Lecture 13
Time Series & Dynamic Programming

Ryan McWay[†]

[†] *Applied Economics,*
University of Minnesota

Mathematics Review Course, Summer 2023
University of Minnesota
August 23rd, 2023

Last Lecture Review

- Statistics:
  - Population, Parameters, and Distributions
  - Discrete & Continuous Variables
  - Law of Iterated Expectations
  - Sampling
  - Estimate, Estimator, & Estimand
  - Conditional Expectation Function
  - Law of Large Numbers
  - Central Limit Theorem
  - Continuous Mapping Theorem
  - Delta Method
  - Hypothesis Testing

Daily Icebreaker

- Attendance via prompt:
  - Name
  - Daily Icebreaker: The department puts on a talent show. What is your talent?

Time Series

Motivation

- General background
  - The analysis of statistics when units vary over time.
- Why do economists' care?
  - Many aggregated measures of the economy come in time series formats (e.g., growth, inflation, unemployment, etc.).
- Application in this career
  - In macroeconomic analysis.

Motivation

- General background
  - The analysis of statistics when units vary over time.
- Why do economists' care?
  - Many aggregated measures of the economy come in time series formats (e.g., growth, inflation, unemployment, etc.).
- Application in this career
  - In macroeconomic analysis.

Motivation

- General background
  - The analysis of statistics when units vary over time.
- Why do economists' care?
  - Many aggregated measures of the economy come in time series formats (e.g., growth, inflation, unemployment, etc.).
- Application in this career
  - In macroeconomic analysis.

Overview

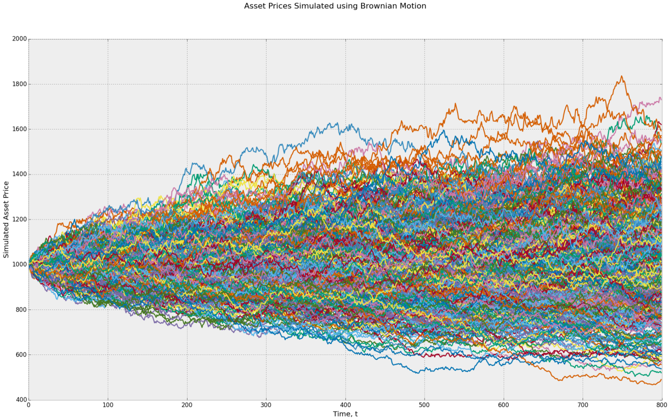
1. Stochastic Processes
2. Discrete Time Markov Chain
3. Continuous Time Markov Chain
4. Poisson Processes
5. System Reliability
6. Stationarity
7. Ergodicity
8. Unit Root or Random Walk

1. Stochastic Processes

- Stochastic: Randomly determined.
- Stochastic Process: Sequence of random variables indexed by time.
- Increment: Time between two index values
- Sample function (realization): Stochastic process may have many outcomes (due to randomness) with only one outcome realized.

$$\{X(t) : t \in T\}$$

1. Stochastic Processes



2. Discrete Time Markov Chains

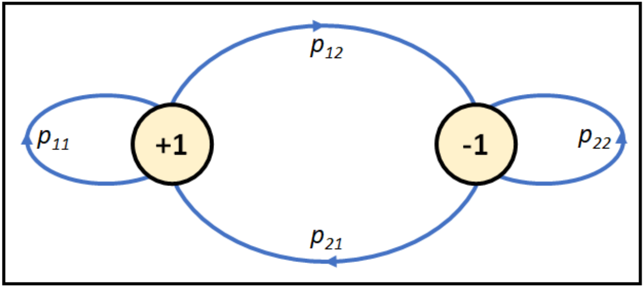
- Describes behavior that jumps between two (or more) states with known probabilities which depend on the current state of the system.
- Conditional Probabilities:

$$p_{i,j} = Pr(X_t = j | X_{t-1} = i)$$

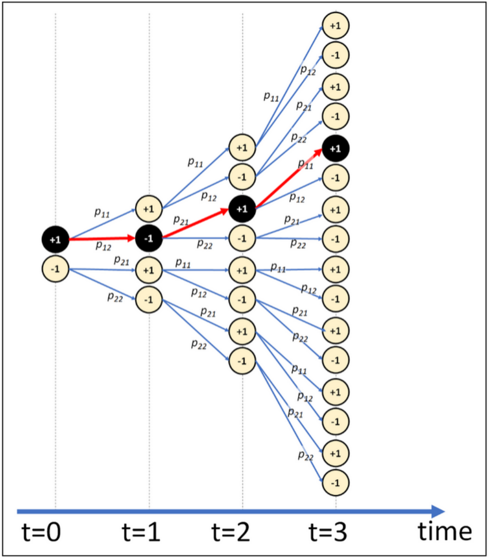
- Markov Property: That current state probability **only** depends on previous time index and **not** earlier time indices.

$$Pr(X_t = j | X_{t-1} = i) = Pr(X_t = j | X_{t-1} = i; X_{t-2} = i_2; \ldots, X_0 = i_0)$$

2. Discrete Time Markov Chains



2. Discrete Time Markov Chains

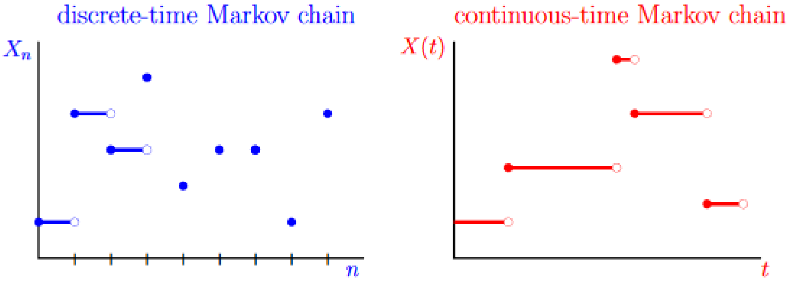


3. Continuous Time Markov Chains

- Rather than transitioning about integer times, switch to exponentially distributed states.
- $T_i \sim Exp(v_i)$
- $\forall s, t \geq 0$ and $\forall i, j \geq 0$ and $x(u) : 0 < u < s$
- Now probability of the state depends on previous states **and** current state.
- Conditional probability:

$$Pr(X(t+s) = j | X(s) = i, X(u) = x(u), 0 \geq u < s) =$$
$$Pr(X(t+s) = j | X(s) = i)$$

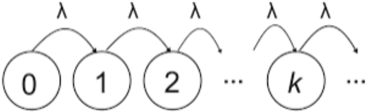
3. Continuous Time Markov Chains



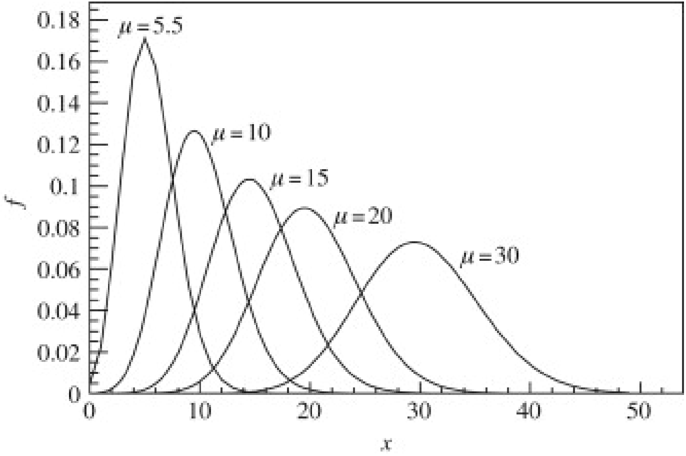
4. Poisson Processes

- Poisson Point Process: Points are randomly located independent of one another.
- A collection of Poisson points in a finite space can be described as a random variable with a Poisson distribution (e.g., count data).
- λ: Rate of intensity or average density of points in a region of space.
- Poisson Distribution: Probability of event occurring n times given an interval of time or space determined by the mean number of events λ.

$$Pr\{N = n\} = \frac{\lambda^n}{n!} e^{-\lambda}$$

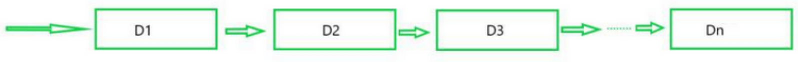


4. Poisson Processes



5. System Reliability

- Consider the process that a system has (i.e., stages of development) has many components (e.g., GDP, inflation, etc.) operating in a series of stages.
- Reliability r_i is the probability that each stage will be successful.
- System Reliability is the geometric product: $\pi(r_i) = \prod_i^n r_i$



6. Stationarity

- Y_t is a random draw (sample) from the distribution.
- The joint distribution of $Y_t, Y_{t+1}, \ldots, Y_{t+l}$ has some mean. We want to know that this mean is constant in the population.
- E.g., The means and variances for the **distribution** are the same at all times t.

6. Stationarity

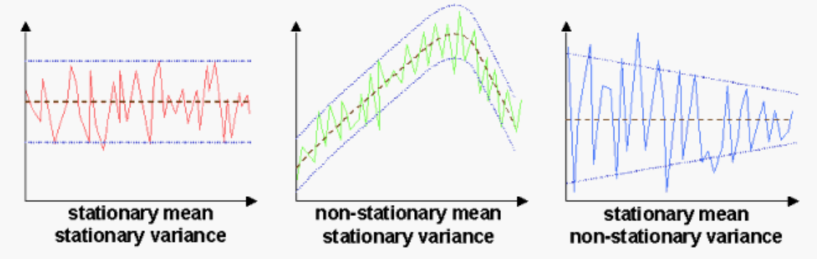
- (Weak) Covariance stationarity when the mean and variance are finite and do not depend on t.

$$\mu = \mathbb{E}[Y_t]$$
$$\Sigma = Var(Y_t) = \mathbb{E}[(Y_t - \mu)(Y_t - \mu)^T]$$

- And that the autocovariances do not depend on $t \forall k$.
$$\Gamma(k) = Cov(Y_t, Y_{t-k}) = \mathbb{E}[(Y_t - \mu)(Y_{t-k} - \mu)^T]$$

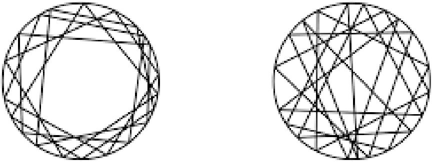
- Strict staionarity asserts that the joint distribution of $Y_t, Y_{t+1}, \ldots, Y_{t+l}$ does not depend on $t \forall l$.

6. Stationarity



7. Ergodicity

- Stationarity alone does not allow us to use the Law of Large Numbers and the Central Limit Theorem.
- An issue is that our expected mean $\mathbb{E}[Y_t] = Z$ may not converge as $n \to \infty$.
- Ergodic system is if all invariant events (i.e., not a function of t) are trivial (i.e., $Pr(x) = \{0, 1\}$ – never occur or always occur).
- The time series does not get ‘stuck’ in the sample space as it passes through **all** parts of the sample distribution.



A. Non-ergodic   B. Ergodic

7. Ergodicity

- Ergodic Theorem: If a vector Y_t is strictly stationary, ergodic, and $\mathbb{E}[||Y||] < \infty$, then as $n \to \infty$:

$$\mathbb{E}[||\bar{Y} - \mu||] \to 0$$
$$\bar{Y} \xrightarrow{p} \mu$$

- We can consistently estimate the mean of a time series variable (or their transformations).

8. Unit Root or Random Walk

- Consider that the time series variable is desribed as an auto-regressive process (i.e., determined by previous values)

$$Y_t = \alpha_0 + \alpha_1 Y_{t-1} + \varepsilon_t$$

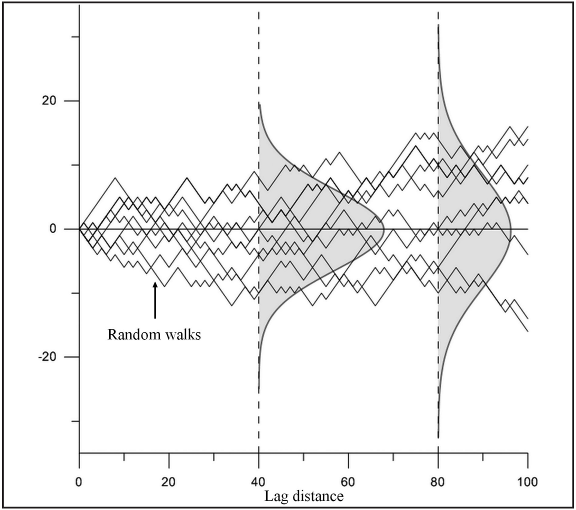
- Random Walk: $\alpha_0 = 0$ and $\alpha_1 = 1$
- It is driven by the error term in each period ε_t

$$Y_t = Y_{t-1} + \varepsilon_t$$

- This is a non-convergent sequence.
- Even if we take a starting point Y_0 and an infinite number of error terms, we cannot describe (predict) Y_t.

$$Y_t = Y_0 + \sum_{j=1}^{t} \varepsilon_j$$

8. Unit Root or Random Walk



Dynamic Programming

Motivation

- General background
  - When you can break a problem into smaller versions of the problem, then you can solve the smaller problems and repeat this recursively to solve the bigger problem.
  - Turn a single optimization problem into many optimization problems.
  - Typically happens when you have a dynamic optimization problem as compared to a static optimization problem.
- Why do economists' care?
  - Used for optimization problems in time series.
- Application in this career
  - In macroeconomics and resource allocation.

Motivation

- General background
  - When you can break a problem into smaller versions of the problem, then you can solve the smaller problems and repeat this recursively to solve the bigger problem.
  - Turn a single optimization problem into many optimization problems.
  - Typically happens when you have a dynamic optimization problem as compared to a static optimization problem.
- Why do economists' care?
  - Used for optimization problems in time series.
- Application in this career
  - In macroeconomics and resource allocation.

Motivation

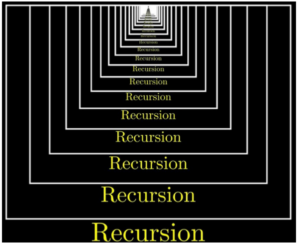
- General background
  - When you can break a problem into smaller versions of the problem, then you can solve the smaller problems and repeat this recursively to solve the bigger problem.
  - Turn a single optimization problem into many optimization problems.
  - Typically happens when you have a dynamic optimization problem as compared to a static optimization problem.
- Why do economists' care?
  - Used for optimization problems in time series.
- Application in this career
  - In macroeconomics and resource allocation.

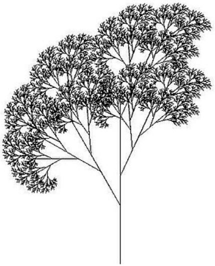
Overview

1. Recursion
2. Memoization
3. Tabulation
4. Overlapping Sub-problems
5. Optimal Sub-structure
6. Dynamic Programming Problem
7. Theory of the Maximum
8. Optimizing the Value Function
9. Bellman Equation with Finite Horizon
10. Bellman's Principle of Optimality
11. Backward Induction
12. Bellman Equation with Infinite Horizon
13. Metric Space
14. Blackwell Sufficient Conditions
15. Contraction Mapping Theorem
16. Value Function Iteration

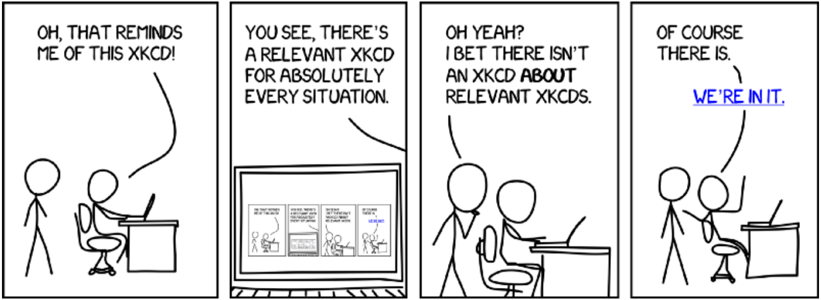
1. Recursion

- Dynamic programming is optimization over plain recursion.
- Recursion: A function that repeats or uses previous terms to calculate subsequent terms.
- Arithmetic sequence: $a_n = a_{n-1} + a_1$
- Geometric sequence: $a_n = r \times a_{n-1}$

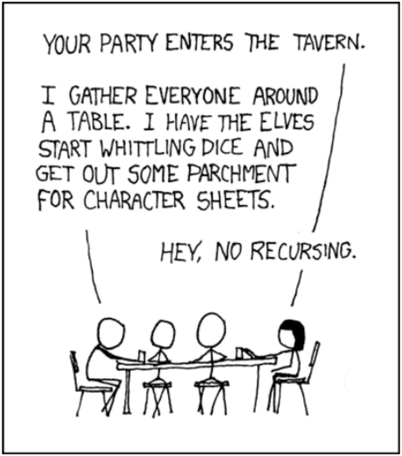




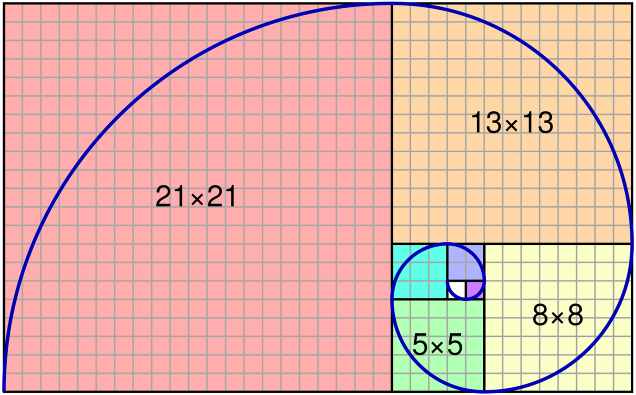
1. Recursion


OH, THAT REMINDS ME OF THIS XKCD!
YOU SEE, THERE'S A RELEVANT XKCD FOR ABSOLUTELY EVERY SITUATION.
OH YEAH? I BET THERE ISN'T AN XKCD ABOUT RELEVANT XKCDS.
OF COURSE THERE IS.
WE'RE IN IT.

1. Recursion (Again?)



Application: Fibonacci Sequence



2. Memoization

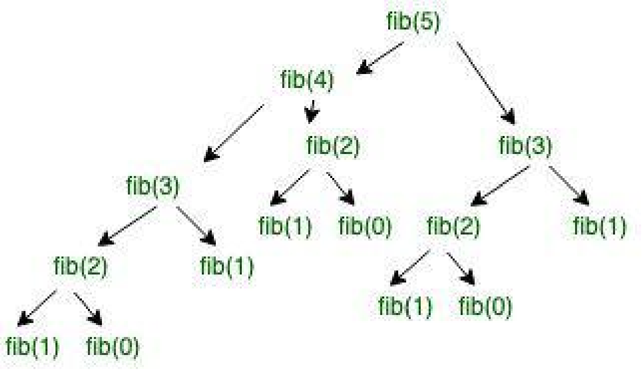
- Top-down approach
- Cache the results of a sub-problem and call them again as needed.
- Used when there are overlapping sub-problems.

3. Tabulation

- Bottom-up approach
- Store the results of a sub-problem in a table. Build the table to solve the larger problem.
- Used in sequential problems without overlapping sub-problems.

4. Overlapping Sub-problems

- Divide and conquer.
- Determine the sub-problems that are used throughout the problem and compute them to be stored for later use.

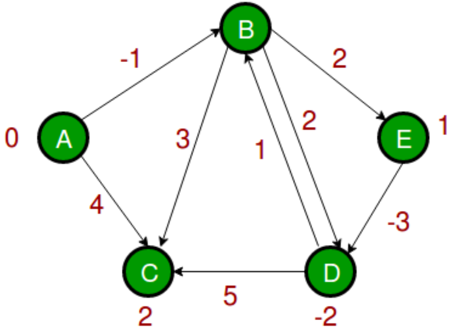


5. Optimal Sub-structure

- Optimal Sub-structure: The given problem can be obtained by using the optimal solution to its sub-problems instead of attempting all possible ways to solve the sub-problems.
- The Shortest Path: If x lies on the path between nodes U and V, then the shortest path p_{UV} from $U \to V$ is $U \to x$ and $x \to V$.
- Distance may vary, so we can apply some weight $w(\cdot)$ to adjust.
- The Longest Path: The longest simple path (i.e., without cycling) between two nodes.
- Iterate from origin adding 1 edge at a time to determine shortest path from i to any other j.

5. Optimal Sub-structure

A	B	C	D	E
0	∞	∞	∞	∞
0	-1	∞	∞	∞
0	-1	4	∞	∞
0	-1	2	∞	∞
0	-1	2	∞	1
0	-1	2	1	1
0	-1	2	-2	1



6. Dynamic Programming Problem

- Markov transition function: $Q(z', z) = Pr(z_{t+1} \leq z' | z_t = z)$
- Assume z_t is known and z_{t+1} is unknown.
- Instantaneous return (utility) function: $u(x_t, c_t)$
- State variables: $x_t \in X \forall t$
- Control variables: $c_t \in C(x_t, z_t) \forall t$.
- Law of motion: $x_{t+1} = f(x_t, z_t, c_t)$
- Discount factor: $\beta < 1$
- Conditional Expectation at $t = 0$: $\mathbb{E}_0$
- Objective function (s.t., the law of motion and stochastic process):

$$\mathbb{E}_0 \sum_{t=0}^{T} \beta^t u(x_t, c_t)$$

6. Dynamic Programming Problem

- State vector (x_t, z_t) completely describes that state at every t.
- Additive separability of objective function implies c_t depends only on current states through a time-varying function:

$$
\begin{aligned}
g_t : & X \times Z \to C, \forall t \\
c_t = & g_t(x_t, z_t)
\end{aligned}
$$

- g_t is a decision rule that maps the state vector into choices.
- The sequence $\pi_T = \{g_0, g_1, \ldots, g_T\}$ is a policy.
- Expected discounted value for a given policy π_T:

$$W_T(x_0, z_0, \pi_T) = \mathbb{E}_0 \sum_{t=0}^{T} \beta^t u(x_t, g_t(x_t, z_t))$$

6. Dynamic Programming Problem

- The maximization problem.
- Individual maximizes:

$$\max_{g_t(x_t,z_t)\in C(x_t,z_t)} W_T(x_0, z_0, \pi_T)$$

- subject to the law of motion:

$$x_{t+1} = f(x_t, z_t, g_t(x_t, z_t))$$

- given the initial state and the transition function:

$$x_0, z_0, Q(z', z)$$

7. Theory of the Maximum

- If
  - The constraint set $C(x_t, z_t)$ is non-empty, compact, and continuous
  - $u(\cdot)$ is continuous and bounded
  - $f(\cdot)$ is continuous
  - Q satisfies the Feller property (sub-set of Markov processes)
- Then
  - Exists a solution (optimal policy) for the problem: $\pi_T^* = \{g_0^*, g_1^*, \ldots, g_T^*\}$
  - The value function $V_T(x_0, z_0) = W_T(x_0, z_0, \pi_T^*)$ is continuous

8. Optimizing the Value Function

- The Value Function: Expected discounted present value of optimal policy π_T^*

$$V_T(x_0, z_0) = \mathbb{E}_0 \sum_{t=0}^{T} \beta^t u(x_t, g_t(x_t, z_t))$$

- By the Theory of the Maximum and the Law of Iterated Expectations, we can rearrange this:

$$V_T(x_0, z_0) = \max_{\pi_T} \mathbb{E}_0\{u(x_0, x_0) + \sum_{t=1}^{T} \beta^t u(x_t, c_t)\}$$

$$V_T(x_0, z_0) = \max_{c_0} \mathbb{E}_0\{u(x_0, x_0) + \beta \max_{\pi_{T-1}} W_{T-1}(x_1, z_1, \pi_{T-1})\}$$

- Where $\pi_{T-1} = \{c_1, c_2, \ldots, c_T\}$

8. Optimizing the Value Function

- Redefine the value function for $T-1$:
$$V_{T-1}(x_1, z_1) = W_{T-1}(x_1, z_1, \pi^*_{T-1})$$

- Now suppose that we have $s \in \{1, 2, \ldots, T\}$ time periods to go.
- Then, our optimization of the value function is
$$V_s(x, z) = \max_{c \in C(x,z)} u(x, c) + \beta \mathbb{E} V_{s-1}(x', z')$$

9. Bellman Equation with Finite Horizon

- Using this basis, let use define $x = x_{T-s}$, $z = z_{T-s}$, and $z' = z_{T-s+1}$.
- The Bellman Equation:

$$V_s(x, z) = \max_{c \in C(x,z)} u(x, c) + \beta \int_Z V_{s-1}(f(x, z, c), z') dQ(z', z)$$

- This reduces the sequence of decision rules into a sequence of choices for control variables.
- E.g., The dynamic problem is now a series of static optimization problems.

Demonstration: Bellman Equation

Question:
$max \sum_{t=0}^{T} \beta^t u(c_t)$ s.t. $c_t + k_{t+1} = f(k_t) + (1-\delta)k_t$, k_0 is given, and $0 < \delta < 1$.

Answer:
Bellman Equation:

$$
\begin{aligned}
v(k_t) =& max[u(c_t) + \beta v(k_{t+1})] \text{ s.t. } c_t + k_{t+1} = f(k_t) + (1-\delta)k_t \\
c_t =& f(k_t) + (1-\delta)k_t - k_{t+1} \\
v(k_t) =& max[u(f(k_t) + (1-\delta)k_t - k_{t+1}) + \beta v(k_{t+1})]
\end{aligned}
$$

Demonstration: Bellman Equation

Question:
$max \sum_{t=0}^{T} \beta^t u(c_t)$ s.t. $c_t + k_{t+1} = f(k_t) + (1-\delta)k_t$, k_0 is given, and $0 < \delta < 1$.

Answer:
Bellman Equation:

$$\begin{aligned} v(k_t) =& max[u(c_t) + \beta v(k_{t+1})] \text{ s.t. } c_t + k_{t+1} = f(k_t) + (1-\delta)k_t \\ c_t =& f(k_t) + (1-\delta)k_t - k_{t+1} \\ v(k_t) =& max[u(f(k_t) + (1-\delta)k_t - k_{t+1}) + \beta v(k_{t+1})] \end{aligned}$$

Demonstration: Bellman Equation

Answer:

FOC w/ k_{t+1}:

$$\begin{aligned}\frac{\partial v(k_t)}{\partial k_{t+1}} =& u'(c_t) \cdot (-1) + \beta v'(k_{t+1}) = 0 \\ \implies u'(c_t) =& \beta v'(k_{t+1})\end{aligned}$$

Apply envelope theorem:

$$\begin{aligned}\frac{\partial v(k_t)}{\partial k_t} =& u'(c_t) \cdot [f'(k_t) + (1-\delta) - \frac{\partial k_{t+1}}{\partial k_t}] + \beta v'(k_{t+1}) \frac{\partial k_{t+1}}{\partial k_t} \\ =& u'(c_t) \cdot f'(k_t) + u'(c_t) \cdot (1-\delta) - u'(c_t) \cdot \frac{\partial k_{t+1}}{\partial k_t} + \beta v'(k_{t+1}) \frac{\partial k_{t+1}}{\partial k_t} \\ =& u'(c_t)[f'(k_t) + (1-\delta)] + \frac{\partial k_{t+1}}{\partial k_t} [\underbrace{\beta v'(k_{t+1}) - u'(c_t)}_{\text{FOC}=0}]\end{aligned}$$

Demonstration: Bellman Equation

Answer:

$$v'(k_t) = u'(c_t)[f'(k_t) + (1 - \delta)]$$

Apply backward induction: $v'(k_t) \to v'(k_{t+1})$

$$v'(k_{t+1}) = u'(c_{t+1})[f'(k_{t+1}) + (1 - \delta)]$$

Plug into FOC:

$$\begin{aligned} u'(c_t) =& \beta v'(k_{t+1}) \\ u'(c_t) =& \beta u'(c_{t+1})[f'(k_{t+1}) + (1 - \delta)] \end{aligned}$$

10. Bellman's Principle of Optimality

- Time Consistent policies:
- If the sequence of functions $\pi_T^* = \{g_0^*, g_1^*, \ldots, g_T^*\}$ is the optimal policy that maximizes $W_T(x_0, z_0, \pi_T)$
- Then after $j : j + s = T$ periods, $\pi_s^* = \{g_{T-s}^*, g_{T-s+1}^*, \ldots, g_T^*\}$ is the optimal policy that maximizes $W_s(x_j, z_j, \pi_s)$
- E.g., We can find policies that are optimal in the future.

11. Backward Induction

- A way to solve the finite Bellman problem.
- Start with the last period $s = 0$.
- So the static problem is:

$$V_0(x_T, z_T) = \max_{c_T \in C(x_T, z_T)} u(x_T, c_T)$$

- This optimizes at $g_T^*(x_T, z_T)$
- Now go back one period to $s = 1$ and use the law of motion $x_T = f(x_{T-1}, z_{T-1}, c_{T-1})$ and the transition function Q.

$$V_1(x_{T-1}, z_{T-1}) = \max_{c_{T-1} \in C(x_{T-1}, z_{T-1})} u(x_{T-1}, c_{T-1}) + \beta \int_Z V_0(f(x_{T-1}, z_{T-1}, c_{T-1}), z_T) dQ(z_T, z_{T-1})$$

- Continue until $s = T$. Now you have the optimal path for the policy.

12. Bellman Equation with Infinite Horizon

- What if time goes on forever: $T \to \infty$?
- Can't use backward induction.
- But now the problem is the same at every time period because you have ∞ periods to go at each state.
- Now the environment is now stationary $\to$ you can treat the value function as time **invariant** $V(x,z)$

$$V(x,z) = \max_{c \in C(x,z)} u(x,c) + \beta \int_Z V(f(x,z,c), z') dQ(z', z)$$

- The stationary decision rule solution is $c^* = g^*(x,z)$

12. Bellman Equation with Infinite Horizon

- We want to know:

1. Does the value function satisfy a fixed point property $V = T(V)$?
2. Can we treat the infinite case as the limit of a finite horizon as $s \to \infty$?

- This can be solved with Value Function Iteration, but we need to introduce some concepts:
  - Metric space
  - Blackwell Sufficient Conditions
  - Contraction Mapping Theorem

13. Metric Space

- Metric Space $(\mathcal{M}, d)$: Set $\mathcal{M}$ with metric (i.e., distance) $d : \mathcal{M} \times \mathcal{M} \to \mathbb{R}_+$ satisfies the following conditions $\forall \varphi, \phi, \psi \in \mathcal{M}$:

1. $d(\varphi, \phi) = 0 \iff \varphi = \phi$
2. $d(\varphi, \phi) = d(\phi, \varphi)$
3. $d(\varphi, \psi) \leq d(\varphi, \phi) + d(\phi, \psi)$

- Operator: Function T mapping metric space into itself.
- Contraction Mapping: T is a contraction with modulus β if $\exists \beta \in (0, 1)$:

$$\forall(\varphi, \phi) \in (\mathcal{M}, d), d(T(\varphi), T(\phi)) \leq \beta d(\varphi, \phi)$$

14. Blackwell Sufficient Conditions

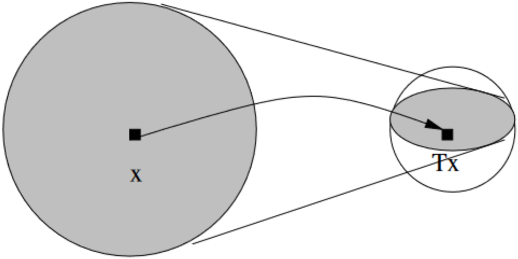
- Let T be an operator in metric space $(\mathcal{M}, d_\infty)$ where $\mathcal{M}$ is a space function with domain X and d_∞ is a supremum metric.
- Then, T is a contraction mapping with modulus β if it satisfies:

1. Monotonicity: $\varphi \leq \phi \rightarrow T(\varphi) \leq T(\phi), \forall \varphi, \phi \in \mathcal{M}$
2. Discounting: $T(a + \varphi) \leq a\beta + T(\varphi), \forall a > 0, \varphi \in \mathcal{M}$

15. Contraction Mapping Theorem

- Ensures a fixed point exists and is unique that can be computed by iteration (e.g., backward induction).
- Let's the value function be a fixed point.
- Let $(\mathcal{M}, d)$ be a complete metric space, and let T be a contraction mapping with modulus β. Then

1. T is a unique fixed point $\varphi^* \in \mathcal{M}$
2. $\forall \varphi^0 \in \mathcal{M}$, the sequence $\varphi^{n+1} = T(\varphi^n)$ starting at φ^0 converging to φ^* in metric d.



16. Value Function Iteration

- To solve an infinite Belleman equation.

1. For unknown V, we can start iterating from an initial ϕ_0 which is certain to converge to a solution V.
2. Let $V_0 = \zeta$ be an initial guess at the value function. Iterate $V_1 = T(\zeta), V_2 = T(V_1), \ldots, V_{n+1} = T(V_n)$ converging over N iterations to V^*.

Review

Review: Times Series

1. Stochastic Processes
2. Discrete Time Markov Chain
3. Continuous Time Markov Chain
4. Poisson Processes
5. System Reliability
6. Stationarity
7. Ergodicity
8. Unit Root or Random Walk

Review: Dynamic Programming

1. Recursion
2. Memoization
3. Tabulation
4. Overlapping Sub-problems
5. Optimal Sub-structure
6. Dynamic Programming Problem
7. Theory of the Maximum
8. Optimizing the Value Function
9. Bellman Equation with Finite Horizon
10. Bellman's Principle of Optimality
11. Backward Induction
12. Bellman Equation with Infinite Horizon
13. Metric Space
14. Blackwell Sufficient Conditions
15. Contraction Mapping Theorem
16. Value Function Iteration