Preliminary and Incomplete

Overview of Dynamic Programming: Theory

Russell Cooper

February 12, 2013

1 Overview

These notes build from Adda-Cooper *Dynamic Economics*. Material from Stokey-Lucas *Recursive Methods in Economic Dynamics* and Ljungqvist-Sargent *Recursive Macroeconomic Theory* are used, and referenced, as well. These references are in the reading list.

Our presentation is by design not as formal as say that provided in *Bertsekas* or *Stokey-Lucas*. The reader interested in more mathematical rigor is urged to review those texts and their many references. We begin with a presentation of the non-stochastic problem and then add uncertainty to the formulation.

2 Non-Stochastic Case

We immediately adopt the dynamic program approach and consider the following equation, called Bellman's equation:

$$V(s) = \max_{c \in C(s)} \tilde{\sigma}(s, c) + \beta V(s') \tag{1}$$

for all $s \in S$, where $s' = \tau(s, c)$. Here time subscripts are eliminated, reflecting the stationarity of the problem. Instead, current variables are unprimed while future ones are denoted by a prime ($'$). Stokey-Lucas discuss the relationship between this recursive problem and the associated sequence problem.

The payoff function in a period is given by $\tilde{\sigma}(s, c)$. The first argument of the payoff function is termed the **state vector**, s. This represents a set of variables that influences the agent's return within the period but, by assumption, these variables are outside of the agent's control within period t. The state variables evolve over time in a manner that may be influenced by the **control vector** c, the second argument of the payoff function. The connection between the state variables over time is given by the transition equation:

$$s' = \tau(s, c).$$

So, given the current state and the current control, the state vector for the subsequent period is determined.

Note that the state vector has a very important property: it completely summarizes all of the information from the past that is needed to make a forward-looking decision. While preferences and the transition equation are certainly dependent on the past, this dependence is represented by s: other variables from the past do not affect current payoffs or constraints and thus cannot influence current decisions. This may seem restrictive but it is not: the vector s may include many variables so that the dependence of current choices on the past can be quite rich.

While the determination of the state vector is effectively determined by preferences and the transition equation, the researcher has some latitude in choosing the control vector. That is, there may be multiple ways of representing the same problem with alternative specifications of the control variables.

We assume that $c \in C$ and $s \in S$. In some cases, the control is restricted to be in subset of C which depends on the state vector: $c \in C(s)$. Finally assume that $\tilde{\sigma}(s, c)$ is bounded for $(s, c) \in S \times C$. [1]

As in *Stokey-Lucas*, the problem can be formulated as

$$V(s) = \max_{s' \in \Gamma(s)} \sigma(s, s') + \beta V(s') \tag{2}$$

for all $s \in S$. This is a more compact formulation and we will use it for our presentation.[2]

Let the policy function that determines the optimal value of the control (the future state) given the state be given by $s' = \phi(s)$. Our interest is ultimately in the policy function since we generally observe the actions of agents rather than their levels of utility. Still, to determine $\phi(s)$ we need to "solve" (2). That is, we need to find the value function that satisfies (2). It is important to realize that while the payoff and transition equations are primitive objects that models specify *a priori,* the value function is derived as the solution of the functional equation, (2).

[1] Ensuring that the problem is bounded is an issue in some economic applications, such as the growth model. Often these problems are dealt with by bounding the sets C and S.

[2] Essentially, this formulation inverts the transition equation and substitutes for c in the objective function. This substitution is reflected in the alternative notation for the return function.

There are two final properties of the agent's dynamic optimization problem worth specifying: **stationarity** and **discounting**. Note that neither the payoff nor the transition equations depend explicitly on time. True the problem is dynamic but time *per se* is not of the essence. In a given state, the optimal choice of the agent will be the same regardless of "when" he optimizes. Stationarity is important both for the analysis of the optimization problem and for empirical implementation. In fact, because of stationarity we can dispense with time subscripts as the problem is completely summarized by the current values of the state variables.

There are many results in the lengthy literature on dynamic programming problems on the existence of a solution to the functional equation. Here, we present one set of sufficient conditions. The reader is referred to *Bertsekas*, *LS* (chpt 20) and *Stokey-Lucas* for additional theorems under alternative assumptions about the payoff and transition functions.

Assumption I

1. S is a convex subset of R^l and $\Gamma(s)$ is nonempty, compact-valued and continuous,

2. $\sigma(\cdot)$ is real-valued, continuous and bounded,

3. $0 < \beta < 1$.

Theorem 1 *If Assumption I holds, then there exists a unique value function $V(s)$ that solves (2) and for any v_0, $||T^n v_0 - V|| \leq \beta^n ||v_0 - V||$.*

Proof: See *Stokey-Lucas*,[Theorem 4.6].

Instead of a formal proof, we give an intuitive sketch. The key component in the analysis is the definition of an operator, commonly denoted as T, defined by:

$$T(W)(s) = \max_{s' \in \Gamma(s)} \sigma(s, s') + \beta W(s') \text{ for all } s \in S.$$

The notation dates back at least to *Bertsekas*. This operator maps continuous bounded functions into continuous bounded functions (as in Theorem 4.6 of *Stokey-Lucas*).

So, this mapping takes a guess on the value function and, working through the maximization for all s, produces another value function, $T(W)(s)$. Clear, any $V(s)$ such that $V(s) = T(V)(s)$ for all $s \in S$ is a solution to (2). So, we can reduce the analysis to determining the fixed points of $T(W)$.

The fixed point argument proceeds by showing the $T(W)$ is a contraction using a pair of sufficient conditions from *blackwell65*. These conditions are: (i) monotonicity and (ii) discounting of the mapping $T(V)$. Your notes from Professor Zhou on continuity, Theorem 2.8, covers this material.

Monotonicity means that if $W(s) \geq Q(s)$ for all $s \in S$, then $T(W)(s) \geq T(Q)(s)$ for all $s \in S$. This property can be directly verified from the fact that $T(V)$ is generated by a maximization problem. So that if one adopts the choice of $c_Q(s)$ obtained from

$$\max_{s' \in \Gamma(s)} \sigma(s, s') + \beta Q(s') \text{ for all } s \in S.$$

When the proposed value function is $W(s)$ then:

$$\begin{aligned} T(W)(s) &= \max_{s' \in \Gamma(s)} \sigma(s, s') + \beta W(s') \ \geq \sigma(s, c_Q(s)) + \beta W(c_Q(s)) \\ &\geq \sigma(s, c_Q(s)) + \beta Q(c_Q(s)) \equiv T(Q)(s) \end{aligned}$$

for all $s \in S$.

Discounting means that adding a constant to W leads $T(W)$ to increase by less than this constant. That is, for any constant k, $T(W + k)(s) \leq T(W)(s) + \beta k$ for all $s \in S$ where $\beta \in [0, 1)$. The term discounting reflects the fact that β must be less than 1. This property is easy to verify in the dynamic programming problem:

$$T(W + k) = \max_{s' \in \Gamma(s)} \sigma(s, s') + \beta[W(s') + k] = T(W) + \beta k, \text{for all } s \in S$$

since we assume that the discount factor is less than 1.

The fact that $T(W)$ is a contraction allows us to take advantage of the contraction mapping theorem.[3] This theorem implies that: (i) there is a unique fixed point and (ii) this fixed point can be reached by an iteration process using an arbitrary initial condition. The first property is reflected in the theorem given above.

The second property is used extensively as a means of finding the solution to (2). To better understand this, let $V_0(s)$ for all $s \in S$ be an initial guess of the solution to (2). Consider $V_1 = T(V_0)$. If $V_1 = V_0$ for all $s \in S$, then we have the solution. Else, consider $V_2 = T(V_1)$ and continue iterating until $T(V) = V$ so that the functional equation is satisfied. Of course, in general, there is no reason to think that this iterative process will converge. However, if $T(V)$ is a contraction, as it is for our dynamic programming

[3]See *Stokey-Lucas* for a statement and proof of this theorem.

framework, then the $V(s)$ that satisfies (2) can be found from the iteration of $T(V_0(s))$ for any initial guess, $V_0(s)$. This procedure is called **value function iteration** and will be a valuable tool for applied analysis of dynamic programming problems.

The value function which satisfies (2) may inherit some properties from the more primitive functions that are the inputs into the dynamic programming problem: the payoff and transition equations. As we shall see, the property of strict concavity is useful for various applications. The results are given formally by:

Theorem 2 *If Assumption I holds, $\sigma(s, s')$ is increasing in s for every s' and $\Gamma(\cdot)$ is monotone in s, then the unique solution to (2) is strictly increasing.*

Proof: See Theorem 4.7 in *Stokey-Lucas*.

Theorem 3 *If Assumption I holds and $\sigma(s, s')$ is concave and the constraint set is convex, then the unique solution to (2) is strictly concave. Further, $\phi(s)$ is a continuous, single-valued function.*

Proof: See Theorem 4.8 in *Stokey-Lucas*.

Here the concavity of $\sigma(s, s')$ means $\sigma(\lambda(s_1, s_1') + (1-\lambda)(s_2, s_2')) \geq \lambda\sigma(s_1, s_1') + (1-\lambda)\sigma(s_2, s_2'))$. And the inequality is strict if $s_1 \neq s_2$. This is for all (s, s') such that $s' \in \Gamma(s)$.

The proof of these two theorems rely on showing that a property of $V(\cdot)$, such as strict concavity or strictly increasing, is preserved by $T(V)(s)$. To see how these last two results work together, suppose that $v(s)$ is strictly increasing and strictly concave for $s \in S$. We want to show these $V(s) = (Tv)(s)$ inherits these properties where:

$$V(s) \equiv \max_{s' \in \Gamma(s)} \sigma(s, s') + \beta v(s'). \tag{3}$$

To show that $V(s)$ is strictly increasing in s, consider $s_1 > s_2$. Then whatever policy $s'(s_2)$ was chosen when $s = s_2$ is feasible when $s = s_1$ by the monotonicity of $\Gamma(s)$. Thus $V(s_2) > V(s_1)$ since $s'(s_2)$ could be chosen in state s_1 and $\sigma(s, s')$ is increasing in s.

As for strictly concavity, let $s_\theta \equiv \theta s_0 + (1-\theta)s_1$ be the convex combination of two states. Let s'_{s_i} be the choices in state s_i for $i = 0, 1$ that solve (3). Let $s'_\theta \equiv \theta s'_0 + (1-\theta)s'_1$.

Then $V(s_\theta) \geq \sigma(s_\theta, s'_\theta) + \beta v(s'_\theta) > \theta[\sigma(s_0, s'_{s_0}) + \beta v(s_0)] + (1-\theta)[\sigma(s_1, s'_{s_1}) + \beta v(s_1)] = \theta V(s_0) + (1-\theta)V(s_1)$. The first inequality is because the choice in state s_θ could be better than s'_θ. The second inequality is due to the concavity of $\sigma(\cdot)$ and the asserted concavity of $v(\cdot)$.

As noted earlier, our interest is in the policy function. Note that from this last theorem, there is a stationary policy **function** which depends only on the state vector. This result is important for econometric application since stationarity is often assumed in characterizing the properties of various estimators.

In some cases, taking a derivative of the value function will be useful, along with Theorem 3, in characterizing the policy function. Theorem 4.11 of *Stokey-Lucas* uses a theorem by Benveniste and Scheinkman to describe the differentiability of the value function. If the return function is continuously differentiable, then the value function is continuously differentiable on the interior of the set of states and the choice set. Note that this result requires the assumptions used earlier as Assumption I.

2.1 Stochastic Dynamic Programming

While the non-stochastic problem is perhaps a natural starting point, in terms of applications it is necessary to consider stochastic elements. Clearly the stochastic growth model, consumption/savings decisions by households, factor demand by firms, pricing decisions by sellers, search decisions all involve the specification of dynamic stochastic environments.

Further, empirical applications rest upon shocks that are not observed by the econometrician. In many applications, the researcher appends a shock to an equation prior to estimation without being explicit about the source of the error term. This is not consistent with the approach of stochastic dynamic programming: shocks are part of the state vector of the agent. Of course, the researcher may not observe all of the variables that influence the agent and/or there may be measurement error. Nonetheless, being explicit about the source of error in empirical applications is part of the strength of this approach.

While stochastic elements can be added in many ways to dynamic programming problems, we consider the following formulation which is used in our applications. Letting ε represent the current value of a vector of "shocks"; i.e. random variables that are partially determined by nature. Let $\varepsilon \in \Psi$ which is assumed to be a finite set.[4] Then using the notation developed above, the functional equation becomes:

[4] As noted earlier, this structure is stronger than necessary but accords with the approach we will take in our empirical implementation. The results reported in *Bertsekas* require that Ψ is countable. *Stokey-Lucas* impose other restrictions on the stochastic process.

$$V(s,\varepsilon) = \max_{s' \in \Gamma(s,\varepsilon)} \sigma(s,s',\varepsilon) + \beta E_{\varepsilon'|\varepsilon} V(s',\varepsilon') \tag{4}$$

for all (s,ε).

Further, we have assumed that the stochastic process itself is purely exogenous as the distribution of ε' depends on ε but is independent of the current state and control. Note too that the distribution of ε' depends on only the realized value of ε : i.e. ε follows a first-order Markov process. This is not restrictive in the sense that if values of shocks from previous periods were relevant for the distribution of ε', then they could simply be added to the state vector.

Finally, note that the distribution of ε' conditional on ε, written as $\varepsilon'|\varepsilon$, is time invariant. This is analogous to the stationarity properties of the payoff and transition equations.. In this case, the conditional probability of $\varepsilon'|\varepsilon$ are characterized by a transition matrix, Π. The element π_{ij} of this matrix is defined as:

$$\pi_{ij} \equiv \ \mathrm{Prob}(\varepsilon' = \varepsilon_j | \varepsilon = \varepsilon_i)$$

which is just the likelihood that ε_j occurs in the next period, given that ε_i occurs today. Thus this transition matrix is used to compute the transition probabilities in (4). Throughout we assume that $\pi_{ij} \in (0,1)$ and $\sum_j \pi_{ij} = 1$ for each i. With this structure:

Theorem 4 *If $\sigma(s,s',\varepsilon)$ is real-valued, continuous, concave and bounded, $0 < \beta < 1$ and the constraint set is compact and convex, then:*

1. there exists a unique value function $V(s,\varepsilon)$ that solves (4)

2. there exists a stationary policy function, $\phi(s,\varepsilon)$.

Proof: As in the proof of Theorem 2, this is a direct application of Blackwell's Theorem. That is, with $\beta < 1$, discounting holds. Likewise, monotonicity is immediate as in the discussion above. See also the proof of Proposition 2 in *Bertsekas*, Chp. 6.