Programação Dinâmica - I

Mauro Rodrigues (USP)

2020

Preliminares

- Modelo neoclássico de crescimento
  - Tempo discreto: $t \in \{0, 1, 2, ...\}$
  - Agente representativo – contínuo de agentes idênticos distribuídos uniformemente no intervalo $[0, 1]$
  - Sem incerteza; não há crescimento populacional
  - Resolver usando planejador central

- Preferência: agentes vivem para sempre e valorizam consumo (c_t) e lazer (l_t):

$$U = \sum_{t=0}^{\infty} \beta^t u(c_t, l_t)$$

  - $\beta \in (0, 1)$ - fator de desconto
  - $u_c > 0,\ u_l > 0,\ u_{cc} < 0,\ u_{ll} < 0$

Tecnologia

- Restrição de tempo: indivíduos têm 1 unidade de tempo em todo período, que pode ser dividida entre trabalho (n_t) e lazer (l_t):

$$n_t + l_t = 1$$

- Função de produção:
  - Produto gerado usando trabalho e capital:

$$y_t = F(k_t, n_t)$$

  - Função F satisfaz retornos constantes de escala
  - F satisfaz as condições de Inada:
    - $F(k, 0) = F(0, n) = 0$
    - $F_k > 0, \; F_n > 0$
    - $F_{kk} < 0, \; F_{nn} < 0$
    - $\lim_{k \to 0} F_k = \lim_{n \to 0} F_n = \infty$
    - $\lim_{k \to \infty} F_k = \lim_{n \to \infty} F_n = 0$

Tecnologia

- Produto pode ser usado para consumo e investimento:
$$c_t + i_t = y_t = F(k_t, n_t)$$
- Investimento amplia estoque de capital no período seguinte; estoque de capital corrente deprecia à taxa $\delta \in (0,1]$:
$$k_{t+1} = (1-\delta)k_t + i_t$$
- Restrição de recursos em t:
$$c_t + k_{t+1} - (1-\delta)k_t = F(k_t, n_t)$$
- Capital é a **variável de estado**: no início de cada período seu valor é dado
  - $k_0 > 0$ dado

Problema sequencial

- Problema do planejador central

$$\max_{\{c_t, n_t, k_{t+1}\}_{t=0}^{\infty}} \left\{ \sum_{t=0}^{\infty} \beta^t u(c_t, 1 - n_t) \right\}$$

s.t.

$$c_t + k_{t+1} - (1-\delta)k_t \leq F(k_t, n_t), \text{para todo } t \geq 0$$

$$k_0 > 0 \text{ dado}$$

- **Problema sequencial**: resolver as trajetórias ótimas inteiras uma única vez
  - Encontrar sequências ótimas de consumo e capital $\{c_t, n_t, k_{t+1}\}_{t=0}^{\infty}$ que maximizam U, dadas as restrições

Condições de primeira ordem

- Lagrangeano:

$$\mathscr{L} = \sum_{t=0}^{\infty} \beta^t \{u(c_t, 1-n_t) + \lambda_t [F(k_t, n_t) - c_t - k_{t+1} + (1-\delta)k_t]\}$$

- Condições de primeira ordem

$$c_t : \beta^t \{u_{ct} - \lambda_t\} = 0$$

$$n_t : \beta^t \{-u_{lt} + \lambda_t F_{nt}\} = 0$$

$$k_{t+1} : -\beta^t \lambda_t + \beta^{t+1} \lambda_{t+1} \{F_{k,t+1} + (1-\delta)\} = 0$$

- Reescrevendo:

$$u_{ct} = \lambda_t$$

$$u_{lt} = u_{ct} F_{nt}$$

$$u_{ct} = \beta u_{c,t+1} \{F_{k,t+1} + (1-\delta)\} \text{ [Equação de Euler]}$$

Condição de transversalidade

- Como a função de produção é homogênea de grau 1, então $F(k,n) = F_k k + F_n n$. Reescrevendo Lagrangeano:

$$\mathscr{L} = \sum_{t=0}^{\infty} \beta^t \{u(c_t, 1-n_t) + \lambda_t [F_{kt} k_t + F_{nt} n_t - c_t - k_{t+1} + (1-\delta) k_t]\}$$

$$= \sum_{t=0}^{\infty} \beta^t \{u(c_t, 1-n_t) + \lambda_t [F_{nt} n_t - c_t]\}$$

$$+ \sum_{t=0}^{\infty} \beta^t \lambda_t \{[F_{kt} + (1-\delta)] k_t - k_{t+1}\}$$

  - Abrindo a segunda soma do lado direito:

$$\sum_{t=0}^{\infty} \beta^t \lambda_t \{[F_{kt} + (1-\delta)] k_t - k_{t+1}\} = \lambda_0 \{[F_{k0} + (1-\delta)] k_0 - \lambda_0 k_1$$
$$+ \beta \lambda_1 [F_{k1} + (1-\delta)] k_1 - \beta \lambda_1 k_2 + \ldots +$$
$$+ \beta^t \lambda_t \{[F_{kt} + (1-\delta)] k_t - \beta^t \lambda_t k_{t+1} +$$
$$+ \beta^{t+1} \lambda_{t+1} \{[F_{k,t+1} + (1-\delta)] k_{t+1}$$
$$- \beta \lambda_{t+1} k_{t+2} + \ldots$$

Condição de transversalidade

- Rearranjando a soma do slide anterior

$$\lambda_0[F_{k0} + (1-\delta)]k_0 +$$
$$+ \sum_{t=0}^{\infty} \{\beta^{t+1}\lambda_{t+1}[F_{k,t+1} + (1-\delta)] + \beta^t \lambda_t\} k_{t+1}$$
$$- \lim_{t\to\infty} \beta^t \lambda_t k_{t+1}$$

  - Como k_0 é dado, o primeiro termo permanece
  - O segundo termo é nulo por causa da equação de Euler
  - No ótimo, o planejador escolhe a trajetória de modo a tornar o último termo o menor possível. Como o capital não pode ser negativo, segue que:

$$\lim_{t\to\infty} \beta^t u_{ct} k_{t+1} = 0 \text{ (condição de transversalidade)}$$

Problema mais geral

- Maximizar a soma:

$$\max_{\{x_{s+1},u_s\}_{s=t}^{\infty}} \left\{ \sum_{s=t}^{\infty} \beta^{s-t} r(x_s, u_s) : x_{s+1} = g(u_s) \right\}$$

$$x_t \text{ dado}$$

- x : variável de estado (dada em t); u : variável de controle
- O problema acima está na forma sequencial (note que t é o período inicial)
  - Escolhem-se as trajetórias inteiras de x e u, dado o valor inicial da variável de estado

Função Valor

- Colocando o problema explicitamente em função do estado inicial (ainda formulação sequencial)

$$V(x_t) = \max_{\{x_{s+1},u_s\}_{s=t}^{\infty}} \left\{ \sum_{s=t}^{\infty} \beta^{s-t} r(x_s, u_s) : x_{s+1} = g(u_s) \right\}$$

- Quebrando o problema em duas partes: o valor do payoff em $t = 0$, e o resto da soma

$$\begin{aligned} V(x_t) &= \max_{\{x_{s+1},u_s\}_{s=t}^{\infty}} \left\{ r(x_t, u_t) + \sum_{s=t+1}^{\infty} \beta^{s-t} r(x_s, u_s) : x_{s+1} = g(u_s) \right\} \\ &= \max_{\{x_{s+1},u_s\}_{s=t}^{\infty}} \left\{ r(x_t, u_t) + \beta \sum_{s=t+1}^{\infty} \beta^{s-(t+1)} r(x_s, u_s) : x_{s+1} = g(u_s) \right\} \end{aligned}$$

- A soma do lado direito é exatamente o mesmo problema, porém da perspectiva de $t + 1$
  - Estado relevante é x_{t+1}

Formulação Recursiva

- Problema na **forma recursiva** (Equação de Bellman):

$$V(x_t) = \max_{x_{t+1}, u_t} \{r(x_t, u_t) + \beta V(x_{t+1}) : x_{t+1} = g(u_t)\}$$

  - Maximização ocorre em todo t
  - Em determinado período, escolhem-se apenas o controle corrente e o estado no período seguinte, levando em conta que amanhã haverá um problema semelhante (que é afetado pela escolha do estado)

- Equação acima vale para quaisquer t e $t+1$. De modo geral:

$$V(x) = \max_{x', u} \left\{ r(x, u) + \beta V(x') : x' = g(u) \right\}$$

  - Variáveis sem linha referem-se ao período corrente; variáveis com linha referem-se ao próximo período

Formulação Recursiva

- Condição de ótimo (substituindo restrição) – derivada com relação a variáveis de escolha deve ser nula

$$\frac{\partial r(x,u)}{\partial u} + \beta V'(x')g'(u) = 0$$

- Teorema do envelope (derivar com relação à variável de estado x)

$$V'(x) = \frac{\partial r(x,u)}{\partial x}$$

  - Adiantando 1 período:

$$V'(x') = \frac{\partial r(x',u')}{\partial x}$$

- Portanto:

$$\frac{\partial r(x,u)}{\partial u} + \beta \frac{\partial r(x',u')}{\partial x} g'(u) = 0$$

Crescimento neoclássico (novamente)

- Formulação sequencial

$$\max_{\{c_s, n_s, k_{s+1}\}_{s=t}^{\infty}} \left\{ \sum_{s=t}^{\infty} \beta^{s-t} u(c_s, 1-n_s) \right\}$$

s.t.

$$c_s + k_{s+1} - (1-\delta)k_s \leq F(k_s, n_s), \text{para todo } s \geq t$$

$$k_t > 0 \text{ dado}$$

- Função Valor (substituindo restrição):

$$V(k_t) = \max_{\{n_s, k_{s+1}\}_{s=t}^{\infty}} \left\{ \sum_{s=t}^{\infty} \beta^{s-t} u[F(k_s, n_s) - k_{s+1} + (1-\delta)k_s, 1-n_s] \right\}$$

  - Quebrando a soma:

$$V(k_t) = \max_{\{n_s, k_{s+1}\}_{s=t}^{\infty}} \{u[F(k_t, n_t) - k_{t+1} + (1-\delta)k_t, 1-n_t] +$$

$$+ \beta \sum_{s=t+1}^{\infty} \beta^{s-(t+1)} u[F(k_s, n_s) - k_{s+1} + (1-\delta)k_s, 1-n_s]\}$$

Crescimento neoclássico (novamente)

- Formulação recursiva (equação de Bellman):

$$V(k_t) = \max_{n_t, k_{t+1}} \{u[F(k_t, n_t) - k_{t+1} + (1-\delta)k_t, 1 - n_t] + \beta V(k_{t+1})\}$$

$$V(k) = \max_{n,k'} \{u[F(k,n) - k' + (1-\delta)k, 1 - n] + \beta V(k')\}$$

- Condições de primeira ordem:

$$n : u_c F_n - u_l = 0$$

$$k' : -u_c + \beta V'(k') = 0$$

- Teorema do envelope:

$$V'(k) = u_c[F_k + (1-\delta)]$$

- Adiantando 1 período:

$$V'(k') = u_{c'}[F_{k'} + (1-\delta)]$$

Crescimento neoclássico (novamente)

- Portanto:

$$u_c F_n = u_l$$

$$u_c = \beta u_{c'} [F_{k'} + (1 - \delta)]$$

- Ou:

$$u_{ct} F_{nt} = u_{lt}$$

$$u_{ct} = \beta u_{c,t+1} [F_{k,t+1} + (1 - \delta)]$$

Resolução da equação de Bellman

- Equação de Bellman

$$V(x) = \max_{x',u} \{ r(x,u) + \beta V(x') : x' = g(u) \}$$

- Incógnita da equação de Bellman é a função V
  - Se soubermos V, podemos resolver o problema de otimização e encontrar as **regras de decisão** (em função da variável de estado):

$$u = h(x)$$

$$x' = \phi(x)$$

- Dada uma condição inicial x_t, podemos então obter toda a trajetória futura das variáveis endógenas:

$$u_t = h(x_t), x_{t+1} = \phi(x_t)$$

$$u_{t+1} = h(x_{t+1}), x_{t+2} = \phi(x_{t+1})$$

$$\ldots$$

Resolução da equação de Bellman

- Encontrar ponto fixo = colocar uma função do lado direito, resolver o problema de otimização, e obter a mesma V

$$V(x) = \max_{x',u} \{r(x,u) + \beta V(x') : x' = g(u)\}$$

- Procedimento iterativo
  - Chutar $V^0(x)$
  - Resolver problema de otimização com $V^0(x)$

$$\max_{x',u} \{r(x,u) + \beta V^0(x') : x' = g(u)\}$$

  - Regras de decisão correspondentes (não são as verdadeiras):

$$u = h^0(x)$$

$$x' = \phi^0(x)$$

Resolução da equação de Bellman

- Atualizar chute:

$$V^1(x) = r(x, h^0(x)) + \beta V^0(\phi^0(x))$$

- Refazer procedimento com chute atualizado $V^1(x)$
  - Resolver problema de otimização

$$\max_{x',u} \{ r(x,u) + \beta V^1(x') : x' = g(u) \}$$

  - Regras de decisão correspondentes:

$$u = h^1(x)$$

$$x' = \phi^1(x)$$

  - Atualizar:

$$V^2(x) = r(x, h^1(x)) + \beta V^1(\phi^1(x))$$

Resolução da equação de Bellman

- De modo geral, na j-ésima iteração, com a função $V^j(x)$, resolver

$$\max_{x',u} \left\{ r(x,u) + \beta V^j(x') : x' = g(u) \right\}$$

  - Regras de decisão correspondentes:

$$u = h^j(x)$$

$$x' = \phi^j(x)$$

  - Atualizar:

$$V^{j+1}(x) = r(x, h^j(x)) + \beta V^j(\phi^j(x))$$

Resolução da equação de Bellman

- Com isso obtemos sequências $\{V^j(x)\}$, $\{h^j(x)\}$, $\{\phi^j(x)\}$
- Sob certas condições (a serem discutidas):

$$\{V^j(x)\} \rightarrow V(x) \text{ [ponto fixo único]}$$

- Além disso:

$$\{h^j(x)\} \rightarrow h(x)$$

$$\{\phi^j(x)\} \rightarrow \phi(x)$$

Alguns resultados importantes

- Definições:
  - X: espaço de funções
  - $x \in X$ tem domínio limitado e imagem nos reais, i.e., $x : [0, \bar{t}] \to \mathbb{R}$
  - Norma do sup; distância entre funções é a maior distância ponto a ponto. Para $x, y \in X$:

$$d(x,y) = \sup_{0 \leq t \leq \bar{t}} |x(t) - y(t)|$$

- **Contração**: Seja (X, d) um espaço métrico e $f : X \to X$. f é uma **contração** se existe $k \in \mathbb{R}$, $0 \leq k < 1$, tal que:

$$d[f(x), f(y)] \leq kd(x,y), \text{para todo } x, y \in X$$

Teorema

- Seja (X, d) um espaço métrico <u>completo</u> e $f : X \to X$ uma <u>contração</u>. Então existe um único ponto fixo $f(x_0) = x_0$.
- Além disso, se $x \in X$ é um ponto qualquer e $\{x_n\}$ é definida de acordo com:

$$
\begin{aligned}
x_1 &= f(x) \\
x_2 &= f(x_1) \\
&\dots \\
x_{n+1} &= f(x_n)
\end{aligned}
$$

  - Então $\{x_n\} \to x_0 \in X$

Prova

- Primeiro note que $x_n = f(x_{n-1}) = f(f(x_{n-2})) = f^2(x_{n-2})$
  - Logo: $x_n = f^m(x_{n-m})$ e $x_n = f^n(x)$
- Para dados n, m, com $n > m$:

$$
\begin{aligned}
d(x_n, x_m) &= d[f^n(x), f^m(x)] = d[f^m(x_{n-m}), f^m(x)] \\
&= d[f(f^{m-1}(x_{n-m})), f(f^{m-1}(x))] \leq kd[f^{m-1}(x_{n-m}), f^{m-1}(x)]
\end{aligned}
$$

  - Dado que f é uma contração $(0 \leq k < 1)$
- De maneira similar:

$$
\begin{aligned}
d[f^{m-1}(x_{n-m}), f^{m-1}(x)] &= d[f(f^{m-2}(x_{n-m})), f(f^{m-2}(x))] \\
&\leq kd[f^{m-2}(x_{n-m}), f^{m-2}(x)]
\end{aligned}
$$

- Logo:

$$
d(x_n, x_m) \leq k^2 d[f^{m-2}(x_{n-m}), f^{m-2}(x)]
$$

Prova

- Repetindo o procedimento m vezes:

$$d(x_n, x_m) \leq k^m d(x_{n-m}, x) \; (*)$$

  - Guarde essa expressão

- Usando a desigualdade triangular sucessivamente:

$$\begin{aligned} d(x_n, x_m) &\leq k^m d(x_{n-m}, x) \\ &\leq k^m [d(x_{n-m}, x_{n-m-1}) + d(x_{n-m-1}, x)] \\ &\leq k^m [d(x_{n-m}, x_{n-m-1}) + d(x_{n-m-1}, x_{n-m-2}) + d(x_{n-m-2}, x)] \end{aligned}$$

- Repetindo procedimento m vezes:

$$d(x_n, x_m) \leq k^m [d(x_{n-m}, x_{n-m-1}) + d(x_{n-m-1}, x_{n-m-2}) + \ldots + d(x_1, x)]$$

Prova

- Recupere expressão (*), trocando n, m por i, j:

$$d(x_i, x_j) \leq k^j d(x_{i-j}, x) \ (*)$$

  - Para $i = n - m, j = n - m - 1 : d(x_{n-m}, x_{n-m-1}) \leq k^{n-m-1} d(x_1, x)$
  - Para $i = n - m - 1, j = n - m - 2 : d(x_{n-m-1}, x_{n-m-2}) \leq k^{n-m-2} d(x_1, x)$
  - E assim por diante

- Lembrando que:

$$\begin{aligned} d(x_n, x_m) &\leq k^m [d(x_{n-m}, x_{n-m-1}) + d(x_{n-m-1}, x_{n-m-2}) + ... + d(x_1, x)] \\ &\leq k^m [k^{n-m-1} d(x_1, x) + k^{n-m-2} d(x_1, x) + ... + d(x_1, x)] \\ &= k^m (1 + k + k^2 + ... + k^{n-m-1}) d(x_1, x) \end{aligned}$$

- Substituindo a soma finita por uma soma infinita:

$$d(x_n, x_m) \leq k^m (1 + k + k^2 + ...) d(x_1, x) = \frac{k^m}{1 - k} d(x_1, x)$$

Prova

- Assim:

$$d(x_n, x_m) \leq \frac{k^m}{1-k} d(x_1, x) \to 0, \text{ quando } n, m \to \infty$$

  - Portanto, $\{x_n\}$ é uma sequência de Cauchy
  - Como o espaço métrico é completo, $\{x_n\} \to x_0 \in X$
- Lembre que $x_{n+1} = f(x_n)$. No limite, $x_{n+1} = x_n = x_0$.
  - Logo $x_0 = f(x_0)$. Ou seja, x_0 é um ponto fixo.
- Mostrar que ponto fixo é único.
  - Suponha que há dois pontos fixos, x_0, y_0, com $x_0 \neq y_0$. Então:

$$\begin{aligned} 0 < d(x_0, y_0) &= d[f(x_0), f(y_0)] \\ &\leq k d(x_0, y_0) < d(x_0, y_0) \end{aligned}$$

  - Temos, portanto, uma contradição. O ponto fixo é único.

Condições suficientes de Blackwell

- Condições para que um operador T seja uma contração
  - X : espaço de funções
  - Norma do sup:

$$d(x,y) = \sup_{t \in D} |x(t) - y(t)|, x, y \in X$$

  - Operador $T : (X,d) \to (X,d)$

- Considere as seguintes propriedades:
  1. (Monotonicidade) Para todo $x, y \in X$, se $x \geq y$, então $T(x) \geq T(y)$
  2. ("Discounting") Seja **c** uma função constante igual a $c \in \mathbb{R}$, para todo os pontos no domínio das funções em X. Para todo $c > 0$ e todo $x \in X$:

$$T(x + \mathbf{c}) \leq T(x) + \beta c, \text{ para algum } \beta \in [0,1)$$

- Se T satisfizer as duas propriedades, então T é uma contração

Prova

- Observação: $x \geq y$ significa que $x(t) \geq y(t)$ para todo $t \in D$
- Para $x, y \in X$:

$$d(x,y) = \sup_{t \in D} |x(t) - y(t)| \geq |x(t) - y(t)|, \forall t \in D$$
$$\geq x(t) - y(t), \forall t \in D$$

  - Logo $x(t) \leq y(t) + d(x,y), \forall t \in D$; ou $x \leq y + d(x,y)$
- Usando Propriedade 1:

$$T(x) \leq T(y + d(x,y))$$

- E pela Propriedade 2, com $c = d(x,y)$:

$$T(y + d(x,y)) \leq T(y) + \beta d(x,y)$$

- Portanto:

$$T(x) - T(y) \leq \beta d(x,y)$$

Prova

- Repetindo mesmo procedimento, mas trocando x e y

$$d(x,y) = d(y,x) = \sup_{t \in D} |y(t) - x(t)| \geq |x(t) - y(t)|, \forall t \in D$$
$$\geq y(t) - x(t), \forall t \in D$$

  - Logo $y(t) \leq x(t) + d(x,y), \forall t \in D$; ou $y \leq x + d(x,y)$
- Usando Propriedade 1:

$$T(y) \leq T(x + d(x,y))$$

- E pela Propriedade 2, com $c = d(x,y)$:

$$T(x + d(x,y)) \leq T(x) + \beta d(x,y)$$

- Portanto:

$$T(y) - T(x) \leq \beta d(x,y)$$

Prova

- Temos:

$$T(x) - T(y) \leq \beta d(x,y) \Leftrightarrow T(x(t)) - T(y(t)) \leq \beta d(x,y), \forall t \in D$$

$$T(y) - T(x) \leq \beta d(x,y) \Leftrightarrow T(y(t)) - T(x(t)) \leq \beta d(x,y), \forall t \in D$$

- Segue então que:

$$|T(x(t)) - T(y(t))| \leq \beta d(x,y), \forall t \in D$$

- Portanto:

$$\sup_{t \in D} |T(x(t)) - T(y(t))| \leq \beta d(x,y)$$

- O lado esquerdo é a própria distância entre $T(x)$ e $T(y)$. Logo, T é uma contração

$$d(T(x), T(y)) \leq \beta d(x,y), \beta \in [0,1)$$

Voltando à equação de Bellman

- Procedimento iterativo:
  - Aplicar sucessivamente a mesma transformação em uma função
  - Encontrar ponto fixo
- Dado um chute inicial $V^0(x)$ para a função valor
  - Na iteração j, com a função $V^j(x)$, o operador T consiste em:

$$V^{j+1}(x) = T[V^j(x)] = \max_{u,x'} \{ r(x,u) + \beta V^j(x') : x' = g(u) \}$$

- Se T é uma contração, em um espaço métrico completo, para um dado chute inicial $V^0(x)$:

$$\{ T^n[V^0(x)] \} \to V(x), \text{ quando } n \to \infty$$

  - Ponto fixo único
  - V pode ser obtida da seguinte forma:

$$V^1(x) = T[V^0(x)] = \max_{u,x'} \{ r(x,u) + \beta V^0(x') : x' = g(u) \}$$

$$V^2(x) = T[V^1(x)] = \max_{u,x'} \{ r(x,u) + \beta V^1(x') : x' = g(u) \}$$

$$\ldots$$

Voltando à equação de Bellman

- Agora mostraremos que T satisfaz as condições suficientes de Blackwell

1. Monotonicidade: Sejam duas funções $V^0(x)$ e $W^0(x)$, com $V^0(x) \geq W^0(x)$ para todo $x \in D$. Então:

$$r(x,u) + \beta V^0(x') \geq r(x,u) + \beta W^0(x')$$

$$\max_{x',u} \{ r(x,u) + \beta V^0(x') : x' = g(u) \} \geq$$
$$\max_{x',u} \{ r(x,u) + \beta W^0(x') : x' = g(u) \}$$

$$T[V^0(x)] \geq T[W^0(x)]$$

2. "Discounting": Seja $\mathbf{c}$ uma função constante igual a $c > 0$. Dada a função $V^0(x)$

$$\begin{aligned} T[V^0(x) + c] &= \max_{x',u} \{ r(x,u) + \beta [V^0(x') + c] : x' = g(u) \} \\ &= \max_{x',u} \{ r(x,u) + \beta V^0(x') : x' = g(u) \} + \beta c \end{aligned}$$

$$T[V^0(x) + c] = T[V^0(x)] + \beta c$$

Voltando à equação de Bellman

- T satisfaz ambas propriedades
  - T é uma contração
- Se estivermos em um espaço métrico completo, $\{T^n[V^0(x)]\} \to V(x)$
- X : espaço de funções, com as seguintes propriedades:
  - $D \subseteq \mathbb{R}^l$
  - $f \in X$ se f contínua, limitada e $f : D \to \mathbb{R}$
  - Norma do sup: $d(x,y) = \sup_{t \in D} \left| x(t) - y(t) \right|$
- Espaço métrico (X, d) é completo (ver Stokey & Lucas, p.47)

Outras propriedades

- Além disso, se r côncava, V também será côncava

$$V(x) = \max_{x',u} \left\{ r(x,u) + \beta V(x') : x' = g(u) \right\}$$

- Se r e g forem difereciáveis, então V também será difereciável nos pontos interiores do domínio
  - Derivada dada pelo Teorema do Envelope
- **Condição de Benveniste-Scheinkman**: se r e g são diferenciáveis e x_0 é um ponto interior do domínio D, então:
  - V é diferenciável em x_0
  - Derivada dada por:

$$V'(x_0) = \frac{\partial r}{\partial x}(x_0, h(x_0))$$

Condição de Benveniste-Scheinkman

- Ideia:

$$V(x) = \max_{x',u} \left\{ r(x,u) + \beta V(x') : x' = g(u) \right\}$$
$$= r(x, h(x)) + \beta V(\phi(x))$$

- Em particular, $V(x_0) = r(x_0, h(x_0)) + \beta V(\phi(x_0))$
- Agora perturbe x ao redor de x_0, sem alterar as escolhas ótimas $h(x_0)$ e $\phi(x_0)$, para obter a função W

$$W(x) = r(x, h(x_0)) + \beta V(\phi(x_0))$$

  - Isso claramente não é ótimo, caso contrário seria a própria função V
- Dado que r é diferenciável, $W(x)$ também será. Derivada dada por:

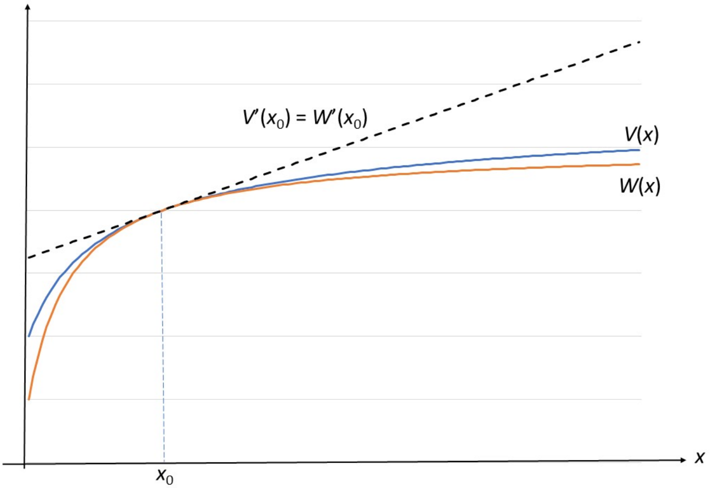
$$W'(x) = \frac{\partial r}{\partial x}(x, h(x_0))$$

Condição de Benveniste-Scheinkman

- Além disso, $W(x) \leq V(x)$, com igualdade se $x = x_0$
  - W tangencia V em x_0
- Portanto
$$V'(x_0) = W'(x_0) = \frac{\partial r}{\partial x}(x_0, h(x_0))$$

Condição de Benveniste-Scheinkman

Figure: Condição de Benveniste-Scheinkman



Exemplo

- Modelo neoclássico de crescimento:
  - Preferência: log no consumo, sem utilidade com relação ao lazer $(n_t = 1)$
  $$U = \sum_{t=0}^{\infty} \beta^t \ln c_t$$
  - Depreciação completa: $\delta = 1$
  - Tecnologia Cobb-Douglas: $y_t = k_t^{\alpha} n_t^{1-\alpha}$
  - Restrição de recursos em t:
  $$c_t + k_{t+1} \leq k_t^{\alpha}$$
- Problema tem solução fechada
- Equação de Bellman:

$$V(k) = \max_{k'} \left\{ \ln(k^{\alpha} - k') + \beta V(k') \right\}$$

Exemplo

- Chute inicial: $V^0(k) = 0$
- Iteração 1

$$\max_{k'}\{\ln(k^\alpha - k') + \beta V^0(k')\} = \max_{k'}\{\ln(k^\alpha - k')\}$$

  - Regra de decisão correspondente:

$$k' = \phi^0(k) = 0$$

  - Atualização:

$$V^1(k) = \ln(k^\alpha) = \alpha \ln k$$

- Como $V^1(k) \neq V^0(k)$, não houve convergência.
  - Refazer procedimento usando $V^1(k)$ como novo chute

Exemplo

- Iteração 2:

$$\max_{k'}\{\ln(k^{\alpha}-k')+\beta V^{1}(k')\} = \max_{k'}\{\ln(k^{\alpha}-k')+\beta\alpha\ln k'\}$$

  - Regra de decisão correspondente:

$$-\frac{1}{k^{\alpha}-k'}+\beta\alpha\frac{1}{k'}=0$$

$$k'=\phi^{1}(k)=\frac{\alpha\beta}{1+\alpha\beta}k^{\alpha}$$

  - Atualização:

$$V^{2}(k)=\ln\left(k^{\alpha}-\frac{\alpha\beta}{1+\alpha\beta}k^{\alpha}\right)+\alpha\beta\ln\left(\frac{\alpha\beta}{1+\alpha\beta}k^{\alpha}\right)$$

$$V^{2}(k)=A_{2}\ln k+B_{2}$$

  - Em que $A_2=\alpha(1+\alpha\beta)$ e B_2 são constantes
  - $V^2(k)\neq V^1(k)$; refazer procedimento usando $V^2(k)$ como novo chute

Exemplo

- Iteração 3:

$$\max_{k'}\{\ln(k^\alpha - k') + \beta V^2(k')\} = \max_{k'}\{\ln(k^\alpha - k') + \beta(A_2 \ln k' + B_2)\}$$

  - Regra de decisão correspondente:

$$-\frac{1}{k^\alpha - k'} + \beta A_2 \frac{1}{k'} = 0$$

$$k' = \phi^2(k) = \frac{\beta A_2}{1+\beta A_2} k^\alpha$$

  - Atualização:

$$V^2(k) = \ln\left(k^\alpha - \frac{\beta A_2}{1+\beta A_2} k^\alpha\right) + \beta\left[A_2 \ln\left(\frac{\beta A_2}{1+\beta A_2} k^\alpha\right) + B_2\right]$$

$$V^2(k) = A_3 \ln k + B_3$$

  - Em que $A_3 = \alpha(1+\beta A_2)$ e B_3 são constantes

Exemplo

- Função valor não convergiu, mas houve convergência na forma funcional
  - Nas próximas iterações, a forma funcional permanecerá a mesma, só se alterarão os coeficientes

- Iteração j

$$V^j(k) = A_j \ln k + B_j,\ \ k' = \phi^j(k) = \frac{\beta A_j}{1+\beta A_j} k^\alpha$$

- Iteração $j+1$

$$V^{j+1}(k) = A_{j+1} \ln k + B_j,\ \ k' = \phi^{j+1}(k) = \frac{\beta A_{j+1}}{1+\beta A_{j+1}} k^\alpha$$

  - Em que $A_{j+1} = \alpha(1+\beta A_j)$

Exemplo

- Quando $j \to \infty, \{V^j(k)\} \to V(k)$
  - Portanto: $\{A^j\} \to A$ e $\{B^j\} \to B$
  - $k' = \phi(k) = \frac{\beta A}{1+\beta A} k^\alpha$
- No limite, $A_{j+1} = A_j = A$
- Dado que $A_{j+1} = \alpha(1 + \beta A_j)$:

$$A = \alpha(1 + \beta A)$$

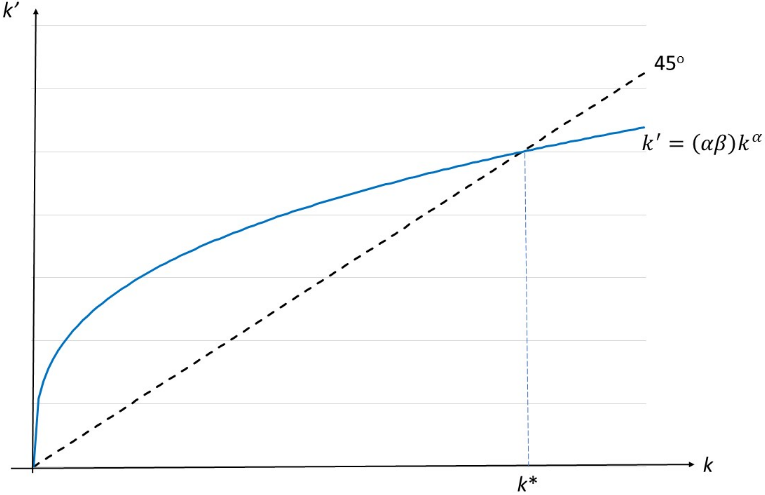
$$A = \frac{\alpha}{1 - \alpha\beta}$$

- Lei de movimento do capital:

$$k' = \frac{\beta A}{1 + \beta A} k^\alpha = (\alpha\beta) k^\alpha$$

Exemplo

Figure: Lei de movimento de k



Estratégia computacional

- Resolver função valor em um grid discreto
- Mesmo problema que anterior, porém com utilidade CRRA e depreciação incompleta ($0 < \delta < 1$)

$$U = \sum_{t=0}^{\infty} \beta^t \frac{c_t^{1-\gamma} - 1}{1-\gamma}$$

$$c_t + k_{t+1} - (1-\delta)k_t \leq k_t^{\alpha}$$

- Função valor:

$$V(k) = \max_{c,k'} \left\{ \frac{c^{1-\gamma} - 1}{1-\gamma} + \beta V(k') : c = k^{\alpha} + (1-\delta)k - k' \right\}$$

Estratégia computacional

- Equação de Euler:

$$c^{-\gamma} = \beta c'^{-\gamma}[\alpha k'^{\alpha-1} + (1-\delta)]$$

- Em estado estacionário:

$$c = c' = c^*$$

$$k = k' = k^*$$

- Logo:

$$1 = \beta[\alpha (k^*)^{\alpha-1} + (1-\delta)]$$

$$k^* = \left\{ \frac{1}{\alpha} \left[\frac{1}{\beta} - (1-\delta) \right] \right\}^{1/(\alpha-1)}$$

Algoritmo

- Passo 1 – montar grid discreto em que a função valor será calculada

$$k \in \{0+\varepsilon, 0+2\varepsilon, ..., \bar{k}\} \equiv K$$

  - Quanto menor ε, maior a precisão; porém mais tempo leva para concluir procedimento
  - Usar k^* para determinar $\bar{k}$ (em nosso caso, $\bar{k} = 1.5k^*$)
  - nk = número de pontos no grid

Algoritmo

- Passo 2 – para cada para $(k,k') \in K \times K$, computar:

$$c(k,k') = k^{\alpha} + (1-\delta)k - k'$$

  - Obter utilidade instantânea:

$$u\left(c(k,k')\right) = \begin{cases} \frac{c(k,k')^{1-\gamma}-1}{1-\gamma}, & \text{se } c(k,k') > 0 \\ -\infty, & \text{se } c(k,k') \leq 0 \end{cases}$$

  - Estocar resultados na matriz u $(nk \times nk)$
    - Linha = k'; coluna = k

Algoritmo

- Passo 3 – Processo iterativo:
  - Chutar $V^0(k)$
    - Vetor v, de dimensão $1 \times nk$ (conjunto de valores, um para cada ponto no grid)
  - Computar:
    $$W^0(k,k') = u\left(c(k,k')\right) + \beta V^0(k')$$
  - Estocar na matriz W $(nk \times nk)$
    - Para montar matriz correspondente a $V^0(k')$, transpor vetor v e replicar nk vezes
    - Transposta serve para obter um vetor coluna, que varia apenas na dimensão de k'
    - Isso gera matriz $nk \times nk$, que pode ser somada à matriz u

Algoritmo

- Passo 4 – Maximizar e atualizar chute

$$V^1(k) = \max_{k'} \left\{ W^0(k, k') \right\}$$

  - Para cada k, encontrar k' ótimo
  - Maximizar dentro de cada coluna da matriz W
  - Isso gerará dois vetores, de dimensão $1 \times nk$
    - *vpr* – valores que maximizam cada coluna da matriz W
    - *ikpr* – posição dos elementos que maximizam cada coluna da matriz W
  - *vpr* será o novo chute, na próxima rodada do procedimento iterativo
  - Com *ikpr* é possível obter as regras de decisão correspondentes
    - Buscar posições do vetor que define grid

Algoritmo

- Passo 5 – Testar convergência
  - Distância entre função antiga e função nova:

$$d = \max_{k \in K} \{ V^1(k) - V^0(k) \}$$

    - Maior distância ponto a ponto no domínio de k (consistente com a norma do sup)
  - Para um número pequeno $\tau > 0$
    - Se $d \leq \tau$, parar (convergência alcançada)
    - Se $d > \tau$, repetir passo 4 com chute atualizado