_0.

Algorithm 2: Generation of P-Markov (X_t) with initial condition ψ_0

```
t ← 0
X_t ← a draw from ψ_0
while t < ∞ do
    X_{t+1} ← a draw from the distribution P(X_t, ·)
    t ← t + 1
end
```

2.1.1.4 Application: S-s Dynamics

As an example, let us consider a firm whose inventory behavior follows S-s dynamics, meaning that the firm waits until its inventory of a given product falls below some level $s > 0$ and then replenishes by buying some fixed amount. This kind of behavior is reasonable if ordering inventory involves a fixed cost. (Later, in §4.2.1, we will show how S-s behavior arises naturally in a model where the firm chooses its inventory path to maximize its present value.)

To implement S-s dynamics, we suppose that a firm's inventory $(X_t)_{t \geq 0}$ of a given product obeys

$$X_{t+1} = \max\{X_t - D_{t+1}, 0\} + S \mathbb{1}\{X_t \leq s\}, \tag{2.5}$$

where

- $(D_t)_{t \geq 1}$ is an exogenous IID demand process with $D_t \stackrel{d}{=} \varphi \in \mathscr{D}(\mathbb{Z}_+)$ for all t and
- S is the amount of stock ordered every time that inventory falls below s.

For the distribution φ of the demand process we take the geometric distribution, so that $\varphi(d) = \mathbb{P}\{D_t = d\} = p(1-p)^d$ for $d \in \mathbb{Z}_+$.

Exercise 2.1.5. A suitable state space for this model is $\mathsf{X} := \{0, \ldots, S + s\}$, since

$$X_t \in \mathsf{X} \implies \mathbb{P}\{X_{t+1} \in \mathsf{X}\} = 1$$

for all t. Verify this claim.

If we define

$$h(x, d) = \max\{x - d, 0\} + S\mathbb{1}\{x \leqslant s\},$$

so that $X_{t+1} = h(X_t, D_{t+1})$ for all t, then the transition matrix can be expressed as

$$P(x, x') = \mathbb{P}\{h(x, D_{t+1}) = x'\} = \sum_{d \geqslant 0} \mathbb{1}\{h(x, d) = x'\}\varphi(d)$$

for $(x, x') \in \mathsf{X} \times \mathsf{X}$. In calculations we can truncate the infinite sum and still obtain a good approximation to P.

Listing 7 provides Julia code that implements the model, simulates inventory paths and computes other objects of interest. Since the state space $\mathsf{X} = \{x_1, \ldots, x_n\}$ corresponds to $\{0, \ldots, S + s\}$, we have $x_i = i - 1$. This convention is used when computing `P[i, j]`, which corresponds to $P(x_i, x_j)$. The code in the listing is used to produce the simulation of inventories in Figure 2.1.

The function `compute_mc` returns an instance of a `MarkovChain` object, which can store both the state X and the transition probabilities. The `QuantEcon.jl` library defines this data type and provides functions that act on it, in order to facilitate simulation of Markov chains, computation of stationary distributions and other related tasks.

2.1.1.5 Higher Order Transition Matrices

Given a finite state space X and transition matrix P, define $(P^k)_{k \geqslant 0}$ by $P^{k+1} = PP^k$ for all k, with the understanding that $P^0 = I$ = the identity matrix. In other words, for each k, the matrix P^k is the k-th power of P. If we spell out the matrix product $P^{k+1} = PP^k$ element-by-element, we get

$$P^{k+1}(x, x') := \sum_{z} P(x, z)P^k(z, x') \qquad (x, x' \in \mathsf{X}, \ k \in \mathbb{N}). \tag{2.6}$$

```julia
using Distributions, IterTools, QuantEcon

function create_inventory_model(; S=100,  # Order size
                                  s=10,   # Order threshold
                                  p=0.4)  # Demand parameter

    ϕ = Geometric(p)
    h(x, d) = max(x - d, 0) + S*(x <= s)
    return (; S, s, p, ϕ, h)
end

"Simulate the inventory process."
function sim_inventories(model; ts_length=200)
    (; S, s, p, ϕ, h) = model
    X = Vector{Int32}(undef, ts_length)
    X[1] = S  # Initial condition
    for t in 1:(ts_length-1)
        X[t+1] = h(X[t], rand(ϕ))
    end
    return X
end

"Compute the transition probabilities and state."
function compute_mc(model; d_max=100)
    (; S, s, p, ϕ, h) = model
    n = S + s + 1  # Size of state space
    state_vals = collect(0:(S + s))
    P = Matrix{Float64}(undef, n, n)
    for (i, j) in product(1:n, 1:n)
        P[i, j] = sum((h(i-1, d) == j-1)*pdf(ϕ, d) for d in 0:d_max)
    end
    return MarkovChain(P, state_vals)
end

"Compute the stationary distribution of the model."
function compute_stationary_dist(model)
    mc = compute_mc(model)
    return mc.state_values, stationary_distributions(mc)[1]
end
```

Listing 7: An implementation of S-s inventory dynamics (`inventory_sim.jl`)

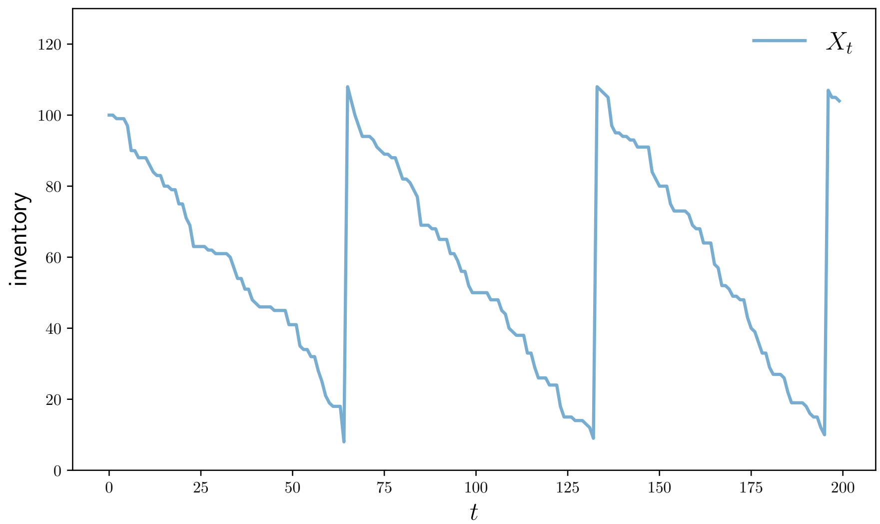


Figure 2.1: Inventory simulation (`inventory_sim.jl`)

Since the set of stochastic matrices is closed under multiplication (Exercise 2.1.4), P^k is a stochastic matrix on X for all $k \in \mathbb{N}$.

In this context, P^k is called the **k-step transition matrix** corresponding to P. The k-step transition matrix has the following interpretation: If (X_t) is P-Markov, then, for any $t, k \in \mathbb{N}$ and $x, x' \in \mathsf{X}$,

$$P^k(x, x') = \mathbb{P}\{X_{t+k} = x' \,|\, X_t = x\}. \tag{2.7}$$

In other words, P^k provides the k-step transition probabilities for the P-Markov chain (X_t), as suggested by its name.

This claim can be verified by induction. Fix $t \in \mathbb{N}$ and $x, x' \in \mathsf{X}$. The claim is true by definition when $k = 1$. Suppose the claim is also true at k and now consider the case $k + 1$. By the law of total probability, we have

$$\mathbb{P}\{X_{t+k+1} = x' \,|\, X_t = x\} = \sum_z \mathbb{P}\{X_{t+k+1} = x' \,|\, X_{t+k} = z\} \mathbb{P}\{X_{t+k} = z \,|\, X_t = x\}.$$

The induction hypothesis allows us to use (2.7), so the last equation becomes

$$\mathbb{P}\{X_{t+k+1} = x' \,|\, X_t = x\} = \sum_z P(z, x') P^k(x, z) = P^{k+1}(x, x').$$

This completes our proof by induction.

We can now give the following useful characterization of irreducibility:

Lemma 2.1.4. *Let P be a stochastic matrix on X. The following statements are equivalent:*

(i) *P is irreducible.*

(ii) *For any P-chain (X_t) and any $x, y \in \mathsf{X}$, there exists a $k \geqslant 0$ such that*

$$\mathbb{P}\{X_{t+k} = y \mid X_t = x\} > 0.$$

In other words, irreducibility of P is equivalent to the statement that P-chains eventually visit any state from any other state with positive probability.

Proof of Lemma 2.1.4. Let P be a stochastic matrix on X and let $P^k(x, x')$ denote the (x, x')-th element of P^k. Recall that P is irreducible if and only if $\sum_{k \geqslant 0} P^k \gg 0$. This is equivalent to the statement that, for each $(x, x') \in \mathsf{X} \times \mathsf{X}$, there exists a $k \geqslant 0$ such that $P^k(x, x') > 0$, which is in turn equivalent to part (ii) of Lemma 2.1.4. □

Exercise 2.1.6. Using Lemma 2.1.4, prove that the stochastic matrix associated with the S-s inventory dynamics in §2.1.1.4 is irreducible.

2.1.1.6 Stationarity and Ergodicity

Fix a stochastic matrix P on X and let (X_t) be a P-chain. Let ψ_t be the distribution of X_t for all t. For each $t \geqslant 0$, these distributions obey the recursion

$$\psi_{t+1}(x') = \sum_{x \in \mathsf{X}} P(x, x') \psi_t(x) \quad \text{for all } x \in \mathsf{X}. \tag{2.8}$$

This just states that

$$\mathbb{P}\{X_{t+1} = x'\} = \sum_{x \in \mathsf{X}} \mathbb{P}\{X_{t+1} = x' \mid X_t = x\} \mathbb{P}\{X_t = x\}$$

for all $x \in \mathsf{X}$, which is true by the law of total probability. Using matrix algebra, with each ψ_t regarded as a row vector, (2.8) can also be written as $\psi_{t+1} = \psi_t P$. Iterating on this equation, we get $\psi_t = \psi_0 P^t$ for all t. In summary,

$$(X_t)_{t \geqslant 0} \text{ is } P\text{-Markov with } X_0 \stackrel{d}{=} \psi_0 \implies X_t \stackrel{d}{=} \psi_0 P^t \text{ for all } t \geqslant 0. \tag{2.9}$$

Note 2.1.1. The fundamental relation $\psi_{t+1} = \psi_t P$ and the result (2.9) require that each ψ_t is a row vector. In what follows, we always treat marginal distributions of $(X_t)_{t \geqslant 0}$ as row vectors.

Consistent with our definition of stationary distributions in §2.1.1.2, a distribution $\psi^* \in \mathscr{D}(\mathsf{X})$ is called **stationary** for P if

$$\sum_{x \in \mathsf{X}} P(x, x')\psi^*(x) = \psi^*(x') \quad \text{for all } x \in \mathsf{X}.$$

Since distributions are regarded as row vectors, we can write this expression more simply as $\psi^* P = \psi^*$. In view of (2.8), if ψ^* is stationary and X_t has distribution ψ^*, then so does X_{t+1}, and hence X_{t+k} for all $k \geqslant 1$.

The next result states some key facts concerning the asymptotic properties of Markov chains.

Theorem 2.1.5. *If P is a stochastic matrix on a finite set X, then P has at least one stationary distribution ψ^*. If P is irreducible, then that stationary distribution is unique in $\mathscr{D}(\mathsf{X})$ and, moreover, for any P-Markov chain (X_t) and any $y \in \mathsf{X}$, we have*

$$\mathbb{P}\left\{\lim_{k \to \infty} \frac{1}{k} \sum_{t=0}^{k-1} \mathbb{1}\{X_t = y\} = \psi^*(y)\right\} = 1. \tag{2.10}$$

Property (2.10) tells us that, with probability one (i.e., for almost every P-Markov chain that we generate), the fraction of time that the chain spends in any given state is, in the limit, equal to the probability assigned to that state by the stationary distribution. Markov chains with this property are sometimes said to be **ergodic**.

The existence and uniqueness claims in Theorem 2.1.5 follow directly from Exercise 2.1.4, since ψ^* is stationary for P if and only if $\psi^* \in \mathscr{D}(X)$ and $\psi^* = \psi^* P$. A proof of (2.10) can be found in Brémaud (2020).

Since the S-s inventory model from §2.1.1.4 is irreducible, the uniqueness and ergodicity results from Theorem 2.1.5 apply. In particular, the model has only one stationary distribution ψ^* in $\mathscr{D}(\mathsf{X})$, where $\mathsf{X} = \{0, \ldots, S + s\}$, and (2.10) is valid whenever (X_t) is generated by the model. Figure 2.2 illustrates this by plotting both the stationary distribution ψ^* (which is computed using the code in Listing 7), and the value $m(y) := \frac{1}{k} \sum_{t=0}^{k-1} \mathbb{1}\{X_t = y\}$ at each $y \in \mathsf{X}$. The value of k is set to $1,000,000$. As predicted by the theorem, the fraction of time spent by the chain in each state is close to the probability assigned by ψ^*.

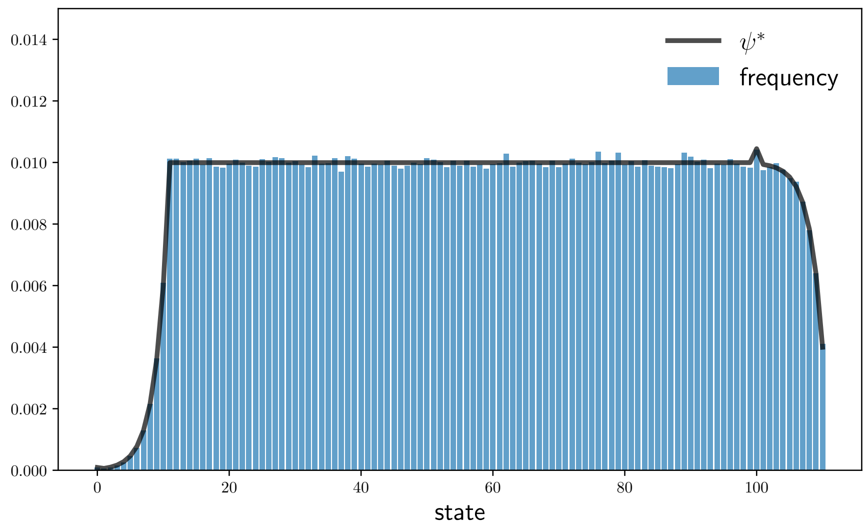


Figure 2.2: Ergodicity (`inventory_sim.jl`)

EXERCISE 2.1.7. Let (X_t) be P-Markov on X with $X_0 \stackrel{d}{=} \psi_0$. Show that

$$\mathbb{E}h(X_t) = \psi_0 P^t h = \langle \psi_0 P^t, h \rangle \quad \text{for all } t \in \mathbb{N}. \tag{2.11}$$

2.1.1.7 Application: Employment and Unemployment

Suppose that a day laborer is either unemployed ($X_t = 1$) or employed ($X_t = 2$) in each period. In state 1 he is hired with probability α. In state 2 he is fired with probability β. The corresponding state space and transition matrix are

$$\mathsf{X} = \{1, 2\} \quad \text{and} \quad P_w = \begin{pmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{pmatrix}.$$

Listing 8 provides a function to perform one update, from X_t to X_{t+1}, using the fact that `rand()` generates a draw from the uniform distribution on $(0, 1]$.

EXERCISE 2.1.8. Explain why Listing 8 updates the current state according to the probabilities in P_w.

One can show that, under the condition $\alpha + \beta > 0$, the stochastic matrix P_w has the

```
function laborer_update(x, α=0.3, β=0.2)
    if x == 1  # Unemployed case
        if rand() < α
            x = 2
        end
    else       # Employed case
        if rand() < β
            x = 1
        end
    end
    return x
end
```

Listing 8: Updating the state of the day laborer (`laborer_sim.jl`)

unique stationary distribution

$$\psi^* = \frac{1}{\alpha + \beta} \begin{pmatrix} \beta & \alpha \end{pmatrix} \tag{2.12}$$

in $\mathscr{D}(\mathsf{X})$ and, moreover, that $\psi P^t \to \psi^*$ as $t \to \infty$ for any $\psi \in \mathscr{D}(\mathsf{X})$. (One way to prove this is to show that P is irreducible aperiodic and then appeal to an extended version of the Perron–Frobenius theorem. See, for example, Chapter 4 of Sargent and Stachurski (2022).)

Exercise 2.1.9. Show algebraically (i.e., with pencil and paper) that ψ^* is indeed stationary for P_w.

Exercise 2.1.10. Fix $\alpha = 0.3$ and $\beta = 0.2$. Compute the sequence (ψP^t) for different choices of ψ and confirm that your results are consistent with the claim that $\psi P^t \to \psi^*$ as $t \to \infty$ for any nonnegative row vector ψ that sums to unity.

It is also true that, under the condition $\alpha + \beta > 0$, this Markov chain is ergodic, since P is irreducible in this case.

Exercise 2.1.11. Simulate a long realization Markov of a P_w-Markov chain from an arbitrary initial condition and confirm that your results are consistent with (2.10).

2.1.2 Approximation

It can be helpful to reduce continuous state Markov models to finite state models in order to simplify numerical calculations. The most common targets for this form of discretization are the linear Gaussian models, which we discuss in detail in §6.1.1. Here we review the one-dimensional case, where $(X_t)_{t \geqslant 0}$ evolves in $\mathbb{R}$ according to

$$X_{t+1} = \rho X_t + b + \nu \varepsilon_{t+1}, \quad (\varepsilon_t) \overset{\text{IID}}{\sim} N(0,1). \tag{2.13}$$

This is a **linear Gaussian AR(1)** model. Here we discuss one technique for discretizing (2.13), often called **Tauchen's method**, and use it to illustrate concepts related to stationarity.

We assume throughout that $|\rho| < 1$. Under this assumption, (2.13) has a unique **stationary distribution** ψ^* given by

$$\psi^* = N(\mu_x, \sigma_x^2) \quad \text{with} \quad \mu_x := \frac{b}{1-\rho} \quad \text{and} \quad \sigma_x^2 := \frac{\nu^2}{1-\rho^2}.$$

This means that ψ^* has the following property:

$$X_t \overset{d}{=} \psi^* \text{ and } X_{t+1} = \rho X_t + b + \nu \varepsilon_{t+1} \text{ implies } X_{t+1} \overset{d}{=} \psi^*.$$

EXERCISE 2.1.12. Suppose that $X_t \overset{d}{=} \psi^*$, $\varepsilon_{t+1} \overset{d}{=} N(0,1)$ and X_t and ε_{t+1} are independent. Prove that $\rho X_t + b + \nu \varepsilon_{t+1}$ has distribution ψ^*. Is this always true if we drop the independence assumption made above?

When $|\rho| < 1$, this model is ergodic in a similar sense to (2.10) on page 56: on average, realizations of the process spend most of their time in regions of the state where the stationary distribution puts high probability mass. (You can check this via simulations if you wish.) Hence, in the discretization that follows, the discrete state space will be centered in this area.

EXERCISE 2.1.13. Set $b = 0$ in (2.13) and let F be the $N(0, \nu^2)$ CDF. Show that

$$\mathbb{P}\{t - \delta < X_{t+1} \leqslant t + \delta \,|\, X_t = x\} = F(t - \rho x + \delta) - F(t - \rho x - \delta) \tag{2.14}$$

for all $\delta, t \in \mathbb{R}$.

We start with the case $b = 0$. As a first step, we choose n as the number of states for the discrete approximation and m as an integer that parameterizes the width of the state space. Then we create a state space $\mathsf{X} := \{x_1, \ldots, x_n\} \subset \mathbb{R}$ as an linear grid that brackets the stationary mean on both sides by m standard deviations:

- set $x_1 = -m\,\sigma_x$,
- set $x_n = m\,\sigma_x$ and
- set $x_{i+1} = x_i + s$ where $s = (x_n - x_1)/(n-1)$ and i in $[n-1]$.

The next step is to create an $n \times n$ matrix P computed to approximate the dynamics in (2.13). For $i, j \in [n]$,

(i) if $j = 1$, then set $P(x_i, x_j) = F(x_1 - \rho x_i + s/2)$.

(ii) If $j = n$, then set $P(x_i, x_j) = 1 - F(x_n - \rho x_i - s/2)$.

(iii) Otherwise, set $P(x_i, x_j) = F(x_j - \rho x_i + s/2) - F(x_j - \rho x_i - s/2)$.

The first two are boundary rules and the third applies Exercise 2.1.13.

EXERCISE 2.1.14. Prove that $\sum_{j=1}^{n} P(x_i, x_j) = 1$ for all $i \in [n]$.

Finally, if $b \neq 0$, then we shift the state space to center it on the mean μ_x of the stationary distribution $N(\mu_x, \sigma_x^2)$. This is done by replacing x_i with $x_i + \mu_x$ for each i.

Julia routines for computing X and P can be found in the library QuantEcon.jl.

Figure 2.3 compares the continuous stationary distribution ψ^* and the unique stationary distribution of the discrete approximation when X and P are constructed as above, under the parameterization $\rho = 0.9$, $b = 0.0$, $\nu = 1.0$. The discretization parameters were set to $n = 15$ and $m = 3$.

2.1.3 Expectations

In this section we discuss how to take conditional expectations with respect to Markov chains. The theory will be essential for the study of finite MDPs, since, in these models, lifetime rewards are expectations of flow reward functions with respect to Markov chains.

2.1.3.1 Conditional Expectations

Let P be any stochastic matrix on $\mathsf{X} \times \mathsf{X}$. For each $h \in \mathbb{R}^{\mathsf{X}}$, we define

$$(Ph)(x) = \sum_{x' \in \mathsf{X}} h(x') P(x, x') \qquad (x \in \mathsf{X}). \tag{2.15}$$

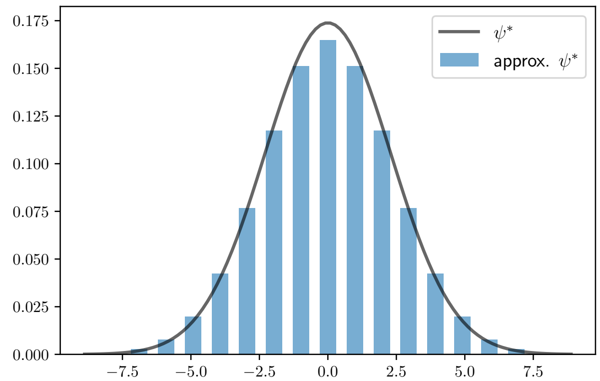


Figure 2.3: Comparison of $\psi^* = N(\mu_x, \sigma_x^2)$ and its discrete approximant

Noting that $P(x, \cdot)$ is the distribution of X_{t+1} given $X_t = x$, we can equivalently write

$$(Ph)(x) = \mathbb{E}[h(X_{t+1}) \mid X_t = x], \tag{2.16}$$

where (X_t) is any P-Markov chain on X.

In terms of matrix algebra, viewing h has an $n \times 1$ column vector, the expression $(Ph)(x)$ is one element of the vector Ph obtained by premultiplying h by P. Some authors refer to the map $h \mapsto Ph$ as the **backward operator**, by analogy with the Kolmogorov backward equation (which arises in continuous time Markov process theory).

The interpretation in (2.16) extends to powers of P. In particular, we have

$$(P^k h)(x) = \sum_{x' \in \mathsf{X}} h(x') P^k(x, x') = \mathbb{E}[h(X_{t+k}) \mid X_t = x]. \tag{2.17}$$

EXERCISE 2.1.15. Show that

(i) Every constant function $h \in \mathbb{R}^{\mathsf{X}}$ is a fixed point of P (i.e., $Ph = h$).

(ii) $\max_x |Ph(x)| \leqslant \max_x |h(x)|$ for all $h \in \mathbb{R}^{\mathsf{X}}$.

EXERCISE 2.1.16. Given Markov matrix P and constant $\varepsilon > 0$, prove the following result: There exists no $h \in \mathbb{R}^{\mathsf{X}}$ with $Ph \geqslant h + \varepsilon$.

2.1.3.2 The Law of Iterated Expectations

The **law of iterated expectations** appears time and again in dynamic modeling, particularly in economics and finance. One common version of the law is that if X and Y are two random variables, then $\mathbb{E}[\mathbb{E}[Y \mid X]] = \mathbb{E}[Y]$. Let's show this in the Markov case when predicting future values.

Let (X_t) be P-Markov with $X_0 \stackrel{d}{=} \psi_0$. Fix $t, k \in \mathbb{N}$. Set $\mathbb{E}_t := \mathbb{E}[\cdot \mid X_t]$. We claim that

$$\mathbb{E}[\mathbb{E}_t[h(X_{t+k})]] = \mathbb{E}[h(X_{t+k})] \quad \text{for any } h \in \mathbb{R}^{\mathsf{X}}. \tag{2.18}$$

To see this, recall that $\mathbb{E}[h(X_{t+k}) \mid X_t = x] = (P^k h)(x)$. Hence $\mathbb{E}[h(X_{t+k}) \mid X_t] = (P^k h)(X_t)$. Therefore,

$$\mathbb{E}[\mathbb{E}_t[h(X_{t+k})]] = \mathbb{E}[(P^k h)(X_t)] = \sum_{x'} (P^k h)(x')\psi_t(x') = \sum_{x'} (P^k h)(x')(\psi_0 P^t)(x').$$

Since $\psi_0 P^t$ is a row vector, we can write the last expression as

$$\left\langle \psi_0 P^t, P^k h \right\rangle = \psi_0 P^t P^k h = \psi_0 P^{t+k} h.$$

Connecting the final expression with (2.9), we see that the right-hand side is precisely $\mathbb{E} h(X_{t+k})$. Hence (2.18) holds.

2.1.4 Monotonicity

Let X be a finite set partially ordered by $\preceq$. In what follows, $i\mathbb{R}^{\mathsf{X}}$ is the set of increasing functions in $\mathbb{R}^{\mathsf{X}}$. Thus, for $h \in \mathbb{R}^{\mathsf{X}}$,

$$h \in i\mathbb{R}^{\mathsf{X}} \quad \iff \quad x, y \in \mathsf{X} \text{ and } x \preceq y \text{ implies } h(x) \leqslant h(y).$$

Example 2.1.1. If $\mathsf{X} = \{1, \ldots, n\}$ and $\preceq$ is the usual order $\leqslant$ on $\mathbb{R}$, then $x \mapsto 2x$ and $x \mapsto \mathbb{1}\{2 \leqslant x\}$ are in $i\mathbb{R}^{\mathsf{X}}$ but $x \mapsto -x$ and $x \mapsto \mathbb{1}\{x \leqslant 2\}$ are not.

The next exercise shows that an increasing function can be represented as the sum of increasing binary functions. This fact will be valuable when we characterize orders over distributions, in §2.1.4.1.

EXERCISE 2.1.17. Let $\mathsf{X} = \{x_1, \ldots, x_n\}$ where $x_k \preceq x_{k+1}$ for all k. Show that, for any $u \in i\mathbb{R}^{\mathsf{X}}$, there exist $s_1, \ldots, s_n$ in $\mathbb{R}_+$ such that $u(x) = \sum_{k=1}^{n} s_k \mathbb{1}\{x \geqslant x_k\}$ for all $x \in \mathsf{X}$.

2.1.4.1 Stochastic Dominance

Given $\varphi, \psi \in \mathscr{D}(\mathsf{X})$, we say that ψ **stochastically dominates** φ and write $\varphi \preceq_{\mathrm{F}} \psi$ if

$$\sum_x u(x)\varphi(x) \leqslant \sum_x u(x)\psi(x) \quad \text{for every } u \text{ in } i\mathbb{R}^{\mathsf{X}}$$

The relation $\preceq_{\mathrm{F}}$ is also called **first order stochastic dominance** to differentiate it from other forms of stochastic order. (We discuss these extensions in §**??**.)

One way to understand the definition of first order stochastic dominance is as follows: Suppose we have an agent whose preferences over outcomes in X are determined by a utility function $u \in \mathbb{R}^{\mathsf{X}}$. Suppose in addition that the agent prefers more to less, in the sense that $u \in i\mathbb{R}^{\mathsf{X}}$, and that the agent ranks lotteries over X according to expected utility. In other words, the agent evaluates $\varphi \in \mathscr{D}(\mathsf{X})$ according to $\sum_x u(x)\varphi(x)$. Then the agent (weakly) prefers ψ to φ whenever $\varphi \preceq_{\mathrm{F}} \psi$.

We can go one step further. Consider now the class $\mathscr{A}$ of *all* agents who (a) have preferences over outcomes in X, (b) prefer more to less, and (c) rank lotteries over X according to expected utility. Then $\varphi \preceq_{\mathrm{F}} \psi$ if and only if every agent in $\mathscr{A}$ prefers ψ to φ.

Remark 2.1.1. The last paragraph helps illustrate the significance of stochastic dominance in economics. It is standard to assume that economic agents have increasing utility functions and use expected utility to evaluate lotteries. In such an environment, a policy maker who can engineer an upward shift in a lottery, as measured by stochastic dominance, will make all agents better off. Such a change is unambiguously welfare enhancing.

EXERCISE 2.1.18. The simplest setting in which we can study stochastic dominance is where $\mathsf{X} = \{1, 2\}$ and X is partially ordered by $\leqslant$. In this case, $\varphi \preceq_{\mathrm{F}} \psi$ if and only φ puts more mass on 1 than ψ, and, equivalently, less mass on 2. That is,

$$\varphi \preceq_{\mathrm{F}} \psi \iff \psi(1) \leqslant \varphi(1) \iff \varphi(2) \leqslant \psi(2).$$

Verify the equivalence of these statements.

There is another way to represent stochastic dominance that can be easier to visualize. To state it, we first introduce the notation

$$G^{\varphi}(y) := \sum_{x \geqslant y} \varphi(x) \qquad (\varphi \in \mathscr{D}(\mathsf{X}),\ y \in \mathsf{X}).$$

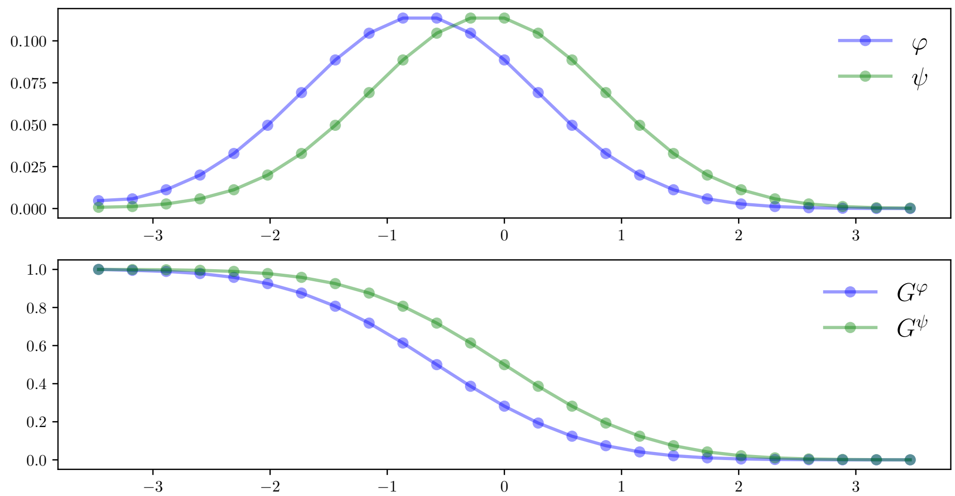


Figure 2.4: Visualization of $\varphi \preceq_{\mathrm{F}} \psi$

For a given distribution φ, the function G^{φ} is sometimes called the **counter CDF** (counter cumulative distribution function) of φ.

Lemma 2.1.6. *For each $\varphi, \psi \in \mathscr{D}(\mathsf{X})$, the following statements hold:*

(i) $\varphi \preceq_F \psi \implies G^{\varphi} \leqslant G^{\psi}$.

(ii) *If X is totally ordered by $\preceq$, then $G^{\varphi} \leqslant G^{\psi} \implies \varphi \preceq_F \psi$.*

The proof is given below. Figure 2.4 helps to illustrate. Here $\mathsf{X} \subset \mathbb{R}$ and φ and ψ are distributions on X. We can see that $\varphi \preceq_{\mathrm{F}} \psi$ because the counter CDFs are ordered, in the sense that $G^{\varphi} \leqslant G^{\psi}$ pointwise on X.

Proof of Lemma 2.1.6. Regarding (i), fix $\varphi, \psi \in \mathscr{D}(\mathsf{X})$ with $\varphi \preceq_{\mathrm{F}} \psi$. Pick any $y \in \mathsf{X}$. By transitivity of partial orders, the function $u(x) := \mathbb{1}\{y \preceq x\}$ is in $i\mathbb{R}^{\mathsf{X}}$. Hence $\sum_x u(x)\varphi(x) \leqslant \sum_x u(x)\psi(x)$. Given the definition of u, this is equivalent to $G^{\varphi}(y) \leqslant G^{\psi}(y)$. As y was chosen arbitrarily, we have $G^{\varphi} \leqslant G^{\psi}$ pointwise on X.

Regarding (ii), let $\varphi, \psi \in \mathscr{D}(\mathsf{X})$ be such that $G^{\varphi} \leqslant G^{\psi}$ and let X be totally ordered by $\preceq$. We can write X as $\{x_1, \ldots, x_n\}$ with $x_i \preceq x_{i+1}$ for all i. Pick any $u \in i\mathbb{R}^{\mathsf{X}}$ and let $\alpha_i = u(x_i)$. By Exercise 2.1.17, we can write u as $u(x) = \sum_{i=1}^{n} s_i \mathbb{1}\{x \succeq x_i\}$ at each $x \in \mathsf{X}$, where $s_i \geqslant 0$ for all i. Hence

$$\sum_{x \in \mathsf{X}} u(x)\varphi(x) = \sum_{x \in \mathsf{X}} \sum_{i=1}^{n} s_i \mathbb{1}\{x \succeq x_i\}\varphi(x) = \sum_{i=1}^{n} s_i \sum_{x \in \mathsf{X}} \mathbb{1}\{x \succeq x_i\}\varphi(x) = \sum_{i=1}^{n} s_i\, G^{\varphi}(x_i).$$

A similar argument gives $\sum_{x \in \mathsf{X}} u(x)\psi(x) = \sum_{i=1}^{n} s_i G^{\psi}(x_i)$. Since $G^{\varphi} \leqslant G^{\psi}$, we have

$$\sum_{x \in \mathsf{X}} u(x)\varphi(x) = \sum_{i=1}^{n} s_i G^{\varphi}(x_i) \leqslant \sum_{i=1}^{n} s_i G^{\psi}(x_i) = \sum_{x \in \mathsf{X}} u(x)\psi(x).$$

We conclude that $\varphi \preceq_{\mathrm{F}} \psi$, as was to be shown. □

Exercise 2.1.19. Prove that $\preceq_{\mathrm{F}}$ is a partial order on $\mathscr{D}(\mathsf{X})$.

2.1.4.2 Monotone Markov Chains

A stochastic matrix P on $\mathsf{X} \times \mathsf{X}$ is called **monotone increasing** if

$$x, y \in \mathsf{X} \text{ and } x \preceq y \quad \implies \quad P(x, \cdot) \preceq_{\mathrm{F}} P(y, \cdot).$$

In other words, P is monotone increasing if shifting up the current state shifts up the next period state, in the sense that its distribution increases in the stochastic dominance ordering on $\mathscr{D}(\mathsf{X})$. Below, we will see that monotonicity of stochastic matrices is closely related to monotonicity in value functions in some important applications.

Monotonicity in stochastic matrices is related to positive autocorrelation. To illustrate the idea, consider the AR(1) model $X_{t+1} = \rho X_t + \sigma \varepsilon_{t+1}$ from §2.1.2 and suppose we apply Tauchen discretization, mapping the parameters ρ, σ and a discretization size n into an $n \times n$ stochastic matrix P on state space $\mathsf{X} = \{x_1, \ldots, x_n\} \subset \mathbb{R}$. If $\rho \geqslant 0$, so that positive autocorrelation holds, then P is monotone increasing.

Exercise 2.1.20. Verify this claim.

Exercise 2.1.21. In §2.1.1.7 we discussed a setting where

$$\mathsf{X} = \{1, 2\} \quad \text{and} \quad P_w = \begin{pmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{pmatrix}$$

for some $\alpha, \beta \in [0, 1]$. Show that P_w is monotone increasing if and only if $\alpha + \beta \leqslant 1$.

Exercise 2.1.22. Prove that P is monotone increasing if and only if P is invariant on $i\mathbb{R}^{\mathsf{X}}$; that is, if $h \in i\mathbb{R}^{\mathsf{X}}$ implies $Ph \in i\mathbb{R}^{\mathsf{X}}$.

Exercise 2.1.23. Prove: If P is monotone increasing then so is P^t for all $t \in \mathbb{N}$.

2.2 Valuations and Forecasts

Roadmap.

2.2.1 Fixed Discount Rates

A common task in Markov settings is computing the expectation of a discounted sum of future measurements. These sums take the form

$$v(x) := \mathbb{E}_x \sum_{t \geqslant 0} \beta^t h(X_t) := \mathbb{E}\left[\sum_{t \geqslant 0} \beta^t h(X_t) \,|\, X_0 = x\right] \tag{2.19}$$

for some constant $\beta \in \mathbb{R}_+$ and $h \in \mathbb{R}^{\mathsf{X}}$, where (X_t) is P-Markov on finite set X. Here $\mathbb{E}_x$ indicates we are conditioning on $X_0 = x$.

Example 2.2.1. Suppose (X_t) represents business conditions, $(h(X_t))_{t \geqslant 0}$ is a given cash flow and β is a discount factor associated with a given discount rate. Then $v(x)$ in (2.19) is the expected present value of this cash flow.

Lemma 2.2.1. *If (X_t) is P-Markov and $\beta \in (0, 1)$, then $v(x)$ is finite for all $x \in \mathsf{X}$, the matrix $I - \beta P$ is invertible and the vector v obeys*

$$v = \sum_{t \geqslant 0} \beta^t (P^t h) = (I - \beta P)^{-1} h. \tag{2.20}$$

Proof. Under the stated conditions we have

$$\mathbb{E}_x \sum_{t \geqslant 0} \beta^t h(X_t) = \sum_{t \geqslant 0} \beta^t \mathbb{E}_x h(X_t) = \sum_{t \geqslant 0} \beta^t (P^t h)(x). \tag{2.21}$$

The last equality in (2.21) follows from (2.17) and the assumption that (X_t) is P-Markov starting at x.[2] Now observe that $\sum_{t \geqslant 0} (\beta P)^t = (I - \beta P)^{-1}$ by the Neumann Series Lemma (p. 12) applied to the matrix βP. The lemma is applicable because $r(\beta P) = \beta r(P) = \beta < 1$, as follows from Exercise 2.1.4. □

[2] In general, care must be taken when pushing mathematical expectations through sums (as in the first equality) whenever the sums are infinite. In the present setting, justification can be provided by appealing to the dominated convergence theorem, which is one of the fundamental results of measure theory. Such discussions are deferred until later in the text.

2.2.2 Application: Valuation of Firms

Consider a firm that receives profit stream $(\pi_t)_{t \geqslant 0}$. For a shareholder, the total valuation of the firm is the expected present of its profit stream. In this section we investigate how to compute this valuation under different hypotheses.

2.2.2.1 Fixed Interest Rates

Suppose first that the interest rate is constant at $r > 0$. With $\beta := 1/(1+r)$, total valuation is

$$V_0 = \mathbb{E} \sum_{t=0}^{\infty} \beta^t \pi_t. \tag{2.22}$$

To compute this value, we need a model of how profits will evolve. A common strategy is to set $\pi_t = \pi(X_t)$ for some fixed $\pi \in \mathbb{R}^{\mathsf{X}}$, where $(X_t)_{t \geqslant 0}$ is a state process. After the function π and the dynamics of (X_t) have been estimated, the value V_0 in (2.22) can be computed.

Here we assume that (X_t) is P-Markov for some stochastic matrix P defined on a finite set X. Then, conditioning on $X_0 = x$, we can write the value as

$$v(x) := \mathbb{E}_x \sum_{t=0}^{\infty} \beta^t \pi_t := \mathbb{E} \left[\sum_{t=0}^{\infty} \beta^t \pi_t \,|\, X_0 = x \right].$$

By Lemma 2.2.1 on page 66, the value $v(x)$ is finite and the function $v \in \mathbb{R}^{\mathsf{X}}$ can be obtained by

$$v = \sum_{t \geqslant 0} \beta^t P^t \pi = (I - \beta P)^{-1} \pi.$$

It seems natural that valuation will be increasing if higher states generate higher profits and also predict higher states in the future. The next exercise confirms this.

Exercise 2.2.1. Let X be partially ordered and suppose that $h \in i\mathbb{R}^{\mathsf{X}}$ and that P is monotone increasing. (See §2.1.4 for terminology and notation.) Prove that, under these conditions, v is increasing on X.

2.2.2.2 Time-Varying Interest Rates

One of the limitations of preceding discussion is that discounting is constant and geometric. A quick look at the data shows that this assumption is problematic. Interest

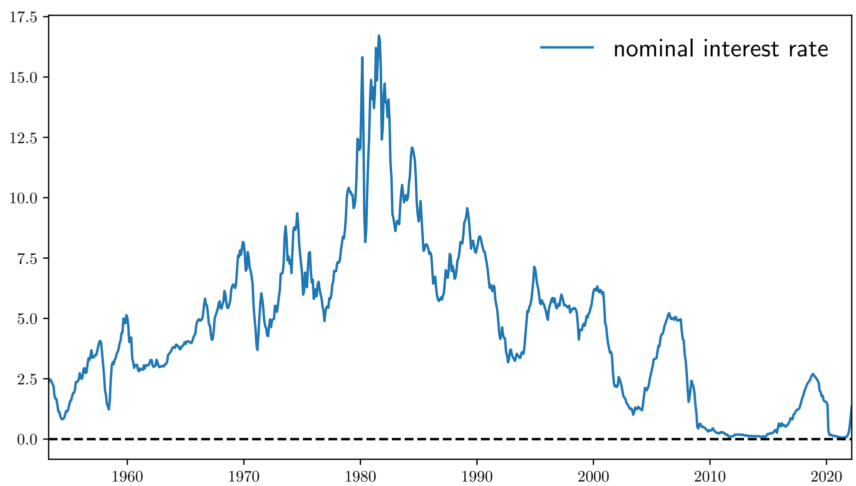


Figure 2.5: Nominal US interest rates (`plot_interest_rates_nominal.jl`)

rates are stochastic and time-varying, even for safe assets like US Treasury bills. To illustrate this, Figure 2.5 shows the nominal interest rate on 1 Year Treasury bills since the 1950s, while Figure 2.6 shows dynamics of the real interest rate for 10 year T-bills since 2012. Clearly both the nominal and the real interest rate are significantly time varying.

Whether a given firm's profit stream should be discounted by nominal or real interest rates depends on the costs and revenue stream of the firm, and how closely they co-move with inflation. Either way, firm valuations respond to anticipated time paths for interest rates.

Example 2.2.2. When a period of rising interest rates is anticipated by the market, the share prices of newer and more technology-heavy firms typically face strong headwinds. This is because the profit streams from such firms are usually biased towards the future, in the sense that dividends are initially low or zero (while profits are reinvested) and eventually high (if the business model is successful). A period of rising interest rates indicates that such profit streams should be heavily discounted.

With this motivation, let us consider an extension of the firm valuation problem where the interest rate is permitted to follow a stochastic process $(r_t)_{t \geqslant 0}$. Under the convention that the interest rate over the period between t and $t+1$ is known at time t and written as r_t, the time zero expected present value of time t profit π_t is

$$\mathbb{E}\left\{\beta_0 \cdots \beta_{t-1} \cdot \pi_t\right\} \qquad \text{where } \beta_t := \frac{1}{1+r_t}.$$

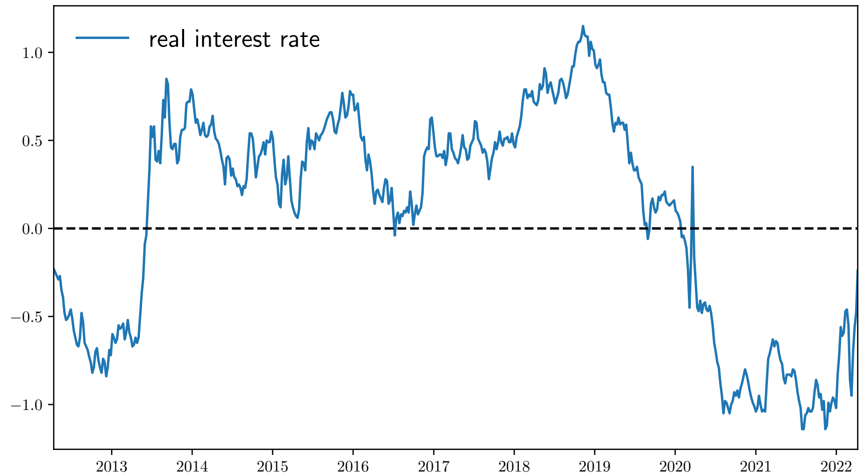


Figure 2.6: Real US interest rates (`plot_interest_rates_real.jl`)

The expected present value of the firm is

$$V_0 = \mathbb{E}\left\{\sum_{t=0}^{\infty}\left[\prod_{i=0}^{t-1}\beta_i\right]\pi_t\right\} \qquad \text{where } \prod_{i=0}^{-1}\beta_i := 1. \tag{2.23}$$

To simplify the problem, we suppose that $\beta_t = \beta(X_t)$ for some $\beta \in \mathbb{R}^{\mathsf{X}}$, so that randomness in interest rates is a function of the same Markov state that influences profits. There is very little loss of generality in making this assumption. (In fact, the two processes can still be completely independent. For example, if we take X_t to have the form $X_t = (Y_t, Z_t)$, where (Y_t) and (Z_t) are independent Markov chains, then we can take β_t to be a function of Y_t and π_t to be a function of Z_t. The resulting interest and profit processes are independent.)

Conditioning on $X_0 = x$, the value in (2.23) becomes

$$v(x) = \mathbb{E}\left\{\sum_{t=0}^{\infty}\left[\prod_{i=0}^{t-1}\beta(X_i)\right]\pi(X_t)\right\}. \tag{2.24}$$

Here are some immediate questions:

- Is $v(x)$ finite for all x?
- How should we compute the valuation function v?

In order to answer these and other questions, we present and prove a general result on geometric sums in the next section. Then we return to the firm valuation problem in §2.2.3.1 and answer the questions posed above.

2.2.3 Generalized Geometric Sums

Throughout this section, we work in the following setting:

- X is a finite set and P is a stochastic matrix on X.
- h is in $\mathbb{R}^{\mathsf{X}}$ and b is a map from $\mathsf{X} \times \mathsf{X}$ to $\mathbb{R}$.
- $(X_t)_{t \geqslant 0}$ is P-Markov, $H_t = h(X_t)$ and $B_t = b(X_{t-1}, X_t)$.
- K is the matrix on X defined by $K(x, x') := b(x, x')P(x, x')$.

Given $x \in \mathsf{X}$ we write $\mathbb{E}_x$ for $\mathbb{E}[\cdot \,|\, X_0 = x]$ and $\mathbb{E}_t$ for $\mathbb{E}[\cdot \,|\, X_t]$. With the convention $\prod_{i=1}^{0} B_i := 1$, we have the following key result.

Theorem 2.2.2. *If $r(K) < 1$, then the function v on X defined by*

$$v(x) = \mathbb{E}_x \left\{ \sum_{t=0}^{\infty} \left[\prod_{i=1}^{t} B_i \right] H_t \right\} \tag{2.25}$$

is finite-valued and is the only function in $\mathbb{R}^{\mathsf{X}}$ that satisfies the recursion

$$v(x) = h(x) + \sum_{x'} v(x') K(x, x') \quad \text{for all } x \in \mathsf{X}. \tag{2.26}$$

Moreover, $I - K$ is nonsingular and $v = (I - K)^{-1} h$.

Theorem 2.2.2 generalizes Lemma 2.2.1 on page 66. Indeed, if $b \equiv \beta \in (0, 1)$, then $r(K) = \beta r(P) = \beta < 1$, and the result in Theorem 2.2.2 reduces to Lemma 2.2.1.

To prove Theorem 2.2.2, we begin with the following result concerning expectations over products.

Lemma 2.2.3. *For each $t \in \mathbb{Z}_+$ and $x \in \mathsf{X}$, we have*

$$\mathbb{E}_x \left\{ \left[\prod_{i=1}^{t} B_i \right] h(X_t) \right\} = \sum_{x' \in \mathsf{X}} K^t(x, x') h(x'). \tag{2.27}$$

Proof. We verify the claim in Lemma 2.2.3 using induction on t. The claim holds at $t = 1$ because, for any such h and x,

$$\mathbb{E}_x [B_1 H_1] = \sum_{x'} b(x, x')h(x')P(x, x') = \sum_{x'} K(x, x')h(x').$$

Now suppose it holds at t. We claim it also holds at $t + 1$. To show this we apply the law of iterated expectations to obtain

$$\mathbb{E}_x [B_1 \cdots B_{t+1} H_{t+1}] = \mathbb{E}_x [\mathbb{E}_t [B_1 \cdots B_{t+1} H_{t+1}]] = \mathbb{E}_x [B_1 \cdots B_t \mathbb{E}_t [B_{t+1} H_{t+1}]] .$$

Since $\mathbb{E}_t B_{t+1} H_{t+1} = \sum_y b(X_t, y)h(y)P(X_t, y)$, we can now write

$$\mathbb{E}_x [B_1 \cdots B_{t+1} H_{t+1}] = \mathbb{E}_x [B_1 \cdots B_t f(X_t)] \quad \text{where} \quad f(x) := \sum_y K(x, y)h(y). \tag{2.28}$$

Applying the induction hypothesis (2.27) to the right-hand side of the first equation in (2.28) (with $h = f$), we now have

$$\mathbb{E}_x [B_1 \cdots B_{t+1} H_{t+1}] = \sum_{x'} K^t(x, x')f(x') = \sum_{x'} K^t(x, x') \sum_y K(x', y)h(y).$$

But $\sum_{x'} K^t(x, x')K(x', y) = K^{t+1}(x, y)$, so (2.27) holds at $t + 1$ as well. The proof is now complete. □

Now we can complete the proof of Theorem 2.2.2.

Proof of Theorem 2.2.2. We fix $x \in \mathsf{X}$ and use Lemma 2.2.3 to obtain

$$v(x) = \mathbb{E}_x \left\{ \sum_{t=0}^{\infty} \left[\prod_{i=1}^{t} B_i \right] H_t \right\} = \sum_{t=0}^{\infty} \mathbb{E}_x \left\{ \left[\prod_{i=1}^{t} B_i \right] H_t \right\} = \sum_{t=0}^{\infty} (K^t h)(x). \tag{2.29}$$

Writing (2.29) pointwise gives, $v = \sum_{t \geqslant 0} K^t h$.[3] By the Neumann series lemma and $r(K) < 1$, this sum converges and equals $(I - K)^{-1} h$. The recursive expression (2.26) follows from $v = (I - K)^{-1} h$, since premultiplying both sides by $I - K$ gives $v = h + Kv$. Finally, if w is an element of $\mathbb{R}^{\mathsf{X}}$ satisfying $w = h + Kw$, then, by the uniqueness component of the Neumann series lemma, $w = v$. In other words, v defined in (2.25) is the only function that satisfies the recursion (2.26). □

[3]In (2.29) we again passed expectations through an infinite sum. This operation takes some care but is valid under the assumption $r(K) < 1$. Footnote 2 on page 66 provides more information.

2.2.3.1 Back to the Firm Problem

Now let's return to the firm valuation problem and use Theorem 2.2.2 to answer the questions posed at the end of §2.2.2.2. In doing so we set

$$K(x, x') := \beta(x)P(x, x') \qquad ((x, x') \in \mathsf{X} \times \mathsf{X}).$$

Proposition 2.2.4. *If $r(K) < 1$, then vector v of state-contigent firm valuation in* (2.24) *is finite for all $x \in \mathsf{X}$ and satisfies*

$$v(x) = \pi(x) + \beta(x) \sum_{x'} v(x')P(x, x'). \tag{2.30}$$

Moreover, $v = (I - K)^{-1}\pi$.

Exercise 2.2.2. Verify Proposition 2.2.4 via Theorem 2.2.2.

The next exercise provides conditions under which valuation is increasing in x.

Exercise 2.2.3. Let X be partially ordered and assume $r(K) < 1$. Prove that v is in $i\mathbb{R}^{\mathsf{X}}$ whenever P is monotone increasing and $\beta, \pi \in i\mathbb{R}^{\mathsf{X}}$.

2.3 Application: Asset Pricing

In this section we provide a brief introduction to the standard theory of asset pricing in a Markov environment. The topic of asset pricing is fascinating in its own right. Here we include it mainly to provide additional practice in dealing with valuation problems. Readers who lack interest in asset pricing and wish to push ahead with their study of dynamic programming can safely jump to §2.4.

2.3.1 Introduction to Asset Pricing

In this section we first discuss risk-neutral pricing and then show why this assumption is typically implausible. Next, we introduce stochastic discount factors and stationary assest pricing.

2.3.1.1 Risk Neutral Pricing?

Consider the problem of assigning a current price Π_t^G to an asset that confers on its owner the right to payoff G_{t+1}. The payoff is stochastic and realized next period. One simple idea is to use the **risk neutral pricing**, which implies that

$$\Pi_t^G = \mathbb{E}_t \, \beta \, G_{t+1} \tag{2.31}$$

for some constant discount fact $\beta \in (0, 1)$. If the payoff is in k periods, then we modify the price to $\mathbb{E}_t \, \beta^k \, G_{t+k}$. In essence, risk neutral pricing says that cost equals expected reward, discounted to present value using a constant rate of discount.

Example 2.3.1. Let S_t be the price of a stock at each point in time t. A **European call option** gives its owner the right to purchase the stock at price K at time $t + k$. There is no obligation to exercise the option, so the payoff at $t + k$ is $\max\{S_{t+k} - K, 0\}$. Under risk neutral pricing, the time t price of this option is

$$\Pi_t^O = \mathbb{E}_t \, \beta^k \, \max\{S_{t+k} - K, 0\}.$$

Unfortunately, assuming risk neutrality for all investors is not appropriate for a general model of asset pricing. The most important reason is that such a model is inconsistent with the data.

To give one example, suppose that we take the asset that pays G_{t+1} in (2.31) and replace it with another asset that pays $H_{t+1} = G_{t+1} + \varepsilon_{t+1}$, where ε_{t+1} is independent of G_{t+1}, $\mathbb{E}_t \, \varepsilon_{t+1} = 0$ and $\operatorname{Var} \varepsilon_{t+1} > 0$. In effect, that we are adding risk to the original payoff without changing its mean.

Under risk neutrality, the price of this new asset is

$$\Pi_t^H = \mathbb{E}_t \, \beta \, [G_{t+1} + \varepsilon_{t+1}] = \Pi_t^G + \beta \, \mathbb{E}_t \, \varepsilon_{t+1} = \Pi_t^G.$$

Thus, H_{t+1} and G_{t+1} are priced identically, even though their means are both $\mathbb{E}_t G_{t+1}$ and their variances satisfy

$$\operatorname{Var} H_{t+1} = \operatorname{Var} G_{t+1} + \operatorname{Var} \varepsilon_{t+1} > \operatorname{Var} G_{t+1}.$$

This outcome contradicts the fact that, in asset markets, investors typically demand some compensation for bearing risk.

A helpful way to think about the same point is to consider the rate of return $r_{t+1} := (G_{t+1} - \Pi_t^G)/\Pi_t^G$ on holding an asset with payoff G_{t+1}. From (2.31) we have $\mathbb{E}_t \, \beta(1 +$

$r_{t+1}) = 1$, or

$$\mathbb{E}_t\, r_{t+1} = \frac{1-\beta}{\beta}.$$

Since the right-hand side does not depend on G_{t+1}, risk neutrality implies that all assets have the same expected rate of return. But this contradicts the fact that, on average, riskier assets tend to have higher rates of return—which are needed to incentivize investors to bear risk.

Example 2.3.2. The **risk premium** on a given asset is defined as the expected rate of return minus the rate of return on a risk-free asset. If we assume risk-neutrality then, by the preceding discussion, the risk premium is zero for all assets. However, calculations based on post-war US data show that the average risk premium for equities is around 8% per annum (see, e.g., Cochrane (2009)).

2.3.1.2 A Stochastic Discount Factor

To go beyond risk neutral pricing, let's start with a model containing one asset and one agent. Due to the simplicity of the model, we will find it straightforward to price the asset and compare it to the risk neutral case.

In the model, a representative agent takes the price Π_t of a risky asset as given and solves

$$\max_{0 \leqslant \alpha \leqslant 1} \{u(C_t) + \beta \mathbb{E}_t u(C_{t+1})\}$$
$$\text{subject to} \quad C_t = E_t - \Pi_t \alpha \quad \text{and} \quad C_{t+1} = E_{t+1} + \alpha G_{t+1}.$$

Here

- u is a flow utility function,
- G_{t+1} is the payoff of the asset and Π_t is the time-t price,
- β is a constant discount factor measuring impatience of the agent,
- E_t and E_{t+1} are endowments and
- α is the share of the asset purchased by the agent.

Rewriting as $\max_\alpha \{u(E_t - \Pi_t \alpha) + \beta \mathbb{E}_t u(E_{t+1} + \alpha G_{t+1})\}$ and differentiating with respect to α leads to the first order condition

$$u'(E_t - \Pi_t \alpha)\Pi_t = \beta \mathbb{E}_t u'(E_{t+1} + \alpha G_{t+1}) G_{t+1}.$$

Rearranging gives us

$$\Pi_t = \mathbb{E}_t \left[\beta \frac{u'(C_{t+1})}{u'(C_t)} G_{t+1} \right]. \tag{2.32}$$

We call the term

$$M_{t+1} := \beta \frac{u'(C_{t+1})}{u'(C_t)} \tag{2.33}$$

the **Lucas stochastic discount factor** (Lucas SDF) to recognize the seminal contribution in Lucas (1978).

Example 2.3.3. If u is linear, so that $u(c) = ac + b$ for some $a, b \in \mathbb{R}$, then $u'(c) = a$ for all c, so $M_{t+1} = \beta$. In other words, if utility has no curvature, then pricing is risk neutral.

Example 2.3.4. If utility has the CRRA form $u(c) = c^{1-\gamma}/(1-\gamma)$ for some $\gamma > 0$, then the Lucas SDF takes the form

$$M_{t+1} = \beta \left(\frac{C_{t+1}}{C_t} \right)^{-\gamma}, \tag{2.34}$$

which we can also write as $M_{t+1} = \beta \exp(-\gamma g_{t+1})$ when $g_{t+1} := \ln(C_{t+1}/C_t)$ is the growth rate of consumption. Thus the SDF is a positive random variable taking relatively small values in states of the world where consumption growth is high.

In the CRRA case, the Lucas SDF applies heavier discounting to assets that concentrate payoffs in states of the world where the agent is already enjoying strong consumption growth. Conversely, the agent attaches higher weights to future payoffs that occur when consumption growth is low. This is because such payoffs hedge against the risk of drawing low consumption states.

2.3.1.3 A General Specification

The standard neoclassical theory theory of asset pricing generalizes the Lucas discounting specification by assuming only that there exists a positive random variable M_{t+1} such that the price of an asset with payoff G_{t+1} is

$$\Pi_t = \mathbb{E}_t \, M_{t+1} \, G_{t+1} \qquad (t \geqslant 0). \tag{2.35}$$

The random variable M_{t+1} is called a **stochastic discount factor** (SDF). Equation 2.35 generalizes (2.32) by refraining from placing a specification on the SDF (apart from assuming positivity).

In fact, it can be shown that there exists an SDF M_{t+1} such that (2.35) is always valid under relatively weak assumptions. In particular, a single SDF M_{t+1} can be used to price *any* asset in the market, so if H_{t+1} is a second stochastic payoff then the current price of an asset with this payoff is $\mathbb{E}_t \, M_{t+1} \, H_{t+1}$.

We skip a proof of these claims, since our main interest is in understanding forward looking equations in Markov environments, which are needed for our discussion of dynamic programming below. References for asset pricing theory with full proofs are listed in §2.5.

2.3.1.4 Markov Pricing

A common assumption in quantitative applications is that all underlying randomness is driven by a Markov model. In this spirit, we take (X_t) to be P-Markov on finite state X, where P is a given stochastic matrix, and suppose further that the SDF and payoff have the forms

$$M_{t+1} = m(X_t, X_{t+1}) \quad \text{and} \quad G_{t+1} = g(X_t, X_{t+1})$$

for fixed functions m, g mapping $\mathsf{X} \times \mathsf{X}$ to $\mathbb{R}_+$. Since m is arbitrary at this point, we are not assuming any particular specification for the SDF.

In this setting, conditioning on $X_t = x$, the standard asset pricing equation $\Pi_t = \mathbb{E}_t \, M_{t+1} \, G_{t+1}$ becomes

$$\pi(x) = \sum_{x' \in \mathsf{X}} m(x, x') g(x, x') P(x, x') \qquad (x \in \mathsf{X}), \tag{2.36}$$

where $\pi(x)$ is the price of the asset conditional on $X_t = x$. (That is, $\Pi_t = \pi(X_t)$.)

2.3.1.5 Pricing a Stationary Dividend Stream

Now we are ready to look at pricing a stationary cash flow over an infinite horizon. This is on of the most fundamental problems in asset pricing. We will apply the Markov structure assumed in §2.3.1.4. In all that follows, (X_t) is P-Markov.

We seek the time t price, denoted by Π_t, for an **ex-dividend contract** on the dividend stream $(D_t)_{t \geqslant 0}$. The contract provides the owner with the right to the dividend stream. The "ex-dividend" component means that, should the contract be traded at time t, the dividend paid at time t goes to the seller rather than the buyer. As a result, purchasing at t and selling at $t+1$ pays $\Pi_{t+1} + D_{t+1}$. Hence, applying the fundamental asset pricing equation, the time t price Π_t of the contract must satisfy

$$\Pi_t = \mathbb{E}_t \, M_{t+1} (\Pi_{t+1} + D_{t+1}). \tag{2.37}$$

We assume the existence of a $d \in \mathbb{R}_+^{\mathsf{X}}$ such that $D_t = d(X_t)$ for all t. Using (2.36), we can write this as

$$\pi(x) = \sum_{x'} m(x, x')(\pi(x') + d(x'))P(x, x') \qquad (x \in \mathsf{X}), \tag{2.38}$$

or, equivalently,

$$\pi = A\pi + Ad \quad \text{when } A(x, x') := m(x, x')P(x, x'). \tag{2.39}$$

By the Neumann series lemma, the solution to this system of equations is

$$\pi^* = (I - A)^{-1} A d = \sum_{k=1}^{\infty} A^k d \quad \text{when } r(A) < 1. \tag{2.40}$$

The vector π^* is called an **equilibrium price function**

Exercise 2.3.1. As discussed in §2.3.1.1, the case $m \equiv \beta$ for some $\beta \in \mathbb{R}_+$ is called the risk-neutral case. Provide a condition on β under which $r(A) < 1$.

Exercise 2.3.2. Confirm that $(\Pi_t)_{t \geqslant 0}$ generated by $\Pi_t = \pi^*(X_t)$ solves (2.37).

Remark 2.3.1. A is often called the **Arrow–Debreu discount operator**. Its powers apply discounting: the valuation of any random payoff g in k periods is $A^k g$.

Exercise 2.3.3. Derive the price for a **cum-dividend contract** on the dividend stream $(D_t)_{t \geqslant 0}$, with the model otherwise unchanged. Under this contract, should the right to the dividend stream be traded at time t, the dividend paid at time t goes to the buyer rather than the seller.

2.3.1.6 Forward Sum Representation

Asset prices can be expressed as infinite sumps under the assumptions stated above. Let's show this for cum-dividend contracts (although the case of ex-dividend contracts is similar). In Exercise 2.3.3 you found that the state-contingent price vector π for a cum-dividend contract on the dividend stream $(D_t)_{t \geqslant 0}$ obeys

$$\pi = d + A\pi \quad \text{when } A(x, x') := m(x, x')P(x, x'). \tag{2.42}$$

As before, $D_t = d(X_t)$ and $(X_t)_{t \geqslant 0}$ is P-Markov. Applying the uniqueness component of Theorem 2.2.2, we see that the function π also obeys

$$\pi(x) = \mathbb{E}_x \left\{ \sum_{t=0}^{\infty} \left[\prod_{i=1}^{t} M_i \right] D_t \right\}$$

where $M_{t+1} := m(X_t, X_{t+1})$ and $\prod_{i=1}^{0} M_i := 1$. This expression agrees with our intuition: The price of the contract is the expected present value of the dividend stream, with the time t dividend discounted by the composite factor $M_1 \cdots M_t$.

2.3.1.7 Pricing a Consol

To be added.

2.3.2 Nonstationary Dividends

Until now, our discussion of asset pricing has assumed that dividends are stationary. However, dividends typically grow over time, along with other economic measures such as GDP. In this section we solve for the price of a dividend stream when dividends exhibit random growth.

2.3.2.1 Price-Dividend Ratios

A standard model of dividend growth is

$$\ln \frac{D_{t+1}}{D_t} = \kappa(X_t, \eta_{t+1}) \qquad t = 0, 1, \ldots,$$

where κ is a fixed function, (X_t) is the state process and (η_t) is IID. We let φ be the density of each η_t and assume that (X_t) is P-Markov on a finite set X. Let's suppose as before that the SDF obeys $M_{t+1} = m(X_t, X_{t+1})$ for some positive function m.

Since dividends grow over time, so will the price of the asset. As such, we should no longer seek a fixed function π such that $\Pi_t = \pi(X_t)$ for all t, since the resulting price process (Π_t) will fail to grow. Instead, we try to solve for the **price-dividend ratio** $V_t := \Pi_t / D_t$, which we hope will be stationary.

EXERCISE 2.3.4. Using $\Pi_t = \mathbb{E}_t \left[M_{t+1}(D_{t+1} + \Pi_{t+1}) \right]$, show that

$$V_t = \mathbb{E}_t \left[M_{t+1} \exp(\kappa(X_t, \eta_{t+1})) \, (1 + V_{t+1}) \right]. \tag{2.43}$$

After conditioning on $X_t = x$, (2.43) leads us to conjecture existence of a function v such that

$$v(x) = \sum_{x' \in \mathsf{X}} m(x, x') \int \exp(\kappa(x, \eta))\varphi(\mathrm{d}\eta) \, [1 + v(x')] \, P(x, x') \tag{2.44}$$

for all $x \in \mathsf{X}$. We understand (2.44) as an equation to be solved for the unknown object $v \in \mathbb{R}^{\mathsf{X}}$. If we can find a solution v^* to (2.44), then setting $V_t = v^*(X_t)$ yields a process (V_t) that obeys (2.43).

Exercise 2.3.5. Let

$$A(x, x') := m(x, x') \int \exp(\kappa(x, \eta))\varphi(\mathrm{d}\eta) P(x, x') \qquad (x, x' \in \mathsf{X}). \tag{2.45}$$

Show that (2.43) has a unique solution v^* in $\mathbb{R}^{\mathsf{X}}$ when $r(A) < 1$, and

$$v^* = (I - A)^{-1} A \mathbb{1} = \sum_{t \geqslant 1} A^t. \tag{2.46}$$

The price-dividend process (V_t^*) defined by $V_t^* = v^*(X_t)$ solves (2.43). The price can be recovered via $\Pi_t = V_t^* D_t$.

2.3.2.2 Application: Markov Growth with a Lucas SDF

As an example, suppose that dividend growth obeys

$$\kappa(X_t, \eta_{d,t+1}) = \mu_d + X_t + \sigma_d \, \eta_{d,t+1}$$

where $(\eta_{d,t})_{t \geqslant 0}$ is IID and standard normal. Consumption growth is given by

$$\ln \frac{C_{t+1}}{C_t} = \mu_c + X_t + \sigma_c \, \eta_{c,t+1},$$

where $(\eta_{c,t})_{t \geqslant 0}$ is also IID and standard normal. We use the Lucas SDF in (2.34), implying that

$$M_{t+1} = \beta \left(\frac{C_{t+1}}{C_t} \right)^{-\gamma} = \beta \exp(-\gamma(\mu_c + X_t + \sigma_c \eta_{c,t+1}))$$

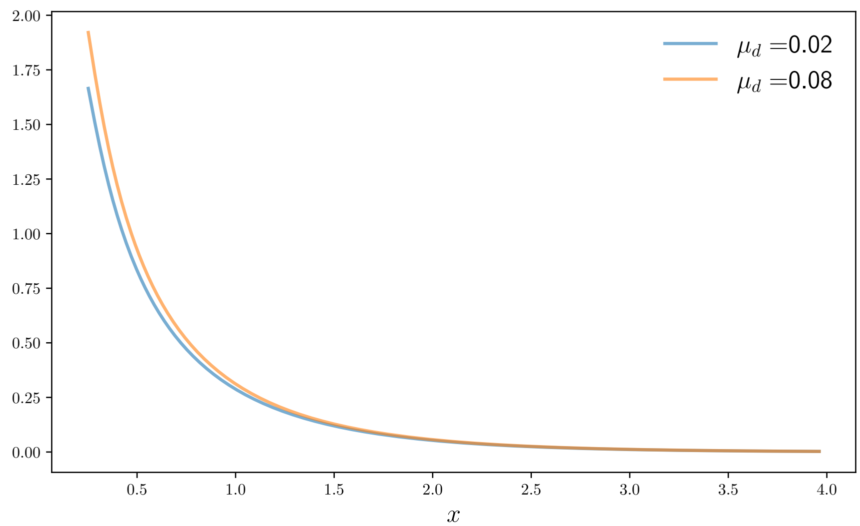


Figure 2.7: Price-dividend ratio as a function of the state

Exercise 2.3.6. Using (2.45), show that

$$A(x, x') = \beta \exp\left(-\gamma \mu_c + \mu_d + (1-\gamma)x + \frac{\gamma^2 \sigma_c^2 + \sigma_d^2}{2}\right) P(x, x').$$

Figure 2.7 shows the price-dividend ratio function v^* for the specification given in Listing 9, as well as for an alternative mean dividend growth rate μ_d. The state process is a Tauchen discretization of an AR(1) process with positive autocorrelation. An increase in the state predicts higher dividends, which tends to increase the price. At the same time, higher x also predicts higher consumption growth, which acts negatively on the price. For values of γ greater than 1, the second effect dominates and the price-dividend ratio slopes down.

Exercise 2.3.7. Complete the code in Listing 9 and replicate Figure 2.7. Add a test to your code that checks $r(A) < 1$ before computing the price-dividend ratio.

2.4 Nonlinear Problems

Add roadmap.

```
using QuantEcon, LinearAlgebra

"Creates an instance of the asset pricing model with Markov state."
function create_asset_pricing_model(;
        n=200,       # state grid size
        ρ=0.9,       # state persistence
        ν=0.2,       # state volatility
        β=0.99,      # discount factor
        γ=2.5,       # preference parameter
        μ_c=0.01,    # mean consumption growth
        σ_c=0.02,    # consumption volatility
        μ_d=0.02,    # mean dividend growth
        σ_d=0.1      # dividend volatility
    )
    mc = tauchen(n, ρ, ν)
    x_vals, P = exp.(mc.state_values), mc.p
    return (; x_vals, P, β, γ, μ_c, σ_c, μ_d, σ_d)
end
```

Listing 9: Asset pricing model with Lucas SDF (`pd_ratio.jl`)

2.4.1 Incomplete Markets

In §2.3.1.5, the problem of solving for the equilibrium price vector π was treated using the Neumann series lemma. However, there are various modifications to the basic model where nonlinearities make use of the Neumann series lemma impossible. Considering the technical issues that arise in these problems will be helpful for us, since the Bellman operators that arise from dynamic programming problems are also nonlinear.

For example, Harrison and Kreps (1978) analyze a setting with heterogeneous beliefs and incomplete markets, leading to failure of the standard asset pricing equation. In the model there are two types of agents. Type i believes that the state updates according to stochastic matrix P_i for $i = 1, 2$. In addition, agents are risk-neutral, so $m(x, y) \equiv \beta \in (0, 1)$. Harrison and Kreps (1978) show that, for their model, the equilibrium condition (2.38) becomes

$$\pi(x) = \max_i \beta \sum_{x'} [\pi(x') + d(x')] P_i(x, x') \tag{2.47}$$

for $x \in S$ and $i \in \{1, 2\}$. Setting aside the details that lead to this equation, our objective is simply: obtain a vector of prices π that solves (2.47).

As a first step, we introduce an operator $T \colon \mathbb{R}^n_+ \to \mathbb{R}^n_+$ that maps π to $T\pi$ via

$$(T\pi)(x) = \max_i \beta \sum_{x'} [\pi(x') + d(x')] P_i(x, x') \qquad (x \in S). \tag{2.48}$$

We are assuming $d \geqslant 0$, so T is indeed a self-map on $\mathbb{R}^n_+$.

By construction, a vector $\pi \in \mathbb{R}^n_+$ is a fixed point of T if and only if it is a vector of prices that solves (2.47). Hence, we have successfully converted our equilibrium problem into a fixed point problem.

We aim to show that T is a contraction. To this end, pick any $p, q \in \mathbb{R}^n_+$. Applying the inequality from Lemma 1.3.2 on page 34, we obtain

$$|(Tp)(x) - (Tq)(x)| \leqslant \beta \max_i \left| \sum_{x'} [p(x') + d(x')] P_i(x, x') - \sum_{x'} [q(x') - d(x')] P_i(x, x') \right|.$$

Using the triangle inequality and canceling terms leads to

$$|(Tp)(x) - (Tq)(x)| \leqslant \beta \max_{i \in \{1,2\}} \sum_{x'} |p(x') - q(x')| P_i(x, x') \leqslant \beta \|p - q\|_\infty.$$

Since this bound holds for all x, we can take the maximum with respect to x and obtain

$$\|Tp - Tq\|_\infty \leqslant \beta \|p - q\|_\infty.$$

In other words, on $\mathbb{R}^n_+$, the map T is a contraction of modulus β with respect to the sup norm.

Since $\mathbb{R}^n_+$ is a closed subset of $\mathbb{R}^n$, we conclude that T has a unique fixed point in this set. Hence, the system (2.47) has a unique solution π^* in $\mathbb{R}^n_+$, representing equilibrium prices. This fixed point can be computed by successive approximation.

2.4.2 Job Search with Markov State

As our next application, let's return to the job search problem from Chapter 1. We introduce one extension: wage draws are allowed to be correlated rather than IID. This will bring us closer to the models typically used in quantitative analysis.

2.4.2.1 Value Function Iteration

We assume that the wage process (W_t) is P-Markov on finite set $\mathsf{W} \subset \mathbb{R}_+$, where P is a stochastic matrix. The Bellman equation (1.24) on page 38 becomes

$$v^*(w) = \max\left\{\frac{w}{1-\beta},\; c + \beta \sum_{w' \in \mathsf{W}} v^*(w') P(w, w')\right\} \qquad (w \in \mathsf{W}). \tag{2.49}$$

The corresponding Bellman operator is

$$(Tv)(w) = \max\left\{\frac{w}{1-\beta},\; c + \beta \sum_{w' \in \mathsf{W}} v(w') P(w, w')\right\} \qquad (w \in \mathsf{W}).$$

Let $\mathscr{V} := \mathbb{R}^{\mathsf{W}}_+$ and endow $\mathscr{V}$ with the supremum norm, so that the distance between two elements f, g of $\mathscr{V}$ is measured by $\|f - g\| = \max_{w \in \mathsf{W}} |f(w) - g(w)|$.

EXERCISE 2.4.1. Prove that T is an order-preserving self-map on $\mathscr{V}$.

EXERCISE 2.4.2. Prove that T is a contraction of modulus β on $\mathscr{V}$.

We recommend you read the proof of the next lemma, since the same style of argument is repeated many times in the text.

Lemma 2.4.1. *The value function v^* is increasing on* W *whenever P is monotone increasing.*

Proof. Let $i\mathscr{V}$ be the increasing functions in $\mathscr{V}$ and suppose that P is monotone increasing. The operator T is a self-map on $i\mathscr{V}$ in this setting, since $v \in i\mathscr{V}$ implies $h(w) := c + \beta \sum_{w' \in \mathsf{W}} v(w')P(w, w')$ is in $i\mathscr{V}$. Hence, for such a v, both h and the stopping value function $e(w) := w/(1 - \beta)$ are in $i\mathscr{V}$. It follows that $Tv = \max\{h, e\}$ is in $i\mathscr{V}$.

Since $i\mathscr{V}$ is a closed subset of $\mathscr{V}$ and T is a self-map on $i\mathscr{V}$, the fixed point v^* is in $i\mathscr{V}$ (cf., Exercise 1.2.8 on page 17). □

In view of the contraction property established in Exercise 2.4.2, we can use value function iteration to solve for the value function and compute the optimal policy. Code for implementing this procedure is shown in Listing 10. The definition of a v-greedy policy is analogous to that for the IID case (see (1.28) on page 41).

2.4.2.2 Continuation Values

The continuation value h^* from the IID case (defined on page 38) is now replaced by a **continuation value function**, given by

$$h^*(w) := c + \beta \sum_{w' \in \mathsf{W}} v^*(w')P(w, w') \qquad (w \in \mathsf{W}).$$

The continuation value depends on w because the current wage offer helps predict the wage offer next period, which in turn affects the value of continuing. The functions $w \mapsto w/(1 - \beta)$, h^* and v^* corresponding to the default model in Listing 10 are shown in Figure 2.8.

EXERCISE 2.4.3. Explain why the continuation value function is increasing in Figure 2.8. If possible, provide a mathematical explanation and economic intuition.

EXERCISE 2.4.4. Using the Bellman equation (2.49), show that h^* obeys

$$h^*(w) := c + \beta \sum_{w' \in \mathsf{W}} \max\left\{\frac{w'}{1 - \beta}, h^*(w')\right\} P(w, w') \qquad (w \in \mathsf{W}).$$

```julia
using QuantEcon, LinearAlgebra
include("s_approx.jl")

"Creates an instance of the job search model with Markov wages."
function create_markov_js_model(;
        n=200,       # wage grid size
        ρ=0.9,       # wage persistence
        ν=0.2,       # wage volatility
        β=0.98,      # discount factor
        c=1.0        # unemployment compensation
    )
    mc = tauchen(n, ρ, ν)
    w_vals, P = exp.(mc.state_values), mc.p
    return (; n, w_vals, P, β, c)
end

" The Bellman operator Tv = max{e, c + β P v} with e(w) = w / (1-β)."
function T(v, model)
    (; n, w_vals, P, β, c) = model
    h = c .+ β * P * v
    e = w_vals ./ (1 - β)
    return max.(e, h)
end

" Get a v-greedy policy."
function get_greedy(v, model)
    (; n, w_vals, P, β, c) = model
    σ = w_vals / (1 - β) .>= c .+ β * P * v
    return σ
end

"Solve the infinite-horizon Markov job search model by VFI."
function vfi(model)
    v_init = zero(model.w_vals)
    v_star = successive_approx(v -> T(v, model), v_init)
    σ_star = get_greedy(v_star, model)
    return v_star, σ_star
end
```

Listing 10: Job search with Markov state (`markov_js.jl`)

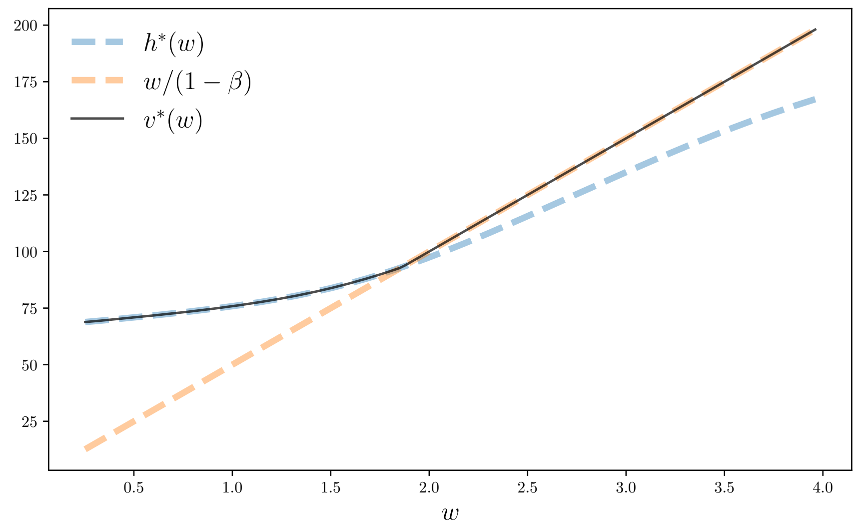


Figure 2.8: Value, stopping and continuation for Markov job search

EXERCISE 2.4.5. Let Q be the operator on $\mathscr{V}$ defined at $h \in \mathscr{V}$ by

$$(Qh)(w) := c + \beta \sum_{w' \in \mathsf{W}} \max \left\{ \frac{w'}{1-\beta}, \, h(w') \right\} P(w, w') \qquad (w \in \mathsf{W}). \tag{2.50}$$

Prove that Q is (a) an order-preserving self-map on $\mathscr{V}$ and (b) a contraction of modulus β on $\mathscr{V}$ under the supremum norm.

Exercise 2.4.5 suggests to us a way to solve the job search problem without using value function iteration: iterate with Q to obtain the continuation value function h^* and then use the policy

$$\sigma^*(w) = \mathbb{1}\left\{ \frac{w'}{1-\beta} \geqslant h^*(w) \right\} \qquad (w \in \mathsf{W}),$$

which tells the agent to accept when the current stopping value exceeds the current continuation value.

In this particular case, the two approaches (iterating with T vs iterating with Q) are relatively similar, and, in general, neither offers any particular advantage over the other. However, we already saw in the IID case that the approach based on continuation values can be much more efficient in some settings (see the discussion in §1.4.2.2). We will investigate the relative merits of the two approaches more systematically in Chapter 3.

2.4.3 Job Search with Separation

As a final application for this chapter, we continue the job search problem with correlated wage draws discussed in §2.4.2 with one natural extension: jobs are not permanent. Instead, workers separate from jobs at fixed rate α. In other words, an existing match between worker and firm terminates with probability α every period.

Once separation enters the picture, the agent comes to view the loss of a job as a capital loss, and a spell of unemployment as an investment in searching for an acceptable job. In what follows, the wage process and discount factor are unchanged from §2.4.2. As before, $\mathscr{V} := \mathbb{R}_+^{\mathsf{W}}$ is endowed with the supremum norm.

The value function for an unemployed worker facing current wage offer w is

$$v_u(w) = \max\left\{ v_e(w),\; c + \beta \sum_{w' \in \mathsf{W}} v_u(w') P(w, w') \right\}. \tag{2.51}$$

The difference from the previous case we studied, where jobs were permanent, is that we now have two value functions. One is v_u, which gives the value of being unemployed at each possible wage offer, conditional on optimal behavior. The second is v_e, which solves

$$v_e(w) = w + \beta \left[\alpha \sum_{w'} v_u(w') P(w, w') + (1-\alpha) v_e(w) \right] \qquad (w \in \mathsf{W}). \tag{2.52}$$

This equation states that value accruing to an employed agent is current wage plus the discounted expected value next of being either employed or unemployed next period, weighted by their probabilities.

We claim that, when $0 < \alpha, \beta < 1$, the equations (2.51) and (2.52) both have unique solutions in $\mathscr{V}$. To help prove this, we solve (2.52) in terms of $v_e(w)$ to obtain

$$v_e(w) = \frac{1}{1 - \beta(1-\alpha)} \left(w + \alpha\beta (P v_u)(w') \right). \tag{2.53}$$

(Recall $(Ph)(w) := \sum_{w'} h(w') P(w, w')$ for $h \in \mathbb{R}^{\mathsf{W}}$ and $w \in \mathsf{W}$.) Substituting into (2.51) yields

$$v_u(w) = \max\left\{ \frac{1}{1 - \beta(1-\alpha)} \left(w + \alpha\beta (P v_u)(w) \right),\; c + \beta\, (P v_u)(w) \right\}. \tag{2.54}$$

Exercise 2.4.6. Prove that there exists a unique $v_u \in \mathscr{V}$ that solves (2.54). Propose a convergent method for solving for both v_u and v_e. [Hint, if you need it: Look at Exercise 1.3.9 on page 34.]

2.5 Chapter Notes

The fundamental theory of asset pricing is discussed in many places, including Hansen and Renault (2010). Textbook introductions can be found in Cochrane (2009), Duffie (2010) and Campbell (2017).

The systematic study of monotone Markov chains was initiated by Daley (1968). Monotone Markov methods have many important applications in economics. See, for example, Hopenhayn and Prescott (1992), Kamihigashi and Stachurski (2014), Jaśkiewicz and Nowak (2014), Balbus et al. (2014), Foss et al. (2018) and Hu and Shmaya (2019).

Chapter 3

Optimal Stopping

A large variety of decision making problems involve choosing when to act in the face of risk and uncertainty. The job search model we studied in Chapters 1–2 is one example. Others include if or when to exit or enter a market, bring a new product to market, default on a loan, exploit some new technology or business opportunity, or exercise a real or financial option. All of these problems can be modeled in a common framework and solved using dynamic programming. Moreover, they have common features that allow us to find sharp characterizations of optimality. Finally, they offer an excellent introduction to dynamic programming because the binary choice (stop or continue) makes the recursive representations particularly clear and insightful.

In this chapter we discuss theory and applications of infinite-horizon optimal stopping problems in settings where the exogenous state variable is discrete.

3.1 Introduction to Optimal Stopping

We begin with the standard theory of optimal stopping in discrete time. We then consider alternative approaches, based around continuation values and threshold policies. One key objective is to provide a rigorous discussion of optimality, which improves on our intuitive analysis in the context of job search in §1.4.

3.1.1 Theory

In this section we set out the fundamental theory of discrete time infinite-horizon optimal stopping problems.

3.1.1.1 The Stopping Problem

Let X be a finite set. An **optimal stopping problem** with state space X consists of

- a stochastic matrix P on X,
- a discount factor $\beta \in (0, 1)$,
- a **continuation reward function** $c \in \mathbb{R}^{\mathsf{X}}$, and
- a **exit reward function** $e \in \mathbb{R}^{\mathsf{X}}$.

Given a P-Markov chain $(X_t)_{t \geqslant 0}$, the problem evolves as follows: An agent observes the state X_t in each period and decides whether to continue or stop. If she chooses to stop, she receives $e(X_t)$ and the process terminates. If she decides to continue, then she receives $c(X_t)$ and the process repeats next period. Lifetime rewards are given by

$$\mathbb{E} \sum_{t \geqslant 0} \beta^t R_t,$$

where R_t equals $c(X_t)$ while the agent continues, $e(X_t)$ when the agent stops, and zero thereafter.

Example 3.1.1. In the infinite-horizon job search problem from Chapter 1, the wage offer process (W_t) is IID with common distribution φ on finite set W, and the choice is between stopping (accepting the job offer) and continuing (receiving unemployment compensation and waiting till next period). This is an optimal stopping problem with state space $\mathsf{X} = \mathsf{W}$ and stochastic matrix P having all rows equal to φ, so that all draws are IID from φ. The exit reward function is $e(x) = x/(1 - \beta)$ and the continuation reward function is constant and equal to unemployment compensation.

Example 3.1.2. Consider an infinite horizon American call option, which provides the right to buy a given asset at strike price K at every future point in time. The market price of the asset is given by $S_t = s(X_t)$, where (X_t) is P-Markov on finite set X. The interest rate is $r > 0$. The decision of when to exercise is an optimal stopping problem, with exit corresponding to exercise of the option. The discount factor is $1/(1+r)$, the exit reward function is $e(x) = s(x) - K$ and the continuation reward is zero.

As for the job search problem, the actions of the agent will be expressed in terms of a **policy function**, which is a map σ from X to $\{0, 1\}$. The interpretation is that, on observing state x at any given time, the agent responds with action $\sigma(x)$, where 0 means "continue" and 1 means "stop." Implicit in this formulation is the assumption that the current state contains enough information for the agent to decide whether or not to stop.

Let Σ be the set of functions from X to $\{0, 1\}$. Let $v_\sigma(x)$ denote the expected lifetime value of following policy σ now and in every future period, given current state $x \in \mathsf{X}$. We call v_σ the **σ-value function**. Below, in §3.1.1.3, we show that v_σ is well defined and describe how to calculate it.

The function v_σ is an essential object in what follows, since our aim is to choose a policy that maximizes lifetime value. In particular, a policy $\sigma^* \in \Sigma$ is called **optimal** if

$$v_{\sigma^*}(x) = \max_{\sigma \in \Sigma} v_\sigma(x) \quad \text{for all } x \in \mathsf{X}. \tag{3.1}$$

3.1.1.2 Policy Valuation

Fixing $\sigma \in \Sigma$, let us think about how to pin down the σ-value function v_σ. Recall that $v_\sigma(x)$ is the lifetime value of following σ conditional on state x. Some thought will convince you that v_σ must satisfy

$$v_\sigma(x) = \sigma(x)e(x) + (1 - \sigma(x))\left[c(x) + \beta \sum_{x' \in \mathsf{X}} v_\sigma(x')P(x, x')\right] \quad \text{for all } x \in \mathsf{X}. \tag{3.2}$$

To see this, suppose first that $\sigma(x) = 1$. In this case, (3.2) states that $v_\sigma(x) = e(x)$, which is what we expect: choosing to stop yields the exit reward. If, on the other hand, the agent chooses to continue, $\sigma(x) = 0$ and we have

$$v_\sigma(x) = c(x) + \beta \sum_{x' \in \mathsf{X}} v_\sigma(x')P(x, x'). \tag{3.3}$$

This statement also makes sense. Since σ is followed in every period, the value of continuing is the current continuation reward plus the discounted expected reward obtained by continuing with policy σ next period.

Now all that remains is to solve (3.2) for the function v_σ. To do this, we set

$$r_\sigma(x) := \sigma(x)e(x) + (1 - \sigma(x))c(x) \quad \text{and} \quad K_\sigma(x, x') := \beta(1 - \sigma(x))P(x, x').$$

With this notation, we can write (3.2) pointwise as $v_\sigma = r_\sigma + K_\sigma v_\sigma$. If $r(K_\sigma) < 1$, then we have

$$v_\sigma = (I - K_\sigma)^{-1} r_\sigma. \tag{3.4}$$

Exercise 3.1.1. Confirm that $r(K_\sigma) < 1$ holds for any optimal stopping problem.

By Exercise 3.1.1 and the Neumann series lemma, v_σ is uniquely defined by (3.4).

3.1.1.3 Policy Operators

For the proofs below, it is helpful to view v_σ as the fixed point of a certain operator. In particular, we pair each $\sigma \in \Sigma$ with a corresponding **policy operator**, denoted by T_σ, and defined at $v \in \mathbb{R}^{\mathsf{X}}$ by

$$(T_\sigma v)(x) = \sigma(x)e(x) + (1 - \sigma(x)) \left[c(x) + \beta \sum_{x' \in \mathsf{X}} v(x')P(x, x') \right] \tag{3.5}$$

for each $x \in \mathsf{X}$.

Exercise 3.1.2. Prove that, for any $\sigma \in \Sigma$, the operator T_σ is order-preserving with respect to the pointwise partial order $\leqslant$ on $\mathbb{R}^{\mathsf{X}}$.

Using the notation defined in §3.1.1.2, a function $v \in \mathbb{R}^{\mathsf{X}}$ is a fixed point of T_σ if and only if $v = r_\sigma + K_\sigma v$. Thus, by $r(K_\sigma) < 1$ and the Neumann series lemma, the policy value function v_σ is the unique fixed point of point of T_σ in $\mathbb{R}^{\mathsf{X}}$. The next result shows that, in addition, iterates of T_σ converge to v_σ.

Proposition 3.1.1. *For any $\sigma \in \Sigma$, the policy operator T_σ is a contraction of modulus β on $\mathbb{R}^{\mathsf{X}}$ under to the supremum norm.*

Exercise 3.1.3. Prove Proposition 3.1.1.

3.1.1.4 The Value Function

In the job search problem, we found the optimal policy by computing the fixed point of the Bellman operator. Here we will do the same. We will also explain more carefully the relationship between optimality and the fixed point of the Bellman operator.

First we define the **value function** v^* of the optimal stopping problem as

$$v^*(x) := \max_{\sigma \in \Sigma} v_\sigma(x) \qquad (x \in \mathsf{X}). \tag{3.6}$$

In particular, $v^*(x)$ is the maximal lifetime value that can be obtained by an agent facing current state x.

Remark 3.1.1. When we discussed the infinite-horizon job search problem in §1.4, we gave a "lazy" definition of the value function: we simply defined it as the solution to the Bellman equation. The correct definition is in fact (3.6). The only reason we omitted the correct definition in that discussion is that we did not yet have a formal way of defining lifetime value.

As we will see below, using the correct definition (3.6) for the value function makes the full optimality theory easier and clearer. At the same time, it leaves us with the problem of obtaining the value function, since solving the maximization in (3.6) is, in general, a hard problem.

Here is our approach: To obtain v^*, we

(i) introduce the Bellman equation for the optimal stopping problem, which is

$$v(x) = \max\left\{e(x), c(x) + \beta \sum_{x' \in \mathsf{X}} v(x')P(x,x')\right\} \qquad (x \in \mathsf{X}), \tag{3.7}$$

(ii) prove that the Bellman equation has a unique solution in $\mathbb{R}^{\mathsf{X}}$, and, finally,

(iii) show that this solution equals the value function, as defined in (3.6).

These steps are completed in §3.1.1.5 below.

3.1.1.5 The Bellman Operator

The **Bellman operator** for the optimal stopping problem is the operator T such that any fixed point of T solves the Bellman equation and vice versa. This is true by construction for T defined by $v \mapsto Tv$,

$$(Tv)(x) = \max\left\{e(x),\ c(x) + \beta \sum_{x' \in \mathsf{X}} v(x')P(x,x')\right\} \qquad (x \in \mathsf{X}). \tag{3.8}$$

EXERCISE 3.1.4. Prove that T is an order preserving self-map on $\mathbb{R}^{\mathsf{X}}$.

Here is the main result for this section:

Proposition 3.1.2. *For the optimal stopping problem defined in §3.1.1.1,*

(i) *T is a contraction map of modulus β on $\mathbb{R}^{\mathsf{X}}$, under the supremum norm $\|\cdot\|_\infty$ and*

(ii) *the unique fixed point of T on $\mathbb{R}^{\mathsf{X}}$ is equal to the value function v^* defined in (3.6).*

EXERCISE 3.1.5. As a first step for proving Proposition 3.1.2, show that T is a contraction of modulus β on $\mathbb{R}^{\mathsf{X}}$. (Extend the proof of contractivity of the Bellman operator in the job search case.)

Now we can complete the proof of Proposition 3.1.2.

Proof of Proposition 3.1.2. With the result of Exercise 3.1.5 in hand, we need only show that the unique fixed point of T in $\mathbb{R}^{\mathsf{X}}$, denoted by $\bar{v}$, is equal to $v^* = \max_{\sigma \in \Sigma} v_\sigma$. We show $\bar{v} \leqslant v^*$ and then $\bar{v} \geqslant v^*$.

For the first inequality, let $\sigma \in \Sigma$ be defined by

$$\sigma(x) = \mathbb{1}\left\{e(x) \geqslant c(x) + \beta \sum_{x' \in \mathsf{X}} \bar{v}(x')P(x, x')\right\} \quad \text{for all } x \in \mathsf{X}.$$

Observe that, for this choice of σ, we have, for any $w \in \mathsf{X}$,

$$\begin{aligned}(T_\sigma v)(x) &= \sigma(x)e(x) + (1 - \sigma(x))\left[c(x) + \beta \sum_{x' \in \mathsf{X}} \bar{v}(x')P(x, x')\right] \\ &= \max\left\{e(x),\ c(x) + \beta \sum_{x' \in \mathsf{X}} \bar{v}(x')P(x, x')\right\} = (T\bar{v})(x) = \bar{v}(x).\end{aligned}$$

In particular, $T_\sigma \bar{v} = T\bar{v} = \bar{v}$. But the only fixed point of T_σ in $\mathbb{R}^{\mathsf{X}}$ is v_σ, so it must be the case that $\bar{v} = v_\sigma$. But then $\bar{v} \leqslant v^*$, by the definition of v^*. This is our first inequality.

Regarding the second inequality, if we fix $v \in \mathbb{R}^{\mathsf{X}}$ and $\sigma \in \Sigma$, then, since $Tv \geqslant T_\sigma v$ holds and since both T and T_σ are order preserving, we obtain $T_\sigma^k v \leqslant T^k v$ for all k. Taking the limit of this expression yields $v_\sigma \leqslant \bar{v}$. Taking the supremum over $\sigma \in \Sigma$ implies $v^* \leqslant \bar{v}$. □

3.1.1.6 Optimal Policies

Paralleling the definition provided in the discussion of job search (§1.4), for each $v \in \mathbb{R}^{\mathsf{X}}$, we call $\sigma \in \Sigma$ **v-greedy** if

$$\sigma(x) = \mathbb{1}\left\{e(x) \geqslant c(x) + \beta \sum_{x' \in \mathsf{X}} v(x')P(x, x')\right\} \quad \text{for all } x \in \mathsf{X}. \tag{3.9}$$

A v-greedy policy uses v to assign values to states and then chooses to stop or continue based on the action that generates a higher payoff.

With this language in place, our informal argument in §1.1.2.1 that optimal choices can be made using the value function becomes precise in the next proposition.

Proposition 3.1.3. *Policy $\sigma \in \Sigma$ is optimal if and only if it is v^*-greedy.*

Proposition 3.1.3 is a version of **Bellman's principle of optimality**.

EXERCISE 3.1.6. Show that $\sigma \in \Sigma$ is v^*-greedy if and only $T_\sigma v^* = v^*$.

Proof of Proposition 3.1.3. From Exercise 3.1.6 we need to show that $T_\sigma v^* = v^*$ is equivalent to optimality. But the unique fixed point of T_σ in $\mathbb{R}^{\mathsf{X}}$ is v_σ, so $T_\sigma v^* = v^*$ is equivalent to $v^* = v_\sigma$. The last equality coincides with the definition of optimality for σ. □

Corollary 3.1.4. *The optimal stopping problem has exactly one optimal policy.*

Proof. This follows directly from Proposition 3.1.3, since, given v^*, the greedy policy

$$\sigma^*(x) := \mathbb{1}\left\{e(x) \geqslant c(x) + \beta \sum_{x' \in \mathsf{X}} v^*(x') P(x, x')\right\} \qquad (x \in \mathsf{X}) \tag{3.10}$$

is clearly uniquely defined. □

3.1.2 Monotonicity

You will recall that the optimal policy in the IID job search problem takes the form $\sigma^*(w) = \mathbb{1}\{w \geqslant w^*\}$ for all $w \in \mathsf{W}$, where $w^* := (1-\beta)h^*$ is the reservation wage and h^* is the continuation value (see page 41). This optimal policy is of threshold type: once the wage offer exceeds the threshold, the agent always stops.

Since threshold policies are convenient, let us now pause to characterize them. To do so, we again take v^* to be the value function of the optimal stopping problem and introduce the corresponding **continuation value function** h^*, which is defined by

$$h^*(x) := c(x) + \beta \sum_{x' \in \mathsf{X}} v^*(x') P(x, x') \qquad (x \in \mathsf{X}). \tag{3.11}$$

Throughout this section, we take X to be a subset of $\mathbb{R}$. Elements of X are ordered by $\leqslant$, the standard order on $\mathbb{R}$.

EXERCISE 3.1.7. Prove that the optimal policy σ^* is decreasing on X whenever e is decreasing on X and h^* is increasing on X.

EXERCISE 3.1.8. Prove that the optimal policy σ^* is increasing on X whenever e is increasing on X and h^* is decreasing on X.

Example 3.1.3. In the IID job search problem, $e(w) = w/(1-\beta)$ is increasing and h^* is constant. Hence the result in Exercise 3.1.8 applies. This is why the optimal policy $\sigma^*(w) = \mathbb{1}\{w \geqslant (1-\beta)h^*\}$ is increasing. The agent accepts all sufficiently large wage offers.

In the settings of Exercises 3.1.7–3.1.8, the optimal policy is either increasing or decreasing. Since X is totally ordered, monotonicity implies that the policy is of threshold type. For example, if σ^* is increasing, then we take x^* to be the smallest $x \in \mathsf{X}$ such that $\sigma^*(x) = 1$. For such an x^* we have

$$x < x^* \implies \sigma^*(x) = 0 \quad \text{and} \quad x \geqslant x^* \implies \sigma^*(x) = 1.$$

Remark 3.1.2. The conditions in Exercises 3.1.7–3.1.8 are sufficient but not necessary for monotone policies. For example, Figure 2.8 on 86 provides an example of a setting where the policy is increasing (the agent accepts for sufficiently large wage offers) even though both $e(x) = x/(1-\beta)$ and h^* are strictly increasing.

The conditions in Exercises 3.1.7–3.1.8 are placed on the continuation value function h^*. Ideally, conditions should be placed on primitives rather than derived objects or solutions. For this reason, the results in the following exercises can be useful.

EXERCISE 3.1.9. Let $i\mathbb{R}^{\mathsf{X}}$ be the increasing functions in $\mathbb{R}^{\mathsf{X}}$. Prove that $h^* \in i\mathbb{R}^{\mathsf{X}}$ whenever $e, c \in i\mathbb{R}^{\mathsf{X}}$ and P is monotone increasing.

EXERCISE 3.1.10. Show that the conditions of Exercise 3.1.7 hold when e is constant on X, c is increasing on X and P is monotone increasing.

3.1.3 Continuation Values

In §1.4.2.2 we used a "continuation value" approach to solving the job search problem with IID draws, which involved computing the continuation value h^* directly and then setting the optimal policy to $\sigma^*(w) = \mathbb{1}\{w/(1-\beta) \geqslant h^*\}$. We saw that this approach is more efficient than first computing the value function, since the continuation value is one-dimensional rather than $|\mathsf{W}|$-dimensional.

In §2.4.2.2, we tried the same approach for the job search problem with Markov state, where wage draws are correlated. We found that there is no clear benefit to the continuation value approach in that setting, since the continuation value function has the same dimensionality as the value function.

These observations motivate us to explore continuation value methods more carefully. In this section, we state the continuation value approach for the general optimal stopping problem and verify convergence. We will see that, while all relevant state components must be included in the value function, purely transitory components do not affect continuation values. Hence the continuation value approach is at least as efficient and sometimes radically more so.

Another asymmetry between value functions and continuation value functions is that the latter are typically smoother. For example, in job search problems, the value function is usually kinked at the reservation wage, while the continuation value function is smooth. Relative smoothness comes from taking expectations over stochastic transitions, since integration is a smoothing operation. Like lower dimensionality, increased smoothness helps with both analysis and computation.

3.1.3.1 Methodology

Let h^* be the continuation value function for the optimal stopping problem, as defined in (3.11). To compute h^* directly we begin with the optimal stopping Bellman equation evaluated at v^* and rewrite it as

$$v^*(x') = \max\{e(x'), h^*(x')\} \qquad (x' \in \mathsf{X}). \tag{3.12}$$

Taking expectations of both sides of this equation conditional on current state x produces $\sum_{x' \in \mathsf{X}} v^*(x') P(x, x') = \sum_{x' \in \mathsf{X}} \max\{e(x'), h^*(x')\} P(x, x')$. Multiplying by β, adding $c(x)$, and using the definition of h^*, we get

$$h^*(x) = c(x) + \beta \sum_{x' \in \mathsf{X}} \max\{e(x'), h^*(x')\} P(x, x') \qquad (x \in \mathsf{X}). \tag{3.13}$$

This expression motivates us to introduce the **continuation value operator** $C \colon \mathbb{R}^{\mathsf{X}} \to \mathbb{R}^{\mathsf{X}}$ via

$$(Ch)(x) = c(x) + \beta \sum_{x' \in \mathsf{X}} \max\{e(x'), h(x')\} P(x, x') \qquad (x \in \mathsf{X}). \tag{3.14}$$

Proposition 3.1.5. *The operator C is a contraction of modulus β on $\mathbb{R}^{\mathsf{X}}$. Moreover, the unique fixed point of C in $\mathbb{R}^{\mathsf{X}}$ is the continuation value function h^*.*

Proposition 3.1.5 provides us with an alternative method to compute the optimal policy, which does not involve value function iteration:

(i) Use successive approximation with C to compute h^* (at least approximately) and

(ii) Calculate σ^* via $\sigma^*(x) = \mathbb{1}\{e(x) \geqslant h^*(x)\}$ for each $x \in \mathsf{X}$.

In §3.1.3.2 we discuss settings where this approach is advantageous.

Proof of Proposition 3.1.5. Fix $f, g \in \mathbb{R}^{\mathsf{X}}$ and $x \in \mathsf{X}$. By the triangle inequality and the bound $|\alpha \vee x - \alpha \vee y| \leqslant |x - y|$ from page 39, we have

$$
\begin{aligned}
|(Cf)(x) - (Cg)(x)| &\leqslant \beta \sum_{x' \in \mathsf{X}} |\max\{e(x'), f(x')\} - \max\{e(x'), g(x')\}| \, P(x, x') \\
&\leqslant \beta \sum_{x' \in \mathsf{X}} |f(x') - g(x')| \, P(x, x').
\end{aligned}
$$

The right-hand side is dominated by $\beta \|f - g\|_\infty$. Taking the maximum on the left-hand side gives

$$\|Cf - Cg\|_\infty \leqslant \beta \|f - g\|_\infty,$$

which confirms that C is a contraction of modulus β on $\mathbb{R}^{\mathsf{X}}$.

From the contraction property, we know that C has exactly one fixed point in $\mathbb{R}^{\mathsf{X}}$. Let $\bar{h}$ be this function. We claim that $\bar{h} = h^*$.

Let $\bar{v} := r \vee \bar{h}$. (We use functional notation here and for the rest of the proof, so that operations and relations are all pointwise.) To show that $\bar{h} = h^*$, it suffices to show that $\bar{v} = v^*$. Indeed, if $\bar{v} = v^*$, then

$$\bar{h} = C\bar{h} = c + \beta P(r \vee \bar{h}) = c + \beta P\bar{v} = c + \beta P v^* = h^*.$$

To see that $\bar{v} = v^*$, we again use $\bar{h} = C\bar{h}$ to obtain

$$r \vee \bar{h} = r \vee (C\bar{h}) = r \vee \left[c + \beta P(r \vee \bar{h})\right].$$

Using $\bar{v} = r \vee \bar{h}$, we can write this as $\bar{v} = r \vee [c + \beta P\bar{v}]$. This is just the Bellman equation in functional notation. As v^* is the only solution to the Bellman equation in $\mathbb{R}^{\mathsf{X}}$ (Proposition 3.1.3), we have $\bar{v} = v^*$, as claimed. □

3.1.3.2 Dimensionality Reduction

In the discussion at the start of §3.1.3, we mentioned that switching to from value function iteration to continuation value iteration can greatly reduced the dimensionality of the problem in some cases. Here we try to pin down the cases where this works.

To begin, let W and Z be two finite sets and suppose that $\varphi \in \mathscr{D}(\mathsf{W})$ and Q is a stochastic matrix on Z. Let (W_t) be IID with distribution φ and let (Z_t) be an Q-Markov chain on Z. If (W_t) and (Z_t) are independent, then (X_t) defined by $X_t = (W_t, Z_t)$ is P-Markov on X, where

$$P(x, x') = P((w, z), (w', z')) = \varphi(w')Q(z, z').$$

Suppose that the continuation reward depends only on z. In this case, we can write the Bellman operator as

$$(Tv)(w, z) = \max\left\{e(w, z),\ c(z) + \beta \sum_{w' \in \mathsf{W}} \sum_{z' \in \mathsf{Z}} v(w', z')\varphi(w')Q(z, z')\right\}. \tag{3.15}$$

Since the right-hand side depends on both w and z, the Bellman operator acts in an n-dimensional space, where $n := |\mathsf{X}| = |\mathsf{W}| + |\mathsf{Z}|$.

However, if we inspect the right-hand side of (3.15), we see that the continuation value function depends only on z. Dependence on w is absent because w does not help predict w'. Thus, the continuation value function is an object in $(n - |\mathsf{W}|)$-dimensional space. The continuation value operator

$$(Ch)(z) = c(z) + \beta \sum_{w' \in \mathsf{X}} \sum_{z' \in \mathsf{X}} \max\left\{e(w', z'), h(z')\right\} P(z, z') \qquad (z \in \mathsf{Z})$$

acts in this lower dimensional-space.

Example 3.1.4. We can embed the IID the job search problem into this setting by taking (W_t) to be the wage offer process and (Z_t) to be constant. This is why the IID case offers a large dimensionality reduction when we switch to continuation values.

More examples of dimensionality reduction are shown in the applications below.

3.2 Applications

Add roadmap.

3.2.1 American Options

American options were introduced briefly in Example 3.1.2 on page 90. Here we investigate this class of derivatives more carefully. We focus on American call options, which provide the right to buy a given asset (e.g., 1,000 shares in some underlying equity) at any time during some specified period at some fixed **strike price** K. The market price of the asset at time t is denoted by S_t.

The infinite horizon case was discussed in Example 3.1.2. However, options without termination dates—also called perpetual options—are rare in practice. Hence we focus on the finite-horizon case. We are interested in computing the expected value of holding the option when discounting with a fixed interest rate. This is a standard approach to pricing American options.

Finite horizon American options can be solved by backwards induction, analogous to the finite horizon job search problem discussed in Chapter 1. Alternatively, we can embed finite horizon options into the theory of infinite-horizon optimal stopping. This second approach is convenient for us, since the theory of infinite-horizon optimal stopping has already been presented.

To perform this embedding, we take $T \in \mathbb{N}$ to be a fixed integer indicating the date of expiration. The option is purchased at $t = 0$ and can be exercised at $t \in \mathbb{N}$ with $t \leqslant T$. To include t in the current state, we set

$$\mathsf{T} := \{1, \dots, T + 1\} \quad \text{and} \quad m(t) := \min\{t + 1, T + 1\} \text{ for all } t \in \mathsf{T}.$$

The idea is that time is updated via $t' = m(t)$, so that time increments at each update until $t = T + 1$. After that we hold t constant. Bounding time at $T + 1$ keeps the state space finite.

We assume that the stock price S_t evolves according to

$$S_t = Z_t + W_t \quad \text{where} \quad (W_t)_{t \geqslant 0} \overset{\text{IID}}{\sim} \varphi \in \mathscr{D}(\mathsf{W}).$$

Here $(Z_t)_{t \geqslant 0}$ is Q-Markov on finite set Z for some stochastic matrix Q, W is a finite set and s is a parameter that controls volatility. This means that the share price is affected by a persistent and purely transient component. We choose parameters such that $(Z_t)_{t \geqslant 0}$ is close to a random walk, implying that price changes are difficult to predict.[1]

To convert these update rules into an optimal stopping problem, as defined in §3.1.1.1, we need to specify the state and clarify the stochastic matrix P on X that maps to the state process. We set the state space to $\mathsf{X} := \mathsf{T} \times \mathsf{W} \times \mathsf{Z}$ and

$$P((t, w, z), (t', w', z')) := \mathbb{1}\{t' = m(t)\} \varphi(w') Q(z, z').$$

[1]Random walks are discussed in depth in Chapter 6.

In other words, time updates deterministically via $t' = m(t)$ and z' and w' are drawn independently from $Q(z, \cdot)$ and φ respectively.

As in the perpetual option case, the continuation reward is zero and the discount rate is $\beta := 1/(1+r)$, where $r > 0$ is a fixed risk-free rate. The exit reward can be expressed as $\mathbb{1}\{t \leqslant T\}(S_t - K)$. In other words, exercise at time t earns the owner $S_t - K$ up to expiry and zero thereafter. In terms of the state (t, z), the exit reward is

$$e(t, w, z) := \mathbb{1}\{t \leqslant T\}[z + w - K].$$

The Bellman equation can be written as

$$v(t, w, z) = \max\left\{e(t, w, z),\ \beta \sum_{w'} \sum_{z'} v(t', w', z')\varphi(w')Q(z, z')\right\},$$

where $t' = m(t)$. This relationship neatly captures the value of the option: It is the maximum of current exercise value and the discounted expected value of of carrying the option over to the next period.

Since the problem described above is an optimal stopping problem in the sense of §3.1.1.1, all of the optimality results described above apply. In particular, iterates of the Bellman operator converge to the value function v^* and, moreover, a policy is optimal if and only if it is v^*-greedy.

We can do better than value function iteration. Since $(W_t)_{t \geqslant 0}$ is IID and appears only in the exit reward, we can reduce dimensionality by switching to the continuation value operator, which, in this case, can be expressed as

$$(Ch)(t, z) = \beta \sum_{z'} \sum_{w'} \max\left\{e(t', w', z'),\ h(t', z')\right\} Q(z, z'). \tag{3.16}$$

As proved in §3.1.3, the unique fixed point of C is the continuation value function h^*, and $C^k h \to h^*$ as $k \to \infty$ for all $h \in \mathbb{R}^{\mathsf{X}}$. With the fixed point in hand, we can compute the optimal policy as

$$\sigma^*(t, w, z) = \mathbb{1}\left\{e(t, w, z) \geqslant h^*(t, z)\right\}.$$

Here $\sigma^*(t, w, z) = 1$ indicates exercise of the option at time t.

Figure 3.1 provides a visual representation of optimal actions under the default parameterization described in Listing 11. Each of the three figures show contour lines of the net exit reward $f(t, w, z) := e(t, w, z) - h^*(w, z)$, viewed as a function of (w, z), when t is held fixed. The date t for each subfigure is shown in the title. The optimal policy exercises the option when $f(t, w, z) \geqslant 0$.

```julia
using QuantEcon, LinearAlgebra, IterTools

"Creates an instance of the option model with log S_t = Z_t + W_t."
function create_american_option_model(;
        n=100, μ=10.0,  # Markov state grid size and mean value
        ρ=0.98, ν=0.2,  # persistence and volatility for Markov state
        s=0.3,          # volatility parameter for W_t
        r=0.01,         # interest rate
        K=10.0, T=200)  # strike price and expiration date
    t_vals = collect(1:T+1)
    mc = tauchen(n, ρ, ν)
    z_vals, Q = mc.state_values .+ μ, mc.p
    w_vals, φ, β = [-s, s], [0.5, 0.5], 1 / (1 + r)
    e(t, i_w, i_z) = (t ≤ T) * (z_vals[i_z] + w_vals[i_w] - K)
    return (; t_vals, z_vals, w_vals, Q, φ, T, β, K, e)
end
```

Listing 11: Pricing and American option (`american_option.jl`)

In each subfigure, the **exercise region**, which is the set (w, z) such that $f(t, w, z) \geqslant 0$, correspond to the northeast part of the figure, where w and z are both large. The boundary between exercise and continuing is the zero contour line, which is shown in black. Notice that the size of the the exercise region expands with t. This is because the value of waiting decreases when the set of possible exercise dates declines.

Figure 3.2 provides some simulations of the stock price process $(S_t)_{t \geqslant 0}$ over the lifetime of the option, again using the default parameterization described in Listing 11. The blue region in the top part of each subfigure is the values of the stock price $S_t = Z_t + W_t$ such that $S_t \geqslant K$. An option traded in this configuration (where the price of the underlying exceeds the strike price) is said to be "in the money." The figure also shows the optimal exercise date in each of the simulations, which is the first t such that $e(t, W_t, Z_t) \geqslant h^*(W_t, Z_t)$.

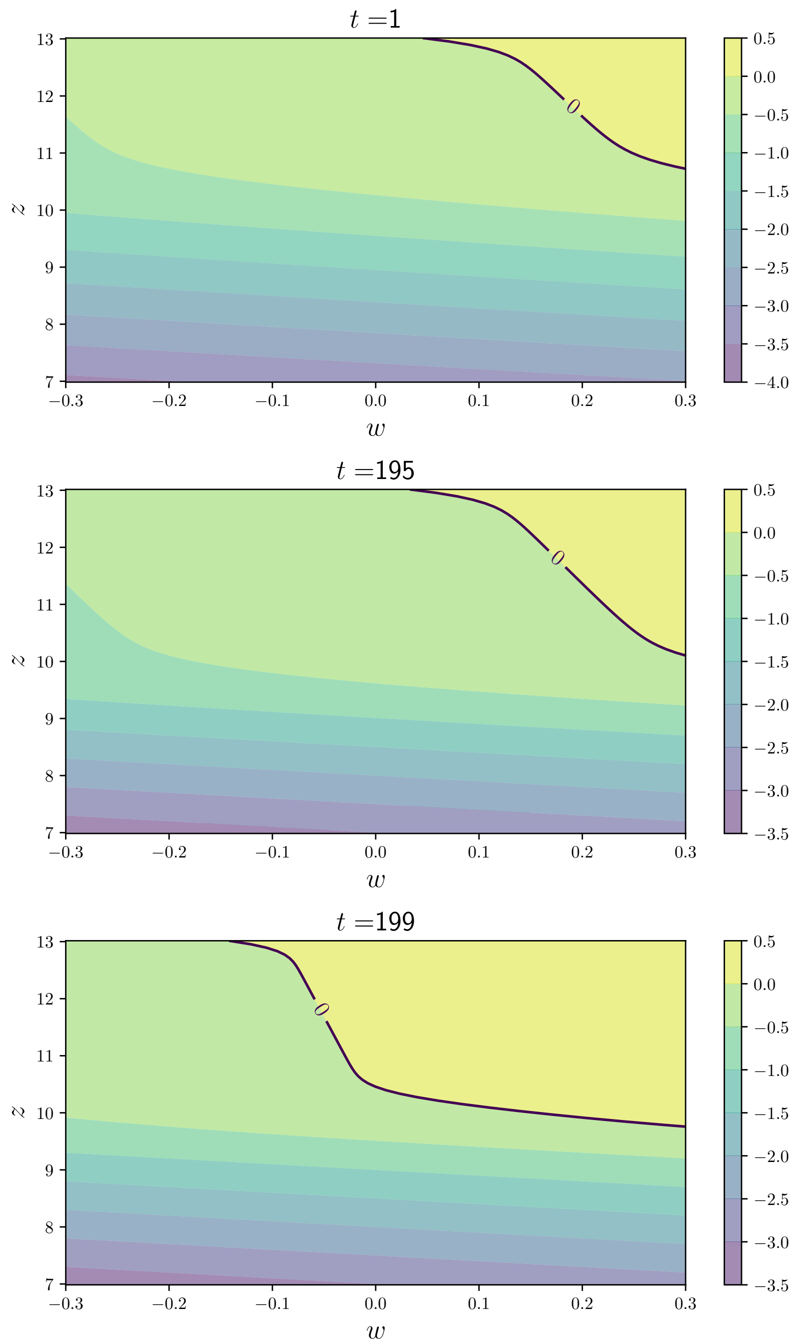


Figure 3.1: Exercise region for the American option

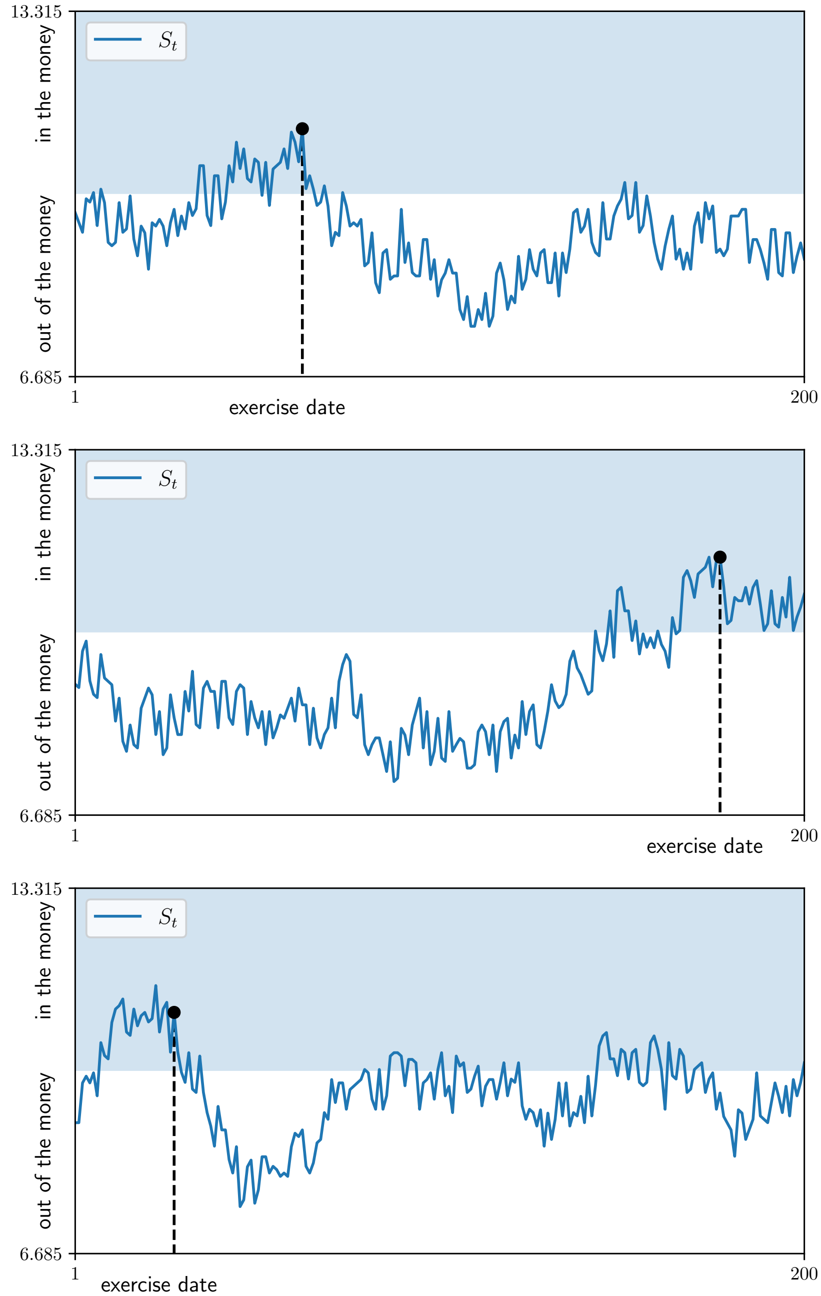


Figure 3.2: Simulations for the American option process

3.2.2 Firm Valuation with Exit

A basic firm entry-exit model.

$$v(z, p) = \max \left\{ s, \max_{\ell \geqslant 0} \pi(\ell, z, p) + \beta \sum_{p'} \sum_{z'} v(z', p') Q(z, z') R(p, p') \right\} \tag{3.17}$$

3.2.3 Further Applications

Shorter applications. Add roadmap.

3.2.3.1 Research and Development

Consider a firm that engages in costly research and development (R&D) in order to develop a new product. The dynamic problem faced by the firm is whether to hold back and continue investing in the project or stop and bring the product to market. For simplicity, we assume here that the value of bringing the product to market is a one-off payoff $\pi_t = \pi(X_t)$, where (X_t) is Markov chain on finite set X with stochastic matrix P. The flow cost of investing in R&D is c per period. Future payoffs are discounted at rate $r > 0$ and we set $\beta := 1/(1 + r)$.

This is an optimal stopping problem with exit reward $e = \pi$ and constant continue reward $-c$. The Bellman equation problem is

$$(Tv)(x) = \max \left\{ \pi(x), -c + \beta \sum_{x'} v(x') P(x, x') \right\} \qquad (x \in \mathsf{X}). \tag{3.18}$$

Exercise 3.2.1. Write down the continuation value operator for this problem. Prove that the continuation value function h^* is increasing in x whenever $\pi \in i\mathbb{R}^{\mathsf{X}}$ and P is monotone increasing.

Exercise 3.2.2. Prove that the optimal policy σ^* is increasing whenever π is increasing and (X_t) is IID (so that all rows of P are identical). Provide economic intuition for this result.

Monotonicity wrt β and c.

3.2.3.2 Replacement Problems

Bus engines, etc.

3.3 Chapter Notes

For interesting applications of optimal stopping in economics and finance, see, for example, McCall (1970), **?**, Hopenhayn (1992), **?**, **?**, **?**, Arellano (2008), **?**, Fajgelbaum et al. (2017) and **?**.

Chapter 4

Finite Markov Decision Processes

In this chapter we summarize a class of discrete time, infinite horizon dynamic programs called finite state Markov decision processes (finite MDPs). This class of problems is broad enough to encompass a very large range of applications, including the job search model we studied in §1.4. It also provides the standard departure point for reinforcement learning, which combines statistical and artificial intelligence methods with dynamic programming in order to handle real-world settings where information on the underlying model is incomplete.

4.1 Finite MDPs

In this section we defined finite MDPs and investigate their fundamental properties.

4.1.1 The Finite MDP Model

Finite MDPs are characterized by two features: rewards are additively separable and the state and action space are finite. Additive separability of rewards will be explained when we contrast it with other cases in Chapter 5. Finiteness of state and action spaces is self-explanatory. The finite case is routinely used in quantitative applications it facilitates computation.[1]

In what follows we require the following definition: A **correspondence** Γ from one set X to another set A is a function from X into $\wp(A)$, the set of all subsets of A. The

[1] In principle, finite states and actions can closely approximate the continuous case. For example, in the interval $[0, 1]$, there are more than one billion 64-bit floating point numbers.

correspondence is called **nonempty** if $\Gamma(x) \neq \emptyset$ for all $x \in \mathsf{X}$. For example, the map Γ defined by $\Gamma(x) = [-x, x]$ is a nonempty correspondence from $\mathbb{R}$ to $\mathbb{R}$.

4.1.1.1 Finite Markov Decision Process

We study a controller who interacts with a state process $(X_t)_{t \geqslant 0}$ by choosing an action path $(A_t)_{t \geqslant 0}$ to maximize expected discounted rewards

$$\mathbb{E} \sum_{t \geqslant 0} \beta^t r(X_t, A_t), \tag{4.1}$$

taking an initial state X_0 as given. As with the all dynamic programs, we insist that the controller is not clairvoyant: he or she cannot choose actions that depend on future states.

To formalize the problem, we take as given a finite set X, henceforth called the **state space**, and a finite set A, henceforth called the **action space**. Given X and A, we define a **finite Markov decision process** to be a tuple (Γ, β, r, P) where

(i) Γ is a correspondence from $\mathsf{X} \to \mathsf{A}$,

(ii) β is a constant in $(0, 1)$,

(iii) r is a function from G to $\mathbb{R}$, where $\mathsf{G} := \{(x, a) \in \mathsf{X} \times \mathsf{A} : a \in \Gamma(x)\}$, and

(iv) P is a **stochastic kernel** from G to X; that is, P is a map from $\mathsf{G} \times \mathsf{X}$ to $\mathbb{R}_+$ satisfying

$$\sum_{x' \in \mathsf{X}} P(x, a, x') = 1 \quad \text{for all } (x, a) \text{ in } \mathsf{G}.$$

In the sequel, Γ is called the **feasible correspondence**, β is called the **discount factor**, and r is called the **reward function**. The set G is called the set of **feasible state-action pairs**.

The feasible correspondence restricts actions, in the sense that $\Gamma(x) \subset \mathsf{A}$ is the set of actions available to the controller in state x. Given a feasible state-action pair (x, a), reward $r(x, a)$ is received and the next period state x' is selected from $P(x, a, \cdot)$, which is an element of $\mathscr{D}(\mathsf{X})$. The dynamics and reward flow are summarized in Algorithm 3.

The **Bellman equation** associated with this problem is

$$v(x) = \max_{a \in \Gamma(x)} \left\{ r(x, a) + \beta \sum_{x' \in \mathsf{X}} v(x') P(x, a, x') \right\} \qquad (x \in \mathsf{X}). \tag{4.2}$$

Algorithm 3: State, actions, and rewards

$t \leftarrow 0$
input x_0
while $t < \infty$ **do**
  observe X_t
  choose action A_t
  receive reward $r(X_t, A_t)$
  draw X_{t+1} from $P(X_t, A_t, \cdot)$
  $t \leftarrow t + 1$
end

This can be understood as an equation in the unknown function $v \in \mathbb{R}^{\mathsf{X}}$. Below we define the value function v^* as maximal lifetime rewards and show that v^* is the unique solution to the Bellman equation in $\mathbb{R}^{\mathsf{X}}$.

The idea behind the Bellman equation is that current actions influence the two terms on the right hand side: current rewards and expected discounted value from future states. In every case we examine, there is a trade-off between maximizing current rewards and influencing the distribution $P(x, a, \cdot)$ of the next period state in order to obtain high future rewards.

4.1.1.2 Example: Cake Eating

Many dynamic programming problems in economics involve a trade-off between current and future consumption. The simplest example in this class is the "cake eating" problem, where initial wealth is given but no labor income is received. We begin with a version where wealth evolves according to

$$W_{t+1} = RW_t - C_t \qquad (t = 0, 1, \ldots). \tag{4.3}$$

Here R is a gross rate of interest, so that investing d dollars today returns Rd next period, and C_t is current consumption. The agent seeks to maximize

$$\mathbb{E} \sum_{t \geqslant 0} \beta^t u(C_t) \quad \text{given } W_0 = w.$$

We assume that $C_t \geqslant 0$ and $W_t \geqslant 0$, so that the agent cannot borrow.

This model can easily be framed as a finite MDP. We assume that wealth takes values in a finite set W. We take wealth as the state variable and W as the state space.

The action is the savings decision $S_t = RW_t - C_t$, which is also equal to next-period wealth. Thus, the action space is also W.

Since consumption is nonnegative, the feasible correspondence is

$$\Gamma(w) = \{s \in \mathsf{W} : s \leqslant Rw\}.$$

The current reward is utility of consumption, or

$$r(w, s) = u(Rw - s)$$

The stochastic kernel is $P(w, s, w') = \mathbb{1}\{w' = s\}$. In other words, next period wealth w' is equal to savings s with probability one.

The Bellman equation is

$$v(w) = \max_{s \in \Gamma(w)} \left\{ r(w, s) + \beta \sum_{w' \in \mathsf{W}} v(w') P(w, s, w') \right\}.$$

We can eliminate s and write this more simply as

$$v(w) = \max_{w' \in \Gamma(w)} \{u(Rw - w') + \beta v(w')\}. \tag{4.4}$$

When we discuss optimality below, the agent will use this equation to trade-off current utility of consumption against the value of future wealth.

4.1.1.3 Example: Job Search

The optimal stopping problem we studied in Chapter 3 can be framed as a finite MDP. Here we show this for the job search problem of §1.4, leaving the general optimal stopping case as an additional exercise.

For the state space we take $\mathsf{X} = \{0, 1\} \times \mathsf{W}$, where W is the set of wage outcomes. A typical element is (e, w), with e representing unemployment ($e = 0$) or employment ($e = 1$) and w being the current wage offer. The action $a \in \mathsf{A} := \{0, 1\}$ indicates rejection or acceptance of the wage offer.

To reflect the assumption that jobs are permanent, the feasible correspondence is

$$\Gamma(x) = \Gamma(e, w) = \{a \in \{0, 1\} : a \geqslant e\}.$$

The set of feasible state-action pairs is, therefore, $\mathsf{G} = \{((e, w), a) \in \mathsf{X} \times \mathsf{A} : a \geqslant e\}$. The reward function is

$$r(x,a) = r((e,w),a) = aw + (1-a)c.$$

and the stochastic kernel $P[((e,w),a),\cdot]$ on G is the distribution corresponding to the random variable $x' := (e',w')$ described in Algorithm 4.

Algorithm 4: Updating the state conditional on current state and action

take current state (e,w) as input ;
if $e = 0$ *and* $a = 0$ **then** draw $w' \sim \varphi$ and $x' \leftarrow (0,w')$;
else $x' \leftarrow (1,w)$;
return x'

On one hand, $e = 1$ implies $a = 1$ and $e' = 1$, so the Bellman equation (4.2) at current state $x = (1,w)$ translates to

$$v(1,w) = w + \beta \mathbb{E} v(1,w).$$

Hence $v(1,w) = w/(1-\beta)$. On the other hand, at $e = 0$, the Bellman equation is

$$v(0,w) = \max_{a \in \{0,1\}} \left\{ aw + (1-a)c + \beta \mathbb{E}\left[av(1,w) + (1-a)v(0,w') \right] \right\}.$$

When $a = 1$, the right hand side is $w + \beta v(1,w) = w/(1-\beta)$. When $a = 0$ it is $c + \beta \mathbb{E} v(0,w')$. As a result, we can write

$$v(0,w) = \max\left\{ \frac{w}{1-\beta},\, c + \beta \sum_{w' \in \mathsf{W}} v(0,w')\varphi(w') \right\}. \tag{4.5}$$

This is exactly the Bellman equation (1.24) for an unemployed agent from the job search problem.

4.1.2 Optimality

We return to the general finite state MDP setting of §4.1.1.1. As was the case for job search, actions will be governed by policies, which are maps from states to actions (see, in particular, §1.4.1.3, where policies were introduced). The set of **feasible policies** is

$$\Sigma := \{\sigma \in \mathsf{A}^{\mathsf{X}} : \sigma(x) \in \Gamma(x) \text{ for all } x \in \mathsf{X}\}. \tag{4.6}$$

If we select a particular policy σ from Σ, it is understood that we respond to state X_t with action $A_t := \sigma(X_t)$ at every date t.

What happens if we commit to a policy σ in Σ for the lifespan of the problem? Now the state evolves by drawing X_{t+1} from $P(X_t, \sigma(X_t), \cdot)$ at every point in time. In other words, the state updates according to the transition matrix on $\mathsf{X} \times \mathsf{X}$ given by

$$P_\sigma(x, x') := P(x, \sigma(x), x') \qquad (x, x' \in \mathsf{X}).$$

Given initial condition $X_0 = x$, this process is a P_σ-Markov with initial condition x. Fixing a policy "closes the loop" in the state transition process and sets a given Markov chain for the state.

Under the policy σ, rewards at state x are $r(x, \sigma(x))$. If we introduce the notation

$$r_\sigma(x) := r(x, \sigma(x)) \qquad (x \in \mathsf{X})$$

and $\mathbb{E}_x := \mathbb{E}[\cdot \,|\, X_0 = x]$, then the expected time t reward is

$$\mathbb{E}_x \, r(X_t, A_t) = \mathbb{E}_x \, r_\sigma(X_t) = (P_\sigma^t \, r_\sigma)(x). \tag{4.7}$$

The lifetime value of following σ starting from state x can be written as

$$v_\sigma(x) = \mathbb{E}_x \sum_{t \geqslant 0} \beta^t r(X_t, \sigma(X_t)) = \sum_{t \geqslant 0} \beta^t \mathbb{E}_x r(X_t, \sigma(X_t)). \tag{4.8}$$

Using (4.7) and switching to vector notation, with v_σ and r_σ viewed as column vectors, we get

$$v_\sigma = \sum_{t \geqslant 0} \beta^t P_\sigma^t \, r_\sigma. \tag{4.9}$$

Analogous to the job search case, we call v_σ the **σ-value function**.[2] By Exercise 2.1.4, the spectral radius of βP_σ is just β, so, by the Neumann series lemma, we can also write

$$v_\sigma = (I - \beta P_\sigma)^{-1} \, r_\sigma. \tag{4.10}$$

The **value function** is defined as

$$v^*(x) = \sup_{\sigma \in \Sigma} v_\sigma(x) \qquad (x \in \mathsf{X}). \tag{4.11}$$

This is consistent with our definition of the value function in the optimal stopping case (see page 92). It is the maximal lifetime value we can extract from each state using optimal behaviour.

[2]Pushing the expectation through the sum is justified here because $\beta < 1$ and r is bounded. See Footnote 2.

The **Bellman operator** for the finite state MDP is the self-map T on $\mathbb{R}^{\mathsf{X}}$ defined by

$$(Tv)(x) = \max_{a \in \Gamma(x)} \left\{ r(x, a) + \beta \sum_{x' \in \mathsf{X}} v(x') P(x, a, x') \right\} \qquad (x \in \mathsf{X}). \tag{4.12}$$

Obviously $Tv = v$ if and only if v satisfies the Bellman equation (4.2).

Given $v \in \mathbb{R}^{\mathsf{X}}$, a policy $\sigma \in \Sigma$ is called ***v*-greedy** if

$$\sigma(x) \in \operatorname*{argmax}_{a \in \Gamma(x)} \left\{ r(x, a) + \beta \sum_{x' \in \mathsf{X}} v(y) P(x, a, x') \right\} \tag{4.13}$$

for all $x \in \mathsf{X}$. A policy $\sigma \in \Sigma$ is called **optimal** if $v_\sigma = v^*$. In other words, a policy is optimal if its lifetime value is maximal at each state. Here is our main result for this section.

Proposition 4.1.1. *For the finite state MDP described in §4.1.1.1,*

(i) *the value function v^* is the unique solution to the Bellman equation in $\mathbb{R}^{\mathsf{X}}$.*

(ii) *T is a contraction of modulus β on $\mathbb{R}^{\mathsf{X}}$ under the norm $\| \cdot \|_\infty$.*

(iii) *A feasible policy is optimal if and only it is v^*-greedy.*

(iv) *At least one optimal policy exists.*

Parts (i) and (ii) together imply that $Tv^* = v^*$ and $\|T^k v - v^*\|_\infty = O(\beta^k)$ for every $v \in \mathbb{R}^{\mathsf{X}}$. Hence, for finite MDPs, we can compute v^* by successive approximation.

A full proof of Proposition 4.1.1 can be constructed using arguments similar to those we used for the optimal stopping problem in Chapter 3. We provide a complete proof in a more general setting in Chapter 5.

EXERCISE 4.1.1. Prove that (iii) implies (iv).

It is important to understand the significance of (iii). Greedy policies are relatively easy to compute, in the sense that solving (4.33) at each x is easier than trying to directly solve the problem of maximizing lifetime value, since Σ is in general far larger than $\Gamma(x)$. Part (iii) tells us that solving the overall problem reduces to computing a v-greedy policy with the right choice of v. As for the job search problem, that choice is the value function v^*. Intuitively, v^* assigns the "correct" value to each state, in the sense of maximal lifetime value the controller can extract, so using v^* to calculate greedy policies leads to the optimal outcome.

4.1.3 Algorithms

In §1.4.2.1, where we considered solving the job search model, we presented an algorithm called value function iteration. In this section we present a generalization suitable for aribtrary MDPs. We also discuss two other methods, both of which can be faster in certain applications.

4.1.3.1 Value Function Iteration

The **value function iteration** for finite MDPs is very similar to that for the job search model (Algorithm 1). The general procedure is given by Algorithm 5.

Algorithm 5: Value function iteration for finite MDPs

---

input $v_0 \in \mathbb{R}^{\mathsf{X}}$, an initial guess of v^*
input τ, a tolerance level for error
$\varepsilon \leftarrow \tau + 1$
$k \leftarrow 0$
while $\varepsilon > \tau$ **do**
  for $x \in \mathsf{X}$ **do**
    $v_{k+1}(x) \leftarrow (Tv_k)(x)$
  end
  $\varepsilon \leftarrow \|v_k - v_{k+1}\|_\infty$
  $k \leftarrow k + 1$
end
Compute a v_k-greedy policy σ
return σ

In other words, we use successive approximation on T to compute an approximation v_k to the value function v^* and then take the v_k-greedy policy.

4.1.3.2 Howard Policy Iteration

Another algorithm for computing the optimal policy is **Howard policy iteration algorithm**. In essence, this method iterates between computing the value of a given policy and computing the greedy policy associated with that value. The full technique is described in Algorithm 6. A visualization of the algorithm is given in Figure 4.1.

One attractive feature of Howard policy function method is that, in a finite state setting, it always converges to the exact optimal policy in a finite number of steps.

Algorithm 6: Howard policy iteration for finite MDPs

input $\sigma_0 \in \Sigma$, an initial guess of σ^*
$k \leftarrow 0$
$\varepsilon \leftarrow 1$
while $\varepsilon > 0$ **do**
  $v_k \leftarrow$ the σ_k-value function $(I - \beta P_{\sigma_k})^{-1} r_{\sigma_k}$
  $\sigma_{k+1} \leftarrow$ a v_k greedy policy
  $\varepsilon \leftarrow \|\sigma_k - \sigma_{k+1}\|_\infty$
  $k \leftarrow k + 1$
end
return σ_k

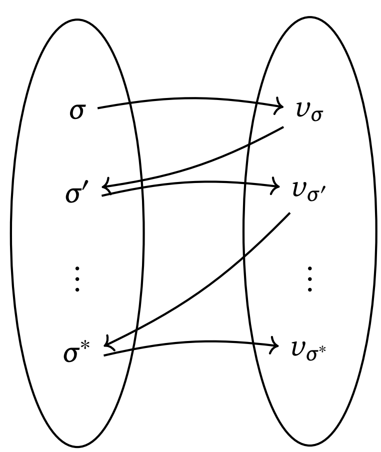


Figure 4.1: Howard policy function iteration algorithm

We prove this fact in a more general setting in Chapter 5. The basic intuition is that the value difference $v_{k+1}(x) - v_k(x)$ is strictly positive at at least one point in the state space when the current policy σ_k is not optimal. In other words, when the policy is not optimal there is always some strict improvement in value. Thus, the sequence of policies (σ_k) generated by the algorithm does not cycle. Since there are only finitely many policies in Σ, convergence is guaranteed.

4.1.3.3 Computing the Value of a Policy

One of the steps in Howard policy iteration routine is computing the value v_σ of a given policy σ. We have already noted that v_σ has both a geometric sum and a matrix inverse representation:

$$v_\sigma = \sum_{t \geqslant 0} \beta^t P_\sigma^t r_\sigma = (I - \beta\, P_\sigma)^{-1} r_\sigma \tag{4.14}$$

In terms of computation, matrix inversion is preferable when X is small. However, it is easy to write down dynamic programming problems where X is very large (see, e.g., Example 1.0.2 on page 2). If, say, X has 10^6 elements, then $I - \beta\, P_\sigma$ is $10^6 \times 10^6$. Matrices of this size are difficult invert—or even store in memory.

Another way to compute v_σ is by making use of the **policy operator** T_σ defined at $v \in \mathbb{R}^{\mathsf{X}}$ by

$$(T_\sigma\, v)(x) = r(x, \sigma(x)) + \beta \sum_{x' \in \mathsf{X}} v(x') P(x, \sigma(x), x') \qquad (x \in \mathsf{X}). \tag{4.15}$$

In vector notation, we can write this as

$$T_\sigma\, v = r_\sigma + \beta P_\sigma v. \tag{4.16}$$

(This definition is analogous to the policy operator defined for the optimal stopping problem in §3.1.1.3.) The next exercise shows how T_σ can be put to work.

Exercise 4.1.2. Fix σ in Σ. Using Banach's contraction mapping theorem, prove that the σ-value function v_σ is the unique fixed point of T_σ in $\mathbb{R}^{\mathsf{X}}$ and, in addition, $T_\sigma^k v \to v_\sigma$ as $k \to \infty$ for all $v \in \mathbb{R}^{\mathsf{X}}$.

Computationally, this means that we can pick $v \in \mathbb{R}^{\mathsf{X}}$ and iterate with T_σ to obtain an approximation to v_σ. This method is often feasible even when solving $(I - \beta P_\sigma)^{-1} r_\sigma$ is not possible.

EXERCISE 4.1.3. Prove that, when the initial condition for iteration is $v \equiv 0 \in \mathbb{R}^{\mathsf{X}}$, the k-th iterate $T_\sigma^k v$ is equal to the truncated sum $\sum_{t=0}^k \beta^t P_\sigma^t r_\sigma$.

4.1.3.4 Optimistic Policy Iteration

Optimistic policy iteration is an algorithm that borrows from both value function iteration and Howard policy iteration. In short, the algorithm is the same as policy iteration except that, instead of computing the full value v_σ of a given policy, the approximation $T_\sigma^k v$ discussed in Exercise 4.1.2 is used instead. Algorithm 7 provides details.

Algorithm 7: Optimistic policy iteration for finite MDPs

---

input $v_0 \in \mathbb{R}^{\mathsf{X}}$, an initial guess of v^*
input τ, a tolerance level for error
input $m \in \mathbb{N}$, a step size
$k \leftarrow 0$
$\varepsilon \leftarrow \tau + 1$
while $\varepsilon > \tau$ **do**
  $\sigma_k \leftarrow$ a v_k-greedy policy
  $v_{k+1} \leftarrow T_{\sigma_k}^m v_k$
  $\varepsilon \leftarrow \|v_k - v_{k+1}\|_\infty$
  $k \leftarrow k + 1$
end
return σ_k

In the algorithm, the policy operator T_{σ_k} is applied m times to generate an approximation of v_{σ_k}. The constant step size m can also be replaced with a sequence $(m_k) \subset \mathbb{N}$. In either case, for finite MDPs, convergence to an optimal policy is guaranteed. We prove this in a more general setting in Chapter 5.

Notice that, as $m \to \infty$, the algorithm increasingly approximates Howard policy iteration, since $T_{\sigma_k}^m v_k$ converges to v_{σ_k}. Moreover, if $m = 1$, the algorithm is essentially the same as value function iteration (the only difference being that the sequence of policies is not explicitly computed).

In many instances, a good choice of m will lead to faster convergence than both value function iteration and policy iteration. We investigate these ideas in the applications below.

4.1.4 Rates of Convergence

Quadratic convergence rates under Howard policy iteration.

4.2 Applications

This section gives several applications of the finite MDP model to economic problems. The applications illustrate the ease with which finite MDPs can be implemented numerically and solved on a computer (provided that the state and action spaces are relatively small).

4.2.1 Optimal Inventories

In §2.1.1.4 we studied a firm whose inventory behavior follows S-s dynamics. In this section we show how S-s behavior arises naturally in optimizing model, where the firm chooses its inventory path to maximize profits in each period. To keep the state space small, we assume for now that the firm only sells one product.

4.2.1.1 Environment

Given a demand shock process $(D_t)_{t \geqslant 0}$, inventory $(X_t)_{t \geqslant 0}$ of the product obeys

$$X_{t+1} = m(X_t - D_{t+1}) + A_t, \quad \text{where } m(y) := y \vee 0. \tag{4.17}$$

The term A_t is units of stock ordered this period, which take one period to arrive. We assume that the firm can store at most K items at one time, so the state space is $\mathsf{X} := \{0, \ldots, K\}$.

Profits are given by

$$\pi_t := X_t \wedge D_{t+1} - cA_t - \kappa \mathbb{1}\{A_t > 0\}.$$

We take the minimum of current stock and demand because orders in excess of inventory are assumed to be lost rather than backfilled. Here c is unit product cost and κ is a fixed cost of ordering inventory.

We assume IID demand shocks with common probability mass function $\varphi \in \mathscr{D}(\mathbb{Z}+)$.

With $\beta \in (0, 1)$ as the discount rate, the value of the firm is

$$V_0 = \mathbb{E} \sum_{t \geqslant 0} \beta^t \pi_t \tag{4.18}$$

The owners of the firm instruct management to maximize shareholder value. Let's now consider their optimization problem.

4.2.1.2 Optimization

The Bellman equation for this dynamic program is

$$v(x) = \max_{a \in \Gamma(x)} \left\{ \sum_{d \geqslant 0} (x \wedge d) \varphi(d) - ca - \kappa \mathbb{1}\{a > 0\} + \beta \sum_{d \geqslant 0} v(m(x - d) + a) \varphi(d) \right\}$$

at each $x \in \mathsf{X}$, where

$$\Gamma(x) := \{0, \ldots, K - x\} \tag{4.19}$$

is the set of feasible actions a when the current inventory state is x. The Bellman equation states that optimal value is attained when the firm chooses a to balance current expected profits with the value of a higher inventory next period.

Exercise 4.2.1. Write down the Bellman operator for this model and prove that this operator is a contraction of modulus β on $\mathbb{R}^{\mathsf{X}}$ when paired with the supremum norm $\|v\|_\infty := \sup_{x \in \mathsf{X}} |v(x)|$.

The unique solution to the Bellman equation is the optimal value function v^*, which, for each x in X, gives the maximal amount of lifetime value that can be extracted from that initial condition.

4.2.1.3 Representation as an MDP

We can map our inventory problem into a finite state MDP with state space X and action space $\mathsf{A} := \mathsf{X}$. The feasible correspondence Γ is as given in (4.19) and the reward function is current profits, or

$$r(x, a) := \sum_{d \geqslant 0} (x \wedge d) \varphi(d) - ca - \kappa \mathbb{1}\{a > 0\}.$$

The stochastic kernel from the set of feasible state-action pairs G induced by Γ is, in view of (4.17),

$$P(x, a, x') = \mathbb{P}\{m(x - D_{t+1}) + a = x'\}. \tag{4.21}$$

EXERCISE 4.2.2. Write down an expression for the stochastic kernel (4.21) using only x, a, x' and the parameters of the model. Assume that the demand shock has geometric distribution on $\mathbb{Z}_+$ with parameter p.

4.2.1.4 Computation

Let's now solve this model numerically. As in Exercise 4.2.2, we take φ to be the geometric distribution on $\mathbb{Z}_+$ with parameter p. We use the default parameter values shown in Listing 12. The code listing also presents an implementation of the Bellman operator. We use the `OffSetArrays` package to index arrays on the custom set `0:K`, since this corresponds to the state space.

Because the model is an MDP, the fixed point of the Bellman operator is the value function v^*, and a policy σ^* is optimal if and only if σ^* is v^*-greedy.

Figure 4.2 exhibits an approximation v^* of the value function, computed by iterating with T starting at $v \equiv 1$. Figure 4.2 also shows the approximate optimal policy, obtained as a v^*-greedy policy:

$$\sigma^*(x) \in \operatorname*{argmax}_{a \in \Gamma(x)} \left\{ r(x, a) + \beta \sum_{d \geqslant 0} v^*(m(x - d) + a)\varphi(d) \right\}$$

The plot of the optimal policy shows that there is a threshold region below which the firm orders large batches and above which the firm orders nothing. This is intuitive, since the firm wishes to economize on the fixed cost of ordering. Figure 4.3 shows a simulation of inventory dynamics under the optimal policy, starting from $X_0 = 0$. The time path closely approximates the S-s dynamics discussed in §2.1.1.4.

EXERCISE 4.2.3. Compute the optimal policy by extending the code given in Listing 12. Replicate Figure 4.3, modulo randomness, by sampling from a geometric distribution an implementing the dynamics in (4.17). At each X_t, the action A_t should be chosen according to the optimal policy $\sigma^*(X_t)$.

```julia
using Distributions, OffsetArrays
m(x) = max(x, 0)  # Convenience function

function create_inventory_model(; β=0.98,     # discount factor
                                  K=40,       # maximum inventory
                                  c=0.2, κ=2, # cost paramters
                                  p=0.6)      # demand parameter
    ϕ(d) = (1 - p)^d * p  # demand pdf
    return (; β, K, c, κ, p, ϕ)
end

"The function B(x, a, v) = r(x, a) + β Σ_x′ v(x′) P(x, a, x′)."
function B(x, a, v, model; d_max=100)
    (; β, K, c, κ, p, ϕ) = model
    reward = sum(min(x, d)*ϕ(d) for d in 0:d_max) - c * a - κ * (a > 0)
    continuation_value = β * sum(v[m(x - d) + a] * ϕ(d) for d in 0:d_max)
    return reward + continuation_value
end

"The Bellman operator."
function T(v, model)
    (; β, K, c, κ, p, ϕ) = model
    new_v = similar(v)
    for x in 0:K
        Γx = 0:(K - x)
        new_v[x], _ = findmax(B(x, a, v, model) for a in Γx)
    end
    return new_v
end

"Get a v-greedy policy.  Returns a zero-based array."
function get_greedy(v, model)
    (; β, K, c, κ, p, ϕ) = model
    σ_star = OffsetArray(zeros(Int32, K+1), 0:K)
    for x in 0:K
        Γx = 0:(K - x)
        _, a_idx = findmax(B(x, a, v, model) for a in Γx)
        σ_star[x] = Γx[a_idx]
    end
    return σ_star
end
```

Listing 12: Solving the optimal inventory model (`inventory_dp.jl`)

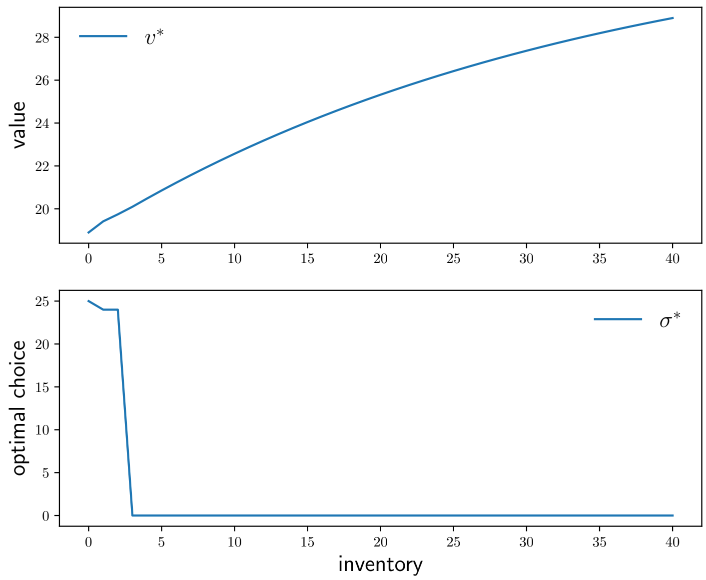


Figure 4.2: The value function and optimal policy for the inventory problem

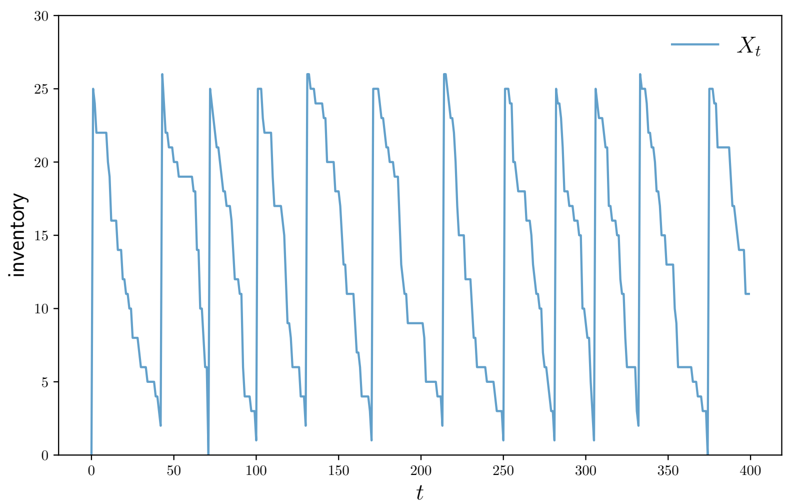


Figure 4.3: Optimal inventory dynamics

4.2.2 Optimal Savings with Labor Income

As our next example of a finite MDP, we modify the cake eating problem in §4.1.1.2 to add labor income. Wealth evolves according to

$$W_{t+1} = RW_t + Y_t - C_t \qquad (t = 0, 1, \ldots), \tag{4.23}$$

where (W_t) takes values in finite set $\mathsf{W} \subset \mathbb{R}_+$ and (Y_t) is a Markov chain on finite set Y with transition matrix Q. Other parts of the problem are unchanged.

4.2.2.1 Implementation

To frame this problem as a finite RDP, we set the state to $x := (w, y)$, representing current wealth and income, taking values in the state space $\mathsf{X} := \mathsf{W} \times \mathsf{Y}$. The feasible correspondence is the set of feasible savings values

$$\Gamma(w, y) = \{s \in \mathsf{W} : s \leqslant Rw + y\}.$$

The current reward is utility of consumption, or $r(w, s) = u(Rw + y - s)$. The stochastic kernel is

$$P((w, y), s, (w', y')) = \mathbb{1}\{w' = s\} Q(y, y').$$

With these definitions, the Bellman operator can be written either in the canonical form (4.12), in terms of state $x \in \mathsf{X}$, or in the simplified form

$$(Tv)(w, y) = \max_{w' \in \Gamma(w,y)} \left\{ u(Rw + y - w') + \beta \sum_{y' \in \mathsf{Y}} v(w', y') Q(y, y') \right\}, \tag{4.24}$$

in terms of state (w, y). The policy operator for given $\sigma \in \Sigma$ is given by

$$(T_\sigma v)(w, y) = u(Rw + y - \sigma(w, y)) + \beta \sum_{y' \in \mathsf{Y}} v(\sigma(w, y), y') Q(y, y'). \tag{4.25}$$

Code for implementing the model and these two operators is given in Listing 13. Income is constructed as a discretized AR(1) process using the method from §2.1.2. Exponentiation is applied to the grid so that income takes positive values.

The function `get_value` in Listing 14 uses the expression $v_\sigma = (I - \beta P_\sigma)^{-1} r_\sigma$ from (4.14) to obtain the value of a given policy σ. The matrix P_σ and vector r_σ take the form

$$P_\sigma((w,y),\sigma(w,y),(w',y')) = \mathbb{1}\{\sigma(w,y)=w'\}Q(y,y')$$
$$r_\sigma(w,y) = u(Rw + y - \sigma(w,y))$$

In order to use regular matrix algebra routines for this computation, we have mapped the indices `i, j` for state (w_i, y_j) into a single index m, as in $x_m = (w_i, y_j)$. The single index m steps through all points in the state space $\mathsf{X} = \mathsf{W} \times \mathsf{Y}$.

Remark 4.2.1. When mapping to a single index, we take into account the fact Julia uses Fortran style **column major** indexing of arrays. This means that when a two-dimensional array `a` with elements `a[i,j]` and indices `i in 1:wn` and `j in 1:yn` is flattened into a linear array `b` with elements `b[m]` and indices `m in 1:(wn*yn)`, the indices of `b` obey `m = i + (j-1) wn`. Visually, this means that the columns of `a` are stacked vertically into one long column. From single index `m` we can recover `i` via `(m-1)%wn + 1` and `j` via `div(m-1, wn) + 1`.

4.2.2.2 Solution and Timings

Since all of the results for finite MDPs (see §4.1.2–§4.1.3) are in effect, we know that the value function v^* is the unique fixed point of the Bellman operator in $\mathbb{R}^\mathsf{X}$, and that value function iteration, Howard policy iteration and optimistic policy iteration all converge. Listing 15 implements these three algorithms. Since the state and action space are finite, Howard policy iteration is guaranteed to return an exact optimal policy.

Figure 4.4 shows the wall time taken to solve the finite optimal savings model under the default parameters when executed on a standard laptop machine. Time is measured in seconds. The horizonal axis corresponds to the step parameter m in optimistic policy iteration (Algorithm 7). The two other algorithms do not depend on m and hence their timings are constant. The figure shows that policy iteration is an order of magnitude faster than value function iteration and optimistic policy iteration is even faster than policy iteration for moderate values of m.

These timings are implementation-dependent and relative speed varies significantly with the way that the algorithms are written, the extent to which parallelization can be exploited and the parameters and description of the problem. (In our case, all implementations aim at clarity rather than speed.) However, most experiment confirm that optimistic policy iteration is faster than value function iteration for many choices of the step size m, as well as being faster than Howard policy iteration for at least some values of m.

```
using QuantEcon, LinearAlgebra, IterTools

function create_savings_model(; R=1.01, β=0.99, γ=2.5,
                                w_min=0.01, w_max=5.0, w_size=200,
                                ρ=0.9, ν=0.1, y_size=5)
    w_grid = LinRange(w_min, w_max, w_size)
    mc = tauchen(y_size, ρ, ν)
    y_grid, Q = exp.(mc.state_values), mc.p
    return (; β, R, γ, w_grid, y_grid, Q)
end

"B(w, y, w′) = u(R*w + y - w′) + β Σ_y′ v(w′, y′) Q(y, y′)."
function B(i, j, k, v, model)
    (; β, R, γ, w_grid, y_grid, Q) = model
    w, y, w′ = w_grid[i], y_grid[j], w_grid[k]
    u(c) = c^(1-γ) / (1-γ)
    c = R*w + y - w′
    @views value = c > 0 ? u(c) + β * dot(v[k, :], Q[j, :]) : -Inf
    return value
end

"The Bellman operator."
function T(v, model)
    w_idx, y_idx = (eachindex(g) for g in (model.w_grid, model.y_grid))
    v_new = similar(v)
    for (i, j) in product(w_idx, y_idx)
        v_new[i, j] = maximum(B(i, j, k, v, model) for k in w_idx)
    end
    return v_new
end

"The policy operator."
function T_σ(v, σ, model)
    w_idx, y_idx = (eachindex(g) for g in (model.w_grid, model.y_grid))
    v_new = similar(v)
    for (i, j) in product(w_idx, y_idx)
        v_new[i, j] = B(i, j, σ[i, j], v, model)
    end
    return v_new
end
```

Listing 13: Discrete optimal savings model (`finite_opt_saving_0.jl`)

```
include("finite_opt_saving_0.jl")

"Compute a v-greedy policy."
function get_greedy(v, model)
    w_idx, y_idx = (eachindex(g) for g in (model.w_grid, model.y_grid))
    σ = Matrix{Int32}(undef, length(w_idx), length(y_idx))
    for (i, j) in product(w_idx, y_idx)
        _, σ[i, j] = findmax(B(i, j, k, v, model) for k in w_idx)
    end
    return σ
end


"Get the value of policy σ."
function get_value(σ, model)
    # Unpack and set up
    (; β, R, γ, w_grid, y_grid, Q) = model
    wn, yn = length(w_grid), length(y_grid)
    n = wn * yn
    u(c) = c^(1-γ) / (1-γ)
    # Function to map from single to multi index when m = i + (j-1)*wn"
    single_to_multi(m) = (m-1)%wn + 1, div(m-1, wn) + 1
    # Allocate and create single index versions of P_σ and r_σ
    P_σ = zeros(n, n)
    r_σ = zeros(n)
    for m in 1:n
        i, j = single_to_multi(m)
        r_σ[m] = u(R * w_grid[i] + y_grid[j] - w_grid[σ[i, j]])
        for m′ in 1:n
            i′, j′ = single_to_multi(m′)
            if i′ == σ[i, j]
                P_σ[m, m′] = Q[j, j′]
            end
        end
    end
    # Solve for the value of σ
    v_σ = (I - β*P_σ) \ r_σ
    # Return as multi-index array
    return reshape(v_σ, wn, yn)
end
```

Listing 14: Discrete optimal savings model (`finite_opt_saving_1.jl`)

```julia
include("s_approx.jl")
include("finite_opt_saving_1.jl")

"Value function iteration routine."
function value_iteration(model, tol=1e-5)
    vz = zeros(length(model.w_grid), length(model.y_grid))
    v_star = successive_approx(v -> T(v, model), vz, tolerance=tol)
    return get_greedy(v_star, model)
end

"Howard policy iteration routine."
function policy_iteration(model)
    wn, yn = length(model.w_grid), length(model.y_grid)
    σ = ones(Int32, wn, yn)
    i, error = 0, 1.0
    while error > 0
        v_σ = get_value(σ, model)
        σ_new = get_greedy(v_σ, model)
        error = maximum(abs.(σ_new - σ))
        σ = σ_new
        i = i + 1
        println("Concluded loop $i with error $error.")
    end
    return σ
end

"Optimistic policy iteration routine."
function optimistic_policy_iteration(model; tolerance=1e-5, m=100)
    v = zeros(length(model.w_grid), length(model.y_grid))
    error = tolerance + 1
    while error > tolerance
        last_v = v
        σ = get_greedy(v, model)
        for i in 1:m
            v = T_σ(v, σ, model)
        end
        error = maximum(abs.(v - last_v))
    end
    return get_greedy(v, model)
end
```

Listing 15: Discrete optimal savings model (`finite_opt_saving_2.jl`)

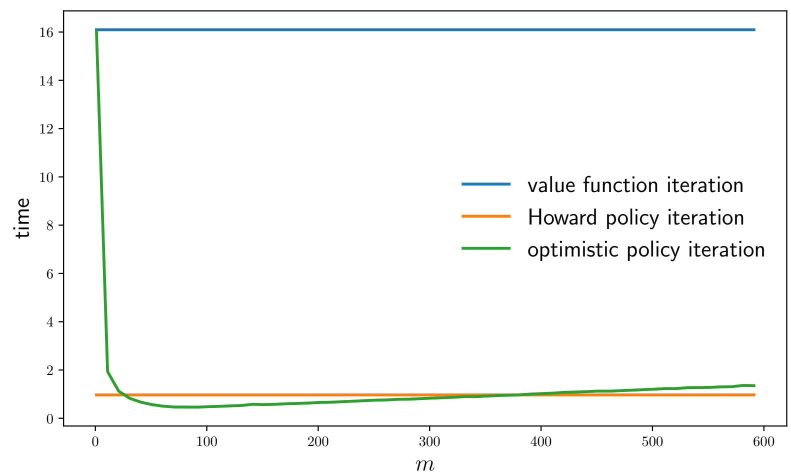


Figure 4.4: Timings for alternative algorithms

4.2.3 Optimal Investment

As our next application, we consider a monopolist facing correlated, stochastically evolving demand and adjustment costs. The trade-off in this dynamic programming problem involves balancing adjustment of capacity to meet demand against the costs associated with that adjustment.

4.2.3.1 Problem Description

We assume that the monopolist produces a single product and faces an inverse demand function of the form

$$P_t = a_0 - a_1 Y_t + Z_t,$$

where a_0, a_1 are positive parameters, Y_t is output, P_t is price and the demand shock Z_t follows

$$Z_{t+1} = \rho Z_t + \sigma \eta_{t+1}, \qquad \{\eta_t\} \stackrel{\text{IID}}{\sim} N(0,1).$$

Current profits are given by

$$\pi_t := P_t Y_t - c Y_t - \gamma (Y_{t+1} - Y_t)^2.$$

Here $\gamma (Y_{t+1} - Y_t)^2$ represents adjustment costs associated with changing production scale, parameterized by γ, and c is unit cost of current production. Costs are convex, so rapid changes to capacity are expensive.

The monopolist chooses (Y_t) to maximize the expected present value of its profit flow, which we write as

$$\mathbb{E} \sum_{t=0}^{\infty} \beta^t \pi_t. \tag{4.26}$$

Here $\beta = 1/(1+r)$, where $r > 0$ is a fixed interest rate.

One way to start thinking about the optimal time path of output is to consider what would happen if $\gamma = 0$. Without adjustment costs there is no intertemporal trade-off, so the monopolist should choose output to maximize current profit in each period. The implied level of output at time t is

$$\bar{Y}_t := \frac{a_0 - c + Z_t}{2a_1}. \tag{4.27}$$

Exercise 4.2.4. Show that $\bar{Y}_t$ maximizes current profit when $\gamma = 0$.

For $\gamma > 0$, we expect the following behavior.

- If γ is close to zero, then the optimal output path Y_t will track the time path of $\bar{Y}_t$ relatively closely, while
- if γ is larger, then Y_t will be significantly smoother than $\bar{Y}_t$, as the monopolist seeks to avoid adjustment costs.

4.2.3.2 Implementation

To implement this as a finite MDP, we let Y be a grid contained in $\mathbb{R}_+$ that lists possible output values. To conform to the finite state setting, we discretize the shock process (Z_t) using Tauchen's method, as described in §2.1.2. For convenience we again use (Z_t) to represent the discrete process, which is a finite Markov chain on $\mathsf{Z} \subset \mathbb{R}$ with transition matrix Q.

The state space for this MDP is $\mathsf{X} = \mathsf{Y} \times \mathsf{Z}$, while the action space is Y. The feasible correspondence is defined by $\Gamma(x) = \mathsf{Y}$, meaning that choice of output is not restricted by the state. The current reward function is current profits, which we can write as

$$r(y, z, y') = (a_0 - a_1 y + z - c)y - \gamma(y' - y)^2.$$

The stochastic kernel is

$$P((y, z), y', (y', z')) = \mathbb{1}\{y = y'\} Q(z, z').$$

With these definitions, the problem defines an MDP and all of the optimality theory for MDPs applies.

We can express the policy operator as

$$(T_\sigma v)(y, z) = r(y, z, \sigma(y, z)) + \beta \sum_{z' \in Z} v(\sigma(y, z), z') Q(z, z').$$

A v-greedy policy is a $\sigma \in \Sigma$ that obeys

$$\sigma(x) = \operatorname*{argmax}_{y' \in Y} \left\{ r(y, z, y') + \beta \sum_{z' \in Z} v(y', z') Q(z, z'). \right\}.$$

By combining iteration with the policy operator and computation of greedy policies, we can implement optimistic policy iteration, compute the optimal policy σ^*, and study the output choices generated by this policy. We are particularly interested in how output responds to randomly generated demand shocks over time.

Figure 4.5 shows the result of a simulation designed to shed light on how output responds to demand. After choosing a initial values (Y_1, Z_1) and generating a Q-Markov chain $(Z_t)_{t=1}^T$ from the transition matrix Q, we simulated optimal output via $Y_{t+1} = \sigma^*(Y_t, Z_t)$. The default parameters are shown in Listing 16. In the figure, the adjustment cost parameter γ is varied as shown in the title. In addition to the optimal output path, the path of $(\bar{Y}_t)$ as defined in (4.27) is also presented.

The figure shows how increasing γ promotes smoothing, as predicted in our discussion above. For small γ, adjustment costs have only minor impact on choices, so output closely follows $(\bar{Y}_t)$, the optimal path when output responds immediately to demand shocks. Conversely, larger values of γ make adjustment expensive, so the operator responds relatively slowly to changes in demand.

4.3 Time-Varying Discount Rates

A version of state-dependent discounting was introduced when we looked at firm valuation with time-varying interest rates (see §2.2.2.2). While optimality theory for models using stochastic discounting involves some challenges, it also opens opportunities: For example, if we can handle state-dependent discounting, then we can bring our models closer to the data and examine interesting questions. For example, how do volatility and persistence in interest rates affect investment, savings and consumption?

Figure 4.5: Simulation of optimal output with different adjustment costs

```julia
using QuantEcon, LinearAlgebra, IterTools

function create_investment_model(;
        r=0.04,                                   # Interest rate
        a_0=10.0, a_1=1.0,                        # Demand parameters
        γ=25.0, c=1.0,                            # Adjustment and unit cost
        y_min=0.0, y_max=20.0, y_size=100,        # Grid for output
        ρ=0.9, ν=1.0,                             # AR(1) parameters
        z_size=25)                                # Grid size for shock
    β = 1/(1+r)
    y_grid = LinRange(y_min, y_max, y_size)
    mc = tauchen(y_size, ρ, ν)
    z_grid, Q = mc.state_values, mc.p
    return (; β, a_0, a_1, γ, c, y_grid, z_grid, Q)
end
```

Listing 16: Discrete optimal investment model (`finite_lq.jl`)

In this section we extend the finite MDP framework to handle state-dependent discounting, provide optimality theory and cover applications. The discussion will be to follow if you have already read the discussion of firm valuation in §2.2.2.2.

4.3.1 MDPs with State-Dependent Discounting

Let's now introduce a variation on finite MDPs that allows for state-dependent discounting. We provide a framework for this setting and optimality results similar to the regular MDP case. The optimality results hold under a relatively weak spectral radius condition.

4.3.1.1 Definition

Let A be a finite set, referred to below as the **action space**. Regarding the state space, we assume that the state X_t can be decomposed into a pair (Y_t, Z_t), where $(Y_t)_{t \geqslant 0}$ is endogenous (i.e., affected by the actions of the controller) and $(Z_t)_{t \geqslant 0}$ is purely exogenous. The state space takes the form $\mathsf{X} = \mathsf{Y} \times \mathsf{Z}$, where Y and Z are finite sets.

Given A and X as defined above, a finite **MDP with state-dependent discounting** is a tuple (Γ, β, r, Q, R) where

(i) Γ is a correspondence from $\mathsf{Y} \to \mathsf{A}$,

(ii) β is a function from Z to $\mathbb{R}_+$,

(iii) r is a function from $\mathsf{G} := \{(y, a) \in \mathsf{Y} \times \mathsf{A} : a \in \Gamma(y)\}$ to $\mathbb{R}$,

(iv) Q is a stochastic matrix on Z and

(v) R is a stochastic kernel from G to Y.

The Bellman equation corresponding to the MDP with state-dependent discounting is

$$v(y, z) = \max_{a \in \Gamma(y)} \left\{ r(y, a) + \beta(z) \sum_{z' \in \mathsf{Z}} \sum_{y' \in \mathsf{Y}} v(y', z') Q(z, z') R(y, a, y') \right\} \tag{4.28}$$

for all $(y, z) \in \mathsf{X}$. The model can be understood as follows.

- (Z_t) is Q-Markov, where Q is a stochastic matrix on finite set Z.
- The **discount factor process** $(\beta_t)_{t \geqslant 0}$ is defined by $\beta_t := \beta(Z_t)$.
- Given $Y_t = y$ and current action a, current reward is $r(y, a)$ and Y_{t+1} is drawn from distribution $R(y, a, \cdot)$.
- Y_{t+1} and Z_{t+1} are updated independently given the time t state and action, which is why we take the product of Q and R in (4.28).

As before, G is called the set of feasible-state action pairs. We let Σ denote the set of feasible policies, which are all $\sigma \colon \mathsf{Y} \to \mathsf{A}$ such that $\sigma(y) \in \Gamma(y)$ for all $y \in \mathsf{Y}$.

4.3.1.2 Lifetime Value

To define lifetime value of a policy $\sigma \in \Sigma$ we introduce the **policy operator**

$$(T_\sigma v)(y, z) = r(y, \sigma(y, z)) + \beta(z) \sum_{z' \in \mathsf{Z}} \sum_{y' \in \mathsf{Y}} v(y', z') Q(z, z') R(y, \sigma(y, z), y') \tag{4.29}$$

for all $(y, z) \in \mathsf{X}$.

To gain a better understanding of T_σ, and to provide a form more suitable for computation, we extend β to all of X via $\beta(x) = \beta(z, y) := \beta(z)$ and, given $\sigma \in \Sigma$, define r_σ, P_σ and K_σ by

- $r_\sigma(x) := r_\sigma(y, z) := r(y, \sigma(y, z))$ and

- $P_\sigma(x, x') := P_\sigma((y, z), (y', z')) := Q(z, z')R(y, \sigma(y, z), y')$.

The stochastic matrix P_σ drives the state process $(X_t)_{t \geqslant 0}$ under policy σ.

We can now state the following key result, which shows that lifetime value is well-defined under a spectral radius condition.

Proposition 4.3.1. *Let L be defined by $L(z, z') := \beta(z)Q(z, z')$. If $r(L) < 1$, then, for each $\sigma \in \Sigma$, the operator T_σ has a unique fixed point $\mathbb{R}^{\mathsf{X}}$, denoted by v_σ. Moreover, for each $x \in \mathsf{X}$, the fixed point v_σ obeys*

$$v_\sigma(x) = \mathbb{E} \left\{ \sum_{t=0}^{\infty} \left[\prod_{i=0}^{t-1} \beta(X_i) \right] r_\sigma(X_t) \right\} \tag{4.30}$$

when (X_t) is P_σ-Markov with initial condition x.

Here, by convention, $\prod_{i=0}^{-1} := 1$. Recalling that $\beta(X_i) = \beta(Z_i)$, the term $\prod_{i=0}^{t-1} \beta(X_i)$ is the discount factor applied to current reward $r_\sigma(X_t)$ under the policy σ. Equation (4.30) tells us that lifetime rewards is the expected value of the sum of these discounted rewards. The proof below exploits the similarity of (4.30) to the expression for firm value given on page 69.

Proof of Proposition 4.3.1. Let $K_\sigma(x, x') := \beta(x)P_\sigma(x, x')$. With this notation, we can write the policy operator T_σ from (4.29) as

$$(T_\sigma\, v)(x) = r_\sigma(x) + \sum_{x'} v(x')K_\sigma(x, x') \qquad (x \in \mathsf{X}), \tag{4.31}$$

Assume for now that $r(K_\sigma) < 1$. We need to show that v_σ in (4.30) is the unique fixed point of T_σ in $\mathbb{R}^{\mathsf{X}}$.

Expression (4.30) is identical to the firm valuation in (2.24) on page 69 after replacing r_σ by π. Hence, via an essentially identical argument to the one provided for Proposition 2.2.4 on page 72, we see that $r(K_\sigma) < 1$ implies v_σ in (4.30) is finite for all x, that $I - K_\sigma$ is nonsingular, and, in addition, that

$$v_\sigma = (I - K_\sigma)^{-1} r_\sigma. \tag{4.32}$$

Rearranging gives $v_\sigma = r_\sigma + K_\sigma\, v_\sigma$. Moreover, by the uniqueness component of the Neumann series lemma, no other $v \in \mathbb{R}^{\mathsf{X}}$ obeys $v = r_\sigma + K_\sigma v$. Since T_σ in (4.31) can be expressed as $T_\sigma\, v = r_\sigma + K_\sigma\, v$, this means that v_σ is the unique fixed point of T_σ in $\mathbb{R}^{\mathsf{X}}$, as was to be shown.

We have established that all the results in Proposition 4.3.1 hold when $r(K_\sigma) < 1$. However, Proposition 4.3.1 assumes only that $r(L) < 1$. Hence, to complete the proof, we still need to verify that $r(L) < 1$ implies $r(K_\sigma) < 1$ for all σ. This is left as an exercise (see below). □

EXERCISE 4.3.1. Prove that $r(K_\sigma) \leqslant r(L)$ for all $\sigma \in \Sigma$.

Notice that the proof of Proposition 4.3.1 also provides us with a convenient way to compute lifetime value, via (4.32). With lifetime value firmly defined, at least when $r(L) < 1$, we can introduce the **value function** v^* via $v^*(x) = \max_{\sigma \in \Sigma} v_\sigma(x)$.

4.3.1.3 Optimality Results for MDPs with State-Dependent Discounting

Given $v \in \mathbb{R}^{\mathsf{X}}$, a policy $\sigma \in \Sigma$ is called **v-greedy** if

$$\sigma(y, z) \in \operatorname*{argmax}_{a \in \Gamma(y)} \left\{ r(y, a) + \beta(z) \sum_{z' \in \mathsf{Z}} \sum_{y' \in \mathsf{Y}} v(y', z') Q(z, z') R(y, a, y') \right\} \tag{4.33}$$

for all $(y, z) \in \mathsf{X}$. A policy $\sigma \in \Sigma$ is called **optimal** if $v_\sigma = v^*$. In other words, a policy is optimal if its lifetime value is maximal at each state.

For the MDP with state dependent discounting described in §4.3.1.1, we can state the following result, which echos the optimality results obtained for ordinary MDPs in §4.1.2.

Proposition 4.3.2. *If $r(L) < 1$, then T is globally stable on $\mathbb{R}^{\mathsf{X}}$ with unique fixed point v^*. Likewise, for each $\sigma \in \Sigma$, T_σ is globally stable on $\mathbb{R}^{\mathsf{X}}$ with unique fixed point v_σ. Moreover, a feasible policy is optimal if and only it is v^*-greedy. At least one optimal policy exists.*

Rather than proving Proposition 4.3.2 here, we will prove a more general result in §5.2.4, in a setting of abstract dynamic programs.

4.3.1.4 Comments on the Conditions

All of theory for MDPs with state-dependent discounting revolves around the assumption $r(L) < 1$. How strict is this condition?

To put this question in context, recall that the key assumption we made for regular MDPs (in §4.1) was: the discount factor is constant and strictly less than one. This assumption gave us contractivity of the Bellman operator and the policy operator, which were central to the optimality results

A natural extension to state-dependent discounting, where β is a function, is to assume the existence of a $b < 1$ such that $\beta(z) \leqslant b$ for all $z \in \mathsf{Z}$. Let's call this condition "strict state-dependent discounting."

EXERCISE 4.3.2. Prove that strict state-dependent discounting implies $r(L) < 1$.

While strict state-dependent discounting is easier to state and understand than the spectral radius condition, there is a good reason to avoid it. The reason is that the real interest rate r_t is sometimes negative, as shown in Figure 2.6 on 69. This means that, when discounting with real rates, the associated discount factor $\beta_t = 1/(1 + r_t)$ is sometimes greater than 1.

The condition $r(L) < 1$ allows the discount rate to exceed one at times, provided that the long-run average is strictly less than one.

Complete discussion.

4.3.2 Application: Inventory Management

The inventory management problem from §4.2.1 used a constant discount rate $\beta = 1/(1+r)$. We can introduce stochastic interest rates by using the results on MDPs with state-dependent discounting. This will allow us to investigate how interest dynamics affect inventories of firms.

To this end, let Z be a finite subset of $\mathbb{R}_+$ and let (Z_t) be Q-Markov on Z for some stochastic matrix Q. Let $\beta_t = \beta(Z_t)$ for some $\beta \in \mathbb{R}_+^{\mathsf{Z}}$. (We model dynamics of (β_t) directly, rather than (r_t), but a more detailed study would probably start by estimating dynamics of the interest rate.)

With other aspects of the problem unchanged, the Bellman equation for this dynamic program is

$$v(x, z) = \max_{a \in \Gamma(x)} \left\{ r(x, a) + \beta(z) \sum_{z'} \sum_{d \geqslant 0} v(m(x - d) + a, z') \varphi(d) Q(z, z') \right\}$$

at each $(x, z) \in \mathsf{X} \times \mathsf{Z}$, where $r(x, a) := \sum_{d \geqslant 0} (x \wedge d) \varphi(d) - ca - \kappa \mathbb{1}\{a > 0\}$ and $m(y) := y \vee 0$. Set $L(z, z') := \beta(z) Q(z, z')$. This model fits the structure of an MDP with state-dependent discounting, and the optimality results in Proposition 4.3.2 apply whenever $r(L) < 1$.

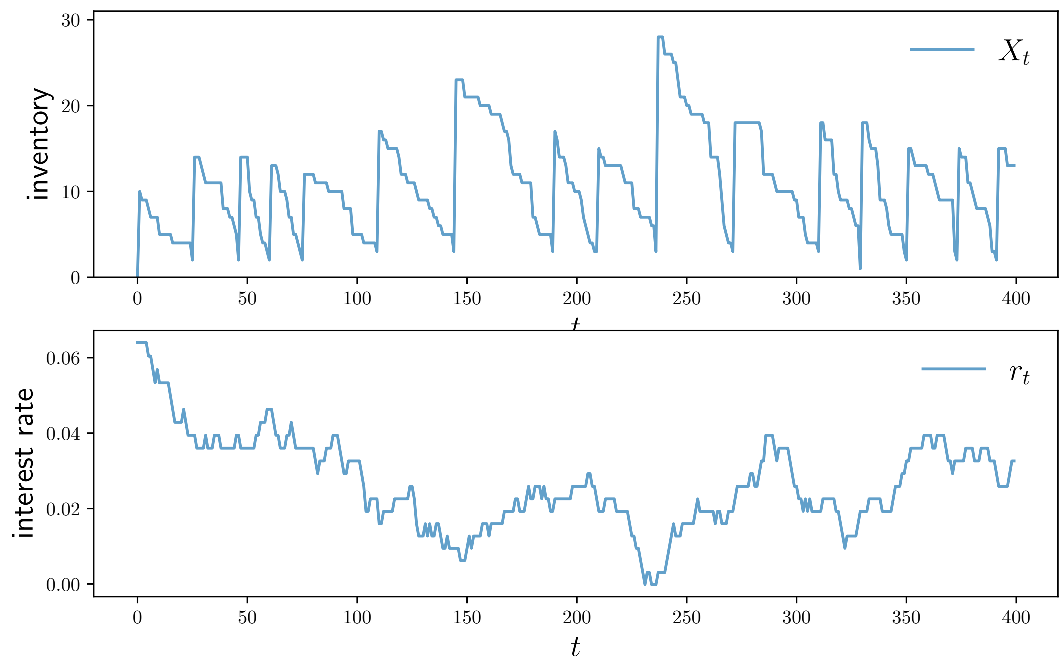


Figure 4.6: Inventory dynamics with time-varying interest rates

Figure 4.6 shows how inventory evolves under the optimal program when the parameters of the problem are as given in Listing 17. (The code in this listing includes a test for $r(L) < 1$.) Here we set $\beta(z) = z$ and take (Z_t) to be a discretization of an AR(1) process. Figure 4.6 was created by simulating (Z_t) according to Q and inventory (X_t) according to $X_{t+1} = m(X_t - D_{t+1}) + A_t$, where A_t follows the optimal policy.

The outcome is similar to Figure 4.3, in the sense that inventory falls slowly and then jumps up. As before, this lumpy behavior is down to fixed costs. However, a new phenomenon is now present: inventories move up or down on average, trending up as interest rates fall and down as interest rates rise. (The interest rate r_t is calculated via $\beta_t = 1/(1 + r_t)$ at each t.) In essence, high interest rates devalue future profits, which in turn encourages managers to economize on stock. Inventory management is one channel through which high interest rates suppress demand and low interest rates promote it.

```julia
using LinearAlgebra, Distributions, OffsetArrays, QuantEcon

function create_sdd_inventory_model(;
        ρ=0.98, ν=0.002, n_z=20, b=0.97,  # Z state parameters
        K=40, c=0.2, κ=0.8, p=0.6)        # firm and demand parameters
    ϕ(d) = (1 - p)^d * p                  # demand pdf
    mc = tauchen(n_z, ρ, ν)
    z_vals, Q = mc.state_values .+ b, mc.p
    rL = maximum(abs.(eigvals(z_vals .* Q)))
    @assert  rL < 1 "Error: r(L) ≥ 1."    # check r(L) < 1
    return (; K, c, κ, p, ϕ, z_vals, Q)
end
```

Listing 17: Investment model with time-varying discounting (`inventory_sdd.jl`)

4.4 Chapter Notes

Extensive discussion of finite MDPs can be found in Puterman (2005), Bertsekas (2012), Stachurski (2022) and Kochenderfer et al. (2022). The treatment in Puterman (2005) is particularly thorough.

The optimal savings problem is a workhorse in macroeconomics and has been treated extensively in the literature. We provide an extended discussion and references in Chapter **??**. The optimal investment problem dates back to Lucas (1978). A textbook treatment can be found in Dixit and Pindyck (2012).

Regarding the S-s inventory model, classic papers in the field include Arrow et al. (1951) and Dvoretzky et al. (1952). Optimality of S-s policies under certain conditions was first established by Scarf (1960). Kelle and Milne (1999) study the impact of S-s inventory policies on the supply chain, including connection to the “bullwhip” effect. The connection between S-s inventory policies and macroeconomic fluctuations is studied in Nirei (2006).

Chapter 5

Recursive Dynamic Programs

The finite MDP model from Chapter 4 is a valuable workhorse. However, economists, financial analysis and researchers from other fields are increasingly pushing past the boundaries of this framework. In this chapter, we examine some of problems where the MDP assumptions fail and introduce more general methods to analyze optimality.

One issue with finite MDPs is additive separability, which means that liftime rewards are a discounted sum of one-period rewards. This makes finite MDPs highly tractable but also limits their applicability.

While there are many possible departures from the MDP assuptions, it turns out that we can handle almost all of these departures within a single framework, by constructing an abstract version of the Bellman equation. In this chapter we consider the finite case, where the Bellman equation is abstract but states and actions are still restricted to be finite. This covers many applications of interest, while minimizing technical distractions. (Later, in Chapter **??**, we will cover the same abstract framework in the general case, where states and actions are allowed to be continuous.)

5.1 Recursive Preferences

The finite MDP problem we studied in Chapter 4 has two separate components: One is to compute the lifetime value of a given policy and the other is to find a policy which maximizes this value. In this chapter we focus purely on the first issue, setting maximization aside. More generally, we focus on computing the lifetime value of a given process for the state.

At the same time, we will allow for far more general specifications of lifetime value. In particular, we consider *recursive preferences*, which provide a much richer way of specifying lifetime rewards. Such preferences are increasingly popular but also involve nontrivial technical problems to compute. We will show how popular specifications of recursive utility can be handled via fixed point theory.

(Once we have understood the process of translating recursive preferences into lifetime value, we will move on to maximizing lifetime value via dynamic programming. This will be the topic of Chapter 5.)

5.1.1 Motivation and Examples

5.1.1.1 The Additively Separable Case

Add roadmap.

5.1.1.2 A Sequential View

Optimal choices over consumption and savings will be a recurring topic in the text. To lay the foundations for these discussions, let's begin to think about how consumers rank different kinds of consumption paths over infinite horizons.

A consumption path is a nonnegative random sequence $(C_t)_{t \geqslant 0}$. A common way to model such a sequence is to assume that there exists a fixed function $c \in \mathbb{R}_+^{\mathsf{X}}$ and a P-Markov chain (X_t) on X such that, at each time t, consumption obeys $C_t = c(X_t)$. Thus, consumption streams are stationary functions of a finite state Markov chain.

In the standard additively separable model of consumer preferences, originally due to Samuelson (1939), the time zero value of a consumption stream (C_t), conditional on current state $X_0 = x \in \mathsf{X}$, is given by

$$v(x) = \mathbb{E}_x \sum_{t \geqslant 0} \beta^t u(C_t), \tag{5.1}$$

where

- $\beta \in (0, 1)$ is a discount factor,
- $\mathbb{E}_x := \mathbb{E}[\,\cdot \mid X_0 = x]$, and
- $u \colon \mathbb{R}_+ \to \mathbb{R}$ is called the **flow utility function**.

Dependence of $v(x)$ on x is due to the fact that the initial condition $X_0 = x$ influences the Markov state process and, therefore, the time path of consumption.

Using our expression for $C_t = c(X_t)$, defining $h := u \circ c$ and applying (2.21), we can rewrite this value function as

$$v(x) = \sum_{t \geqslant 0} \beta^t \mathbb{E}_x u(c(X_t)) = \sum_{t \geqslant 0} [(\beta P)^t h](x). \tag{5.2}$$

By Lemma 2.2.1, this sum is finite and can be expressed as

$$v = (I - \beta P)^{-1} h. \tag{5.3}$$

The left and right hand sides are understood as vectors, which can be evaluated at any $x \in \mathsf{X}$. Primitives (u, c, β, P) can be plugged into (5.3) to obtain a lifetime utility valuation of the consumption path (C_t), starting from any given state $x \in \mathsf{X}$.

5.1.1.3 A Recursive View

The additively separable model of valuation in §5.1.1.2 can also be studied recursively. To see this, suppose that a continuation value of current and future consumption is defined at each point in time t by the recursive expression

$$V_t = u(C_t) + \beta\, \mathbb{E}_t V_{t+1} \tag{5.4}$$

The random functions V_t and V_{t+1} are the unknown objects in this expression. The expectation $\mathbb{E}_t$ is conditional on knowledge of time t state X_t.

Since consumption is a function of (X_t) and, by the Markov property, the future and past are independent after conditioning on the present, it is natural to guess that current value V_t will depend on the whole Markov chain only through X_t and that V_{t+1} will depend only on X_{t+1}. For this reason, we guess there is a solution of the difference equation (5.4) takes the form $V_t = v(X_t)$ for some function v.

Remark 5.1.1. Here v is an *ansatz*, meaning "educated guess." First we guess the form of a solution, then we write down properties such an object must satisfy, and, finally, we try to verify that the guess is correct. So long as we carry out the final step, there is no loss of rigor caused by starting with a guess.

Under this conjecture, (5.4) can be rewritten as $v(X_t) = u(c(X_t)) + \beta \mathbb{E}_t v(X_{t+1})$. Conditioning on $X_t = x$, this becomes

$$v(x) = h(x) + (\beta P v)(x) \qquad (x \in \mathsf{X}), \tag{5.5}$$

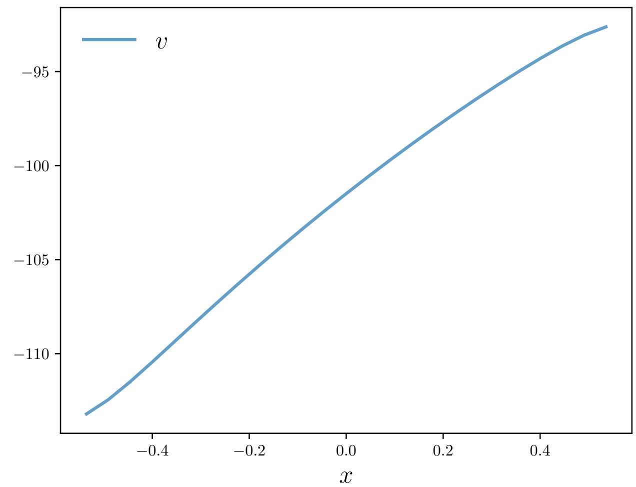


Figure 5.1: The value of $(C_t)_{t \geqslant 0}$ given $X_t = x$

where $h := u \circ c$. In vector form this is $v = h + \beta P v$. From the Neumann Series Lemma, the solution is $v^* = (I - \beta P)^{-1} h$, which is identical to (5.3). Thus, (5.4) and the sequential representation (5.1) specify the same valuation for consumption paths.

Remark 5.1.2. Perhaps the recursive formulation in §5.1.1.3 seems unnecessary, since we are led to the same result that we obtained from using the more direct sequential approach used in §5.1.1.2. However, we will find that, for nonlinear models of preferences over consumption paths, the sequential approach *has no natural counterpart*. In such settings, we are forced to proceed recursively when studying lifetime value. We take up this study in Chapter **??**.

Figure 5.1 shows one example, where

- u has the CRRA specification $u(c) = c^{1-\gamma}/(1-\gamma)$ with $\gamma > 0$,
- $c(x) = \exp(x)$, so that consumption takes the form $C_t = \exp(X_t)$, and
- $(X_t)_{t \geqslant 0}$ is a Tauchen discretization (see §2.1.2) of $X_{t+1} = \rho X_t + \sigma \xi_{t+1}$.

The parameters are $n = 25$, $\beta = 0.98$, $\rho = 0.96$, $\sigma = 0.05$ and $\gamma = 2$. We set $h = u \circ c$ and $v = (I - \beta P)^{-1} h$.

Exercise 5.1.1. Replicate Figure 5.1.

Exercise 5.1.2. The value function in Figure 5.1 is increasing in the state x. Prove that this us always so when $\rho \geqslant 0$.

5.1.1.4 Limitations of Additive Separability

To get a sense of the issues, consider the following scenario. You accept a new job and will be employed by this firm for the rest of your life. Your daily consumption will be determined by your daily wage. Your boss offers you two options:

- (Option A) Your boss will flip a coin at the start of day one. If the coin is heads, you will receive \$10,000 a day for the rest of your life. If the coin is tails, you will receive \$10 per day for the rest of your life.
- (Option B) Your boss will flip a coin at the start of *each* day. If the coin is heads, you will receive \$10,000. If the coin is tails, you will receive \$10.

If you find that you have a strict preference between options A and B, then your utility cannot be modeled using additively separable preferences.

To see why, let φ be a probability distribution that represents the lottery described above, putting mass 0.5 on 10,000 and mass 0.5 on 0. Under option A, consumption is given by $C_t = C_1$ for all t, where $C_1 \sim \varphi$. Under option B, consumption $(C_t)_{t \geqslant 1}$ is an IID sequence drawn from φ. Either way, lifetime utility is

$$\mathbb{E} \sum_{t \geqslant 1} \beta^t u(C_t) = \sum_{t \geqslant 1} \beta^t \mathbb{E} u(C_t) = \frac{\bar{u}}{1-\beta},$$

where $\bar{u} := u(0)/2 + u(10,000)/2$.

The critical part of this argument is the passing of expectations through the sum, which uses additive separability. The implication is that lifetime utility depends only on the marginal distribution of each C_t, rather than on the joint distribution of the stochastic process $(C_t)_{t \geqslant 0}$. As a result, additively separable preferences cannot distinguish between A and B, even though many people have strict preferences between them.

The scenario presented above is complete artificial. Do the deficiencies in additively separability really matter for economic modeling? It is becoming increasingly clear that the answer is affirmative. For example, in analysis of asset pricing, both non-additive preferences and state-dependent discounting play an important role in bringing model outputs closer to the data. §5.4 gives references.

Preferences that fail to be additivity-separable are usually called **recursive preferences**. The terminology refers to the fact that, under these kinds of preferences, lifetime utility is expressed recursively.

The distinction is somewhat confusing, since additively separable preferences also admit recursive specification (recall the recursive expression $V_t = u(C_t) + \beta \mathbb{E}_t V_{t+1}$ from page 141). However, for most forms of recursive preferences, lifetime utility can *only* be expressed recursively. There is no neat expression in terms of an infinite sum.

To give one example, with **Epstein–Zin** preferences, the additively separable relationship $V_t = u(C_t) + \beta \mathbb{E}_t V_{t+1}$ is replaced by

$$V_t = \left\{ u(C_t)^\rho + \beta [\mathbb{E}_t V_{t+1}^\eta]^{\rho/\eta} \right\}^{1/\rho}, \tag{5.6}$$

where η and ρ are nonzero parameters. With preferences such as (5.6), there is no neat "infinite sum" representation like $\mathbb{E} \sum_t \beta^t u(C_t)$ due to the nonlinearities in the expression.

The lack of an infinite sum representation of lifetime utility makes the theory of dynamic programming somewhat trickier when recursive utilities are present. Moreover, just because we write down a recursive expression for lifetime utility doesn't mean that corresponding lifetime utility is actually well defined. (The situation is analogous to nonlinear equations: we can write down any equation we like, but doing so doesn't imply that a solution actually exists.)

In this chapter we provide a framework where many types of recursive preference models can be comfortably handled. Before we can present these results, however, we need to invest in more fixed point theory.

5.1.2 Risk-Sensitive Preferences

To be added.

5.1.3 Convex and Concave Operators

In this section we introduce a different set of sufficient conditions for global stability. These conditions replace contractivity with certain shape properties on the operator that often arise in fields such as economics and finance.

5.1.3.1 The One-Dimensional Case

To build intuition, we start with the one-dimensional case, where the fixed-point problem can be visualized and proofs are relatively simple. We show how concavity and monotonicity can be paired to achieve stability.

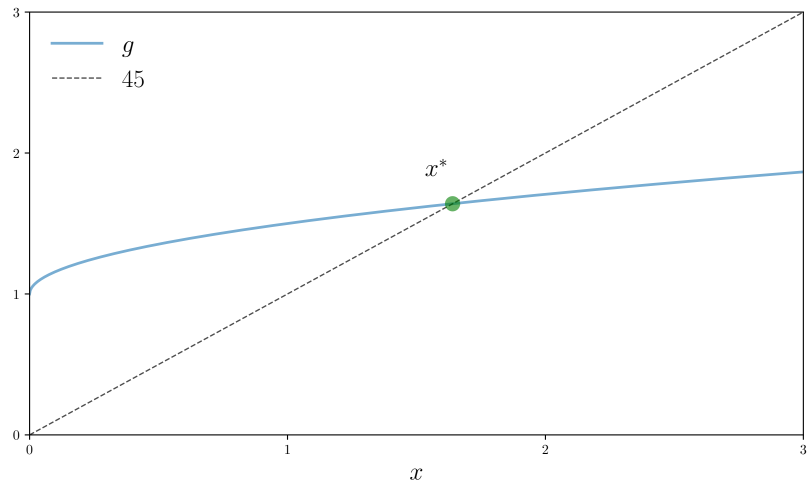


Figure 5.2: Global stability induced by increasing concave functions

We have in fact already seen how this pairing can produce a unique stable fixed point. In §1.2.3.2, we studied a discrete time Solow–Swan model and showed global stability of g on S when $g(k) := sf(k) + (1-\delta)k$ and $S := (0, \infty)$, with $f(k) = Ak^\alpha$, $0 < \alpha, \delta < 1$ and $A > 0$. However, the proof we constructed was quite specialized. Here is a more general result.

Proposition 5.1.1. *Let g be an increasing concave self-map on $S := (0, \infty)$. If, for all $x \in S$, there exists a pair $a, b \in S$ with $a \leqslant x \leqslant b$, $a < g(a)$ and $g(b) \leqslant b$, then g is globally stable on S.*

The proof is below. Figure 5.2 gives one example, where $g(x) = 1 + \sqrt{x}/2$. For a function such as this one, given any positive number x, we can find a number $a < x$ that gets mapped strictly up (i.e., $g(a)$ is above the 45 line) and a point $b > x$ that gets mapped strictly down (i.e., $g(b)$ is below the 45 degree line). Under iteration all trajectories converge to the unique fixed point x^*.

Before reading the proof we recommend you sketch your own examples to see why the different conditions are required.

Exercise 5.1.3. Prove that the map g and set S defined in the discussion of the Solow-Swan model above Proposition 5.1.1 satisfies the conditions of the proposition.

Exercise 5.1.4. Dropping the Cobb-Douglas specification on production, suppose $g(k) = sf(k) + (1-\delta)k$ where $0 < s, \delta < 1$ and f is a strictly positive increasing concave production function on $S = (0, \infty)$ satisfying the **Inada conditions**

$$f'(k) \to \infty \text{ as } k \to 0 \quad \text{and} \quad f'(k) \to 0 \text{ as } k \to \infty,$$

Use Proposition 5.1.1 to prove that g is globally stable on S.

Exercise 5.1.5. Fajgelbaum et al. (2017) study a law of motion for aggregate uncertainty given by

$$s_{t+1} = g(s_t) \quad \text{where} \quad g(s) := \rho^2 \left[\frac{1}{s} + a^2 \frac{1}{\eta} \right]^{-1} + \gamma.$$

Let a, η and γ be positive constants and assume $0 < \rho < 1$. Prove that g is globally stable on $M := (0, \infty)$.

Proof of Proposition 5.1.1. First we prove existence of a fixed point $x^* \in S$. Fix $x \in S$. Suppose first that $x \leqslant g(x)$ Since g is increasing, we then have $g(x) \geqslant g^2(x)$. Continuing in this fashion (or using induction) shows that $(g^n(x))$ is monotone increasing. Moreover, there exists a $b \in S$ such that $x \leqslant b$ and $g(b) \leqslant b$. Hence $g(x) \leqslant g(b) \leqslant b$. Continuing in this fashion (or using induction) yields $g^n(x) \leqslant b$ for all n. We now see that $(g^n(x))$ is increasing and bounded above. Thus, there exists an $x^* \in S$ such that $(x_n) := (g^n(x)$ converges to x^*. Since g is concave and, therefore, continuous on any open set, the result in Exercise 1.2.6 implies that $x^* = g(x^*)$.

We have treated the case $x \leqslant g(x)$ and shown existence of a fixed point. If, instead, $x \geqslant g(x)$, then $(g^n(x))$ is shown to be decreasing and bounded by a symmetric argument. In the same way, we obtain a fixed point x^* with $g^n(x) \to x^*$.

To show the uniqueness of the fixed point, assume $g(x) = x$ and $g(y) = y$ for some $x, y \in S$ with $x \leqslant y$. By assumption, there exists an $a \in S$ such that $a \leqslant x \leqslant y$ and $g(a) > a$. Since $g(x) = x$, the inequality $a < x$ must hold. Because $a < x \leqslant y$, we can take $\lambda \in [0, 1)$ such that $x = \lambda a + (1 - \lambda)y$. Concavity of g implies

$$g(x) = g(\lambda a + (1 - \lambda)y) \geqslant \lambda g(a) + (1 - \lambda)g(y) \geqslant \lambda a + (1 - \lambda)y = x = g(x).$$

In particular, $\lambda g(a) + (1 - \lambda)g(y) = \lambda a + (1 - \lambda)y$. Since $g(y) = y$, we obtain $\lambda g(a) = \lambda a$. But $g(a) > a$, so $\lambda = 0$. Recalling that $x = \lambda a + (1 - \lambda)y$, this yields $x = y$.

We have proved existence of a unique fixed point in S to which every trajectory converges. □

5.1.3.2 The Multidimensional Case

The preceding fixed point result extends naturally to multiple dimensions. Here we present a multidimensional version that covers both convex and concave functions.

First we extend the definition of convexity and concavity to vector-valued self-maps. In fact the conditions look identical to those for scalar-valued functions: a self-map T on a convex subset D of $\mathbb{R}^n$ is called **convex** if

$$T(\lambda u + (1-\lambda)y) \leqslant \lambda Tu + (1-\lambda)Ty \text{ whenever } u, y \in D \text{ and } \lambda \in [0,1];$$

and **concave** if

$$\lambda Tu + (1-\lambda)Ty \leqslant T(\lambda u + (1-\lambda)y) \text{ whenever } u, y \in D \text{ and } \lambda \in [0,1].$$

Here $\leqslant$ is, as usual, the pointwise partial order.

Using the same order, for fixed $\varphi, \psi \in \mathbb{R}^n$, we define the **order-interval** $[\varphi, \psi]$ to be the set of all x in $\mathbb{R}^n$ with $\varphi \leqslant u \leqslant \psi$.

We are now ready to state a fixed point theory for monotone concave vector-valued self-maps. The following result was proved by Du (1990). A proof of can also be found in Theorem 2.1.2 and Corollary 2.1.1 of Zhang (2012).

Theorem 5.1.2 (Du). *Let $I := [\varphi, \psi]$ be a nonempty order interval in $\mathbb{R}^n$ and let T be a self-map on I. If T is order-preserving, then then T is globally stable on I under any one of the condition sets (i)–(iii) below.*

(i) *T is concave and $T\varphi \gg \varphi$, or*

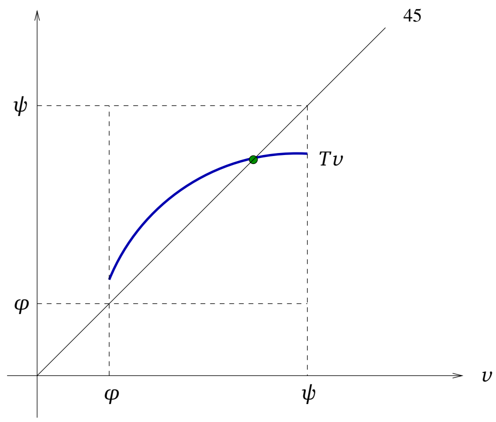
(ii) *T is concave and there exists an $\varepsilon > 0$ such that $T\varphi \geqslant \varphi + \varepsilon(\psi - \varphi)$, or*

(iii) *T is convex and $T\psi \ll \psi$.*

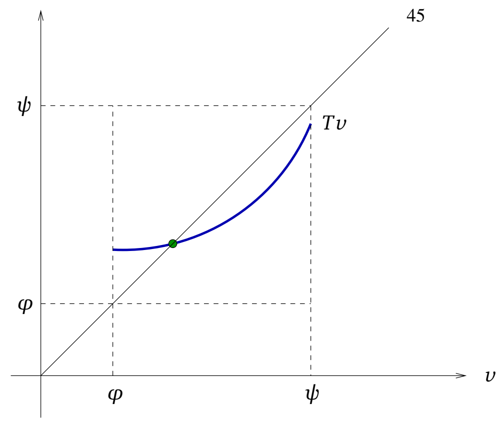
Conditions (i) and (ii) are very similar: both require that T is concave and maps φ strictly up. Condition (iii) replaces concavity with convexity. Figure 5.3 illustrates the convex and the concave versions of the result in the simple case $n = 1$. We encourage you to sketch your own variations to get a feeling for why the different conditions are needed.

5.1.3.3 Application: Negative Discount Rate Optimality

Recalling that β is the discount factor in MDPs, the **discount rate** or **rate of time preference** is the value ρ defined by $\beta = 1/(1+\rho)$. The standard MDP assumption



(a)



(b)

Figure 5.3: Du's theorem: convex and concave cases

$\beta < 1$ implies this rate is positive. The condition $\beta < 1$ is essential to the standard theory of MDPs, since it yields contractivity of Bellman and policy operators.

At the same time, behavior consistent with positive discount rates is not universal. For example, negative rates of time preference are commonly observed when agents face an unpleasant task. Subjects of studies often prefer getting such tasks "over and done with" rather than postponing them. (Negative discount rates are observed in more standard settings as well. §5.4 provides background and references.)

To model senarios where the task is unpleasant and the discount rate is negative, we consider the Bellman equation

$$f(x) = \min_{0 \leqslant x' \leqslant x} \{\ell(x - x') + \beta f(x')\} \tag{5.7}$$

where

- x represents the amount of the task currently remaining,
- x' is the remainder next period, so that $x - x'$ is the amount of the task completed in the current period,
- ℓ is an increasing and strictly convex loss function satisfying $0 = \ell(0) < \ell'(0)$,
- $f(w)$ is represents minimum "cost-to-go" when the agent acts optimally from state w, and
- the discount factor obeys $\beta > 1$.

Because $\beta > 1$, future losses are amplified. Hence the agent wants to complete the task quickly. At the same time, ℓ is strictly convex, so completing too much in any one period is suboptimal. The right hand side of (5.7) captures this trade off between current loss and future loss.

For now, we assume that x and x' take values in a finite set X with $\min \mathsf{X} = 0$ and $\bar{x} := \max \mathsf{X} > 0$. (Later we relax this assumption.)

Denote by T the Bellman operator corresponding to (5.7). For any $f \in \mathbb{R}^{\mathsf{X}}$, the operator T satisfies

$$(Tf)(x) = \min_{x' \in \Gamma(x)} \{\ell(x - x') + \beta f(x')\} \qquad (x \in \mathsf{X}), \tag{5.8}$$

where $\Gamma(x) := \{x \in \mathsf{X} : 0 \leqslant x' \leqslant x\}$. While (5.7) appears at first glance to be a standard Bellman equation, the assumption $\beta > 1$ implies that T is not a contraction with respect to any obvious metric.

To handle T we set $I = [\varphi, \psi] \subset \mathbb{R}^{\mathsf{X}}$ where φ and ψ are functions in $\mathbb{R}^{\mathsf{X}}$ defined by $\varphi(x) = \ell'(0)x$ on X and $\psi = \ell$. We make the following observation, which is proved at the end of this section.

Lemma 5.1.3. *There exists an $\varepsilon > 0$ such that $T\varphi \geqslant \varphi + \varepsilon(\psi - \varphi)$.*

EXERCISE 5.1.6. Prove that T is an order-preserving self-map on I.

EXERCISE 5.1.7. Let p and q be functions from nonempty finite set D into $\mathbb{R}$. Prove that $\min_{x \in D}(p(x) + q(x)) \geqslant \min_{x \in D} p(x) + \min_{x \in D} q(x)$.

EXERCISE 5.1.8. Prove that T is a concave operator on I.

Combining the lemmas and exercises above, we have shown that, under the stated assumptions, T is a concave order-preserving self-map on $I = [\varphi, \psi]$ and, in addition, there exists an $\varepsilon > 0$ such that $T\varphi \geqslant \varphi + \varepsilon(\psi - \varphi)$. From Theorem 5.1.2 we conclude that T is globally stable on I.

EXERCISE 5.1.9. Prove that the unique fixed point of T in I is increasing.

EXERCISE 5.1.10. Since $I = [\varphi, \psi]$ and $\psi = \ell$, the above argument implies that the fixed point f^* obeys $f^* \leqslant \ell$ on X. Provide an intuitive explanation of why this should be true.

Proof of Lemma 5.1.3. Since ℓ is strictly convex, the function $\psi - \varphi$ is increasing, so with

$$\varepsilon := \frac{\psi(h) - \varphi(h)}{\psi(\bar{x}) - \varphi(\bar{x})},$$

where h is the minimum of $|x - x'|$ for all distinct $x, x' \in \mathsf{X}$, we get

$$\psi(x - x') - \varphi(x - x') \geqslant \varepsilon(\psi(x) - \varphi(x)) \quad \text{for all } x, x' \in \mathsf{X} \text{ with } x' < x.$$

Hence, fixing $x \in \mathsf{X}$ and letting x' be the minimizer of $\ell(x - x') + \beta\ell'(0)x'$, we have

$$\begin{aligned}(T\varphi)(x) - \varphi(x) &= \ell(x - x') + \beta\ell'(0)x' - \ell'(0)x \\ &\geqslant \ell(x - x') + \ell'(0)(x' - x) \\ &= \psi(x - x') + \varphi(x - x') \geqslant \varepsilon(\psi(x) - \varphi(x)).\end{aligned}$$

Since $x \in \mathsf{X}$ was arbitrary, we have proved the claim in the lemma. □

5.1.4 Conjugate Operators

Suppose we are concerned with the dynamics induced by an operator T mapping $\mathbb{R}^n$ into itself. For example, we might want to know if a unique fixed point of T exists, or if iterates of T always converge to a fixed point. Discussion above suggests that, in order to address these questions, we should apply fixed point theory to T.

Sometimes, however, there is an easier approach: transform T into a "simpler" operator U and then study the fixed point properties of U. Of course, for this idea to work, we need to be sure that any properties we discover about U can be translated back to properties of T, the operator that we are actually interested in.

This section explains the notion of topological conjugacy, which originates in the field of dynamical systems theory, and can be used effectively for the problem just described. Later in this chapter, we will apply the methodology to operators that arise from dynamic programming problems.

To explain the idea, let M and $\hat{M}$ be two subsets of $\mathbb{R}^n$. A function H from M to $\hat{M}$ is called a **homeomorphism** if it is continuous, a bijection, and its inverse H^{-1} is also continuous.

Example 5.1.1. The map $Hx = \ln x$ from $(0, \infty)$ to $\mathbb{R}$ is a homeomorphism, with continuous inverse $H^{-1}y = \exp(y)$.

Example 5.1.2. Let H be an $n \times n$ matrix. We can regard H as a map sending column vector x into column vector Hx. This map is a homeomorphism from $\mathbb{R}^n$ to itself if and only if H is nonsingular.

A **dynamical system** is a pair (M, T), where M is a subset of $\mathbb{R}^n$ and T is a self-map on M. Two dynamical systems (M, T) and $(\hat{M}, \hat{T})$ are said to be **topologically conjugate** if there exists a homeomorphism H from M into $\hat{M}$ such that $\hat{T} = H \circ T \circ H^{-1}$ on $\hat{M}$. In other words, shifting a point $\hat{x} \in \hat{M}$ to $\hat{T}\hat{x}$ using the map $\hat{T}$ is equivalent to moving $\hat{x}$ into M with H^{-1}, applying T, and then moving the result back using H:

$$
\begin{tikzcd}
x \arrow[r, "T"] & T(x) \arrow[d, "H"] \\
\hat{x} \arrow[u, "H^{-1}"] \arrow[r, "\hat{T}"] & \hat{T}\hat{x}
\end{tikzcd}
$$

Example 5.1.3. Let H be an $n \times n$ **diagonalizable matrix**, meaning that there exists a diagonal matrix D and a nonsingular matrix P such that $A = PDP^{-1}$. (The matrices D and P can be complex-valued.) Analogous to Example 5.1.3, we can regard A as a map on $\mathbb{R}^n$ and D and a map on $\mathbb{C}^n$, the set of complex n-vectors The identity $A = PDP^{-1}$ implies that the dynamical systems $(A, \mathbb{R}^n)$ and $(D, \mathbb{C}^n)$ are topologically conjugate.

Exercise 5.1.11. Let $M := ((0, \infty), |\cdot|)$ and $\hat{M} := (\mathbb{R}, |\cdot|)$. Let $Tx = Ax^\alpha$, where $A > 0$ and $\alpha \in \mathbb{R}$, and let $\hat{T}\hat{x} = \ln A + \alpha\hat{x}$. Show that T and $\hat{T}$ are topologically conjugate under $H := \ln$.

Exercise 5.1.12. Show that if (M, T) and $(\hat{M}, \hat{T})$ are topologically conjugate, then $x \in M$ is a fixed point of T on M if and only if $H(x) \in \hat{M}$ is a fixed point of $\hat{T}$ on $\hat{M}$.

Exercise 5.1.13. Let $x^* \in M$ be a fixed point of T and let x be any point in M. Show, in addition, that $\lim_{k\to\infty} T^k(x) = x^*$ if and only if $\lim_{k\to\infty} \hat{T}^k Hx = Hx^*$.

5.1.5 Esptein–Zin Utility

5.2 Abstract DP Theory

First we introduce an general dynamic decision problem and and analyze optimality in this abstract setting. Then we will show how interested applications can be handled as special cases.

5.2.1 Abstract Decision Processes

In this section we study an abstract dynamic program with Bellman equation

$$v(x) = \max_{a \in \Gamma(x)} B(x, a, v). \tag{5.9}$$

Here x is the state, a is an action, Γ is a feasible correspondence and B is a abstract representation of the right-hand side of a Bellman equation. (Compare with, say, (4.2) on page 108.) The function v assigns values to states and is a member of some class $\mathscr{V}$ contained in the set of real-valued functions on X. A very wide range of dynamic programs can be expressed in this way.

We begin by clarifying the model and then discuss optimality.

5.2.1.1 Finite RDPs

To formalize the abstract decision process described above, let X and A be nonempty finite sets, referred to as the **state space** and **action space** respectively. In what follows, a **feasible correspondence** is any correspondence from X to A such that $\Gamma(x)$ is nonempty for all $x \in \mathsf{X}$. We understand $\Gamma(x)$ as all actions available to the controller in state x.

As was the case with finite MDPs, given such a correspondence, we let G be the **graph** of Γ, also called the set of **feasible state-action pairs**, and defined by $\mathsf{G} = \{(x, a) \in \mathsf{X} \times \mathsf{A} : a \in \Gamma(x)\}$. In addition, we let Σ denote the set of all **feasible policies**, defined as all $\sigma \colon \mathsf{X} \to \mathsf{A}$ such that $\sigma(x) \in \Gamma(x)$ for all $x \in \mathsf{X}$. As with finite MDPs, each $\sigma \in \Sigma$ specifies an action to be taken by the controller at any given state.

Of course G and Σ depend on Γ and our notation does not emphasize this fact. However, in what follows, Γ will typically be fixed and hence the meanings of G and Σ will be clear.

Given X and A, we define a **finite recursive decision process** (finite RDP) to be a triple $(\Gamma, \mathscr{V}, B)$ containing

(i) a **feasible correspondence** Γ with associated graph G and feasible policy set Σ,

(ii) a closed subset $\mathscr{V}$ of $\mathbb{R}^{\mathsf{X}}$ called the set of **candidate value functions**,

(iii) a **value aggregator**

$$B \colon \mathsf{G} \times \mathscr{V} \to \mathbb{R}$$

satisfying the **monotonicity condition**

$$v, w \in \mathscr{V} \text{ and } v \leqslant w \implies B(x, a, v) \leqslant B(x, a, w) \text{ for all } (x, a) \in \mathsf{G}, \tag{5.10}$$

and the **consistency condition**

$$v \in \mathscr{V} \implies w \in \mathscr{V} \quad \text{where } w(x) := B(x, \sigma(x), v) \tag{5.11}$$

We understand $\mathscr{V}$ as a class of functions that assign values to states. The interpretation of the aggregator B is:

> $B(x, a, v)$ = total lifetime rewards, contingent on current action a, current state x and the use of v to evaluate future states.

In other words, $B(x, a, v)$ corresponds to the right hand side of the Bellman equation—the function that we maximize over when choosing an optimal action. Not surprisingly, optimality is contingent on inserting the correct function v into $B(x, a, v)$, so locating and calculating this v will be one of our major concerns.

Some clarifications:

- The order on the left side of (5.10) is the usual pointwise partial order.
- When we say that $\mathscr{V}$ is "closed," we mean that, when each element of $\mathscr{V}$ is regarded as a vector in Euclidean space $\mathbb{R}^{|\mathsf{X}|}$, the resulting set of vectors is closed in $\mathbb{R}^{|\mathsf{X}|}$. Hopefully this identification of functions in $\mathbb{R}^{\mathsf{X}}$ and vectors in $\mathbb{R}^{|\mathsf{X}|}$ is comfortable for you now. (Otherwise see §1.2.5.2.)

The monotonicity condition (5.10) is natural: relatively to v, if rewards are at least as high with w in every future state, then the total rewards we can extract under w should be at least as high. The consistency condition in (5.11) is required to ensure that, when considering the value of different policies, we do not leave the class $\mathscr{V}$ of candidate value functions.

Example 5.2.1 (Every finite MDP is a finite RDP)**.** A finite MDP (see, e.g., §4.1.1.1) is a special case of a finite RDP. To see this, consider an arbitrary finite MDP (Γ, β, r, P) with state space X and action space A. To frame this as a finite RDP, we take Γ as the feasible correspondence for the RDP and $\mathscr{V} = \mathbb{R}^{\mathsf{X}}$ as the class of candidate value functions. The aggregator B is

$$B(x, a, v) = r(x, a) + \beta \sum_{x' \in \mathsf{X}} v(x') P(x, a, x') \qquad ((x, a) \in \mathsf{G},\ v \in \mathscr{V}). \tag{5.12}$$

(This corresponds to the unmaximized right-hand side of the Bellman equation in (4.2) on page 108.) Now $(\Gamma, \mathscr{V}, B)$ forms an RDP. The monotonicity condition (5.10) clearly holds and the consistency condition (5.11) is trivial in this case, since $\mathscr{V}$ is all of $\mathbb{R}^{\mathsf{X}}$.

Example 5.2.2 (State-Dependent Discounting). We can add state-dependent discounting to the last example by changing the aggregator to

$$B(x, a, v) = r(x, a) + \beta(x) \sum_{x' \in \mathsf{X}} v(x') P(x, a, x'), \tag{5.13}$$

where β is some nonnegative function of the state. As before we take $\mathscr{V} = \mathbb{R}^{\mathsf{X}}$. The monotonicity condition continues to hold, since

$$w \leqslant v \quad \implies \quad \beta(x) \sum_{x' \in \mathsf{X}} w(x') P(x, a, x') \leqslant \beta(x) \sum_{x' \in \mathsf{X}} v(x') P(x, a, x') \text{ for all } (x, a) \in \mathsf{G}.$$

Example 5.2.3 (Epstein–Zin Preferences). We can modify the MDP in Example 5.2.1 to use the Epstein–Zin aggregator in (5.6) by setting

$$B(x, a, v) = \left\{ r(x, a)^\rho + \beta \left[\sum_{x' \in \mathsf{X}} v(x')^\eta P(x, a, x') \right]^{\rho/\eta} \right\}^{1/\rho}, \tag{5.14}$$

where η and ρ are nonzero parameters. To avoid undefined exponentiation, we assume here that $m := \min_{(x,a) \in \mathsf{G}} r(x, a)$ is strictly positive and take $\mathscr{V} = \{v \mathbb{R}^{\mathsf{X}} : v \geqslant m\mathbb{1}\}$, where $\mathbb{1}$ is a vector of ones.

Exercise 5.2.1. Confirm that the Epstein–Zin model described in Example 5.2.3 satisfies the monotonicity and consistency conditions in the definition of an RDP. You can assume that η and ρ are nonzero and $\beta \in (0, 1)$.

5.2.2 Optimality Theory

In this section we present optimality theory for finite RDPs. First we define optimality and then we seek to characterize it under conditions on the primitives.

5.2.2.1 Operators

Given a finite RDP $(\Gamma, \mathscr{V}, B)$ with state and action spaces X and A, we introduce, for each $\sigma \in \Sigma$, the **policy operator** T_σ as a map from $\mathscr{V}$ to itself defined by

$$(T_\sigma v)(x) = B(x, \sigma(x), v) \qquad (x \in \mathsf{X}).$$

Lemma 5.2.1. *T_σ is an order-preserving self-map on $\mathscr{V}$ for all $\sigma \in \Sigma$.*

Proof. The claim that T_σ is a self-map on $\mathscr{V}$ follows immediately from the consistency condition in (5.11). The order-preserving property follows from the monotonicity condition in (5.10). □

Given v in $\mathscr{V}$, we say that a policy $\sigma \in \Sigma$ is **v-greedy** for the finite RDP $(\Gamma, \mathscr{V}, B)$ if it satisfies

$$B(x, \sigma(x), v) = \max_{a \in \Gamma(x)} B(x, a, v) \qquad \text{for all } x \in \mathsf{X}. \tag{5.15}$$

In essence, a v-greedy policy treats v as the correct value function and sets all actions accordingly. Since $\Gamma(x)$ is finite and nonempty at each $x \in \mathsf{X}$, at least one v-greedy policy exists.

Finally, we define the **Bellman operator** via

$$(Tv)(x) = \max_{a \in \Gamma(x)} B(x, \sigma(x), v) \qquad (x \in \mathsf{X},\ v \in \mathscr{V}).$$

Example 5.2.4. The Bellman operator associated with the Epstein–Zin RDP in (5.14) is given by

$$(Tv)(x) = \max_{a \in \Gamma(x)} \left\{ r(x, a)^\rho + \beta \left[\sum_{x' \in \mathsf{X}} v(x')^\eta P(x, a, x') \right]^{\rho/\eta} \right\}^{1/\rho} \qquad (x \in \mathsf{X}).$$

In what follows, it will be helpful to note that, for any given RDP $(\Gamma, \mathscr{V}, B)$ and any $v \in \mathscr{V}$, we have the following property, which follows easily from the definitions.

$$Tv = T_\sigma v \iff \sigma \text{ is } v\text{-greedy}. \tag{5.16}$$

Lemma 5.2.2. *The Bellman operator T is an order-preserving self-map on $\mathscr{V}$.*

Exercise 5.2.2. Verify Lemma 5.2.2.

Exercise 5.2.3. Show that, for a given finite RDP $(\Gamma, \mathscr{V}, B)$ and fixed $v \in \mathscr{V}$, the Bellman operator T obeys

$$(T^k v)(x) = \max_{a \in \Gamma(x)} B(x, a, T^{k-1} v) \tag{5.17}$$

for all $k \in \mathbb{Z}_+$ and all $x \in \mathsf{X}$. (For an easier exercise, check that it works when $k = 2$.) Show, in addition, that for any policy $\sigma \in \Sigma$, the policy operator T_σ obeys

$$(T_\sigma^k v)(x) = \max_{a \in \Gamma(x)} B(x, a, T_\sigma^{k-1} v) \tag{5.18}$$

for all $k \in \mathbb{Z}_+$ and all $x \in \mathsf{X}$.

EXERCISE 5.2.4. Prove: The Bellman operator for the state-dependent discounting model in Example 5.2.2 is globally stable on $\mathscr{V} = \mathbb{R}^{\mathsf{X}}$ whenever $0 \leqslant \max_{x \in \mathsf{X}} \beta(x) < 1$.

Remark 5.2.1. The condition for global stability in Exercise 5.2.4 is too strict for applications where the discount rate is can be negative (as motivation, see the negative interest rates in Figure 2.6). In §5.2.4.3 below, we introduce a model that allows for state-dependent discount rate processes taking negative values.

5.2.2.2 Defining Optimality

When we studied finite MDPs, we made extensive use of the fact that T_σ and T are both contraction maps, implying global stability on $\mathbb{R}^{\mathsf{X}}$. In the present setting, assuming contractivity is too restrictive, since many useful models fail to be norm contractions. Instead, we will assume global stability directly, and then show how it can be obtained in various special cases, either via contractivity or through other methods.

To this end, we call a finite RDP $(\Gamma, \mathscr{V}, B)$ with associated Bellman operator T and policy operators $\{T_\sigma\}_{\sigma \in \Sigma}$ **globally stable** if

(i) T is globally stable on $\mathscr{V}$ and

(ii) T_σ is globally stable on $\mathscr{V}$ for all $\sigma \in \Sigma$.

For a globally stable finite RDP, given $\sigma \in \Sigma$, we define the **σ-value function** to be the unique $v_\sigma \in \mathscr{V}$ such that

$$v_\sigma(x) = B(x, \sigma(x), v_\sigma) \quad \text{for all } x \in \mathsf{X}. \tag{5.20}$$

In other words, v_σ is the unique fixed point of the policy operator T_σ. Existence and uniqueness both follow from the assumption of global stability of T_σ.

The function v_σ represents the lifetime value of following the policy σ in each period under the stated RDP. This interpretation is a direct generalization of the one we gave for finite MDPs. Indeed, in §4.1.3.3 we saw that, for a finite MDP, the lifetime value v_σ of following policy σ is the unique fixed point of the corresponding policy operator T_σ. In particular, it is the unique solution to the equation

$$v_\sigma(x) = r(x, \sigma(x)) + \beta \sum_{x' \in \mathsf{X}} v_\sigma(x') P(x, \sigma(x), x') \qquad (x \in \mathsf{X}). \tag{5.21}$$

Equation (5.20) generalizes this idea.

A policy $\sigma^* \in \Sigma$ is called **optimal** for the finite RDP $(\Gamma, \mathscr{V}, B)$ if

$$v_{\sigma^*}(x) \geqslant v_\sigma(x) \quad \text{for all } \sigma \in \Sigma \text{ and all } x \in \mathsf{X}.$$

Thus, an optimal policy is a policy that generates maximal lifetime value from every possible state.

Closely related to optimal policies are value functions. The **value function** associated with our planning problem is the v^* in $\mathbb{R}^{\mathsf{X}}$ defined by

$$v^*(x) = \sup_{\sigma \in \Sigma} v_\sigma(x) \qquad (x \in \mathsf{X}). \tag{5.22}$$

Evidently, a policy σ is optimal if and only if $v_\sigma(x) = v^*(x)$ for all $x \in \mathsf{X}$.

(While it might not be clear that the supremum in (5.22) is finite, we will confirm finiteness is immediately below.)

5.2.2.3 Optimality Results

The next theorem is our main optimality result for dynamic decision problems with finite states and actions. The proof of part (i) is similar to, but more general than, the proof of Proposition **??**, which pertains to the job search model.

Theorem 5.2.3. *For every globally stable finite RDP, the following statements are true:*

(i) *The value function v^* satisfies the Bellman equation.*

(ii) *The value function is the only fixed point of T in $\mathscr{V}$ and*

$$\lim_{k \to \infty} T^k v = v^* \quad \textit{for all } v \in \mathscr{V}.$$

(iii) *A policy $\sigma \in \Sigma$ is optimal if and only if it is v^*-greedy.*

(iv) *At least one optimal policy exists.*

Proof. Since T is globally stable on $\mathscr{V}$, it has a unique fixed point $\bar{v} \in \mathscr{V}$. Our first claim is that $\bar{v}$ is equal to v^*, the value function. We show $\bar{v} \leqslant v^*$ and then $\bar{v} \geqslant v^*$.

For the first inequality, let $\sigma \in \Sigma$ be $\bar{v}$-greedy. Recalling (5.16), we observe that, for this choice of σ, we have $T_\sigma \bar{v} = T\bar{v} = \bar{v}$. Hence $\bar{v}$ is also a fixed point of T_σ. But the only fixed point of T_σ in $\mathscr{V}$ is v_σ, so it must be the case that $\bar{v} = v_\sigma$. But then $\bar{v} \leqslant v^*$, since, by definition, $v^* = \sup_{\sigma \in \Sigma} v_\sigma$. This is our first inequality.

Regarding the second inequality, if we fix $v \in \mathscr{V}$ and $\sigma \in \Sigma$, then, since $Tv \geqslant T_\sigma v$ holds and since both T and T_σ are order preserving, we obtain $T_\sigma^k v \leqslant T^k v$ for all k. Taking the limit of this expression yields $v_\sigma \leqslant \bar{v}$. Taking the supremum over $\sigma \in \Sigma$ implies $v^* \leqslant \bar{v}$.

Hence v^* is a fixed point of T in $\mathscr{V}$. Since T is globally stable on $\mathscr{V}$, the remaining claims in parts (i)–(ii) follow immediately.

Regarding part (iii), it follows from (5.16) and part (i) of this theorem that

$$\sigma \text{ is } v^*\text{-greedy} \quad \iff \quad T_\sigma v^* = Tv^* = v^*.$$

The right hand side of this expression tells us that v^* is a fixed point of T_σ. Since the RDP is globally stable, the only fixed point of T_σ is v_σ, so the right hand side is equivalent to the statement $v_\sigma = v^*$. Hence, by this chain of logic and the definition of optimality,

$$\sigma \text{ is } v^*\text{-greedy} \iff v^* = v_\sigma \iff \sigma \text{ is optimal} \tag{5.23}$$

In other words, σ is v^*-greedy if and only if it is optimal.

Since $\Gamma(x)$ is always finite and nonempty, at least one v^*-greedy policy exists. Since Bellman's principle of optimality holds, each such policy is optimal, so the set of optimal policies is nonempty. □

5.2.3 Contracting RDPs

The previous section showed that globally stable finite RDPs have excellent optimality properties. But what kinds of RDPs are globally stable? In this section we provide one rather strict sufficient condition for global stability of finite RDPs. Later we will deal with more complex cases.

Let $(\Gamma, \mathscr{V}, B)$ be a finite RDP with state space X and action space A. We call $(\Gamma, \mathscr{V}, B)$ **contracting** if there exists a $\beta \in [0, 1)$ such that

$$|B(x, a, v) - B(x, a, w)| \leqslant \beta \|v - w\|_\infty \quad \text{for all } (x, a) \in \mathsf{G} \text{ and } v, w \in \mathscr{V}. \tag{5.24}$$

In line with the terminology for contration maps, we call β the **modulus of contraction** for the RDP when (5.24) holds.

The following proposition tells is that, under this contractivity condition, the RDP is globally stable and all of the results in Theorem 5.2.3 apply.

Proposition 5.2.4. *If a finite RDP is contracting then the associated Bellman and policy operators T and $\{T_\sigma\}_{\sigma \in \Sigma}$ are all contractions of modulus β on $\mathscr{V}$ under the norm $\|\cdot\|_\infty$.*

Proof. Fix $\sigma \in \Sigma$. let v and w be elements of $\mathscr{V}$. By (5.24) we have

$$|(T_\sigma)\, v(x) - (T_\sigma)\, w(x)| = |B(x, \sigma(x), v) - B(x, \sigma(x), w)| \leqslant \beta \|v - w\|_\infty$$

for every $x \in \mathsf{X}$. Taking the supremum over the left hand side proves that T_σ is a contraction of modulus β with respect to the supremum norm. Since $\mathscr{V}$ is a closed subset of $\mathbb{R}^{\mathsf{X}}$, it follows from Banach's contraction mapping theorem that T_σ is globally stable on $\mathscr{V}$.

Similarly, fixing $x \in \mathsf{X}$ and applying (5.24) and the sup inequality in Lemma 1.3.2, we have

$$|(Tv)(x) - (Tw)(x)| \leqslant \max_{a \in \Gamma(x)} |B(x, a, v) - B(x, a, w)| \leqslant \beta \|v - w\|_\infty.$$

Taking the supremum over the left hand side shows that T is also contracting on $\mathscr{V}$, so the Banach contraction mapping theorem applies. Finally, existence of greedy policies is trivial when A is finite and Γ is nonempty. □

EXERCISE 5.2.5. Show that any finite MDP is a contracting RDP. Using this fact, complete the proof of Theorem 5.2.3 on page 157.

Next we introduce a sufficient condition for contractivity that can be very useful in practice. To state the condition, we take $(\Gamma, \mathscr{V}, B)$ to be a finite RDP such that $v \in \mathscr{V}$ implies $v + \lambda \mathbb{1} \in \mathscr{V}$ for every $\lambda \geqslant 0$. (Here $\mathbb{1}$ is the function everywhere equal to unity.) We say that the RDP satisfies **Blackwell's condition** if there exists a $\beta \in [0, 1)$ such that, for any $\lambda \geqslant 0$,

$$B(x, a, v + \lambda \mathbb{1}) \leqslant B(x, a, v) + \beta\lambda \qquad \text{for all } (x, a) \in \mathsf{G}.$$

EXERCISE 5.2.6. Prove the following: Every finite RDP that satisfies Blackwell's condition is contracting with modulus of contraction β.

EXERCISE 5.2.7. Prove that the discrete optimal savings model from §4.2.2 satisfies Blackwell's condition.

5.2.4 Eventually Contracting RDPs

Some RDPs fail to be contracting. One example is the finite MDPs with state-dependent discounting that we discussed in §4.3.1. Other examples involve a combination of recursive preferences and state-dependent discounting. In this section, to handle such models, we introduce a class of RDPs that contract eventually, in a sense to be defined. We show that these "eventually contracting RDPs" are globally stable, so that the optimality results of Theorem 5.2.3 apply.

5.2.4.1 Eventual Contracting Operators

To handle eventually contracting models, we will use a simple but valuable extension to Banach's fixed point theorem, which was introduced on page 28. The result extends Banach's theorem to multi-step contractions.

To state the result we take M to be a subset of $\mathbb{R}^n$ and define a self-map T on M to be **eventually contracting** if there exists a $k \in \mathbb{N}$ and a norm $\| \cdot \|$ such that T^k is a contraction on M under the norm $\| \cdot \|$. Significantly, most of the conclusions of Banach's theorem carry over to the case where T is eventually contracting:

Theorem 5.2.5. *If $M \subset \mathbb{R}^n$ is closed and $T\colon M \to M$ is eventually contracting, then T is globally stable on M.*

EXERCISE 5.2.8. Prove Theorem 5.2.5. [Hint: Theorem 5.2.5 is self-improving in the sense that it implies this seemingly stronger result.]

It is helpful to recognize the connection between Theorem 5.2.5 and the Neumann series lemma. If $M = \mathbb{R}^n$ and $Tx = Ax + b$ with $r(A) < 1$, then

$$\|T^k x - T^k y\|_\infty = \|A^k x - A^k y\|_\infty = \|A^k (x - y)\|_\infty \leqslant \|A^k\|_\infty \|x - y\|_\infty.$$

Since $r(A) < 1$, we can choose k such that $\|A^k\|_\infty < 1$ (see Exercise 1.2.2). Hence T is eventually contracting and Theorem 5.2.5 yields global stability. We do not need to call on the Neumann series lemma.

On one hand, eventual contractions have much wider scope than the Neumann series lemma, since they can also be applied in nonlinear settings. On the other, the Neumann series lemma is preferred when applicable, since it also gives inverse and power series representations of the fixed point.

5.2.4.2 A Condition for Eventual Contraction

Let's now look at providing an eventual contraction condition for finite RDPs. Let $(\Gamma, \mathscr{V}, B)$ be a finite RDP and assume in addition that the state space takes the form

$\mathsf{X} = \mathsf{Z} \times \mathsf{Y}$. We call $(\Gamma, \mathscr{V}, B)$ **eventually contracting** if there exists a nonnegative matrix L on $\mathsf{Z} \times \mathsf{Z}$ such that $r(L) < 1$ and

$$|B(y, z, a, v) - B(y, z, a, w)| \leqslant \sum_{z' \in \mathsf{Z}} \max_{y' \in \mathsf{Y}} |v(y', z') - w(y', z')| L(z, z') \tag{5.25}$$

for all $(y, z, a) \in \mathsf{G}$.

The next exercise shows that contracting RDPs are a special case of eventually contracting RDPs.

Exercise 5.2.9. Prove the following: If $(\Gamma, \mathscr{V}, B)$ is an eventually contracting RDP and, in addition, $L(z, z') = \beta Q(z, z')$ for some $\beta \in (0, 1)$ and stochastic matrix Q on $\mathsf{Z} \times \mathsf{Z}$, then $(\Gamma, \mathscr{V}, B)$ is a contracting RDP.

check!

The main result of this section states that eventually contracting RDPs are globally stable, and hence all of the optimality results in Theorem 5.2.3 apply.

Proposition 5.2.6. *Every eventually contracting RDP is also globally stable.*

In the proof of Proposition 5.2.6, we will use the following lemma.

Lemma 5.2.7. *If $\beta \in \mathbb{R}^{\mathsf{Z}}_+$ and Q is a stochastic matrix on Z, then the operator H on $\mathbb{R}^{\mathsf{X}}$ defined by*

$$(Hg)(y, z) = \sum_{z' \in \mathsf{Z}} \max_{y' \in \mathsf{Y}} g(y', z') L(z, z'),$$

satisfies $H^k g \leqslant \|g\|_\infty L^k \mathbb{1}$ pointwise on X.

Proof. We prove this only for $k = 2$. (The proof for general k is similar.) Fixing $g \in \mathbb{R}^{\mathsf{X}}$, we have

$$\begin{aligned}(H^2 g)(y, z) &= \sum_{z' \in \mathsf{Z}} \max_{y' \in \mathsf{Y}} \left[\sum_{z'' \in \mathsf{Z}} \max_{y'' \in \mathsf{Y}} g(y'', z'') L(z', z'') \right] L(z, z') \\ &\leqslant \|g\|_\infty \sum_{z' \in \mathsf{Z}} \sum_{z'' \in \mathsf{Z}} L(z', z'') L(z, z').\end{aligned}$$

From the definition of matrix multiplication, we now have

$$(H^2 g)(y, z) = \sum_{z''} L^2(z, z'') = (L^2 \mathbb{1})(z).$$

The proof for $k = 2$ is done. $\square$

Proof of Proposition 5.2.6. Let $(\Gamma, \mathscr{V}, B)$ be an eventually contracting RDP with associated Bellman and policy operators T and $\{T_\sigma\}_{\sigma \in \Sigma}$ We aim to show that all of these operators are globally stable on $\mathscr{V}$.

Fix $\sigma \in \Sigma$. let v and w be elements of $\mathscr{V}$. Fix $k \in \mathbb{N}$. By (5.25), at every point in the state space, we have

$$
\begin{aligned}
|(T_\sigma^k)\, v(y,z) - (T_\sigma^k)\, w(y,z)| &= |B(y,z,\sigma(y,z), T_\sigma^{k-1} v) - B(y,z,\sigma(y,z), T_\sigma^k w)| \\
&\leqslant \sum_{z' \in \mathsf{Z}} \max_{y' \in \mathsf{Y}} |(T_\sigma^{k-1} v)(y',z') - (T_\sigma^{k-1} w)(y',z')| L(z,z').
\end{aligned}
$$

(The recursive step in the first line is by (5.18).) Thus, pointwise on the state space, we have

$$
|T_\sigma^k\, v - T_\sigma^k\, w| \leqslant H |T_\sigma^{k-1} v - T_\sigma^{k-1} w|. \tag{5.26}
$$

Since the function L is nonnegative, the operator H is order-preserving on $\mathbb{R}^{\mathsf{X}}$. As a result, we can iterate on (5.26) to obtain

$$
|T_\sigma^k\, v - T_\sigma^k\, w| \leqslant HH |T_\sigma^{k-2} v - T_\sigma^{k-2} w| = H^2 |T_\sigma^{k-2} v - T_\sigma^{k-2} w|.
$$

Continuing in this way yields the pointwise bound $|T_\sigma^k v - T_\sigma^k w| \leqslant H^k |v - w|$. Applying Lemma 5.2.7, we now have $|T_\sigma^k v - T_\sigma^k w| \leqslant L^k \mathbb{1} \|v - w\|_\infty$. Hence, taking the supremum on the right and then the left,

$$
\|T_\sigma^k v - T_\sigma^k w\|_\infty \leqslant \|L^k \mathbb{1}\|_\infty \|v - w\|_\infty \leqslant \|L^k\|_\infty \|v - w\|_\infty.
$$

Since $r(L) < 1$, we have $\|L^k\|_\infty \to 0$ as $k \to \infty$. Hence T_σ is eventually contracting on $\mathscr{V}$ and therefore globally stable.

A similar argument works for T. Fixing $k \in \mathbb{N}$, we have

$$
\begin{aligned}
|(T^k)\, v(y,z) - (T^k)\, w(y,z)| &= |\max_{a \in \Gamma(y,z)} B(y,z,a,T^{k-1} v) - \max_{a \in \Gamma(y,z)} B(y,z,a,T^k w)| \\
&\leqslant \max_{a \in \Gamma(y,z)} |B(y,z,a,T^{k-1} v) - B(y,z,a,T^k w)| \\
&\leqslant \sum_{z' \in \mathsf{Z}} \max_{y' \in \mathsf{Y}} |(T^{k-1} v)(y',z') - (T^{k-1} w)(y',z')| L(z,z').
\end{aligned}
$$

This gives us (5.26) with T replacing T_σ. The rest of the proof is almost identical. □

5.2.4.3 MDPs with State-Dependent Discounting

Recall the definition of finite MDPs with state-dependent discounting. We show that, under suitable regularity conditions, every such model is an eventually contracting RDP. As a result, the optimality results in Theorem 5.2.3 go through.

Consider a finite MDP with state-dependent discounting as defined in §4.3.1.1. We can embed this model into an RDP by taking $\mathsf{X} = \mathsf{Y} \times \mathsf{Z}$, $\mathscr{V} = \mathbb{R}^{\mathsf{X}}$ and

$$B(y, z, a, v) = r(y, a) + \beta(z) \sum_{z' \in \mathsf{Z}} \sum_{y' \in \mathsf{Y}} v(y', z') Q(z, z') R(y, a, y') \tag{5.27}$$

Let $L(z, z') = \beta(z) Q(z, z')$.

Proposition 5.2.8. *The RDP defined above is globally stable RDP whenever $r(L) < 1$.*

Proof. Fix $(y, z) \in \mathsf{X}$, $a \in \Gamma(y)$ and $v, w \in \mathbb{R}^{\mathsf{X}}$. Since $\sum_{y' \in \mathsf{Y}} R(y, a, y') = 1$, we have

$$\begin{aligned} |B(y, z, a, v) - B(y, z, a, w)| &\leqslant \beta(z) \sum_{z' \in \mathsf{Z}} \sum_{y' \in \mathsf{Y}} |v(y', z') - w(y', z')| R(y, a, y') Q(z, z') \\ &\leqslant \sum_{z' \in \mathsf{Z}} \max_{y' \in \mathsf{Y}} |v(y', z') - w(y', z')| L(z, z'). \end{aligned}$$

Hence, condition (5.25) holds. Since $r(L) < 1$, the RDP is eventually contracting, and therefore globally stable, by Proposition 5.2.6. □

All of the optimality results for finite MDPs with state-dependent discounting in §4.3.1.3 follow from Proposition 5.2.8.

5.2.5 Algorithms

In §4.1.3 we studied algorithms for solving finite MDPs. In this section we do the same for finite RDPs. As we will see, the same algorithms can be applied after obvious modifications. Here we take the time to fill in some proofs and details that were deferred during our analysis of the finite MDP case, since the present setting is more general.

5.2.5.1 Value Function Iteration

An error bound for VFI.

5.2.5.2 Policy Function Iteration

Howard policy iteration. Optimistic policy iteration. Convergence results.

5.3 Applications

Classes of RDPs.

5.3.1 Risk-Sensitive MDPs

Intro and roadmap

5.3.1.1 Certainty Equivalents

Let $\mathscr{V}$ be a subset of $\mathbb{R}^{\mathsf{X}}$. We define a **risk-adjusted expectation operator** on $\mathscr{V}$ to be a map $\mathbb{F}$ from $\mathscr{V} \times \mathscr{D}(\mathsf{X})$ to $\mathbb{R}$ such that, for all $v, w \in \mathscr{V}$ and $\varphi \in \mathscr{D}(\mathsf{X})$,

(i) $v \leqslant w$ implies $\mathbb{F}(v, \varphi) \leqslant \mathbb{F}(w, \varphi)$ and

(ii) $\mathbb{F}(\lambda, \varphi) = \lambda$ for all $\lambda \geqslant 0$.

The following example is the canonical and most commonly used type of risk-adjusted expectation:

Example 5.3.1. Let $\mathscr{V} = \mathbb{R}^{\mathsf{X}}$. The expectation operator $\mathbb{F}_E(v, \varphi) := \sum_{x \in \mathsf{X}} v(x)\varphi(x)$ is a risk-adjusted expectation operator on $\mathscr{V}$.

Example 5.3.2. Let $\mathscr{V} = \mathbb{R}^{\mathsf{X}}_+$. The map

$$\mathbb{F}_\alpha(v, \varphi) := \left\{ \sum_{x \in \mathsf{X}} v^\alpha(x)\varphi(x) \right\}^{1/\alpha} \tag{5.28}$$

is a risk-adjusted expectation operator on $\mathscr{V}$ for all $\alpha \neq 0$.[1]

[1]The function v^α is defined by $v^\alpha(x) = (v(x))^\alpha$ for all x. We restrict attention to $\mathscr{V} = \mathbb{R}^{\mathsf{X}}_+$ to ensure that (5.29) is well defined. If $\alpha < 0$ and $v(x) = 0$ for some $x \in \mathsf{X}$ we set $\mathbb{F}_\alpha(v, \varphi) = 0$.

Example 5.3.3. Let $\mathscr{V} = \mathbb{R}_+^{\mathsf{X}}$. The function

$$\mathbb{F}_e(v, \varphi) := \frac{1}{\theta} \ln \left\{ \sum_{x \in \mathsf{X}} \exp(\theta v(x)) \varphi(x) \right\} \tag{5.29}$$

is a risk-adjusted expectation on $\mathscr{V}$ for all $\theta \neq 0$. In the literature,

- $\mathbb{F}_e(v, \varphi)$ is called the **entropic risk measure** of v under φ and
- $\mathbb{F}_e$ is called the **entropic risk-adjusted expectation operator**.

5.3.1.2 Optimality Results

Let (Γ, β, r, P) be a finite MDP with state space X and action space A. Let $\mathscr{V} = \mathbb{R}_+^{\mathsf{X}}$ and let r be nonnegative. Let $\mathbb{F}$ be a certainty equivalent operator on $\mathscr{V}$. For $(x, a) \in \mathsf{G}$ and $v \in \mathscr{V}$, let

$$B(x, a, v) := r(x, a) + \beta \mathbb{F}(v, P(x, a, \cdot))$$

Exercise 5.3.1. Verify that the tuple $(\Gamma, \mathscr{V}, B)$ forms a finite RDP.

We call every finite RDP of this class a **risk-sensitive MDP**. Evidently, if $\mathbb{F}$ is the ordinary expectations operator from Example 5.3.1, the the risk-sensitive MDP reduces to an standard finite MDP.

Proposition 5.3.1. *If $(\Gamma, \mathscr{V}, B)$ is a risk-sensitive MDP and the certainty equivalent operator satisfies the subadditive condition*

$$\mathbb{F}(v + \lambda \mathbb{1}, \varphi) \leqslant \mathbb{F}(v, \varphi) + \lambda \tag{5.30}$$

for all $v \in \mathscr{V}$, $\varphi \in \mathscr{D}(\mathsf{X})$ and $\lambda \in \mathbb{R}_+$, then $(\Gamma, \mathscr{V}, B)$ is contracting, with modulus of contraction β.

In particular, if the conditions of Proposition 5.3.1 hold, then $(\Gamma, \mathscr{V}, B)$ is a globally stable RDP and all of the results in Theorem 5.2.3 apply.

Proof. We show that $(\Gamma, \mathscr{V}, B)$ obeys Blackwell's condition. Fix $v \in \mathscr{V}$, $(x, a) \in \mathsf{G}$, and $\lambda \geqslant 0$. Applying (5.30) gives

$$B(x, a, v + \lambda \mathbb{1}) = r(x, a) + \beta \mathbb{F}(v + \lambda \mathbb{1}, P(x, a, \cdot)) \leqslant r(x, a) + \beta \mathbb{F}(v, P(x, a, \cdot)) + \beta \lambda.$$

The right-hand side equals $B(x, a, v) + \beta \lambda$, so Blackwell's condition is confirmed. The claim in Proposition 5.3.1 now follows from Exercise 5.2.6. □

The subadditive condition (5.30) is nontrivial. However, when $\mathscr{V} = \mathbb{R}_+^{\mathsf{X}}$, it does hold in the following important case:

Lemma 5.3.2. *The entropic risk-adjusted expectation operator $\mathbb{F}_e$ satisfies the subadditive condition* (5.30).

Proof. Fix $v \in \mathscr{V}$, $\varphi \in \mathscr{D}(\mathsf{X})$ and $\lambda \in \mathbb{R}_+$. Let X be a draw from φ. We have

$$\mathbb{F}_e(v + \lambda \mathbb{1}, \varphi) = \frac{1}{\theta} \ln \{\mathbb{E} \exp[\theta(v(X) + \lambda)]\} = \frac{1}{\theta} \ln \{\mathbb{E} \exp[\theta v(X)] \cdot \exp(\theta \lambda)\}.$$

Since $\ln(ab) = \ln a + \ln b$ for $a, b \geqslant 0$, condition (5.30) holds. □

5.3.1.3 An Example Application

5.3.2 Epstein–Zin Utility

Add introduction.

$$B(x, a, v) = \left\{ r(x, a)^\rho + \beta \left(\sum_{x'} v(x')^\eta P(x, a, x') \right)^{\rho/\eta} \right\}^{1/\rho}. \tag{5.31}$$

Let $\min r = \min_{(x,a) \in \mathsf{G}} r(x, a)$ and let $\max r$ be defined analogously.

Assumption 5.3.1. The parameters obey $\min r > 0$, $0 < \beta < 1$ and $\eta < 0 < \rho < 1$.

Set

$$m_1 := \min r \qquad \text{and} \qquad m_2 := \frac{\max r}{(1-\beta)^{1/\rho}}.$$

EXERCISE 5.3.2. Prove: If v is in $\mathscr{V}$, then $m_1 \leqslant B(x, a, v) \leqslant m_2$ for all $(x, a) \in \mathsf{G}$.

Let $\mathbb{1}$ be a vector of ones and let $\mathscr{V}$ be the order interval

$$\mathscr{V} := [v_1, v_2], \quad \text{where } v_i := m_i \cdot \mathbb{1} \text{ for } i = 1, 2.$$

EXERCISE 5.3.3. Show that the Bellman operator is a self-map on $\mathscr{V}$.

Below, for a strictly positive vector v and nonzero scalar α, the exponent v^α is take pointwise (i.e., element-by-element along the vector). With this understanding, let

$$\mathscr{W} := [w_2, w_1] \quad \text{where } w_i := v_i^\eta \text{ for } i = 1, 2,$$

and let U be the operator on $\mathscr{W}$ defined by

$$(Uw)(x) = \min_{a \in \Gamma(x)} B(x, a, v^{1/\eta})^\eta \qquad (x \in \mathsf{X}).$$

Exercise 5.3.4. Prove that U is an order-preserving self-map on $\mathscr{W}$.

Lemma 5.3.3. *The systems* $(\mathscr{V}, T)$ *and* $(\mathscr{W}, U)$ *are topologically conjugate.*

Proof. Fix $v \in \mathscr{V}$. Let H be the map sending strictly positive vector v into v^η. Notice that H maps $v \in \mathscr{V}$ into $\mathscr{W}$, since $v \in \mathscr{V}$ implies $v_1 \leqslant v \leqslant v_2$, and hence $Hv_2 \leqslant Hv \leqslant Hv_1$, which says $Hv \in \mathscr{W}$. In fact, H is a homeomorphism from $\mathscr{V}$ onto $\mathscr{W}$, with continuous inverse $H^{-1}w = w^{1/\eta}$. Moreover, for $v \in \mathscr{V}$ and any $x \in \mathsf{X}$,

$$(UHv)(x) = \min_{a \in \Gamma(x)} B(x, a, (Hv)^{1/\eta})^\eta = \min_{a \in \Gamma(x)} B(x, a, v)^\eta,$$

while

$$(HTv)(x) = [\max_{a \in \Gamma(x)} B(x, a, v)]^\eta = \min_{a \in \Gamma(x)} B(x, a, v)^\eta.$$

Thus, $UHv = HTv$ for all $v \in \mathscr{V}$, or $UH = HT$. Rearranging gives $T = H^{-1}UH$, so $(\mathscr{V}, T)$ and $(\mathscr{W}, U)$ are topologically conjugate, as claimed. □

Lemma 5.3.4. *The operator* U *is a concave order-preserving self-map on* $\mathscr{W}$.

Proof. Since $B(x, a, v)$ is monotone in v, the same is true of $B(x, a, v^{1/\eta})^\eta$, from which it follows easily that the operator U is order-preserving.

Regarding concavity, fix $(x, a) \in \mathsf{G}$ and observe that

$$B(x, a, w^{1/\eta})^\eta = \left\{ r(x, a)^\rho + \beta \left(\sum_{x'} w(x') P(x, a, x') \right)^{1/\theta} \right\}^\theta = f(\ell(w)),$$

where

$$\theta := \frac{\eta}{\rho}, \qquad f(t) := \left\{ r(x, a)^\rho + \beta t^{1/\theta} \right\}^\theta, \qquad \text{and } \ell(w) := \sum_{x'} w(x') P(x, a, x').$$

Since $\eta < 0 < \rho$, we have $\theta < 0$. Simple calculations show that $f'(t) > 0$ and $f''(t) < 0$ for all $t > 0$. Also, ℓ is order-preserving and linear. Hence $f \circ \ell$ is order-preserving and concave. In other words, $w \mapsto B(x, a, w^{1/\eta})^\eta$ is order-preserving and concave for each fixed $(x, a) \in \mathsf{G}$.

These two properties are passed on to U. The order-preserving part is easy to check. Regarding concavity, fixing $\lambda \in [0, 1]$, $w, v \in \mathscr{W}$ and $x \in \mathsf{X}$, we have

$$
\begin{aligned}
[U(\lambda w + (1-\lambda)v)](x) &= \min_{a \in \Gamma(x)} B[x, a, (\lambda w + (1-\lambda)v)^{1/\eta}]^\eta \\
&\geqslant \min_{a \in \Gamma(x)} \left\{ \lambda B(x, a, v^{1/\eta})^\eta + (1-\lambda) B(x, a, v^{1/\eta})^\eta \right\} \\
&\geqslant \lambda \min_{a \in \Gamma(x)} B(x, a, v^{1/\eta})^\eta + (1-\lambda) \min_{a \in \Gamma(x)} B(x, a, v^{1/\eta})^\eta .
\end{aligned}
$$

Since x was arbitrary, we can now write $U(\lambda w + (1-\lambda)v) \geqslant \lambda U w + (1-\lambda) U v$. Hence U is concave, as claimed. □

5.3.3 Two-Player Games

To be added.

5.4 Chapter Notes

To be added. Give a history of abstract dynamic programming.

We mentioned the fact that the discounted additively separable preference structure introduced in §5.1.1.1 is originally due to Samuelson (1939). An axiomatic foundation was supplied by Koopmans (1960). A critical review can be found in Frederick et al. (2002).

Regarding negative discounting, Loewenstein and Sicherman (1991) found that the majority of surveyed workers reported a preference for increasing wage profiles over decreasing ones, even when it was pointed out that the latter could be used to construct a dominating consumption sequence. Loewenstein and Prelec (1991) obtained similar results. In summarizing their study, they argue that, in the context of the choice problems they examined, “sequences of outcomes that decline in value are greatly disliked, indicating a negative rate of time preference” (Loewenstein and Prelec, 1991, p. 351).

Al-Najjar and Shmaya (2019) study the connection between Epstein–Zin utility and parameter uncertainty.

Mention hyperbolic discounting, including KR's work.

Chapter 6

Linear Regulators

In this chapter we study a special class of dynamic programming problems where

(i) the rule that updates the state each period is a *linear* process and

(ii) the current reward function r is *quadratic* in current states and actions.

These kinds of problems are sometimes called **linear-quadratic control problems**, or just **LQ problems**. LQ problems benefit from an extensive optimality theory and are well suited to computer implementation. This makes them particularly popular in applied settings where there is a need to handle many state and actions.

Of course, linearity is always an approximation for real-world phenomena. As such, for systems exhibiting nonlinear behavior, a common perception is that LQ models are unsuited. However, it turns out that even highly nonlinear phenomena can be approximated arbitrarily well by linear systems, provided that we are willing to work in high-dimensional spaces. As a result, LQ models remain very popular, and research at the intersection of nonlinear dynamics and LQ control is now extremely active.

In this chapter we cover the foundations of LQ control with economic applications. Pointers to more advanced theory and applications are provided in §6.3.

6.1 Foundations

As mentioned above, LQ theory uses linear dynamics. In this section we review linear stochastic processes and related topics.

6.1.1 Vector Autoregressions

Vector autoregressions are a standard representation of linear processes, routinely deployed in economic applications. Below we discuss time paths, moments and distribution dynamics for vector autoregressions, as well as foundational objects such as random vectors and conditional expectations.

6.1.1.1 Notes on Random Vectors

The definition of expectation is lifted from random variables to random vectors as follows: If $\xi = (\xi_i)_{i=1}^n$ is a random vector taking values in $\mathbb{R}^n$, the (vector-valued) **expectation** of ξ is the vector of expectations of each ξ_i. That is,

$$\mathbb{E}\,\xi := \begin{pmatrix} \mathbb{E}\,\xi_1 \\ \mathbb{E}\,\xi_2 \\ \vdots \\ \mathbb{E}\,\xi_n \end{pmatrix}.$$

The **variance-covariance matrix** of ξ is

$$\operatorname{Var}\xi := \begin{pmatrix} \operatorname{Cov}[\xi_1, \xi_1] & \operatorname{Cov}[\xi_1, \xi_2] & \cdots & \operatorname{Cov}[\xi_1, \xi_n] \\ \operatorname{Cov}[\xi_2, \xi_1] & \operatorname{Cov}[\xi_2, \xi_2] & \cdots & \operatorname{Cov}[\xi_2, \xi_n] \\ \vdots & & & \vdots \\ \operatorname{Cov}[\xi_n, \xi_1] & \operatorname{Cov}[\xi_n, \xi_2] & \cdots & \operatorname{Cov}[\xi_n, \xi_n] \end{pmatrix}$$

(assuming second moments exist). The elements along the principal diagonal are just the variance of each ξ_i. We say that ξ is

- **zero mean** if $\mathbb{E}\,\xi = 0$ and
- **isotropic** if $\operatorname{Var}\xi$ is the $n \times n$ identity.

For any nonrandom $b \in \mathbb{R}^n$ and $A \in \mathbb{M}^{k \times n}$, we have

$$\mathbb{E}[A\xi + b] = A\mathbb{E}[\xi] + b \quad \text{and} \quad \operatorname{Var}[A\xi + b] = A\operatorname{Var}[\xi]A^\top. \tag{6.1}$$

It is clear that $\operatorname{Var}\xi$ is always symmetric. In fact $\operatorname{Var}\xi$ is also positive semi-definite: From (6.1) we have, for arbitrary $a \in \mathbb{R}^n$,

$$a^\top \operatorname{Var}[\xi]a = \operatorname{Var}[a^\top \xi] \geqslant 0.$$

Expectations of random matrices are defined analogously:

$$Z = (Z_{ij}) \in \mathbb{M}^{n\times k} \quad \Longrightarrow \quad \mathbb{E}Z := (\mathbb{E}Z_{ij}) \in \mathbb{M}^{n\times k}.$$

Here each Z_{ij} is a random variable and expectation is taken element-by-element.

6.1.1.2 Multivariate Gaussians

A scalar random variable ζ is called **standard normal** and we write $\zeta \stackrel{d}{=} N(0,1)$ if the distribution of ζ has density $f(z) = (2\pi)^{-1/2}\exp(-z^2)$. A random variable ξ has a **normal** (or **Gaussian**) distribution and we write $\xi \stackrel{d}{=} N(\mu,\sigma)$ if, for some $\mu \in \mathbb{R}$ and $\sigma \geqslant 0$, $\xi \stackrel{d}{=} \mu + \sigma\zeta$ for some $\zeta \sim N(0,1)$. Note that we allow $\sigma = 0$ here, so ξ can be degenerate.

Now let μ be a vector in $\mathbb{R}^n$ and let Σ be a positive semidefinite element of $\mathbb{M}^{n\times n}$. A random vector ξ taking values in $\mathbb{R}^n$ is called **multivariate Gaussian** and we write $\xi \stackrel{d}{=} N(\mu,\Sigma)$ if the scalar random variable $h^\top\xi$ is normally distributed, with

$$h^\top\xi \stackrel{d}{=} N(h^\top\mu, h^\top\Sigma h) \ \text{ in } \mathbb{R} \text{ for all } h \in \mathbb{R}^n.$$

The first and second moments of ξ are then given by

$$\mathbb{E}\,\xi = \mu \quad \text{and} \quad \operatorname{Var}\xi = \Sigma.$$

Remark 6.1.1. The meaning of $\xi \stackrel{d}{=} N(\mu,\Sigma)$ is sometimes expressed by stating that ξ has density

$$\varphi(x) = \det(2\pi\Sigma)^{-1/2}\exp\left(-\frac{1}{2}(x-\mu)^\top\Sigma^{-1}(x-\mu)\right) \qquad (x \in \mathbb{R}^n).$$

However, this representation requires that Σ is positive definite—a condition that becomes increasingly restrictive when the dimensionality of ξ increases. The definition we used above avoids this problem.

Even if $\xi_1,\ldots,\xi_n$ are normally distributed in $\mathbb{R}$, it does not necessarily follow that $\xi := (\xi_1,\ldots,\xi_n)$ is multivariate Gaussian. However,

Lemma 6.1.1. *If $\xi_1,\ldots,\xi_n$ are independent and normally distributed, then $\xi = (\xi_1,\ldots,\xi_n)$ is multivariate Gaussian.*

A proof can be found in Dudley (2002).

6.1.1.3 The VAR Model

Consider a first order deterministic systems of the form $x_{t+1} = Ax_t + b$, where (x_t) takes values in $\mathbb{R}^n$ and A is $n \times n$. Such systems are typically called **linear**, although the map $x \mapsto Ax + b$ is actually affine rather than linear when $b \neq 0$.

Lemma 6.1.2. *If $(x_t)_{t \geqslant 0}$ obeys $x_{t+1} = Ax_t + b$ and $r(A) < 1$, then $\lim_{t \to \infty} x_t = (I - A)^{-1} b$.*

Exercise 6.1.1. Prove Lemma 6.1.2. [Hint: See Exercise 1.2.7 on page 16.]

A **first order vector autoregression** (first order VAR) takes the system above and adds a noise term:

$$X_{t+1} = AX_t + b + C\xi_{t+1}. \tag{6.2}$$

In what follows, we always assume that

- $(\xi_t)_{t \geqslant 1}$ is a sequence of IID copies of a zero mean, isotropic random vector ξ taking values in $\mathbb{R}^j$.
- X_0 and $(\xi_t)_{t \geqslant 1}$ are independent.

Remark 6.1.2. In (6.2), the state X_t is an $n \times 1$ vector. (The symbol is capitalized because it is random, rather than because it is a matrix.)

Exercise 6.1.2. Show that, under the stated assumptions, $\mathbb{E}[X_t \xi_{t+1}^\top] = 0$.

When we study a system such as (6.2), there are two kinds of questions that usually arise. One is the dynamics of the **sample paths** $(X_t)_{t \geqslant 0}$ across realizations of uncertainty. The second is the dynamics of the *distributions* of each random vector X_t. We will start with the second question.

At first, when considering distribution dynamics, we will confine our attention to dynamics of the first two *moments*, which are

$$\mu_t := \mathbb{E} X_t \quad \text{and} \quad \Sigma_t := \operatorname{Var} X_t \qquad (t \geqslant 0).$$

For what follows it will be convenient to define, when the sums converge,

$$\mu^* := \sum_{i \geqslant 0} A^i b. \quad \text{and} \quad \Sigma^* := \sum_{i \geqslant 0} A^i C C^\top (A^\top)^i. \tag{6.3}$$

To begin our analysis, we take expectations on both sides of (6.2) to obtain

$$\mu_{t+1} = K\mu_t \quad \text{where} \quad K\mu := A\mu + b. \tag{6.4}$$

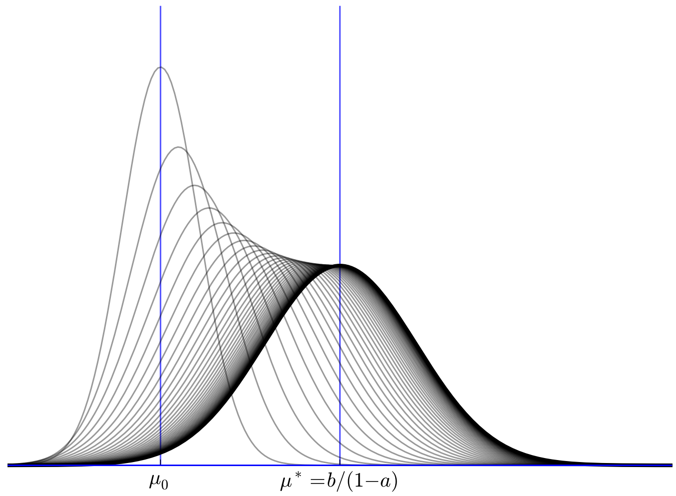


Figure 6.1: Convergence of μ_t to μ^* in the scalar model

By Exercise 1.2.7 on page 16, we have

$$r(A) < 1 \implies K \text{ is global stable on } \mathbb{R}^n \text{ with steady state } \mu^*. \tag{6.5}$$

Figure 6.1 shows convergence of the mean (and of the entire distribution) when $n = j = 1$, $A = a$ and the sequence (ξ_t) is IID and standard normal.

Next we seek a law of motion analogous to (6.4) for the matrix sequence (Σ_t). By definition,

$$\begin{aligned}\Sigma_{t+1} &= \mathbb{E}[(X_{t+1} - \mu_{t+1})(X_{t+1} - \mu_{t+1})^\top] \\ &= \mathbb{E}[(A(X_t - \mu_t) + C\xi_{t+1})(A(X_t - \mu_t) + C\xi_{t+1})^\top].\end{aligned}$$

Expanding out the last expression and using the fact that

$$\mathbb{E}[A(X_t - \mu_t)\xi_{t+1}^\top C^\top] = \mathbb{E}[C\xi_{t+1}(X_t - \mu_t)^\top A^\top] = 0$$

(see Exercise 6.1.2), we can reduce this to

$$\Sigma_{t+1} = \mathbb{E}[A(X_t - \mu_t)(X_t - \mu_t)^\top A^\top] + \mathbb{E}[C\xi_{t+1}\xi_{t+1}^\top C^\top],$$

or

$$\Sigma_{t+1} = L\Sigma_t \quad \text{where} \quad L\Sigma := A\Sigma A^\top + CC^\top. \tag{6.6}$$

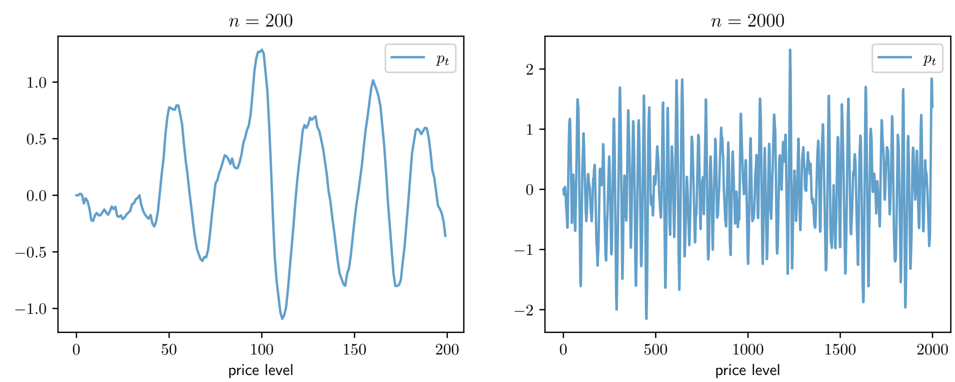


Figure 6.2: Time series of prices

This is a difference equation in matrix space. From Lemma **??** we have

$$r(A) < 1 \implies (\mathbb{M}^{n \times n}, L) \text{ is global stability with steady state } \Sigma^*. \tag{6.7}$$

It is notable that the stability conditions for (μ_t) and (Σ_t) are identical.

Exercise 6.1.3. Mankiw and Reis (2002) consider price dynamics

$$p_{t+1} = \frac{1}{1+\beta}(2p_t - p_{t-1} + \beta m_{t+1}) \tag{6.8}$$

where p_t is a price level, m_t is a measure of money supply and β is a positive parameter. Using techniques analogous to those used in §**??**, convert (6.8) into a first order VAR model as in (6.2).

Exercise 6.1.4. Write down an expression for the spectral radius of A in terms of β. Argue that the stability condition $r(A) < 1$ holds whenever $\beta > 0$.

Exercise 6.1.5. Figure 6.2 illustrate dynamics over a 200 and 2,000 period horizons respectively. In the simulation, β is set to 0.05 and (m_t) is standard normal. Replicate these figures (modulo randomness).

Exercise 6.1.6. Kydland and Prescott (1980) use the second order stochastic difference equation

$$Y_{t+1} = \alpha_1 Y_t + \alpha_2 Y_{t-1} + \varepsilon_{t+1} \tag{6.9}$$

to estimate and analyze the dynamics of detrended log output. Let (ε_t) be iid with zero mean and standard deviation σ. Map (6.9) into a VAR form.

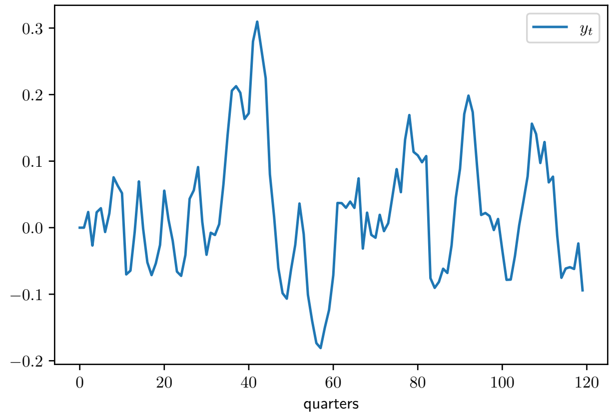


Figure 6.3: Time series of detrended log output

EXERCISE 6.1.7. Provide an expression for the eigenvalues of A in terms of the parameters.

If both eigenvalues are interior to the unit circle in $\mathbb{C}$, then $r(A) < 1$ and stability will hold. Kydland and Prescott (1980) calibrated $\hat{\alpha}_1 = 1.386$ and $\hat{\alpha}_2 = -0.477$.

EXERCISE 6.1.8. Show that, under this parameterization, both eigenvalues are real and $r(A) \approx 0.75$.

Figure 6.3 shows a simulated time series when (ε_t) is $N(0, \sigma^2)$ with $\sigma = 0.05$. The initial conditions are $Y_0 = Y_1 = 0$.

6.1.1.4 Distribution Dynamics: The Gaussian Case

In §6.1.1.3 we studied the dynamics of the first two moments of the vector autoregression $X_{t+1} = AX_t + b + C\xi_{t+1}$ where $(\xi_t)_{t \geqslant 1}$ is IID, zero mean and isotropic. We found that the time t mean and variance-covariance matrix are given by

- $K^t \mu_0$ where $K\mu := A\mu + b$ on $\mathbb{R}^n$ and
- $L^t \Sigma_0$ where $L\Sigma := A'\Sigma A + CC'$ on $\mathbb{M}^{n \times n}$.

These moments (μ_t, Σ_t) tell us something about the distribution of X_t, denoted henceforth by ψ_t. In general, first two moments provide only limited information about ψ_t. There is, however, one case where we can easily extract the full distribution ψ_t at every point in time from the first two moments: the Gaussian case.

To shift the VAR model to the Gaussian case we will assume that

$$(\xi_t)_{t \geqslant 1} \overset{\text{IID}}{\sim} N(0, I) \quad \text{and} \quad X_0 \overset{d}{=} N(\mu_0, \Sigma_0) \tag{6.12}$$

where μ_0 is any vector in $\mathbb{R}^j$ and Σ_0 is any positive semidefinite $j \times j$ matrix. Under these Gaussian conditions we have

$$X_t \overset{d}{=} N\left(K^t \mu_0, L^t \Sigma_0\right) \text{ for all } t \geqslant 0. \tag{6.13}$$

Here the claim that X_t has the first two moments specified in (6.13) has already been verified, while normality can be checked using the definition of multivariate Gaussians and an induction argument.

Exercise 6.1.9. Confirm this. In doing so, you can exploit the fact that any affine combination of *independent* normal random variables in $\mathbb{R}$ is normal.

Proposition 6.1.3. *Let ψ_t be the distribution of X_t for each $t \geqslant 0$. If $r(A) < 1$, then under the Gaussian conditions in* (6.12)*, we have*

$$\psi_t \overset{w}{\to} N(\mu^*, \Sigma^*) \qquad (t \to \infty) \tag{6.14}$$

where

(i) $\mu^* = \sum_{i=0}^{\infty} A^i b$ *and*

(ii) Σ^* *is the unique fixed point of* $\Sigma := A'\Sigma A + CC'$.

Here $\overset{w}{\to}$ means weak convergence of distributions. We discuss weak convergence in more detail in Volume II. For now we note that, for multivariate Gaussians, weak convergence is equivalent to pointwise convergence of the corresponding characteristic functions (see, e.g., Çınlar (2011)).

Proof of Proposition 6.1.3. We show that the characteristic function of the distribution $N(\mu_t, \Sigma_t)$ converges pointwise to that of $N(\mu^*, \Sigma^*)$. In our case, this translates to the claim that, at any fixed $s \in \mathbb{R}^n$,

$$\lim_{t \to \infty} \exp\left(is'\mu_t - \frac{1}{2} s'\Sigma_t s\right) = \exp\left(is'\mu^* - \frac{1}{2} s'\Sigma^* s\right) \tag{6.15}$$

Fixing such an s, to prove (6.15) it suffices to show that

$$s'\mu_t \to s'\mu^* \quad \text{and} \quad s'\Sigma_t s \to s'\Sigma^* s \quad \text{in } \mathbb{R} \text{ as } t \to \infty \tag{6.16}$$

This in turn follows from the converence in (6.5) and (6.7) (see, e.g., Exercise **??**). □

Example 6.1.1. Consider the scalar **AR(1)**case, where $(X_t) \subset \mathbb{R}$ obeys

$$X_{t+1} = aX_t + b + \sigma\varepsilon_{t+1}, \quad (\varepsilon_t) \overset{\text{IID}}{\sim} N(0,1). \tag{6.17}$$

This is a version of the Gaussian VAR with $A = a$ and other obvious identifications. The case $|a| < 1$ is known as the **mean-reverting** case, under which the distribution of X_t converges weakly to

$$\psi^* := N\left(\frac{b}{1-a}, \frac{\sigma^2}{1-a^2}\right) \tag{6.18}$$

Since, in this case $r(A) = |a|$, the stable case in the sense of Proposition 6.1.3 coincides with the mean-reverting case.

6.1.1.5 An Analytical View

We can translate the results from this section into the language of dynamical systems. Let $\mathscr{G}$ be the set of all Gaussian distributions on $\mathbb{R}^n$, endowed with the topology of weak convergence. Let P be the operator on $\mathscr{G}$ defined by

$$\mathscr{G} \ni \psi := N(\mu, \Sigma) \mapsto \psi P := N(A\mu + b, A'\Sigma A + CC') \in \mathscr{G}.$$

Then P updates distributions by one period, in the sense that the marginal distributions $(\psi_t)_{t \geqslant 0}$ of the state process obey $\psi_{t+1} = \psi_t P$. We have written the argument ψ to the left, as in ψP rather than, say, $P\psi$ or $P(\psi)$, so as to tie in with the notation in §**??**, and in particular with (**??**) on page **??**, where an analogous operation is carried out in a discrete setting.

Proposition 6.1.3 tells us that $(\mathscr{G}, P)$ is globally stable whenever $r(A) < 1$. The unique steady state is $\psi^* := N(\mu^*, \Sigma^*)$. In this context, ψ^* is called a **stationary distribution** of P. This object is analogous to the stationary distribution for wealth dynamics discussed in §**??**.

6.1.2 State Space Models

Add roadmap.

6.1.2.1 The Model

Let's now extend the VAR model from (6.2) to the standard **linear state space** model

$$X_{t+1} = AX_t + b + C\xi_{t+1} \tag{6.19}$$
$$Y_t = GX_t + H\zeta_t \tag{6.20}$$

where

- A is $n \times n$, b is $n \times 1$ and C is $n \times j$.
- G is $k \times n$ and H is $k \times \ell$.
- $(\xi_t)_{t \geqslant 0}$ are IID copies of the zero mean isotropic $j \times 1$ random vector ξ.
- $(\zeta_t)_{t \geqslant 0}$ are IID copies of the zero mean isotropic $\ell \times 1$ random vector ζ.

As usual (X_t) is called the **state** process. Its initial condition X_0 is assumed to be independent of (ξ_t) and (ζ_t). The $k \times 1$ process (Y_t) is called the **observation process**. The processes (ξ_t) and (ζ_t) are also independent.

Linear state space models are often used in a setting where we envisage imperfect observation of an economic system, either by an econometrician or an agent within a model. We will discuss an example of this form below. In other settings, the linear state space model is simply a convenient extension of the basic VAR model.

Example 6.1.2. The "canonical linear model" of (log) labor earnings discussed in **?** is

$$Y_t = X_t + h\zeta_t \quad \text{where} \quad X_{t+1} = \rho X_t + b + c\xi_{t+1}$$

and (ξ_t, ζ_t) is IID. Here h, ρ, b, c are parameters with $|\rho| < 1$. This is a scalar linear state space model. X_t is called the **persistent component** of labor income, while (ζ_t) is called the **transitory component**.

Example 6.1.3. The dynamic second order linear model in (6.9) can be reorganized into the first order model

$$X_t := \begin{pmatrix} Y_t \\ Y_{t-1} \end{pmatrix}, \quad A := \begin{pmatrix} \alpha_1 & \alpha_2 \\ 1 & 0 \end{pmatrix}, \quad C := \begin{pmatrix} \sigma \\ 0 \end{pmatrix} \quad \text{and} \quad \xi_t := \frac{1}{\sigma}\varepsilon_t$$

If we now take $G = (1, 0)'$ and $H = 0$, we extract (Y_t) from (X_t).

6.1.2.2 Dynamics

We can easily compute the first two moments of the observation process given our results on the moments of the state process in §6.1.1.3. Recalling that

- $\mu_t = K^t \mu_0$ where $K\mu := A\mu + b$ on $\mathbb{R}^n$ and
- $\Sigma_t = L^t \Sigma_0$ where $L\Sigma := A'\Sigma A + CC'$ on $\mathbb{M}^{n \times n}$

we obtain

$$\mathbb{E} Y_t = G\mu_t \quad \text{and} \quad \operatorname{Var} Y_t = G\Sigma_t G' + HH' \tag{6.21}$$

The evolution of this sequence is determined by $K^t \mu_0$ and $L^t \Sigma_0$. This is natural because the state process is the driver of dynamics in the linear state space model. We know that if $r(A) < 1$, then $K^t \mu_0$ and $L^t \Sigma_0$ converge. In particular, we have

$$\mathbb{E} X_t \to \mu^*, \quad \mathbb{E} Y_t \to G\mu^*, \quad \operatorname{Var} X_t \to \Sigma^*, \text{ and } \operatorname{Var} Y_t \to G\Sigma^* G' + HH' \tag{6.22}$$

as $t \to \infty$, where μ^* and Σ^* are the respective fixed points of F and L.

A common setting for the linear state space model is the Gaussian one, where the assumptions above are supplemented by

Assumption 6.1.1. The random vectors ξ and ζ are multivariate Gaussian.

Provided that X_0 is also Gaussian, the first two moments then pin down the distribution of X_t, which we saw in (6.13), and from that the distribution of Y_t:

$$Y_t \stackrel{d}{=} N\left(G\mu_t, G\Sigma_t G' + HH'\right) \tag{6.23}$$

When $r(A) < 1$, the stationary distribution is Gaussian with the moments provided in (6.22).

Exercise 6.1.10. Use the characteristic function approach found in the proof of Proposition 6.1.3 to show that

$$r(A) < 1 \implies Y_t \xrightarrow{w} N\left(G\mu^*, G\Sigma^* G' + HH'\right).$$

6.1.3 Conditioning and Martingales

Next we review prediction based on conditional expectations. Conditional expectations are themselves a cornerstone of economic theory and empirics, since they describe optimal forecasts based on limited information. Here we provide a brief treatment that suffices for what follows. (Volume II contains more details and proofs.)

Add roadmap. Note that we have already used conditional expectations. We are now adding formal structure.

6.1.3.1 Definition

Let Y and the elements of $\mathscr{G} := \{X_1, \ldots, X_k\}$ be scalar random variables. Consider the problem of predicting Y given $\mathscr{G}$. That is, we wish to form a prediction of the value that Y will take once $X_1, \ldots, X_k$ are known, without any additional information on the state of the world. Another way to say this is that we seek a (deterministic) function $f : \mathbb{R}^k \to \mathbb{R}$ such that

$$\hat{Y} := f(X_1, \ldots, X_k) \text{ is a good predictor of } Y.$$

To find such an f we must define what "good" means. The most common definition in the present context is that **mean squared error** $\mathbb{E}[(\hat{Y} - Y)^2]$ is small. Thus, we have a minimization problem in function space (the set from which f is chosen). Based on projection arguments (see Volume II), it can be shown that there exists an essentially unique $\hat{f}$ in the set of functions from $\mathbb{R}^k$ to $\mathbb{R}$ that solves

$$\hat{f} = \operatorname*{argmin}_f \mathbb{E}[(Y - f(X_1, \ldots, X_k))^2]. \tag{6.24}$$

We call the resulting variable

$$\hat{Y} := \hat{f}(X_1, \ldots, X_k)$$

the **conditional expectation** of Y given $\mathscr{G}$. Common alternative notations for $\hat{Y}$ include

$$\mathbb{E}_{\mathscr{G}} Y =: \mathbb{E}[Y \,|\, \mathscr{G}] =: \mathbb{E}[Y \,|\, X_1, \ldots, X_k]$$

In the present context, $\mathscr{G}$ is often called an **information set**.

6.1.3.2 Properties

In the next proposition, a random variable Y is called **$\mathscr{G}$-measurable** if there exists a function f such that $Y = f(X_1, \ldots, X_k)$. Intuitively, Y is perfectly predictable given the data in $\mathscr{G}$.

Proposition 6.1.4. *Let X and Y be random variables with finite first moment and let $\mathscr{G}$ and $\mathscr{H}$ be information sets. The following properties hold:*

(i) *$\mathbb{E}_{\mathscr{G}} X$ is $\mathscr{G}$-measurable*

(ii) *If $\mathscr{G} \subset \mathscr{H}$, then $\mathbb{E}_{\mathscr{G}}[\mathbb{E}_{\mathscr{H}} Y] = \mathbb{E}_{\mathscr{G}} Y$ and $\mathbb{E}[\mathbb{E}_{\mathscr{G}} Y] = \mathbb{E} Y$.*

(iii) *If Y is independent of the variables in $\mathscr{G}$, then $\mathbb{E}_{\mathscr{G}} Y = \mathbb{E} Y$.*

(iv) *If Y is $\mathscr{G}$-measurable, then* $\mathbb{E}_{\mathscr{G}} Y = Y$.

(v) *If X is $\mathscr{G}$-measurable, then* $\mathbb{E}_{\mathscr{G}}[XY] = X\mathbb{E}_{\mathscr{G}} Y$.

(vi) $\mathbb{E}_{\mathscr{G}}[\alpha X + \beta Y] = \alpha \mathbb{E}_{\mathscr{G}} X + \beta \mathbb{E}_{\mathscr{G}} Y$ *for all* α, β *in* $\mathbb{R}$.

Property (i) states that the linearity of expectations is preserved under conditioning. Property (ii) is called the **law of iterated expectations**, and is shared by all projections. Property (v) is sometimes called **conditional determinism**, since X can be treated like a constant when it is pinned down by the information set.

For a proof see Volume II or Çınlar (2011). The applications below provide practice applying these rules.

6.1.3.3 Vector-Valued Conditional Expectations

If $Y = (Y_1, \ldots, Y_m)$ is a vector, then the conditional expectation of Y is the vector containing the conditional expectation of each element (similar to ordinary vector expectations). Thus, written as column vectors,

$$\mathbb{E}_{\mathscr{G}} \begin{pmatrix} Y_1 \\ \vdots \\ Y_m \end{pmatrix} = \begin{pmatrix} \mathbb{E}_{\mathscr{G}} Y_1 \\ \vdots \\ \mathbb{E}_{\mathscr{G}} Y_m \end{pmatrix},$$

where $\mathscr{G}$ is an arbitrary information set.

EXERCISE 6.1.11. Prove that $\mathbb{E}_{\mathscr{G}}[AY + b] = A\mathbb{E}_{\mathscr{G}}[Y] + b$ for any $b \in \mathbb{R}^n$ and $A \in \mathbb{M}^{k \times n}$.

6.1.3.4 Application: Factoring Multi-Period SDFs

Let $(m_t)_{t \geqslant 0}$ be a sequence of one-period stochastic discount factors (also called **pricing kernels**). By this we mean that, for all $t \geqslant 0$, the time t price of an arbitrary asset that pays g_{t+1} at $t+1$ is

$$\pi_t = \mathbb{E}_t \, m_{t+1} \, g_{t+1}, \tag{6.25}$$

where $\mathbb{E}_t$ is expectation conditioning on all states up to and including date t.

In this section we aim to show that the n-period SDF can be obtained as the n-fold product of the one-period SDFs. In other words, we claim that the time t price of asset that pays g_{t+n} at $t+n$ and zero in other periods is

$$\pi_t = \mathbb{E}_t \prod_{i=1}^{n} m_{t+i} \, g_{t+n}. \tag{6.26}$$

We will prove this claim for the case $n = 2$. The extension to arbitrary n is straightforward (use induction).

By (6.25), the time $t+1$ price of g_{t+2} is $\pi_{t+1} = \mathbb{E}_{t+1}\, m_{t+2}\, g_{t+2}$. If we buy the asset at time t, then we can sell it at $t+1$ at its current price. Thus, π_{t+1} gives the one-period-ahead payoffs in each state obtained by buying the asset at time t. Hence, using (6.25) once more, it will be priced at time t by $\pi_t = \mathbb{E}_t\, m_{t+1}\, \pi_{t+1}$. Putting these equations together gives

$$\pi_t = \mathbb{E}_t\, m_{t+1}\, \mathbb{E}_{t+1}\, m_{t+2}\, g_{t+2}.$$

Applying (v) in Proposition 6.1.4 and then the law of iterated expectations (property (ii)), this simplifies to

$$\pi_t = \mathbb{E}_t\, m_{t+1}\, m_{t+2}\, g_{t+2},$$

which is (6.26) in the case of $n = 2$.

6.1.3.5 Forecasts for State Space Models

Assume the setting of §6.1.2.1 and suppose we wish to forecast geometric sums such as

$$s_X := \mathbb{E}_t \left[\sum_{j=0}^{\infty} \beta^j X_{t+j} \right] \quad \text{and} \quad s_Y := \mathbb{E}_t \left[\sum_{j=0}^{\infty} \beta^j Y_{t+j} \right], \tag{6.27}$$

where $\beta > 0$ and $\mathbb{E}_t$ is expectation conditioning on the information set $X_0, \ldots, X_t$. For example,

- if (Y_t) is a cash flow, then s_Y represents a risk-neutral asset price.
- If (Y_t) is money supply, then s_Y is a model of the price level.

Exercise 6.1.12. Show that $\mathbb{E}_t X_{t+j} = A^j X_t$ and $\mathbb{E}_t Y_{t+j} = G A^j X_t$ for all $j \geqslant 0$.

Exercise 6.1.13. When $r(A) < 1/\beta$, we can pass the expectation through the infinite sum in (6.27). Show that, in this case, we have

$$\mathbb{E}_t \left[\sum_{j=0}^{\infty} \beta^j X_{t+j} \right] = [I - \beta A]^{-1} X_t \quad \text{and} \quad \mathbb{E}_t \left[\sum_{j=0}^{\infty} \beta^j Y_{t+j} \right] = G[I - \beta A]^{-1} X_t.$$

6.1.3.6 Martingales

Stochastic models are usually pieced together from elementary components, such as IID innovations. Another such building block is martingales.

Discuss the importance of martingales in economics. Exchange rates should be relatively unpredictable. Meese and Rogoff 1983. Perhaps connect to equivalent martingale measures in a discrete setting.

To define martingales, we need the notion of a **filtration**, which is a sequence of information sets $(\mathscr{G}_t)_{t \geqslant 0}$ increasing in the sense of set inclusion, so that $\mathscr{G}_t \subset \mathscr{G}_{t+1}$ for all t.

Example 6.1.4. If $(\xi_t)_{t\geqslant 0}$ is a stochastic process, then the sequence $(\mathscr{G}_t)_{t\geqslant 0}$ with $\mathscr{G}_t := \{\xi_0, \ldots, \xi_t\}$ is a filtration. We call this the **filtration generated by $(\xi_t)_{t\geqslant 0}$**.

A stochastic process $(\eta_t)_{t\geqslant 0}$ is said to be **adapted** to a given filtration $(\mathscr{G}_t)_{t\geqslant 0}$ if η_t is $\mathscr{G}_t$-measurable for all $t \geqslant 0$.

A stochastic process $(w_t)_{t\geqslant 1}$ taking values in $\mathbb{R}^n$ is called a **martingale** with respect to a filtration $(\mathscr{G}_t)$ if it is adapted to $(\mathscr{G}_t)$, satisfies the moment condition $\mathbb{E}\|w_t\|_1 < \infty$, and

$$\mathbb{E}[w_{t+1} \,|\, \mathscr{G}_t] = w_t, \qquad \text{for all } t \geqslant 0.$$

In other words, our best forecast of next period's value is the current value.

Example 6.1.5. Consider a scalar **random walk**, which is a sequence (w_t) of the form

$$w_t = \sum_{i=1}^{t} \xi_i, \qquad (\xi_t) \overset{\text{IID}}{\sim} F \text{ and } \mathbb{E}\,\xi_t = \int xF(\mathrm{d}x) = 0.$$

For example, w_t might be a player's wealth over a sequence of fair gambles. Figure 6.4 shows 12 realizations of a random walk when (ξ_t) is standard normal.

Exercise 6.1.14. Prove that (w_t) is a martingale with respect to the filtration generated by (ξ_t).

Exercise 6.1.15. Consider the sequence (w_t) defined by

$$w_t = \prod_{i=1}^{t} \xi_i, \qquad (\xi_t) \overset{\text{IID}}{\sim} F \text{ and } \mathbb{E}\,\xi_t = \int xF(\mathrm{d}x) = 1.$$

Show that this process is a martingale with respect to the filtration generated by (ξ_t).

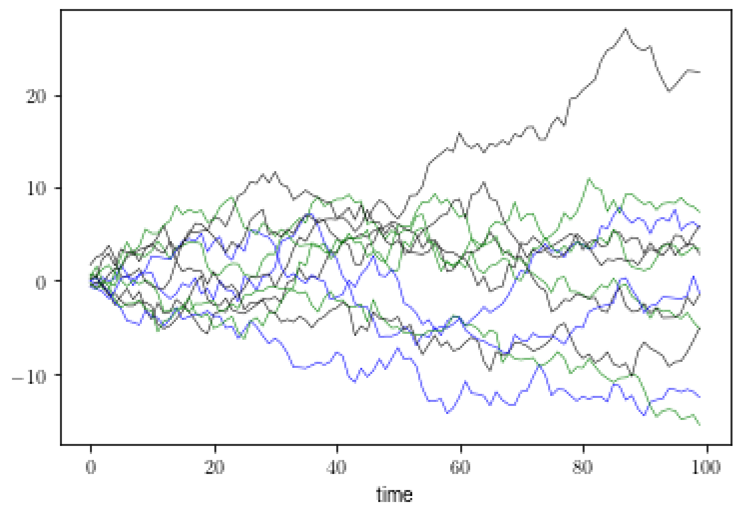


Figure 6.4: Twelve realizations of a random walk

A stochastic process $(w_t)_{t \geqslant 1}$ in $\mathbb{R}^n$ is called a **martingale difference sequence** (or **MDS**) with respect to a filtration $(\mathscr{G}_t)$ if $(w_t)_{t \geqslant 1}$ is adapted to $(\mathscr{G}_t)$, if $\mathbb{E}\|w_t\|_1 < \infty$ and if

$$\mathbb{E}[w_{t+1} \,|\, \mathscr{G}_t] = 0, \qquad \forall\, t \geqslant 1.$$

For example, if (v_t) is a martingale with respect to $(\mathscr{G}_t)$ then the first difference $w_t := v_t - v_{t-1}$ is an MDS with respect to $(\mathscr{G}_t)$, since for any t,

$$\mathbb{E}[w_{t+1} \,|\, \mathscr{G}_t] = \mathbb{E}[v_{t+1} - v_t \,|\, \mathscr{G}_t] = \mathbb{E}[v_{t+1} \,|\, \mathscr{G}_t] - \mathbb{E}[v_t \,|\, \mathscr{G}_t] = v_t - v_t = 0$$

An MDS is a generalization of the idea of a zero mean IID sequence, and is often used in economics and related fields to represent the idea of an "unpredictable" sequence. To see that it is a generalization, suppose that (w_t) is IID with $\mathbb{E}[w_t] = 0$. Then (w_t) is an MDS with respect to the **natural filtration**, which is the filtration generated by itself. This follows from independence, since, with $\mathscr{G}_t = \{w_1, \ldots, w_t\}$, we have $\mathbb{E}[w_{t+1} \,|\, \mathscr{G}_t] = \mathbb{E}[w_{t+1}]$ for all t. The conclusion follows.

EXERCISE 6.1.16. Show that if (w_t) is an MDS with respect to some filtration $(\mathscr{G}_t)$, then $\mathbb{E}[w_t] = 0$ for all t.

EXERCISE 6.1.17. Show that if (w_t) is an MDS with respect to $(\mathscr{G}_t)$, then w_s and w_t are **orthogonal**, in the sense that $\mathbb{E}[w_s w_t'] = 0$ whenever $s \neq t$.

6.1.3.7 A Note on Borel Functions

This section mentions some technical issues related to expectations and conditioning that were glossed over in the presentation above. They can be safely ignored by any reader keen to move on. We mention them only for completeness.

In §6.1.3.2, our definition of measurability was slightly imprecise. It would be more correct to say that Y is $\mathscr{G}$-measurable if there exists a *Borel measurable* function f such that $Y = f(X_1, \ldots, X_k)$. But what does this mean?

Here we give a brief introduction to Borel functions, with a full treatment deferred to Volume II.

When working with functions, we often need to place restrictions on the functions we consider for the problem in question to make sense. For example, it would be embarassing if our proposed solution method for a given problem involved a Taylor expansion and yet the functions we applied this method to had kinks. In this case the "algorithm" is not well defined. At the same time, some interesting functions do have kinks, so we would not want to rule such functions out if no differentiation is involved.

These thoughts lead us to consider classes of "nice" functions, that are well behaved in one way or another. Linear functions on $\mathbb{R}$ are certainly well behaved, as well as being easy to describe. Polynomial functions are a natural generalization. These, in turn, are a special case of the functions on $\mathbb{R}$ that have derivatives of every order. Such functions are special case of the "smooth" functions, which are those functions having continuous first derivative. The smooth functions are contained in the class of Lipschitz functions, which lie inside the class of everywhere continuous functions.

Need we generalize any more? The answer is "yes!" Sometimes economic variables exibit jumps. Agents make sudden changes in behavior. This means that we must admit discontinuities. At the same time, we do not wish to stray too far from continuity, so as to avoid dealing with the wildest functions mathematicians can dream up. This naturally leads us to the Borel functions.

Let M and N be subsets of $\mathbb{R}^d$. Readers familiar with analysis will know that a function f from M to N is continuous on M if and only if $f^{-1}(G)$ is open in M whenever G is open in N. The definition of a **Borel function** (sometimes called a **Borel measurable function**) weakens this restriction by requiring instead that, whenever G is open in N, the preimage $f^{-1}(G)$ lies in a larger class of sets than the open sets called the Borel sets.

The Borel sets are discussed in detail in Volume II JOHN: THIS IS VOLUME 2 but we can define them easily enough. The set of **Borel sets** of M, denoted here by $\mathscr{B}$, is the smallest collection of subsets of M that contains the open sets and is also closed under the taking of complements and countable unions. In other words, $\mathscr{B}$ contains the open sets, satisfies

(i) $M \in \mathscr{B}$,

(ii) $B \in \mathscr{B}$ implies $B^c \in \mathscr{B}$, and

(iii) if $(B_n)_{n \geqslant 1}$ is a sequence contained in $\mathscr{B}$, then $\cup_n B_n \in \mathscr{B}$;

and, in addition, if $\mathscr{A}$ is another collection of subsets of M that contains then open sets and satisfies these three properties, then $\mathscr{B} \subset \mathscr{A}$.

There are three reasons why Borel functions are so important to modern analysis. First, in most settings, the set of Borel functions is much larger than the class of continuous functions, precisely because the class of Borel sets is much larger than the class of open sets. For example, it can be shown that, when $M = N = \mathbb{R}$, the class of Borel functions includes not just the continuous functions but also any increasing function, any convex function, any concave function, any linear combination of these kinds of functions, any continuous transformation of any of these functions, and so on.

Second, the class of Borel functions is closed under all the standard arithmetic and limiting operations. Sums and scalar multiples of Borel functions are Borel functions. Pointwise limits, suprema and infima of Borel functions are Borel functions, and so on. This is important for consistency because it means that we will not inadvertently introduce unpleasant functions through standard operations.

Third, when considering integrals such as $\int f(x)\,\mathrm{d}x$ or $\mathbb{E} f(X)$, we often need to admit discontinuous f but, at the same time, cannot admit arbitrary f while still being confident about basic properties of the integral, such as linearity. It turns out that Borel functions admit a well defined theory of integration with all the usual helpful properties. We will revisit this in Volume II.

6.2 Linear Quadratic Models

Add roadmap. Regarded as old fashioned by some but actually at the forefront of data-driven engineering and control. Same idea: project to high dimensions and then use linear methods. Give cites from journals, survey articles.

6.2.1 LQ Asset Pricing

Add roadmap. A nice warm up for LQ control.

6.2.1.1 LQ Dividends

In §**??** we studied the risk neutral asset pricing formula

$$\pi_t = \beta \, \mathbb{E}_t \, [d_{t+1} + \pi_{t+1}], \tag{6.28}$$

where $\beta \in (0, 1)$ discounts next period values, π_t is price at time t, $\mathbb{E}_t$ is time t conditional expectation, and dividends $d_t = d(X_t)$ is a function of a finite Markov chain (X_t). The price π_t is the endogenous process we wish to solve for.

We now revisit this equation, after replacing the finite Markov assumption with

- $X_{t+1} = AX_t + C\xi_{t+1}$ for all $t \geqslant 0$, where (ξ_t) is an isotropic martingale difference sequence taking values in $\mathbb{R}^j$ and
- $d_t = X_t^\top DX_t$ for some positive semidefinite $D \in \mathbb{M}^{n \times n}$.

In the expressions above, X_t is $n \times 1$, A is $n \times n$ and C is $n \times j$. The restriction on D ensures nonnegative dividends.

Exercise 6.2.1. Under the stated assumptions, verify the following claim, which we use below to forecast quadratic terms:

$$\mathbb{E}_t [x_{t+1}^\top H x_{t+1}] = x_t^\top A^\top H A x_t + \mathrm{trace}(C^\top H C) \quad \text{for all } H \in \mathbb{M}^{n \times n}.$$

6.2.1.2 Prices as Functions of the State

Assume the set up in §6.2.1.1. As in §**??**, we conjecture that π_t is a stationary function of X_t. We will take a second leap and guess that, in the present setting, prices are a quadratic in X_t. In particular, we guess that, for some $P \in \mathbb{M}^{n \times n}$ and $\delta \in \mathbb{R}$,

$$\pi_t = X_t^\top P X_t + \delta \quad \text{for all } t \geqslant 0. \tag{6.29}$$

Our task is to show this price system satisfies the pricing equation for suitable P, δ.

EXERCISE 6.2.2. Suppose there exists a positive semidefinite $P \in \mathbb{M}^{n \times n}$ and $\delta \in \mathbb{R}$ such that, for all $x \in \mathbb{R}^n$,

$$x^\top P x + \delta = \beta x^\top A^\top (D + P) A x + \beta \operatorname{trace}(C^\top (D + P) C) + \beta \delta \tag{6.30}$$

Prove that the pricing equation (6.28) holds when (π_t) is as given in (6.29).

It remains to find conditions under which there exists a pair (P, δ) with the stated properties. We do this in two steps.

EXERCISE 6.2.3. Suppose there exist a $P \in \mathbb{M}^{n \times n}$ such that

$$P = \beta A^\top (D + P) A. \tag{6.31}$$

Prove that, if this is true and

$$\delta := \frac{\beta}{1 - \beta} \operatorname{trace}(C^\top (D + P) C) \tag{6.32}$$

then (P, δ) obeys (6.30) for all $x \in \mathbb{R}^n$.

The last step is to find a positive semidefinite P that satisfies (6.31).

EXERCISE 6.2.4. Prove that a unique positive semidefinite solution to (6.31) exists whenever $r(A) < 1/\sqrt{\beta}$.

Our treatment is complete. Under the stability condition $r(A) < 1/\sqrt{\beta}$, we can solve the Lyapunov equation (6.31) for P, obtain δ via (6.32), and then compute prices via $\pi_t := X_t^\top P X_t + \delta$. These prices satisfies the risk neutral asset pricing equation.

6.2.2 LQ Control

Add roadmap.

6.2.2.1 Dynamics

Linear quadratic (LQ) control problems are a special class of dynamic decision problems in which dynamics are linear and rewards are quadratic. These assumptions facilitate tractability in high dimensions.

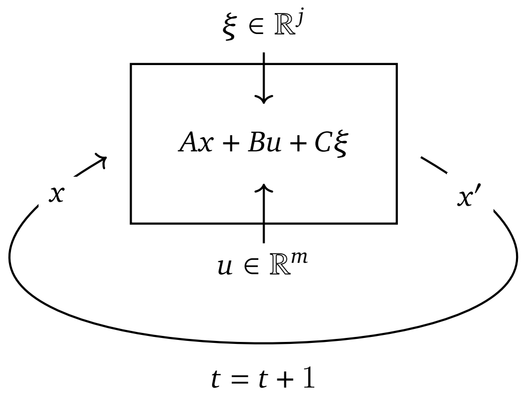


Figure 6.5: State dynamics for LQ control problems

The dynamics of the state process (x_t) are

$$x_{t+1} = Ax_t + Bu_t + C\xi_{t+1} \tag{6.33}$$

with x_0 given, where

- the **state vector** (x_t) takes values in $\mathbb{R}^n$,
- the **control vector** (u_t) takes values in $\mathbb{R}^m$,
- the matrices A and B are $n \times n$ and $n \times m$ respectively, while
- C is $n \times j$ and (ξ_t) is IID, zero mean and isotropic.

Intuitively, a decision maker chooses a control sequence (u_t) to guide the state (x_t) but transitions are buffeted by shocks (ξ_t). Figure 6.5 provides a visualization with x' being the updated state.

For example, consider the law of motion for wealth

$$w_{t+1} = (1 + r)(w_t - c_t) + y_{t+1}$$

that we saw previously in §**??**. For the moment we assume that $y_t = \mu + \sigma\xi_t$ where (ξ_t) is IID $N(0, 1)$ in $\mathbb{R}$. We set $u_t := c_t - \bar{c}$ where $\bar{c}$ is some "ideal" level of consumption (which can be arbitrarily large). Then

$$w_{t+1} = (1 + r)(w_t - u_t - \bar{c}) + \mu + \sigma\xi_{t+1}. \tag{6.34}$$

To write (6.34) in the form of equation (6.33), consider

$$\begin{pmatrix} w_{t+1} \\ 1 \end{pmatrix} = \begin{pmatrix} 1+r & -(1+r)\bar{c}+\mu \\ 0 & 1 \end{pmatrix} \begin{pmatrix} w_t \\ 1 \end{pmatrix} + \begin{pmatrix} -(1+r) \\ 0 \end{pmatrix} u_t + \begin{pmatrix} \sigma \\ 0 \end{pmatrix} \xi_{t+1}$$

The first row is equivalent to (6.34). Moreover, the model is now linear and can be written in the form of (6.33) by setting $x_t = (w_t, 1)^\top$ along with

$$A := \begin{pmatrix} 1+r & -(1+r)\bar{c}+\mu \\ 0 & 1 \end{pmatrix}, \quad B := \begin{pmatrix} -(1+r) \\ 0 \end{pmatrix} \quad \text{and} \quad C := \begin{pmatrix} \sigma \\ 0 \end{pmatrix}$$

6.2.2.2 Rewards

In the LQ model we will aim to minimize a flow of losses, where current loss is given by

$$x_t^\top R x_t + u_t^\top Q u_t. \tag{6.35}$$

Here

- R is $n \times n$ and positive semidefinite.
- Q is $m \times m$ and positive definite.

Example 6.2.1. For the household with budget constraint (6.34), a typical choice of R and Q is

$$Q := 1 \quad \text{and} \quad R := \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

Recalling that $x_t = (w_t, 1)^\top$ and $u_t := c_t - \bar{c}$, this leads to current loss

$$x_t^\top R x_t + u_t^\top Q u_t = u_t^2 = (c_t - \bar{c})^2$$

In particular, the household's current loss is the squared deviation of consumption from the ideal level $\bar{c}$.

6.2.3 Finite Horizon Optimality

We begin studying optimal control in a finite horizon setting, which facilitates injecting time into a model and hence generating time dependent policies. Studies of life cycle savings and consumption are one example, since savings behavior differs across stages of life.

6.2.3.1 Theory

Assuming terminal time $T \in \mathbb{N}$, the problem is to choose a sequence of controls $u_0, \ldots, u_{T-1}$ to minimize

$$\mathbb{E}\left\{\sum_{t=0}^{T-1} \beta^t (x_t^\top R x_t + u_t^\top Q u_t) + \beta^T x_T^\top R_f x_T\right\} \tag{6.36}$$

subject to (6.33) and initial state x_0. Here

- $\beta \in (0, 1]$ is the time discount factor and
- $x^\top R_f x$ gives terminal loss at state x, where R_f is positive semidefinite.

We allow $\beta = 1$ to include the undiscounted case. If x_0 is random then we require it to be independent of $\xi_1, \ldots, \xi_T$.

To solve the finite horizon LQ problem we use backwards induction. Let $J_T(x) := x^\top R_f x$ and consider the problem of the controller in $T-1$. The controller takes x_{T-1} as given—since it can't be changed at this point—and trades off current and final losses by solving

$$\min_u \{x_{T-1}^\top R x_{T-1} + u^\top Q u + \beta \, \mathbb{E} \, J_T(A x_{T-1} + B u + C \xi_T)\} \tag{6.37}$$

Set

$$J_{T-1}(x) := \min_u \{x^\top R x + u^\top Q u + \beta \, \mathbb{E} J_T(A x + B u + C \xi_T)\} \tag{6.38}$$

Stepping back to time $T-2$, the function J_{T-1} now plays a role analogous to that played by the terminal loss $J_T(x) = x^\top R_f x$ at $T-1$, in the sense that $J_{T-1}(x)$ summarizes the future loss associated with moving to state x. Once again, the controller chooses u to trade off current loss against future loss, solving

$$\min_u \{x_{T-2}^\top R x_{T-2} + u^\top Q u + \beta \, \mathbb{E} J_{T-1}(A x_{T-2} + B u + C \xi_{T-1})\} \tag{6.39}$$

Letting

$$J_{T-2}(x) = \min_u \{x^\top R x + u^\top Q u + \beta \, \mathbb{E} J_{T-1}(A x + B u + C \xi_{T-1})\} \tag{6.40}$$

the pattern for backwards induction is now clear. We calculate the **cost-to-go functions** $\{J_t\}$ recursively via

- $J_T(x) = x^\top R_f x$ and
- $J_{t-1}(x) = \min_u \{x^\top R x + u^\top Q u + \beta \, \mathbb{E} J_t(A x + B u + C \xi_t)\}$ for $t = T, T-1, \ldots, 1$.

The function J_t represents the total cost-to-go from time t when the controller behaves optimally. The equations recursively defining (J_t) correspond to the Bellman equations from dynamic programming, specialized to the LQ problem.

The next lemma helps us understand and compute (J_t).

Lemma 6.2.1. *Each J_t has the form*

$$J_t(x) = x^\top P_t x + d_t \quad \textit{where} \quad P_t \in \mathbb{M}^{n\times n} \textit{ and } d_t \in \mathbb{R}.$$

Proof. This is true for $t = T$ with $P_T := R_f$ and $d_T = 0$. Suppose now that it is true at some $t \leqslant T$. We then have, for arbitrary $x \in \mathbb{R}^n$,

$$J_{t-1}(x) = \min_u \{x^\top R x + u^\top Q u + \beta\, \mathbb{E}(Ax + Bu + C\xi_t)^\top P_t (Ax + Bu + C\xi_t) + \beta d_t\}.$$

To obtain the minimizer, we use Lemma **??** on page **??**, which gives

$$u = -(Q + \beta B^\top P_t B)^{-1} \beta B^\top P_t A x. \tag{6.41}$$

Plugging this back into our objective function and rearranging yields

$$J_{t-1}(x) = x^\top P_{t-1} x + d_{t-1}$$

where

$$P_{t-1} = R - \beta^2 A^\top P_t B (Q + \beta B^\top P_t B)^{-1} B^\top P_t A + \beta A^\top P_t A \tag{6.42}$$

and

$$d_{t-1} = \beta(d_t + \mathrm{trace}(C^\top P_t C)). \tag{6.43}$$

□

Exercise 6.2.5. Verify the details of these calculations.

With Lemma 6.2.1 in hand we can compute the cost-to-go functions via (P_t, d_t), as shown in Algorithm 8.

With (P_t, d_t) in hand, we can proceed forward from x_0: At each t, we choose the minimizing control given this pair, which, recalling (6.41), takes the form

$$u_t = -F_t x_t \quad \text{where} \quad F_t := (Q + \beta B^\top P_{t+1} B)^{-1} \beta B^\top P_{t+1} A. \tag{6.44}$$

Then the state updates and we repeat. The resulting sequence of controls solves our finite horizon LQ problem.

Algorithm 8: Computing the cost-to-go functions in finite horizon LQ

set $t = T$;
set $P_t = R_f$;
set $d_t = 0$;
while $t > 0$ **do**
  set $P_{t-1} = R - \beta^2 A^\top P_t B (Q + \beta B^\top P_t B)^{-1} B^\top P_t A + \beta A^\top P_t A$;
  set $d_{t-1} = \beta(d_t + \mathrm{trace}(C^\top P_t C))$;
  set $t = t - 1$
end
return $(P_t, d_t)_{t=0}^T$

Rephrasing this more concisely,

$$x_{t+1} = (A - BF_t)x_t + C\xi_{t+1} \quad \text{and} \quad u_t = -F_t x_t \tag{6.45}$$

for $t = 0, \ldots, T-1$ attains the minimum of (6.36) subject to our constraints.

Emphasize that the solution is a policy—in this case linear.

6.2.3.2 A Life Cycle Problem

Compare outcomes to data.

Early Keynesians assumed that households have a constant marginal propensity to consume from current income but data contradicts this. In response, Milton Friedman and Franco Modigliani and others built models based on a consumer's preference for an intertemporally smooth consumption stream (see, e.g., Friedman (1956) or Modigliani et al. (1954)).

To illustrate the key ideas, consider the wealth dynamics given in (6.34), which we saw can be expressed as

$$x_{t+1} = Ax_t + Bu_t + C\xi_{t+1} \quad \text{with} \;\; x_t = \begin{pmatrix} w_t \\ 1 \end{pmatrix} \text{ and } u_t = c_t - \bar{c},$$

where

$$A := \begin{pmatrix} 1+r & -(1+r)\bar{c} + \mu \\ 0 & 1 \end{pmatrix}, \quad B := \begin{pmatrix} -(1+r) \\ 0 \end{pmatrix} \quad \text{and} \;\; C := \begin{pmatrix} \sigma \\ 0 \end{pmatrix}.$$

We assume in what follows that $(\xi_t) \overset{\text{IID}}{\sim} N(0,1)$.

To convert this into a finite horizon LQ problem we set the objective to

$$\mathbb{E}\left\{\sum_{t=0}^{T-1}\beta^t(c_t-\bar{c})^2+\beta^T q w_T^2\right\}. \tag{6.46}$$

Here q is a large positive constant that induces the consumer to target zero debt at T. (Without such a constraint, the optimal choice is to choose $c_t = \bar{c}$ in each period, letting assets adjust accordingly.)

To match with this state and control, the objective function (6.46) can be written in quadratic form by setting

$$Q := 1, \quad R := \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}, \quad \text{and} \quad R_f := \begin{pmatrix} q & 0 \\ 0 & 0 \end{pmatrix}$$

Now that we have A, B, C, Q, R and R_f, we can either calculate the cost-to-go functions and optimal controls directly or use existing code such as that found in the `QuantEcon` libraries.

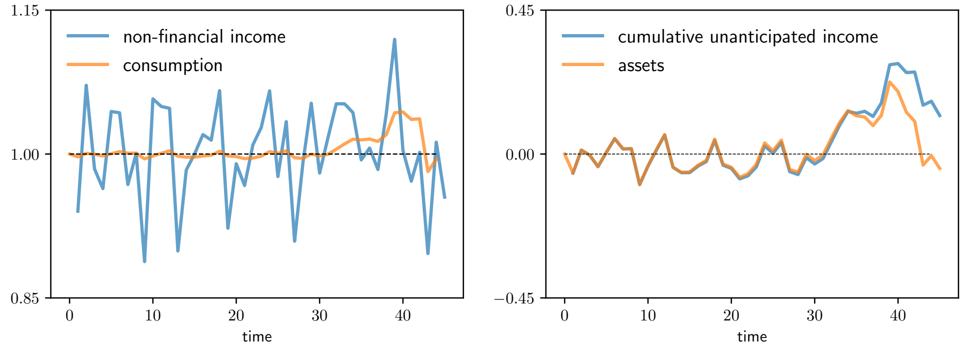
Figure 6.6 gives an illustration of the dynamics via simulation once the optimal controls have been obtained. The baseline parameterization, shown in the top subfigure, is

$$r = 0.04,\ \beta = \frac{1}{1+r},\ \bar{c} = 2,\ \mu = 1,\ \sigma = 0.25,\ T = 45 \text{ and } q = 10^6.$$

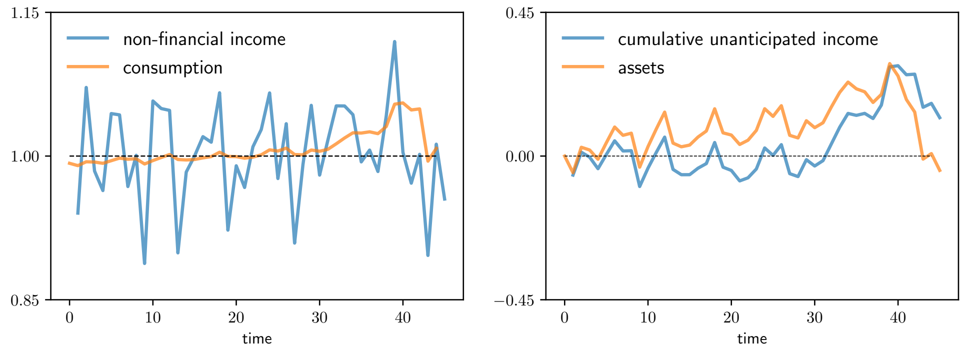
The top left panel shows a simulated time path for consumption c_t and income y_t. As anticipated, the time path of consumption is smoother than that for income, although it becomes more irregular towards the end of the agent's life, when the zero final asset requirement impinges more on consumption choices.

The top right panel shows that the time path of assets w_t is closely correlated with cumulative *unanticipated* income $\sum_{j=0}^{t}\sigma\xi_j$. Hence, unanticipated windfall gains are saved rather than consumed, while unanticipated negative shocks are met by reducing assets. (Again, this relationship breaks down towards the end of life due to the zero final asset requirement)

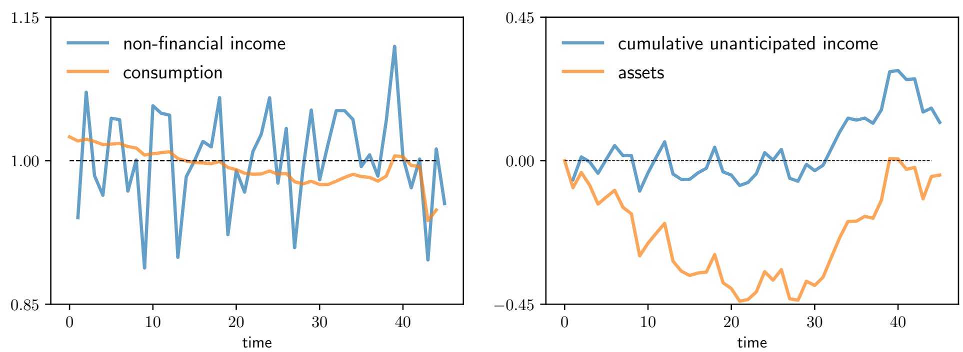
Subfigures (b) and (c) from Figure 6.6 show the same scenario after varying the subjective discount rate β. In (b) the agent is more patient than the market discount rate $1/(1+r)$. Hence she prefers to build up assets and weight consumption towards then end of her life. In (c) the situation is reversed, with $\beta < 1/(1+r)$. Now the agent is impatient relative to the market rate of return and hence prefers to consume early by accruing debt and then repay later.



(a) $\beta = 1/(1+r)$



(b) $\beta > 1/(1+r)$



(c) $\beta < 1/(1+r)$

Figure 6.6: Symmetric and asymmetric networks

6.2.4 Infinite Horizon Problems

Next we shift to an infinite horizon. This setting presents technical challenges because we cannot use backward induction. At the same time, infinite horizon problems are simple in the sense that, in time homogeneous cases, optimal policies are time invariant. The current date does not matter because, at any given point in time, the agent still faces an infinite future.

6.2.4.1 Objective

We maintain the previous dynamics $x_{t+1} = Ax_t + Bu_t + C\xi_{t+1}$ while modifying the objective to

$$\mathbb{E}\left\{\sum_{t=0}^{\infty} \beta^t (x_t^\top R x_t + u_t^\top Q u_t)\right\}. \tag{6.47}$$

Insert stabilizing S such that $r(A + SB) < 1$. Solve. What does the objective look like. Prove finite.

6.2.4.2 Solution

In the infinite horizon case, the value function is time invariant. For LQ problems, this translates to the statement that P_t and d_t discussed in Lemma 6.2.1 are constant. The stationary matrix P is, when it exists, the solution to the discrete time algebraic **Riccati equation**

$$P = R - (\beta B^\top PA)^\top (Q + \beta B^\top PB)^{-1} (\beta B^\top PA) + \beta A^\top PA. \tag{6.48}$$

Equation (6.48), can be understood as the stationary version of (6.42) and is also called the **LQ Bellman equation**. The stationary optimal policy for this model is

$$u = -Fx \quad \text{where} \quad F = (Q + \beta B^\top PB)^{-1} (\beta B^\top PA) \tag{6.49}$$

Equation 6.49 follows from exactly the same reasoning that led us to the finite horizon version F_t in (6.44).

Notice how the time dependent policy sequence (F_t) in (6.45) is replaced by a fixed matrix F from (6.49).

The sequence (d_t) from (6.43) is replaced by the constant value

$$d := \operatorname{trace}(C^\top PC) \frac{\beta}{1 - \beta} \tag{6.50}$$

The state evolves according to the time-homogeneous process

$$x_{t+1} = (A - BF)x_t + C\xi_{t+1} \tag{6.51}$$

The only significant computational difficulty is solving the Riccati equation (6.48). Remaining objects such as F and d are easily calculated once we have P in hand. The following result addresses computation of P. Also, let $\mathscr{R}$ be the self-mapping on $\mathbb{M}^{n\times n}$ defined by

$$\mathscr{R}(P) := R - (\beta B^\top PA)^\top (Q + \beta B^\top PB)^{-1}(\beta B^\top PA) + \beta A^\top PA$$

and let $\mathscr{M}_P$ be the set of positive definite matrices in $\mathbb{M}^{n\times n}$.

Theorem 6.2.2. *If (A, B) is stabilizable and (A, R) is observable, then*

(i) *$(\mathscr{M}_P, \mathscr{R})$ is globally stable with unique fixed point $P^* \in \mathscr{M}_P$ and*

(ii) *The policy*

$$u = -F^* x \;\; \text{where} \;\; F^* := (Q + \beta B^\top P^* B)^{-1}(\beta B^\top P^* A)$$

is the unique optimal policy for infinite horizon control problem.

Skill proof but give references, correct. How to handle the case where stabilizability is not global (e.g., one of the states is a constant, as in the investment problem below)? Or, alternatively, how to change investment problem state so that we don't have this issue? Also, perhaps go through the details of the scalar case, as an exercise?

Linear quadratic control problems of the class discussed above have a special property called **certainty equivalence**, which means that the optimal policy F is not affected by the parameters in C, which specify the shock process. This can be confirmed by inspecting (6.49). In other words, we can ignore uncertainty when solving for optimal behavior, and plug it back in when examining optimal state dynamics.

6.2.5 Investment with Adjustment Costs

For now you can see the output of our calculations in Figures 6.7–6.8, each of which shows a time path for both $\bar{q}_t$ and optimal output q_t. In the second figure, γ is increased by a factor of 5 and the time series for output is significantly smoother.

As well as converting this into a minimization problem, our challenge is to set up the state and control so that

(i) the current payoff can be expressed in the quadratic form (6.35)

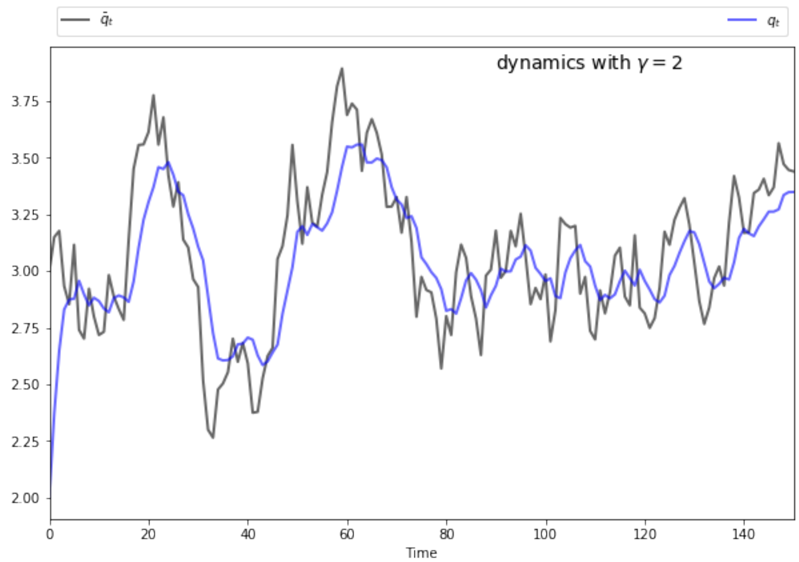


Figure 6.7: Output with adjustment costs when $\gamma = 2$

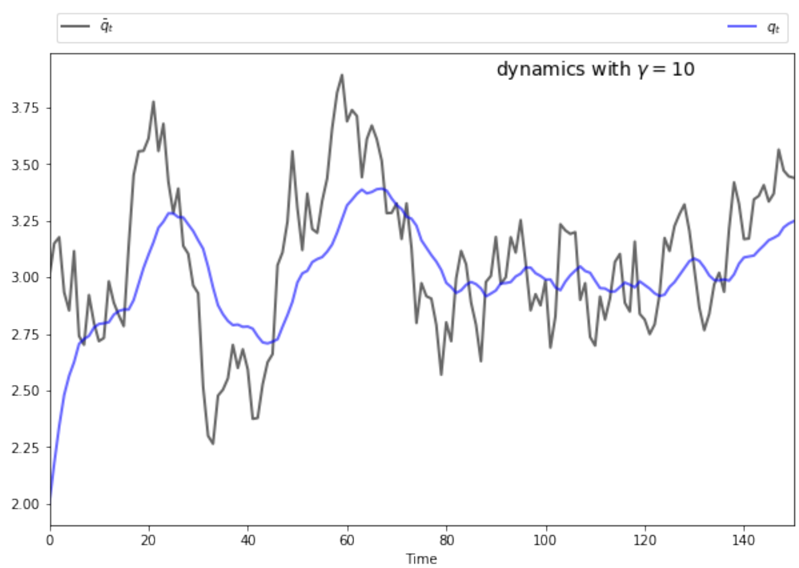


Figure 6.8: Output with adjustment costs when $\gamma = 10$

(ii) the state and control obey the linear dynamics (6.33).

As a first step, let us modify the rewards of the firm to

$$\mathbb{E} \sum_{t=0}^{\infty} \beta^t (\pi_t - a_1 \bar{q}_t^2) \quad \text{where } \bar{q}_t := \frac{a_0 - c + z_t}{2a_1} \tag{6.52}$$

While such a modification alters lifetime value, the optimal production sequence (q_t) will be identical, since

$$\mathbb{E} \sum_{t=0}^{\infty} \beta^t (\pi_t - a_1 \bar{q}_t^2) = \mathbb{E} \sum_{t=0}^{\infty} \beta^t \pi_t - a_1 \mathbb{E} \sum_{t=0}^{\infty} \beta^t \bar{q}_t^2$$

and the second term on the right does not depend on (q_t). Moreover,

$$u_t := q_{t+1} - q_t \qquad \Longrightarrow \qquad \pi_t - a_1 \bar{q}_t^2 = -a_1 (q_t - \bar{q}_t)^2 - \gamma u_t^2,$$

which is already quadratic. Finally, switching to a minimization problem requires us to multiply by -1, so the current loss is

$$\ell_t := a_1 (q_t - \bar{q}_t)^2 + \gamma u_t^2 \tag{6.53}$$

It remains to set up dynamics as linear in state and control, in order to fit with the canonical model (6.33). To this end we take $x_t = (\bar{q}_t, q_t, 1)^\top$ as our state. After setting $m_0 := (a_0 - c)/2a_1$ and $m_1 := 1/2a_1$, we can write $\bar{q}_t = m_0 + m_1 z_t$, and then, with some manipulation

$$\bar{q}_{t+1} = m_0(1 - \rho) + \rho \bar{q}_t + m_1 \sigma \xi_{t+1} \tag{6.54}$$

By our definition of u_t, the dynamics of q_t are $q_{t+1} = q_t + u_t$.

With these observations we can write the dynamic component of the LQ system as $x_{t+1} = Ax_t + Bu_t + C\xi_{t+1}$ when

$$A = \begin{pmatrix} \rho & 0 & m_0(1-\rho) \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \quad \text{and} \quad C = \begin{pmatrix} m_1 \sigma \\ 0 \\ 0 \end{pmatrix}$$

EXERCISE 6.2.6. Complete the LQ specification of the adjustment cost model by expressing (6.53) in the form of (6.35) by suitable choice of R and Q.

6.3 Chapter Notes

To be added.

Sargent (1987) provides a detailed discussion of the relationship between eigenvalues and oscillations in discrete time models.

Part I

Appendices

Chapter 7

Appendix I: Remaining Proofs

blueAdd at end or omit.

Chapter 8

Appendix II: Solutions

Solution to Exercise 1.2.5. Let T and S be as stated in the exercise. Regarding uniqueness, suppose that T has two distinct fixed points x and y in S. Since $T^m x = \bar{x}$ and $T^m y = \bar{x}$, we have $T^m x = T^m y$. But x and y are distinct fixed points, so $x = T^m x$ must be distinct from $y = T^m y$. Contradiction.

Regarding the claim that $\bar{x}$ is a fixed point, we recall that $T^k x = \bar{x}$ for $k \geqslant m$. Hence $T^m \bar{x} = \bar{x}$ and $T^{m+1}\bar{x} = \bar{x}$. But then

$$T\bar{x} = TT^m\bar{x} = T^{m+1}\bar{x} = \bar{x},$$

so $\bar{x}$ is a fixed point of T.

Solution to Exercise 1.2.6. Assume the hypotheses of the exercise and let $u_m := T^m u$ for all $m \in \mathbb{N}$. By continuity and $u_m \to u^*$ we have $Tu_m \to Tu^*$. But the sequence (Tu_m) is just (u_m) with the first element omitted, so, given that $u_m \to u^*$, we must have $Tu_m \to u^*$. Since limits are unique, it follows that $u^* = Tu^*$.

Solution to Exercise 1.2.8. Let the stated hypotheses hold and fix $u \in C$. By global stability we have $T^k u \to u^*$. Since T is invariant on C we have $(T^k u)_{k \in \mathbb{N}} \subset C$. Since C is closed, this implies that the limit is in C. In other words, $u^* \in C$, as claimed.

Solution to Exercise 1.2.12. For $\alpha > 0$ we always have $\|\alpha u\|_0 = \|u\|_0$, which violates positive homogeneity.

Solution to Exercise 1.2.17. By the definition of the operator norm we have $\|Au\| \leqslant \|A\|_o \|u\|$ for all $u \in \mathbb{R}^n$. If $\|A\|_o < 1$, then T is a contraction of modulus $\|A\|_o$, since, for any $x, y \in U$,

$$\|Ax + b - Ay - b\| = \|A(x - y)\| \leqslant \|A\|_o \|x - y\|.$$

Solution to Exercise 1.2.18. By the definition of the derivative, for any $x \in U := (0, \infty)$, we have

$$\lim_{y \to x} \left| \frac{g(y) - g(x)}{y - x} - g'(x) \right| = 0.$$

Hence, by the reverse triangle inequality, for fixed $\varepsilon > 0$, we can take a $\delta > 0$ such that

$$\left| \frac{g(y) - g(x)}{y - x} \right| > |g'(x)| - \varepsilon = g'(x) - \varepsilon$$

for all $y \in (x - \delta, x + \delta)$. Rearranging gives

$$|g(x) - g(y)| > [g'(x) - \varepsilon]|x - y|$$

for all $y \in (x - \delta, x + \delta)$. But $g'(x) = s\alpha x^{\alpha - 1} + 1 - \delta$, which converges to $+\infty$ as $x \to 0$. It follows that, for any $\lambda \in [0, 1)$, we can find a pair x, y such that $|g(x) - g(y)| > \lambda |x - y|$. Hence g is not a contraction map under $|\cdot|$.

Solution to Exercise 1.2.20. From the bound in Exercise 1.2.19, we obtain

$$\|u_m - u_k\| \leqslant \frac{\lambda^m}{1 - \lambda} \lambda^i \|u_0 - u_1\| \qquad (m, k \in \mathbb{N} \text{ with } m < k).$$

Hence (u_m) is Cauchy, as claimed.

Solution to Exercise 1.3.3. Let $M = \{1, 2\}$, let $A = \{1\}$ and let $B = \{2\}$. Then $A \subset B$ and $B \subset A$ both fail. Hence $\subset$ is not a total order on $\wp(M)$.

Solution to Exercise 1.3.6. Fix $a, b \in \mathbb{R}_+$ and $c \in \mathbb{R}$. By (1.21), we have

$$a \wedge c = (a - b + b) \wedge c \leqslant (|a - b| + b) \wedge c \leqslant |a - b| \wedge +b \wedge c.$$

Thus, $a \wedge c - b \wedge c \leqslant |a - b| \wedge c$. Reversing the roles of a and b gives $b \wedge c - a \wedge c \leqslant |a - b| \wedge c$. This proves the claim in Exercise 1.3.6.

Solution to Exercise 1.3.7. Fix $B \in \mathbb{M}^{m \times k}$ with $b_{ij} \geqslant 0$ for all i, j. Pick any $i \in [m]$ and $x \in \mathbb{R}^k$. By the triangle inequality, we have $|\sum_j b_{ij} x_j| \leqslant \sum_j b_{ij} |x_j|$. Stacking these inequalities yields $|Bx| \leqslant B|x|$, as was to be shown.

Solution to Exercise 1.3.9. Let T_1, T_2 be contraction maps on U of modulus λ_1 and λ_2 respectively. Fix $u, v \in U$. We have

$$\|Tu - Tv\|_\infty = \|(T_1 u) \vee (T_2 u) - (T_1 v) \vee (T_2 v)\|_\infty = \max_i |(\max_j (T_j u)_i - \max_j (T_j v)_i|,$$

where i ranges over $1, \ldots, n$ and where j ranges over $1, 2$. Applying Lemma 1.3.2 and reversing the order of maxima gives

$$\|Tu - Tv\|_\infty \leqslant \max_i \max_j |(T_j u)_i - (T_j v)_i|, = \max_j \max_i |(T_j u)_i - (T_j v)_i|.$$

From the definition of the supremum norm and our assumptions on T_1, T_2, this becomes

$$\|Tu - Tv\|_\infty \leqslant \max_j \|T_j u - T_j v\|_\infty, \leqslant \max_j \lambda_j \|u - v\|_\infty.$$

Hence T is a contraction of modulus $\lambda := \max_j \lambda_j$.

Solution to Exercise 1.4.3. Fix $\beta_1 \leqslant \beta_2$. Let g_1 and g_2 be the corresponding fixed point maps, as defined in (1.32). Since $\beta_1 \leqslant \beta_2$, we have $g_1(h) \leqslant g_2(h)$ for all $h \in \mathbb{R}_+$ and, in addition, g_2 is a contraction map (and hence globally stable), Proposition 1.3.3 applies. In particular, the fixed point h_1^* corresponding to β_1 is less than or equal to h_2^*, the fixed point corresponding to β_2.

Solution to Exercise 2.1.1. Fix an $n \times k$ matrix A with $A \geqslant 0$, along with $x, y \in \mathbb{R}^k$. We need to show that $x \leqslant y$ implies $Ax \leqslant Ay$ for any conformable vectors x, y. This holds because if $x \leqslant y$ we have $y - x \geqslant 0$, so $A(y - x) \geqslant 0$. But then $Ay - Ax \geqslant 0$, or $Ax \leqslant Ay$.

Solution to Exercise 2.1.2. Fix square A, B with $0 \leqslant A \leqslant B$. It follows from the rules of matrix multiplication that, for arbitrary nonnegative square matrices E, F, G with $F \leqslant G$, we have $EF \leqslant EG$ and $FE \leqslant GE$. Hence, if $A^k \leqslant B^k$ for some $k \in \mathbb{N}$, then $A^{k+1} = AA^k \leqslant BA^k \leqslant BB^k = B^{k+1}$. Thus, by induction, $A^k \leqslant B^k$ for all $k \in \mathbb{N}$, which verifies the first claim. Regarding the second, it is clear that for nonnegative matrices E, F with $E \leqslant F$ we have $\|E\|_\infty \leqslant \|F\|_\infty$. Hence $\|A^k\|_\infty \leqslant \|B^k\|_\infty$ for all $k \in \mathbb{N}$. Raising both sides to the power $1/k$ and applying Gelfand's lemma verifies $r(A) \leqslant r(B)$.

Solution to Exercise 2.1.3. Let A be as stated and let e be the right eigenvector in (2.1). Since e is nonnegative and nonzero, and since eigenvectors are defined only up to constant multiples, we can and do assume that $\sum_j e_j = 1$. From $Ae = r(A)e$ we have $\sum_j a_{ij} e_j = r(A) e_i$ for all i. Summing with respect to i gives $\sum_j \mathrm{cs}_j(A) e_j = r(A)$.

Since the elements of e are nonnegative and sum to one, $r(A)$ is a weighted average of the column sums. Hence the second pair of bounds in Lemma 2.1.2 holds. The remaining proof is similar (use the left eigenvector).

Solution to Exercise 2.1.4. Let P and Q be as stated. Evidently $PQ \geqslant 0$. Moreover, $PQ\mathbb{1} = P\mathbb{1} = \mathbb{1}$, so PQ is stochastic. That $r(P) = 1$ follows directly from Lemma 2.1.2. By the Perron–Frobenius theorem, there exists a nonzero, nonnegative row vector φ satisfying $\varphi P = \varphi$. Rescaling φ to $\varphi/(\varphi\mathbb{1})$ gives the desired vector ψ.

The final positivity and uniqueness claim is also by the Perron–Frobenius theorem, and its consequences for irreducible matrices. Indeed, if φ is another nonnegative vector satisfying $\varphi\mathbb{1} = 1$ and $\varphi P = \varphi$, then, by the Perron–Frobenius theorem, $\varphi = \alpha\psi$ for some $\alpha > 0$. But then $\alpha\psi\mathbb{1} = 1$ and $\psi\mathbb{1} = 1$, which gives $\alpha = 1$. Hence $\varphi = \psi$.

Solution to Exercise 2.1.5. Let $X_t = x \in S$, so that $X_{t+1} = \max\{x - D_{t+1}, 0\} + S\mathbb{1}\{x \leqslant s\}$. Evidently X_{t+1} is integer-valued and nonnegative. If $x \leqslant s$, then $X_{t+1} \leqslant \max\{s - D_{t+1}, 0\} + S \leqslant s + S$. Similarly, if $s < x \leqslant S + s$, then $X_{t+1} \leqslant \max\{x - D_{t+1}, 0\} \leqslant S + s$. The claim is verified.

Solution to Exercise 2.1.6. Let $x \in \mathsf{X}$ be the current state at time t and suppose first that $s < x$. The next period state X_{t+1} hits s with positive probability, since $\varphi(d) > 0$ for all $d \in \mathbb{Z}_+$. The state X_{t+2} hits $S + s$ with positive probability, since $\varphi(0) > 0$. From $S + s$, the inventory level reaches any point in $\mathsf{X} = \{0, \ldots, S + s\}$ in one step with positive probability. Hence, from current state x, inventory reaches any other state y with positive probability in three steps.

The logic for the case $x \leqslant s$ is similar and left to the reader.

Solution to Exercise 2.1.7. Fix $t \in \mathbb{N}$. Under the stated hypotheses, we have $X_t \stackrel{d}{=} \psi_0 P^t$ (see (2.9)). Hence

$$\mathbb{E}h(X_t) = \sum_{x'} h(x')\mathbb{P}\{X_t = x'\} = \sum_{x'} h(x')(\psi P^t)(x') = \left\langle h, \psi_0 P^t \right\rangle .$$

Solution to Exercise 2.1.13. Since we are conditioning on $X_t = x$, we can replace X_{t+1} with $\rho x + \varepsilon_{t+1}$. The result then follows from $\mathbb{P}\{\alpha < \varepsilon_{t+1} \leqslant \beta\} = F(\beta) - F(\alpha)$.

Solution to Exercise 2.1.16. Let P and ε have the stated properties. Suppose to the contrary that there is a $h \in \mathbb{R}^{\mathsf{X}}$ with $Ph \geqslant h + \varepsilon := Ph \geqslant h + \varepsilon\mathbb{1}_{\mathsf{X}}$. Since P is nonnegative, it is order preserving (cf. Exercise 2.1.2 on page 48), so $P^2 h \geqslant Ph + P\varepsilon =$

$Ph + \varepsilon \geqslant h + 2\varepsilon$. Continuing in this way yields $P^n h \geqslant h + n\varepsilon$ for all $n \in \mathbb{N}$. But P^n is a Markov matrix, so, by Exercise 2.1.15, $P^n h$ is bounded. Contradiction.

Solution to Exercise 2.1.17. Set $\alpha_k := u(x_k)$ for all k and $s_k := \alpha_k - \alpha_{k-1}$ with $\alpha_0 := 0$. Fix $x_j \in \mathsf{X}$. Then

$$\sum_{k=1}^{n} s_k \mathbb{1}\{x_j \geqslant x_k\} = \sum_{k=1}^{j} s_k = (\alpha_1 - \alpha_0) + (\alpha_2 - \alpha_1) + \ldots + (\alpha_j - \alpha_{j-1}) = \alpha_j.$$

In other words, $\sum_{k=1}^{n} s_k \mathbb{1}\{x_j \geqslant x_k\} = u(x_j)$. This completes the proofs.

Solution to Exercise 2.1.18. Fix $\varphi, \psi \in \mathsf{X}$ and suppose that $\varphi \preceq_{\mathrm{F}} \psi$. Let $u \in \mathbb{R}^{\mathsf{X}}$ be defined by $u(1) = 0$ and $u(2) = 1$. Then, by the definition of stochastic dominance, we have $\varphi(2) \leqslant \psi(2)$. Since $\varphi(1) = 1 - \varphi(2)$ and $\psi(1) = 1 - \psi(2)$, this inequality is equivalent to $\psi(1) \leqslant \varphi(1)$. Finally, suppose that $\psi(1) \leqslant \varphi(1)$ and fix $u \in i\mathbb{R}^{\mathsf{X}}$. Let $h = u(2) - u(1) \geqslant 0$. Then

$$\sum_x u(x)\varphi(x) = u(1)\varphi(1) + (u(1) + h)(1 - \varphi(1)) = u(1) + h(1 - \varphi(1)).$$

Similarly, $\sum_x u(x)\psi(x) = u(1) + h(1 - \psi(1))$. Since $h \geqslant 0$ and $\psi(1) \leqslant \varphi(1)$, we have $\sum_x u(x)\varphi(x) \leqslant \sum_x u(x)\psi(x)$. Thus, $\varphi \preceq_{\mathrm{F}} \psi$. This chain of implications proves the equivalences in the exercise.

Solution to Exercise 2.1.20. Using Exercise 2.1.13 and the definition of P, it can be shown that

$$G(x, x_k) := \sum_{k=j}^{n} P(x, x_j) = \mathbb{P}\{x_k - s/2 < X_{t+1} \,|\, X_t = x\}.$$

Rewriting the probability in terms of ε_{t+1}, we get

$$G(x, x_k) = \mathbb{P}\{\varepsilon_{t+1} > (x_k - s/2 - \rho x)/\sigma\}.$$

Since $\rho \geqslant 0$, we can now see that $x \leqslant y$ implies $G(x, x_k) \leqslant G(y, x_k)$ for all k, or, equivalently, $G(x, \cdot) \leqslant G(y, \cdot)$ pointwise on X. By Lemma 2.1.6, this is equivalent to the statement that $P(x, \cdot) \preceq_{\mathrm{F}} P(y, \cdot)$, which confirms that P is monotone increasing.

Solution to Exercise 2.1.21. This matrix P_w is monotone increasing if and only if $(1 - \alpha, \alpha) \preceq_{\mathrm{F}} (\beta, 1 - \beta)$. From Exercise 2.1.18, we know that this is equivalent to $\beta \leqslant 1 - \alpha$, or $\beta + \alpha \leqslant 1$.

Solution to Exercise 2.1.23. Clearly this is true for $t = 1$. Suppose it is also true for arbitrary t. Then, for any $h \in i\mathbb{R}^{\mathsf{X}}$, the function $P^t h$ is again in $i\mathbb{R}^{\mathsf{X}}$. From this it follow that $P^{t+1}h$ is also in $i\mathbb{R}^{\mathsf{X}}$, since P is monotone increasing. This proves that P^{t+1} is invariant on $i\mathbb{R}^{\mathsf{X}}$, and therefore monotone increasing.

Solution to Exercise 2.2.1. Let X be partially ordered and let π and P have the stated conditions. By Exercise 2.1.23, P^t is monotone increasing for all t. By this fact and the assumption $\pi \in i\mathbb{R}^{\mathsf{X}}$, we see that $P^t \pi \in i\mathbb{R}^{\mathsf{X}}$. Hence $v = \sum_{t \geqslant 0} \beta^t P^t \pi$ is also increasing.

Solution to Exercise 2.2.2. Proposition 2.2.4 follows directly from Theorem 2.2.2 when $B_t = b(X_{t-1}, X_t) = \beta(X_{t-1})$ and $h = \pi$.

Solution to Exercise 2.3.3. Under a cum-dividend contract, purchasing at t and selling at $t+1$ pays $D_t + \Pi_{t+1}$. Hence, applying the fundamental asset pricing equation, the time t price Π_t of the contract must satisfy

$$\Pi_t = D_t + \mathbb{E}_t \, M_{t+1} \Pi_{t+1}. \tag{2.41}$$

Proceeding as for the ex-dividend contract, the price conditional on current state x is $\pi(x) = d(x) + \sum_{x'} m(x, x')\pi(x')P(x, x')$. In vector form, this is $\pi = d + A\pi$. Solving out for prices gives $\pi^* = (I - A)^{-1} d$.

Solution to Exercise 2.3.5. We seek a v that solves

$$v(x) = \sum_{x' \in \mathsf{X}} [1 + v(x')] \, A(x, x') \qquad (x, x' \in \mathsf{X}).$$

Treating A as a matrix and v as a column vector, this equation this becomes $v = A\mathbb{1} + Av$, where $\mathbb{1}$ is a column vector of ones. By the Neumann series lemma, $r(A) < 1$ implies that this equation has the unique solution $v^* = (I - A)^{-1} A\mathbb{1}$. By the same lemma, v^* has the alternative representation $v^* = \sum_{t \geqslant 0} A^t (A\mathbb{1}) = \sum_{t \geqslant 1} A^t \mathbb{1}$.

Solution to Exercise 2.4.3. The code in Listing 10 creates a Markov chain via Tauchen approximation of an AR(1) process with positive autocorrelation parameter. By Exercise 2.1.20, P is monotone increasing. Hence, by Lemma 2.4.1, the value function is increasing. Since $h^* = c + \beta P v^*$, it follows that h^* is increasing. Regarding intuition, positive autocorrelation in wages means that high current wages predict high future wages. It follows that the value of waiting for future wages rises with current wages.

Solution to Exercise 2.4.6. Let T be the operator on $\mathscr{V}$ such that $(Tv_u)(w)$ is the right-hand side of (2.54). To solve the exercise, it suffices to prove that T is a contraction map on $\mathscr{V}$. (Then v_u can be obtained, in the limit, by applying successive approximation to T and, once the approximate fixed point is computed, v_e can be obtained via (2.53).) To show that T is a contraction, we let T_1 and T_2 be the operators on $\mathscr{V}$ defined by

$$(T_1 v)(w) = \frac{1}{1 - \beta(1-\alpha)} (w + \alpha\beta (Pv)(w)) \quad \text{and} \quad (T_2 v)(w) = c + \beta \, (Pv)(w).$$

Since $Tv = (T_1 v) \wedge (T_2 v)$, Exercise 1.3.9 on page 34 tells us that T will be a contraction provided that T_1 and T_2 are both contraction maps. For the case of T_2, we have

$$\|T_1 f - T_1 g\|_\infty = \max_w |c + \beta \, (Pf)(w) - c - \beta \, (Pg)(w)| \leqslant \max_w \beta \sum_{w'} |f(w') - g(w')| P(w, w').$$

The last term is dominated by $\beta \|f - g\|_\infty$, so T_1 is a contraction. The proof for T_2 is similar in spirit and left to the reader.

Solution to Exercise 3.1.1. Pointwise on X we have $1 - \sigma \leqslant 1$, so $P_\sigma \leqslant P$. By Exercise 2.1.2, we then have $r(P_\sigma) \leqslant r(P) = 1$. Hence $r(K_\sigma) = \beta \, r(P_\sigma) \leqslant \beta < 1$.

Solution to Exercise 3.1.2. Fix $\sigma \in \Sigma$. If $f, g \in \mathbb{R}^{\mathsf{X}}$, $f \leqslant g$ and $x \in \mathsf{X}$, then

$$\begin{aligned} (T_\sigma g)(x) - (T_\sigma f)(x) &= (1 - \sigma(x))\beta \sum_{x' \in \mathsf{X}} g(x') P(x, x') - \beta \sum_{x' \in \mathsf{X}} f(x') P(x, x') \\ &= (1 - \sigma(x))\beta \sum_{x' \in \mathsf{X}} (g(x') - f(x')) P(x, x'). \end{aligned}$$

Since $g(x') \geqslant f(x')$ for all x' this expression is nonnegative. Hence $(T_\sigma g)(x) \geqslant (T_\sigma f)(x)$ for all x.

Solution to Exercise 3.1.3. Fix $\sigma \in \Sigma$. Given $f, g \in \mathbb{R}^{\mathsf{X}}$ and $x \in \mathsf{X}$, we have

$$\begin{aligned} |(T_\sigma f)(x) - (T_\sigma \, g)(x)| &= \left| (1 - \sigma(x))\beta \sum_{x' \in \mathsf{X}} (g(x') - f(x')) P(x, x') \right| \\ &\leqslant \beta \left| \sum_{x'} [f(x') - g(x')] P(x, x') \right|. \end{aligned}$$

Applying the triangle inequality and $\sum_{x' \in \mathsf{X}} P(x, x') = 1$, we obtain

$$|(T_\sigma f)(x) - (T_\sigma g)(x)| \leqslant \beta \sum_{x'} |f(x') - g(x')| P(x, x') \leqslant \beta \|f - g\|_\infty.$$

Taking the supremum over all x on the left hand side of this expression leads to

$$\|T_\sigma f - T_\sigma g\|_\infty \leqslant \beta \|f - g\|_\infty.$$

Since f, g were arbitrary elements of $\mathbb{R}^{\mathsf{X}}$, the contraction claim is proved.

Solution to Exercise 3.1.5. Take any f, g in $\mathbb{R}^{\mathsf{X}}$ and fix any $w \in \mathsf{X}$. The bound in (1.27) gives

$$\begin{aligned} |(Tf)(x) - (Tg)(x)| &\leqslant \left| c + \beta \sum_{x'} f(x') P(x, x') - \left(c(x) + \beta \sum_{x'} g(x') P(x, x') \right) \right| \\ &= \beta \left| \sum_{x'} [f(x') - g(x')] P(x, x') \right|. \end{aligned}$$

Applying the triangle inequality, we obtain

$$|(Tf)(x) - (Tg)(x)| \leqslant \beta \sum_{x'} |f(x') - g(x')| P(x, x') \leqslant \beta \|f - g\|_\infty.$$

Taking the supremum over all w on the left hand side of this expression leads to

$$\|Tf - Tg\|_\infty \leqslant \beta \|f - g\|_\infty.$$

Since f, g were arbitrary elements of $\mathbb{R}^{\mathsf{X}}$, the contraction claim is verified.

Solution to Exercise 3.1.7. Fix $x, x' \in \mathsf{X}$ with $x \leqslant x'$. Since σ^* is binary, to show σ^* is decreasing it suffices to show that $\sigma^*(x) = 0$ implies $\sigma^*(x') = 0$. Hence we suppose that $\sigma^*(x) = 0$. This in turn implies that $e(x) < h^*(x)$. As $x \leqslant x'$, e is decreasing and h^* is increasing on X, we have $e(x') < h^*(x')$. Hence $\sigma^*(x') = 0$. We conclude that σ^* is decreasing on X, as claimed.

Solution to Exercise 3.1.8. The solution to Exercise 3.1.8 is similar to that of Exercise 3.1.7 and hence omitted.

Solution to Exercise 3.2.1. In view of (3.14), the continuation value opertor for this problem is

$$(Ch)(x) = -c + \beta \sum_{x'} \max\{\pi(x'), h(x')\} P(x, x') \qquad (x \in \mathsf{X}).$$

The monotonicity result for h^* follows from Exercise 3.1.9 on page 3.1.9.

Solution to Exercise 3.2.2. If (X_t) is IID with common distribution φ, then the continuation value h^* is constant; in particular, it is the unique solution to

$$h = -c + \beta \sum_{x'} \max\{\pi(x'), h(x')\} \varphi(x').$$

Since constant functions are (weakly) decreasing, Exercise 3.1.8 applies and σ^* is increasing. Intuitively, the value of waiting is independent of the current state, while the value of bringing the product to market is increasing in the current state. Hence, if the firm brings to the product to market in state x, then it should also do so at any $x' \geqslant x$.

Solution to Exercise 4.1.1. For each $v \in \mathbb{R}^{\mathsf{X}}$, a v-greedy policy exists: simply select a point a_x^* from the nonempty set on the right hand side of (4.33) at every x in X. By (iii), the resulting policy $\sigma(x) := a_x^*$ is optimal when $v = v^*$.

Solution to Exercise 4.1.2. Fix σ in Σ. First let us show that v_σ is a fixed point of T_σ: We have, in vector notation,

$$T_\sigma v_\sigma = r_\sigma + \beta P_\sigma \left(\sum_{t \geqslant 0} \beta^t P_\sigma^t r_\sigma \right) = r_\sigma + \left(\sum_{t \geqslant 1} \beta^t P_\sigma^t r_\sigma \right) = \sum_{t \geqslant 0} \beta^t P_\sigma^t r_\sigma$$

which is v_σ. (The passage of P_σ through the limit associated with the infinite sum is justified here because P_σ is a linear operator acting on a finite dimensional space, and therefore continuous.)

Moreover, T_σ is a contraction of modulus β on $\mathbb{R}^{\mathsf{X}}$. Indeed, for any v, w in $\mathbb{R}^{\mathsf{X}}$ we have

$$\begin{aligned} |T_\sigma v(x) - T_\sigma w(x)| &= \beta \left| \sum_y P(x, \sigma(x), y) v(y) - \sum_y P(x, \sigma(x), y) w(y) \right| \\ &\leqslant \sum_y P(x, \sigma(x), y) \beta \, |v(y) - w(y)| \leqslant \beta \|v - w\|_\infty \end{aligned}$$

Taking the supremum over all $x \in \mathsf{X}$ yields the desired result. This establishes all claims in the lemma.

Solution to Exercise 4.2.1. The Bellman operator is

$$(Tv)(x) = \max_{a \in \Gamma(x)} \left\{ \sum_{d \geqslant 0} (x \wedge d)\varphi(d) - ca + \beta \sum_{d \geqslant 0} v(m(x-d) + Sa)\varphi(d) \right\} \tag{4.20}$$

This operator is a sup norm contraction mapping on $\mathbb{R}^{\mathsf{X}}$ because, in view of Lemma 1.3.2 on page 34, for any v, w in $\mathbb{R}^{\mathsf{X}}$,

$$\begin{aligned} |(Tv)(x)| - (Tw)(x)| &\leqslant \beta \max_{a \in \Gamma(x)} \left| \sum_{d \geqslant 0} \left[v(m(x-d) + Sa) - w(m(x-d) + Sa) \right] \varphi(d) \right| \\ &\leqslant \beta \max_{a \in \Gamma(x)} \sum_{d \geqslant 0} |v(m(x-d) + Sa) - w(m(x-d) + Sa)| \, \varphi(d) \end{aligned}$$

Since $\sum_{d \geqslant 0} \varphi(d) = 1$, it follows that, for arbitrary $x \in \mathsf{X}$,

$$|(Tv)(x)| - (Tw)(x)| \leqslant \beta \|v - w\|_\infty$$

Taking the supremum over all $x \in \mathsf{X}$ yields the desired result.

Solution to Exercise 4.2.2. The stochastic kernel is

$$P(x, a, y) = \begin{cases} 0 & \text{if } y < a \\ (1-p)^x & \text{if } y = a \\ (1-p)^{x+a-y} p & \text{if } y > a \end{cases} \tag{4.22}$$

The middle case is obtained by observing that the next period state hits y when $y = a$ if and only if $D_{t+1} \geqslant x$ and then using the expression for the PMF of the geometric distribution.

Solution to Exercise 4.3.1. Extending L to $\mathsf{X} \times \mathsf{X}$ via $L(x, x') = L((y, z), (y', z')) := L(z, z')$, we have

$$K_\sigma(x, x') = L(x, x') R(y, \sigma(y, z), y') \leqslant L(x, x'),$$

since $R(y, \sigma(y, z), y') \leqslant 1$ for all y, z, y'. The claim now follows from Exercise 2.1.2 on page 48.

Solution to Exercise 5.1.2. Both u and $\exp$ are increasing on X, so $h = u \circ c$ is in $i\mathbb{R}^{\mathsf{X}}$. Since $\rho \geqslant 0$, the stochastic matrix P is monotone increasing (see §2.1.4.2). Clearly βP shares this property. It follows that $\beta P h \in i\mathbb{R}^{\mathsf{X}}$. Applying βP again, we have $(\beta P)^2 h \in i\mathbb{R}^{\mathsf{X}}$. Continuing in this way, we see that $(\beta P)^k h$ is increasing for all k. Hence $\sum_{k \geqslant 0} (\beta P)^k h$ is increasing. By the Neumann series lemma, this sum is equal to v, so $v \in i\mathbb{R}^{\mathsf{X}}$.

Solution to Exercise 5.1.6. We already proved in Lemma 5.1.3 that $T\varphi \geqslant \varphi$. Also, for any $x \in \mathsf{X}$, we have $0 \in \Gamma(x)$, and hence

$$(T\psi)(x) = \min_{x' \in \Gamma(x)} \{\ell(x - x') + \beta \ell(x')\} \leqslant \ell(x) = \psi(x).$$

Thus, $T\psi \leqslant \psi$. Moreover, T is clearly order preserving on I, since for $f, g \in I$ with $f \leqslant g$, the definition of T gives $Tf \leqslant Tg$. Since $T\varphi \geqslant \varphi$ and $T\psi \leqslant \psi$, the order-preserving property implies that T is a self-map on I.

Solution to Exercise 5.1.8. Fix $f, g \in I$ and $\lambda \in [0, 1]$. For any $x \in \mathsf{X}$, we have

$$\begin{aligned}
(T(\lambda f + (1-\lambda) g))(x) &= \min_{x' \in \Gamma(x)} \{\ell(x - x') + \beta(\lambda f + (1-\lambda) g))(x')\} \\
&= \min_{x' \in \Gamma(x)} \{\lambda[\ell(x - x') + \beta f(x')] + (1-\lambda)[\ell(x - x') + \beta g(x')]\} \\
&\geqslant \lambda (Tf)(x) + (1-\lambda)(Tg)(x),
\end{aligned}$$

where the last step used Exercise 5.1.7. Since x was arbitrary, we have shown that T is concave.

Solution to Exercise 5.1.9. Let iI be the set of increasing functions in I. Because weak inequalities are preserved under limits, this set is closed in I. Moreover, T is invariant on iI (check this). Hence, by Exercise 1.2.8 on page 17, the fixed point is in iI.

Solution to Exercise 5.1.10. Suppose that the current state is x. The agent always has the option to do everything in one step, with loss $\ell(x)$. Hence the minimum loss $f^*(x)$, which includes this option, as well as the alternative of spreading effort over time, should be no larger than $\ell(x)$.

Solution to Exercise 5.1.11. To show that $\hat{T} = H \circ T \circ H^{-1}$ holds, we can equivalently prove that $\hat{T} \circ H = H \circ T$. For $x \in \mathbb{R}$, we have $HTx = \ln A + \alpha \ln x$ and $\hat{T} H x = \ln A + \alpha \ln x$. Hence $\hat{T} \circ H = H \circ T$, as was to be shown.

Solution to Exercise 5.1.12. Let (M, T) and $(\hat{M}, \hat{T})$ be topologically conjugate, with $\hat{T} \circ H = H \circ T$. The stated equivalence holds because

$$Tx = x \iff HTx = Hx \iff \hat{T}Hx = Hx.$$

Solution to Exercise 5.1.13. From $\hat{T} = H \circ T \circ H^{-1}$ we have $\hat{T}^2 = H \circ T \circ H^{-1} \circ H \circ T \circ H^{-1} = H \circ T^2 \circ H^{-1}$ and, continuing in the same way (or using induction), $\hat{T}^k = H \circ T^k \circ H^{-1}$ for all $k \in \mathbb{N}$. Equivalently, $\hat{T}^k \circ H = H \circ T^k$ for all $k \in \mathbb{N}$. Hence, using continuity of H and H^{-1},

$$T^k x \to x^* \iff HT^k x \to Hx^* \iff \hat{T}^k Hx \to Hx^*.$$

Solution to Exercise 5.2.2. Regarding the self-map property, fix $v \in \mathscr{V}$ and let σ be v-greedy. As T_σ is a self-map on $\mathscr{V}$, we have $T_\sigma v \in \mathscr{V}$. Since $Tv = T_\sigma v$, we conclude that $Tv \in \mathscr{V}$, as required.

To show that T is order-preserving, we apply monotonicity of B (see (5.10)) which yields $\max_{a \in \Gamma(x)} B(x, a, v) \leqslant \max_{a \in \Gamma(x)} B(x, a, w)$ for all $x \in \mathsf{X}$ whenever $v \leqslant w$.

Solution to Exercise 5.2.3. Here's a proof for T: The statement

$$(T^k v)(x) = \max_{a \in \Gamma(x)} B(x, a, T^{k-1} v) \tag{5.19}$$

is certainly true when $k = 0$ (and T^0 is the identity). Now suppose it is valid at $k - 1$. Then, since $(T^k v)(x) = (T(T^{k-1} v))(x)$ at any given x, we can apply the induction hypothesis to obtain (5.19) for all k. The proof for T_σ is very similar.

Solution to Exercise 5.2.6. Let $(\Gamma, \mathscr{V}, B)$ satisfy Blackwell's condition. Fix $v, w \in \mathscr{V}$ and $(x, a) \in \mathsf{G}$. Observe that $v = w + v - w \leqslant w + \|v - w\|_\infty$. By monotonicity of B and Blackwell's condition, we have

$$B(x, a, v) \leqslant B(x, a, w + \|v - w\|_\infty) \leqslant B(x, a, w) + \beta \|v - w\|_\infty.$$

As a result, $B(x, a, v) - B(x, a, w) \leqslant \beta \|v - w\|_\infty$. Reversing the roles of v and w yields

$$|B(x, a, v) - B(x, a, w)| \leqslant \beta \|v - w\|_\infty.$$

Since $\beta < 1$, the RDP is contracting.

Solution to Exercise 5.2.8. Let M be closed in $\mathbb{R}^n$, let T be a self-map on M and let T^k be a contraction. Let u^* be the unique fixed point of T^k. Fix $\varepsilon > 0$. We can choose n such that $\|T^{nk}Tu^* - u^*\| < \varepsilon$. Then

$$\|TT^{nk}u^* - u^*\| = \|Tu^* - u^*\| < \varepsilon.$$

Since ε was arbitrary we have $\|Tu^* - u^*\| = 0$, implying that u^* is a fixed point of T. The proof that $T^n u \to u^*$ for any u is left to the reader.

Solution to Exercise 5.3.2. Fix $v \in \mathscr{V}$ and all $(x, a) \in \mathsf{G}$. Since $v \geqslant v_1$, the definition of B implies that

$$B(x, a, v) \geqslant \{(\min r)^\rho + \beta(\min r)^\rho\}^{1/\rho} = \min r(1+\beta)^{1/\rho} \geqslant m_1.$$

At the same time,

$$B(x, a, v) \leqslant \left\{(\max r)^\rho + \beta m_2^\rho\right\}^{1/\rho} = ((1-\beta)m_2^\rho + \beta m_2^\rho)^{1/\rho} = m_2.$$

Solution to Exercise 5.3.3. Fix v is in $\mathscr{V}$. In view of Exercise 5.3.2, we have $m_1 \leqslant B(x, a, v) \leqslant m_2$ for all $(x, a) \in \mathsf{G}$. Indeed, if $v_1 \leqslant v \leqslant v_2$, then

$$(Tv)(x) = \max_{a \in \Gamma(x)} B(x, a, v) \leqslant m_2 = v_2(x).$$

Hence $Tv \leqslant v_2$ and, by a similar argument $Tv \geqslant v_1$. Thus, T is a self-map on $\mathscr{V}$.

Solution to Exercise 5.3.4. Pick any $w \in \mathscr{W}$. Since $w \leqslant w_1$ and $\eta < 0$, we have $w^{1/\eta} \geqslant w_1^{1/\eta}$. But then, since B is monotone,

$$B(x, a, w^{1/\eta}) \geqslant B(x, a, w_1^{1/\eta}) = B(x, a, v_1) \geqslant v_1(x)$$

for all $(x, a) \in \mathsf{G}$. Hence

$$(Uw_1)(x) = \min_{a \in \Gamma(x)} B(x, a, w^{1/\eta})^\eta \leqslant v_1(x)^\eta = w_1(x).$$

A similar argument shows that $(Uw_2)(x) \geqslant w_2(x)$ for all $x \in \mathsf{X}$.

Solution to Exercise 6.1.1. The vector x_t can also be written as $x_t = F^t x_0$, where $Fx := Ax + b$. In Exercise 1.2.7 you proved that F is globally stable on $\mathbb{R}^n$ whenever $r(A) < 1$. The unique fixed point is $x^* := (I - A)^{-1}b$, by the Neumann series lemma.

Solution to Exercise 6.1.3. We can reorganize (6.8) into a first order system by setting

$$X_t := \begin{pmatrix} p_t \\ p_{t-1} \end{pmatrix}, \quad A := \frac{1}{1+\beta} \begin{pmatrix} 2 & -1 \\ 1 & 0 \end{pmatrix}, \quad C := \begin{pmatrix} \beta/(1+\beta) \\ 0 \end{pmatrix} \quad \text{and} \quad \xi_t := m_t$$

Solution to Exercise 6.1.4. Computing the modulus of the two eigenvalues leads to $1/(1+\beta)$ in both cases. Hence $r(A) < 1$ whenever $\beta > 0$.

Solution to Exercise 6.1.6. If we set

$$X_t := \begin{pmatrix} Y_t \\ Y_{t-1} \end{pmatrix}, \quad A := \begin{pmatrix} \alpha_1 & \alpha_2 \\ 1 & 0 \end{pmatrix}, \quad C := \begin{pmatrix} \sigma \\ 0 \end{pmatrix} \quad \text{and} \quad \xi_t := \frac{1}{\sigma} \varepsilon_t, \tag{6.10}$$

then the first entry in the two dimensional system

$$X_{t+1} = AX_t + b + C\xi_{t+1}$$

coincides with (6.9).

Solution to Exercise 6.1.7. The eigenvalues of A solve $\det(A - \lambda I) = 0$. The two solutions are, in this case, the roots of the quadratic term $\lambda^2 - \alpha_1 \lambda - \alpha_2$, or

$$\lambda_i = \frac{\alpha_1 \pm \sqrt{\alpha_1^2 + 4\alpha_2}}{2} \qquad i = 1, 2 \tag{6.11}$$

Solution to Exercise 6.1.14. Observe that

$$\mathbb{E}[w_{t+1} \,|\, \mathscr{G}_t] = \mathbb{E}[w_t + \xi_{t+1} \,|\, \mathscr{G}_t] = \mathbb{E}[w_t \,|\, \mathscr{G}_t] + \mathbb{E}[\xi_{t+1} \,|\, \mathscr{G}_t]$$

But $\mathbb{E}[w_t \,|\, \mathscr{G}_t] = w_t$ because $w_t = \sum_{i=1}^t \xi_i$ is $\mathscr{G}_t$-measurable and $\mathbb{E}[\xi_{t+1} \,|\, \mathscr{G}_t] = \mathbb{E}[\xi_{t+1}] = 0$ by independence and the zero mean assumption on ξ_{t+1}. The martingale property now follows.

Solution to Exercise 6.1.16. By the law of iterated expectations, we have $\mathbb{E}[w_t] = \mathbb{E}[\mathbb{E}[w_t \,|\, \mathscr{G}_{t-1}]] = \mathbb{E}[0] = 0$.

Solution to Exercise 6.1.17. Supposing without loss of generality that $s < t$, we have

$$\mathbb{E}[w_s w_t'] = \mathbb{E}[\mathbb{E}[w_s w_t' \mid \mathcal{G}_{t-1}]] = \mathbb{E}[w_s \mathbb{E}[w_t' \mid \mathcal{G}_{t-1}]] = \mathbb{E}[0] = 0$$

Solution to Exercise 6.2.1. We have

$$\mathbb{E}_t[x_{t+1}^\top H x_{t+1}] = \mathbb{E}_t[(Ax_t + C\xi_{t+1})^\top H(Ax_t + C\xi_{t+1})].$$

The right hand side expands to

$$\mathbb{E}_t[x_t^\top A^\top HAx_t] + 2\mathbb{E}_t[x_t^\top A^\top HC\xi_{t+1}] + \mathbb{E}_t[\xi_{t+1}^\top C^\top HC\xi_{t+1}] = I + II + III.$$

Since x_t is known at t we have $I == x_t^\top A^\top HAx_t$. Since $\{\xi_t\}$ is an MDS,

$$II = 2\mathbb{E}_t[x_t^\top A^\top HC\xi_{t+1}] = 2x_t^\top A^\top HC\,\mathbb{E}_t[\xi_{t+1}] = 0.$$

Finally,

$$III = \mathbb{E}_t[\xi_{t+1}^\top C^\top HC\xi_{t+1}] = \operatorname{trace}(C^\top HC).$$

Combining these expressions verifies the claim in the exercise.

Solution to Exercise 6.2.2. Suppose P and δ have the stated properties. Let $\pi_t = X_t^\top PX_t + \delta$ for all t. Applying Exercise 6.2.1 yields

$$\begin{aligned}\pi_t X_t^\top PX_t + \delta &= \beta X_t^\top A^\top (D+P)AX_t + \beta \operatorname{trace}(C^\top (D+P)C) + \beta\delta \\ &= \beta\mathbb{E}_t[X_{t+1}^\top DX_{t+1} + X_{t+1}^\top PX_{t+1} + \delta]\end{aligned}$$

In the present setting, the last expression is $\beta\mathbb{E}_t[d_{t+1} + \pi_{t+1}]$, so the pricing equation is verified.

Solution to Exercise 6.2.3. By hypothesis, $P = \beta A^\top(D+P)A$, so $x^\top Px = \beta x^\top A^\top(D+P)Ax$. It follows that

$$x^\top Px + \delta = \beta x^\top A^\top (D+P)Ax + \delta.$$

To complete the proof, it suffices to show that $\delta = \beta \operatorname{trace}(C^\top(D+P)C) + \beta\delta$. But this is true by definition.

Solution to Exercise 6.2.4. Letting $M := \beta A^\top DA$ and $\Lambda := \sqrt{\beta}A^\top$, we can express (6.31) as $P = \Lambda P\Lambda^\top + M$. This is a discrete Lyapunov equation in P. Since $r(\Lambda) < 1$, a unique solution exists. The solution is positive semidefinite by Exercise **??** in §**??**.

Solution to Exercise 6.2.6. Set $Q = \gamma$ and

$$R = a_1 \begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Bibliography

Al-Najjar, N. I. and Shmaya, E. (2019). Recursive utility and parameter uncertainty. *Journal of Economic Theory*, 181:274–288.

Aliprantis, C. D. and Burkinshaw, O. (1998). *Principles of real analysis*. Academic Press, 3 edition.

Arellano, C. (2008). Default risk and income fluctuations in emerging economies. *American Economic Review*, 98(3):690–712.

Arrow, K. J., Harris, T., and Marschak, J. (1951). Optimal inventory policy. *Econometrica: Journal of the Econometric Society*, pages 250–272.

Balbus, Ł., Reffett, K., and Woźny, Ł. (2014). A constructive study of markov equilibria in stochastic games with strategic complementarities. *Journal of Economic Theory*, 150:815–840.

Bertsekas, D. (2012). *Dynamic programming and optimal control: Volume I*, volume 1. Athena scientific.

Bertsekas, D. P. (2018). *Abstract dynamic programming*. Athena Scientific.

Borovička, J. and Stachurski, J. (2020). Necessary and sufficient conditions for existence and uniqueness of recursive utilities. *The Journal of Finance*.

Brémaud, P. (2020). *Markov Chains: Gibbs Fields, Monte Carlo Simulation and Queues*, volume 31. Springer Nature.

Campbell, J. Y. (2017). *Financial decisions and markets: a course in asset pricing*. Princeton University Press.

Çınlar, E. (2011). *Probability and stochastics*, volume 261. Springer Science & Business Media.

Cochrane, J. H. (2009). *Asset pricing: Revised edition*. Princeton University Press.

Daley, D. (1968). Stochastically monotone markov chains. *Probability Theory and Related Fields*, 10(4):305–317.

Dixit, R. K. and Pindyck, R. S. (2012). *Investment under uncertainty*. Princeton University Press.

Du, Y. (1990). Fixed points of increasing operators in ordered banach spaces and applications. *Applicable Analysis*, 38(01-02):1–20.

Dudley, R. M. (2002). *Real analysis and probability*, volume 74. Cambridge University Press.

Duffie, D. (2010). *Dynamic asset pricing theory*. Princeton University Press.

Dvoretzky, A., Kiefer, J., and Wolfowitz, J. (1952). The inventory problem: I. case of known distributions of demand. *Econometrica*, pages 187–222.

Fajgelbaum, P. D., Schaal, E., and Taschereau-Dumouchel, M. (2017). Uncertainty traps. *The Quarterly Journal of Economics*, 132(4):1641–1692.

Foss, S., Shneer, V., Thomas, J. P., and Worrall, T. (2018). Stochastic stability of monotone economies in regenerative environments. *Journal of Economic Theory*, 173:334–360.

Frederick, S., Loewenstein, G., and O'donoghue, T. (2002). Time discounting and time preference: A critical review. *Journal of economic literature*, 40(2):351–401.

Friedman, M. (1956). *Theory of the consumption function*. Princeton university press.

Hansen, L. P. and Renault, E. (2010). Pricing kernels. *Encyclopedia of Quantitative Finance*.

Harrison, J. M. and Kreps, D. M. (1978). Speculative investor behavior in a stock market with heterogeneous expectations. *The Quarterly Journal of Economics*, 92(2):323–336.

Hopenhayn, H. A. (1992). Entry, exit, and firm dynamics in long run equilibrium. *Econometrica: Journal of the Econometric Society*, pages 1127–1150.

Hopenhayn, H. A. and Prescott, E. C. (1992). Stochastic monotonicity and stationary distributions for dynamic economies. *Econometrica*, pages 1387–1406.

Hu, T.-W. and Shmaya, E. (2019). Unique monetary equilibrium with inflation in a stationary bewley–aiyagari model. *Journal of Economic Theory*, 180:368–382.

Jaśkiewicz, A. and Nowak, A. S. (2014). Stationary markov perfect equilibria in risk sensitive stochastic overlapping generations models. *Journal of Economic Theory*, 151:411–447.

Kamihigashi, T. and Stachurski, J. (2014). Stochastic stability in monotone economies. *Theoretical Economics*, 9(2):383–407.

Kelle, P. and Milne, A. (1999). The effect of (s, s) ordering policy on the supply chain. *International Journal of Production Economics*, 59(1-3):113–122.

Kochenderfer, M. J., Wheeler, T. A., and Wray, K. H. (2022). *Algorithms for decision making*. The MIT Press.

Koopmans, T. C. (1960). Stationary ordinal utility and impatience. *Econometrica: Journal of the Econometric Society*, pages 287–309.

Krasnoselskii, M. (1964). *Positive solutions of operator equations*. Noordhoff, Groningen.

Kreyszig, E. (1978). *Introductory functional analysis with applications*, volume 1. wiley New York.

Kydland, F. and Prescott, E. C. (1980). A competitive theory of fluctuations and the feasibility and desirability of stabilization policy. In *Rational expectations and economic policy*, pages 169–198. University of Chicago Press.

Loewenstein, G. and Prelec, D. (1991). Negative time preference. *The American Economic Review*, 81:347–352.

Loewenstein, G. and Sicherman, N. (1991). Do workers prefer increasing wage profiles? *Journal of Labor Economics*, 9:67–84.

Lucas, R. and Stokey, N. (1989). *Recursive methods in dynamic economics*. Harvard University Press.

Lucas, R. E. (1978). Asset prices in an exchange economy. *Econometrica: Journal of the Econometric Society*, pages 1429–1445.

Mankiw, N. G. and Reis, R. (2002). Sticky information versus sticky prices: a proposal to replace the new keynesian phillips curve. *The Quarterly Journal of Economics*, 117(4):1295–1328.

McCall, J. J. (1970). Economics of Information and Job Search. *The Quarterly Journal of Economics*, 84(1):113–126.

Meyer, C. D. (2000). *Matrix analysis and applied linear algebra*, volume 71. Siam.

Modigliani, F., Brumberg, R., and Kurihara, K. K. (1954). Post-keynesian economics. In *Demography and Financial Markets*, pages 388–436. Rutgers University Press New Brunswick.

Nirei, M. (2006). Threshold behavior and aggregate fluctuation. *Journal of Economic Theory*, 127(1):309–322.

Puterman, M. L. (2005). *Markov decision processes: discrete stochastic dynamic programming*. Wiley Interscience.

Rust, J. (1996). Numerical dynamic programming in economics. *Handbook of computational economics*, 1:619–729.

Samuelson, P. A. (1939). Interactions between the multiplier analysis and the principle of acceleration. *The Review of Economics and Statistics*, 21(2):75–78.

Sargent, T. J. (1987). *Dynamic macroeconomic theory*. Harvard University Press.

Sargent, T. J. and Stachurski, J. (2022). Economic networks: Theory and computation.

Scarf, H. (1960). The optimality of (s, s) policies in the dynamic inventory problem. *Mathematical Methods in the Social Sciences*, pages 196–202.

Stachurski, J. (2022). *Economic dynamics: theory and computation*. MIT Press, 2 edition.

Zhang, Z. (2012). *Variational, topological, and partial order methods with their applications*, volume 29. Springer.