

# Monitoria 1

## MODELO DE SEARCH, CRESCIMENTO E ÁRVORE DE LUCAS

Macroeconomia III

EPGE - FGV

30 de outubro de 2017

# Modelo de Search no Mercado de Trabalho

McCall 1970

## Desemprego?

- ▶ Excesso de demanda
- ▶ Fricções no mercado de trabalho

## Enviroment:

- ▶ Em  $t = 0$  trabalhador encontra-se desempregado
- ▶ Enquanto desempregado recebe oferta de trabalho  $w$ ,  $w \sim^{iid} F(\cdot)$ , definida em  $[0, \bar{w}]$
- ▶ Trabalhador decide se aceita ou não o salário  $w$ 
  - ▶ Se não aceita, recebe transferência  $b$ , e procurará nova oferta no próximo período.
  - ▶ Se aceita  $w$ , tem renda  $w$  enquanto se mantém empregado, podendo ser despedido com probabilidade  $\pi$  ao final de cada período

Monitoria 1

Macroeconomia III

Modelo de Search  
no Mercado de  
Trabalho

Modelo Clássico de  
Crescimento

Árvore de Lucas

Equity Premium  
Puzzle

Sistemas lineares

Método da  
Bisseção

Método de  
Newton-Raphson

Lucas's  
signal-extraction  
model

Imrohorglu

Huggett's Model

Aiyagari's Model

# Modelo de Search no Mercado de Trabalho

McCall 1970

- ▶ se não aceita  $w$ 
  - ▶ ele procura nova oferta  $w'$ :  $w'$  sorteada segundo  $f : [0, \bar{w}] \rightarrow \mathbb{R}_{++}$ .
  - ▶ recebe uma transferência  $b$  neste período
- ▶ se aceita  $w$ 
  - ▶ ele trabalha e auferir renda  $w$  neste período
  - ▶ com probabilidade  $\pi$  ele é despedido no período seguinte
  - ▶ caso seja despedido, ele começa período seguinte com  $w' = 0$
  - ▶ caso não seja despedido, ele começa período seguinte com  $w' = w$

Monitoria 1

Macroeconomia III

Modelo de Search  
no Mercado de  
Trabalho

Modelo Clássico de  
Crescimento

Árvore de Lucas

Equity Premium  
Puzzle

Sistemas lineares

Método da  
Bisseção

Método de  
Newton-Raphson

Lucas's  
signal-extraction  
model

Imrohorglu

Huggett's Model

Aiyagari's Model

# Modelo de Search no Mercado de Trabalho

Preferências representadas por:

$$U(\{c_t\}_{t=0}^{\infty}) = E \left\{ \sum_{t=0}^{\infty} \beta^t u(c_t) \right\}$$

em que

- ▶  $\beta \in (0, 1)$ .
- ▶  $u : \mathbb{R}_+ \rightarrow \mathbb{R}$ .
- ▶  $u(0) = 0$ .
- ▶  $\lim_{c \rightarrow 0} u'(c) = \infty$ .
- ▶  $u'(c) > 0$ ,  $u''(c) < 0 \ \forall c \geq 0$ .

Não existe possibilidade de empréstimo ou poupança. Não há *recall*

# Modelo de Search no Mercado de Trabalho

## Formulação Recursiva

Se o trabalhador aceita  $w$  hoje e **segue a política ótima a partir de amanhã**, então aufere

$$u(w) + \beta [(1 - \pi)v(w) + \pi v(0)]$$

Se o trabalhador escolhe procurar nova oferta hoje e **segue a política ótima a partir de amanhã**, então aufere

$$u(b) + \beta E [v(w')] = u(b) + \beta \int_0^{\bar{w}} v(w') f(w') dw'$$

# Modelo de Search no Mercado de Trabalho

## Formulação Recursiva

Portanto

$$v(w) = \max \{ u(w) + \beta [(1 - \pi)v(w) + \pi v(0)] , \\ u(b) + \beta \int_0^{\bar{w}} v(w') f(w') dw' \}$$

É possível mostrar que  $\exists! v$ , contínua e limitada, que satisfaz a equação funcional acima e é solução do problema sequencial.

Monitoria 1

Macroeconomia III

Modelo de Search  
no Mercado de  
Trabalho

Modelo Clássico de  
Crescimento

Árvore de Lucas

Equity Premium  
Puzzle

Sistemas lineares

Método da  
Bisseção

Método de  
Newton-Raphson

Lucas's  
signal-extraction  
model

Imrohoroglu

Huggett's Model

Aiyagari's Model

# Modelo de Search no Mercado de Trabalho

## Formulação Recursiva

Note que,

$$U = u(b) + \beta \int_0^{\bar{w}} v(w') f(w') dw' = v(0)$$

constante.

Pela continuidade de  $v$ ,  $\exists R$  tal que:

$$u(R) + \beta [(1 - \pi)v(R) + \pi U] = U$$

Assim,

$$v(w) = I(w) [u(w) + \beta [(1 - \pi)v(w) + \pi U]] + (1 - I(w)) U$$

onde  $I(w) = 1$  se  $w \geq R$  e  $I(w) = 0$  c.c.

# Modelo de Search no Mercado de Trabalho

Nota sobre integração numérica

Integração trapezóide:

Utilizando uma aproximação lagrangeana de  $F(x)$  é possível mostrar que

$$\int_a^b F(x)dx \simeq \frac{b-a}{2}(F(a) + F(b)). \quad (1)$$

Quando  $a$  e  $b$  estiverem próximos o bastante a eq. (1) será uma boa aproximação.

Logo basta fazer uma partição fina o bastante do intervalo  $[a, b]$ . Por exemplo, podemos dividir  $[a, b]$  em  $n$  intervalos de mesmo tamanho  $h = (b - a)/n$ .

$$I_n = \frac{h}{2} \left[ F(a) + F(b) + 2 \sum_{i=1}^{n-1} F(x_i) \right]. \quad (2)$$



# Modelo de Search no Mercado de Trabalho

## Nota sobre integração numérica

### Algoritmo:

1. calcule  $h = (b - a)/n$
2. monte um grid:  $x_i = a + hi, i \in \{0, \dots, n\}$
3. compute  $I_n = \frac{h}{2} \left[ F(x_0) + F(x_n) + 2 \sum_{i=1}^{n-1} F(x_i) \right]$

# Modelo de Search no Mercado de Trabalho

## Algoritmo

1. Carregamos todos os parâmetros e funções que o *environment* fornece.
2. Definimos um grid para a variável de estado:  $w$ .
3. Criamos chutes iniciais para  $V$  e  $G$ , respectivamente função valor e função política.
4. Definimos limites de tolerância para nosso código:  $\varepsilon$  pequeno e  $itmax$  grande.
5. Calculamos o payoff de não aceitar a oferta e guardamos em uma variável,  $N$ .
6. Para cada valor do grid de  $w$  calculamos o payoff de aceitar a oferta e guardamos em um vetor,  $A$ .
7. Para cada valor do grid de  $w$  calculamos a nova função valor  $TV = \max\{N, A\}$  e guardamos a função política em  $G$ .
8. Calculamos  $d = |TV - V|$ , atualizamos  $V$  ( $V = TV$ ).
9. Se  $d < \varepsilon$  ou as iterações chegaram a  $itmax$  paramos o código, caso contrário voltamos ao passo 5.

# Modelo de Search no Mercado de Trabalho

## Exemplo

Considere o seguinte exemplo:

►  $(\beta, \pi, \bar{w}, b) = (0.9, 0.3, 10, 0).$

►  $u(x) = \sqrt{x}.$

►  $w \sim U([0, \bar{w}]).$

# Modelo Clássico de Crescimento

## Problema do Planejador

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

restrito ao conjunto definido por

$$\begin{aligned} c_t &\geq 0, k_{t+1} \geq 0 \quad , \quad \forall t \geq 0 \\ c_t + k_{t+1} &\leq f(k_t) + (1 - \delta)k_t \quad , \quad \forall t \geq 0 \\ k_0 &\text{ dado} \end{aligned}$$

- Suponha que  $\delta = 1$

# Modelo Clássico de Crescimento

## Problema do Planejador

Reescrevendo o problema

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

restrito ao conjunto definido por

$$k_{t+1} \in \Gamma(k_t) = [0, f(k_t)] \quad , \quad \forall t \geq 0$$

$k_0$  dado

# Modelo Clássico de Crescimento

## Formulação Recursiva

Reescrevendo o problema

$$v(k) = \max_{k' \in \Gamma(k)} \{u(f(k) - k') + \beta v(k')\}$$

em que  $\Gamma(k) = [0, f(k)]$ .

$k' = g(k)$  função política.

# Modelo Clássico de Crescimento

## Algoritmo

1. Carregamos todos os parâmetros e funções do *environment*.
2. Definimos um grid para a variável de estado:  $k$ , capital.
3. Criamos chutes iniciais para  $V$  e  $Gk$  e  $Gc$ , respectivamente função valor e função políticas. Também definimos  $TV$ .
4. Definimos limites de tolerância para nosso código:  $\varepsilon$  pequeno e  $itmax$  grande. Declaramos também um erro grande inicial  $d = 1$ , e iteração  $it = 0$ .
5. Enquanto  $d < \varepsilon$  e  $it \leq itmax$ .
6. Para cada valor de estado  $k$ 
  - ▶ para cada  $k'$  do grid, computamos  $c$  e  $u(c) + \beta V(k')$ .
  - ▶ Dentre todos os  $k'$ , calculamos  $TV$  (máximo entre todos). E guardamos a função política em  $G$ .
  - ▶ Calculamos  $d = |TV - V|$ , atualizamos  $V$  ( $V = TV$ ).
  - ▶ Se  $d < \varepsilon$  ou as iterações chegaram a  $itmax$  paramos o código, caso contrário voltamos ao passo 5.
7. Uma vez que convirja, achamos  $V^*(k)$  tq  $TV^* = V^*$ . Basta recuperar as respectivas funções políticas de acordo com a posição guardada no loop das iterações.

# Modelo Clássico de Crescimento Estocástico

## Problema do Planejador

Consideramos agora  $f(k_t, z_t) = z_t k_t^\alpha$ , em que  $z_t$  é um processo estocástico que segue uma cadeia de Markov tal que  $P(z_t = \bar{z} | z_{t-1} = \bar{z}) = \xi$  e  $P(z_t = \underline{z} | z_{t-1} = \underline{z}) = \zeta$ .

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

restrito ao conjunto definido por

$$\begin{aligned} c_t &\geq 0, k_{t+1} \geq 0 \quad , \quad \forall t \geq 0 \\ c_t + k_{t+1} &\leq f(k_t, z_t) + (1 - \delta)k_t \quad , \quad \forall t \geq 0 \\ k_0 &\text{ dado} \end{aligned}$$

- Suponha que  $\delta = 1$



# Modelo Clássico de Crescimento Estocástico

## Problema do Planejador

Reescrevendo o problema

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

restrito ao conjunto definido por

$$k_{t+1} \in \Gamma(k_t, z_t) = [0, f(k_t, z_t)] \quad , \quad \forall t \geq 0$$
$$k_0 > 0 \text{ dado}$$

# Modelo Clássico de Crescimento Estocástico

## Formulação Recursiva

Reescrevendo o problema

$$v(k, z) = \max_{k' \in \Gamma(k, z)} \left\{ u(f(k) - k') + \beta \sum_{z'} \pi_{zz'} v(k', z') \right\}$$

em que  $\Gamma(k, z) = [0, f(k, z)]$ .

$k' = g(k)$  função política.

## Enviroment:

- ▶ Economia de trocas, número grande de indivíduos, sem heterogeneidade (agente representativo)
- ▶ Um único ativo durável
- ▶ Possui uma única unidade do ativo (árvore),  $s_0 = 1$ .
- ▶ ativo não sofre depreciação e produz frutos (dividendos) a cada período que evoluem de acordo com um processo estocástico.
- ▶ frutos são perecíveis.

# Árvore de Lucas

Agentes:

- Preferências sobre plano de consumo  $c = \{c_t\}_{t=0}^{\infty}$ :

$$U(c) = \mathbb{E} \left[ \sum_{t=0}^{\infty} \beta^t u(c_t) \right]$$

com  $\beta \in (0, 1)$ ,  $u' > 0$ ,  $u'' < 0$ .

- os gastos dos agentes são restritos pela sua riqueza:

$$w_t = (p_t + x_t)s_t$$

que pode ser utilizada para adquirir mais unidades do ativo árvore.

Problema Sequencial:

$$\max_c \mathbb{E} \sum_t \beta^t u(c_t)$$

$$s.t. \quad c_t + p_t s_{t+1} \leq (p_t + x_t)s_t, \quad \forall t$$

$$c_t, s_{t+1} \geq 0 \quad s_0, x_0 \quad \text{dados}$$

# Árvore de Lucas

## Formulação Recursiva

Reescrevendo...

$$V(s, x) = \max_{c, s' \geq 0} u(c) + \beta E [V(s', x') | x]$$

$$s.t \quad c + p(x)s' \leq [p(x) + x]s$$

onde as variáveis de estado são  $(s, x)$ .

A solução será dada por uma função política  $s' = g(s, x)$ .

Condição de market clearing:  $g(s, x) = 1$ .

# Mehra and Prescott - Equity Premium Puzzle

Ambiente

- ▶ massa unitária de agentes com desconto  $\beta$
- ▶ uma árvore de Lucas ( $s$ )(ativo de risco) e títulos sem risco de um período ( $B$ )
- ▶ a árvore paga dividendos ( $y$ ) que crescem a uma taxa  $x$  (modificação do environment da árvore de Lucas - taxa de crescimento das dotações seguem um processo de Markov).
  - ▶  $x$  segue um processo de markov com  $n$  estados
  - ▶  $\pi(x', x) = P(x_{t+1} = x' | x_t = x)$
  - ▶ taxa de crescimento bruta dos dividendos:  $x' = \frac{y'}{y}$ .

Monitoria 1

Macroeconomia III

Modelo de Search  
no Mercado de  
Trabalho

Modelo Clássico de  
Crescimento

Árvore de Lucas

Equity Premium  
Puzzle

Sistemas lineares

Método da  
Bisseção

Método de  
Newton-Raphson

Lucas's  
signal-extraction  
model

Imrohoroglu

Huggett's Model

Aiyagari's Model

# Mehra and Prescott - Equity Premium Puzzle

Equação de Bellman

$$V(w, x, y) = \max_{s' \geq 0, B' \geq 0} u(c) + \beta \sum_{x'} V(w', x', y') \pi(x', x)$$

$$\text{sa } c + p(x, y)s' + q(x, y)B' \leq w$$

(r.o)

$$w' = [p(x', y') + y']s' + B'$$

$$y' = x'y$$

onde as duas últimas equações referem-se as leis de movimento.

Monitoria 1

Macroeconomia III

Modelo de Search  
no Mercado de  
Trabalho

Modelo Clássico de  
Crescimento

Árvore de Lucas

Equity Premium  
Puzzle

Sistemas lineares

Método da  
Bisseção

Método de  
Newton-Raphson

Lucas's  
signal-extraction  
model

Imrohoroglu

Huggett's Model

Aiyagari's Model

# Mehra and Prescott - Equity Premium Puzzle

## Equilíbrio

### Definition

Um equilíbrio competitivo recursivo é  $\{V, g_s, g_B, p, q\}$  tais que

- ▶ dados  $p$  e  $q$ ,  $V, g_s, g_B$  resolvem o problema de programação dinâmica dos agentes.
- ▶ Market Clearing

$$\begin{aligned}s' &= g_s(w, x, y) = 1 \\ B' &= g_B(w, x, y) = 0\end{aligned}$$

Monitoria 1

Macroeconomia III

Modelo de Search  
no Mercado de  
Trabalho

Modelo Clássico de  
Crescimento

Árvore de Lucas

Equity Premium  
Puzzle

Sistemas lineares

Método da  
Bisseção

Método de  
Newton-Raphson

Lucas's  
signal-extraction  
model

Imrohorglu

Huggett's Model

Aiyagari's Model



# Mehra and Prescott - Equity Premium Puzzle

## Resolvendo o Modelo

Em equilíbrio  $c = y$ . Então

$$p(x, y) = \beta \sum_{x'} \frac{u'(x'y)}{u'(y)} [p(x', x'y) + x'y] \pi(x', x)$$

Suponha que  $u(c) = c^{1-\sigma}/(1-\sigma)$  e  $p(x_i, y) = p_i y$  para todo  $i$ , temos

$$p_i = \beta \sum_{j=1}^n x_j^{1-\sigma} (p_j + 1) \pi(x_j, x_i)$$

Monitoria 1

Macroeconomia III

Modelo de Search  
no Mercado de  
Trabalho

Modelo Clássico de  
Crescimento

Árvore de Lucas

**Equity Premium  
Puzzle**

Sistemas lineares

Método da  
Bisseção

Método de  
Newton-Raphson

Lucas's  
signal-extraction  
model

Imrohoroglu

Huggett's Model

Aiyagari's Model

# Mehra and Prescott - Equity Premium Puzzle

## Resolvendo o Modelo

Ou seja, temos um sistema com  $n$  equações para encontrar  $n$  preços.

- ▶ Definindo a matriz  $n \times n$   $\mathbf{A}$  em que
  - ▶  $a_{ij} = \beta x_j^{1-\sigma} \pi(x_j, x_i)$
- ▶ o vetor  $\mathbf{n}$   $n \times 1$  em que
  - ▶  $b_i = \sum_{j=1}^n x_j^{1-\sigma} \pi(x_j, x_i)$
- ▶ podemos redefinir o sistema de equações como

$$\mathbf{p} = \mathbf{A}\mathbf{p} + \mathbf{b} \Rightarrow \quad (3)$$

$$(\mathbf{I} - \mathbf{A})\mathbf{p} = \mathbf{b} \Rightarrow \quad (4)$$

$$\mathbf{p} = (\mathbf{I} - \mathbf{A})^{-1}\mathbf{b} \quad (5)$$

se  $(\mathbf{I} - \mathbf{A})^{-1}$  existe.

# Mehra and Prescott - Equity Premium Puzzle

## Resolvendo o Modelo

para os títulos,

$$q(x, y) = \beta \sum_{x'} \frac{u'(x'y)}{u'(y)} \pi(x', x)$$

$$q(x_i, y) = \beta \sum_{j=1}^n x_j^{1-\sigma} \pi(x_j, x_i)$$

Monitoria 1

Macroeconomia III

Modelo de Search  
no Mercado de  
Trabalho

Modelo Clássico de  
Crescimento

Árvore de Lucas

**Equity Premium  
Puzzle**

Sistemas lineares

Método da  
Bisseção

Método de  
Newton-Raphson

Lucas's  
signal-extraction  
model

Imrohoroglu

Huggett's Model

Aiyagari's Model

# Mehra and Prescott - Equity Premium Puzzle

Retornos esperados  
para a árvore

$$\hat{r}^e(x_j, x_i) = \frac{p(x_j, x_j y) + x_j y - p(x_i, y)}{p(x_i, y)} \quad (6)$$

$$= \frac{p_j x_j + x_j - p_i}{p_i} \quad (7)$$

Então o retorno esperado condicionais

$$r^e(x_i) = \sum_{j=1}^n \hat{r}^e(x_j, x_i) \pi(x_j, x_i)$$

e o retorno incondicionais

$$\bar{r}^e = \sum_{i=1}^n r^e(x_i) \bar{\pi}(x_i)$$

$\bar{\pi}$  é a distribuição invariante da matriz de Markov.

# Mehra and Prescott - Equity Premium Puzzle

Retornos esperados

para os títulos

$$r^f(x_i) = \frac{1 - q(x_i, y)}{q(x_i, y)} = \frac{1 - q_i}{q_i},$$

e o retorno incondicionais

$$\bar{r}^f = \sum_{i=1}^n r^f(x_i) \bar{\pi}(x_i)$$

Então a média do **prêmio de risco** é

$$\bar{r}^e - \bar{r}^f$$

Monitoria 1

Macroeconomia III

Modelo de Search  
no Mercado de  
Trabalho

Modelo Clássico de  
Crescimento

Árvore de Lucas

Equity Premium  
Puzzle

Sistemas lineares

Método da  
Bisseção

Método de  
Newton-Raphson

Lucas's  
signal-extraction  
model

Imrohoroglu

Huggett's Model

Aiyagari's Model

# Mehra and Prescott - Equity Premium Puzzle

## Calibração

### Mehra and Prescott (1985)

- ▶  $n = 2$
- ▶  $x_1 = 1 + \mu - \delta, x_2 = 1 + \mu + \delta$
- ▶ Matriz de transição

$$\begin{pmatrix} \phi & 1 - \phi \\ 1 - \phi & \phi \end{pmatrix}$$

- ▶  $\mu$ : média do crescimento do consumo per capita
- ▶  $\delta$ : o desvio padrão do crescimento do consumo per capita
- ▶  $\phi$ : autocorrelação de primeira ordem do crescimento do consumo per capita ( $2\phi - 1$ )
- ▶  $\beta \in (0, 1)$  e  $\sigma \in [0, 10]$

# Sistemas Lineares

problema

Queremos resolver sistemas do tipo

$$Ax = b$$

em que

- ▶  $A$  é  $n \times n$
- ▶  $x$  é  $n \times 1$
- ▶  $b$  é  $n \times 1$

# Sistemas Lineares

## Método de Jacobi

### Sistema

$$a_{1,1}x_1 + a_{1,2}x_2 + \dots + a_{1,n}x_n = b_1$$

$$a_{2,1}x_1 + a_{2,2}x_2 + \dots + a_{2,n}x_n = b_2$$

$$\vdots$$

$$a_{n,1}x_1 + a_{n,2}x_2 + \dots + a_{n,n}x_n = b_n$$

Dado  $x$ , podemos calcular

$$x_1 = \frac{1}{a_{1,1}}(b_1 - a_{1,2}x_2 - a_{1,3}x_3 - \dots - a_{1,n}x_n)$$

$$x_2 = \frac{1}{a_{2,2}}(b_2 - a_{2,1}x_1 - a_{2,3}x_3 - \dots - a_{2,n}x_n)$$

$$\vdots$$

$$x_n = \frac{1}{a_{n,n}}(b_n - a_{n,1}x_1 - a_{n,2}x_2 - \dots - a_{n,n-1}x_{n-1})$$



# Sistemas Lineares

## Método de Jacobi

Assim, temos um algoritmo para computar a solução do sistema

- ▶ chute  $x_0$
- ▶ compute  $x$  de acordo com o sistema anterior
- ▶ se  $|x_0 - x| < \epsilon$  pare, temos a solução
- ▶ caso contrário, faça  $x_0 = x$  e volte ao segundo passo

# Sistemas Lineares

## Eliminação de Gauss

$$A = \begin{pmatrix} a_{1,1} & \dots & a_{1,n} \\ \vdots & \ddots & \vdots \\ a_{n,1} & \dots & a_{n,n} \end{pmatrix}, b = \begin{pmatrix} a_{1,n+1} \\ \vdots \\ a_{n,n+1} \end{pmatrix}$$

defina  $E_j = [a_{j,1}, \dots, a_{j,n}, a_{j,n+1}]$ .

Tomando o cuidado para que  $a_{1,1} \neq 0$  fazemos

$$(E_j - (a_{j,1}/a_{1,1})E_1) \rightarrow (E_j)$$

Teremos novas A e b, em que

$$A = \begin{pmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ 0 & a_{2,2} & \dots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n,2} & \dots & a_{n,n} \end{pmatrix}$$

# Sistemas Lineares

## Eliminação de Gauss

Podemos repetir um procedimento similar para as outras linhas. Para  $i = 2, \dots, n-1$

$$(E_j - (a_{j,i}/a_{i,i})E_i) \rightarrow (E_j) \text{ para } j = i+1, \dots, n$$

desde que  $a_{i,i} \neq 0$ , de modo que teremos novas A e b, em que A será triangular superior.

Podemos computar  $x_n = a_{n,n+1}/a_{n,n}$  e recursivamente

$$x_i = \frac{a_{i,n+1} - \sum_{j=i+1}^n a_{i,j}x_j}{a_{i,i}}$$

para  $j = n-1, n-2, \dots, 1$ .

# Método da Bisseção

Um método numérico que permite encontrar  $x$  tal que  $f(x) = 0$ .

## Theorem

*Seja  $A \subseteq \mathbb{R}$ . Se  $f : A \rightarrow \mathbb{R}$  é contínua e existem  $\underline{x} \in A$  e  $\bar{x} \in A$  tais que:*

- ▶  $\underline{x} < \bar{x}$
- ▶  $f(\underline{x})f(\bar{x}) < 0$

*então  $\exists \tilde{x} \in (\underline{x}, \bar{x})$  tal que  $f(\tilde{x}) = 0$ . Adicionalmente, se  $f'(x) < 0$ , para todo  $x \in A$ , então tal solução é única.*

- ▶ Podemos então utilizar esse método para otimizar funções cujo ótimo é interior.
- ▶ Para tanto basta definir  $f(\tilde{x}) = \frac{\partial F(\tilde{x})}{\partial x}$ , em que  $F(\cdot)$  é a função que queremos otimizar.
- ▶ Então, basta aplicar o teorema que acabamos de aprender e encontrar  $\tilde{x}$  tal que  $f(\tilde{x}) = 0$ .

# Método da Bisseção

## Algoritmo

(i) Encontre  $\underline{x}$  e  $\bar{x}$  tais que  $f(\underline{x})f(\bar{x}) < 0$ .

Obs.: sabe-se que  $\tilde{x} \in (\underline{x}, \bar{x})$

(ii) Defina  $x_m = \frac{\bar{x} + \underline{x}}{2}$

(iii) Calcule  $f(x_m)$

- ▶ se  $f(x_m) < 0$ , faça  $\bar{x} = x_m$
- ▶ se  $f(x_m) > 0$ , faça  $\underline{x} = x_m$

(iv) Defina  $x'_m = \frac{\bar{x} + \underline{x}}{2}$

(v) Calcule  $\delta_x = |x'_m - x_m|$  e  $\delta_f = |f(x'_m)|$

(vi) Defina  $\delta \equiv \max\{\delta_x, \delta_f\}$

- ▶ se  $\delta > 0$  retorne ao passo (iii).
- ▶ se  $\delta = 0$ , então  $x_m$  é a solução

# Método da Bisseção

## Propriedades

- (i) Convergência garantida.
- (ii) No entanto, como o método apenas encontra as raízes de uma função, não há garantia de que a solução encontrada é única.

# Método de Newton-Raphson

- ▶ Permite encontrar  $x \in \mathbb{R}$  tal que  $f(x) = 0$ .
- ▶ Sejam  $A$  e  $B$  subconjuntos de  $\mathbb{R}$ .
- ▶ E  $f : A \rightarrow B$  uma função diferenciável.

Usando expansão de Taylor tem-se

$$f(x) \approx f(x_0) + f'(x_0)[x - x_0]$$



# Método de Newton-Raphson

Uma estimativa de  $x$  tal que  $f(x) = 0$  é  $\bar{x}$  tal que

$$f(x_0) + f'(x_0)[\bar{x} - x_0] = 0$$

e portanto

$$\bar{x} = x_0 - \frac{f(x_0)}{f'(x_0)}$$

# Método de Newton-Raphson

## Algoritmo

(i) Defina  $i = 0$  e escolha  $\alpha \in \mathbb{R}$ . Faça  $x_i = \alpha$

(ii) Estime  $x_{i+1}$  segundo

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

(iii) Calcule  $\delta_x = |x_{i+1} - x_i|$  e  $\delta_f = |f(x_{i+1})|$

(iv) Defina  $\delta \equiv \max\{\delta_x, \delta_f\}$

- ▶ se  $\delta > 0$ , faça  $i = i + 1$  e retorne ao passo (ii)
- ▶ se  $\delta = 0$ , então  $x_i$  é a solução

# Método de Newton-Raphson

## Propriedades

(i) Convergência não garantida.

- ▶ se restringirmos a busca num intervalo fechado  $[a, b]$  garantimos a convergência do método.

(ii) No entanto, como o método apenas encontra as raízes de uma função, não há garantia de que a solução encontrada é única.

# Método de Newton multivariado

## Notação

$$x \in \mathbb{R}^n, f : \mathbb{R}^n \rightarrow \mathbb{R}^m$$

$$x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}; f(x) = \begin{pmatrix} f^1(x) \\ \vdots \\ f^m(x) \end{pmatrix};$$

$$J(x) = \begin{pmatrix} \frac{\partial f^1}{\partial x_1} & \cdots & \frac{\partial f^1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f^m}{\partial x_1} & \cdots & \frac{\partial f^m}{\partial x_n} \end{pmatrix}.$$

# Método de Newton multivariado

## Problema

Queremos encontrar  $x^*$  tal que  $f(x^*) = 0$ .

Podemos fazer uma aproximação linear de  $f$  a partir de um certo ponto  $x^0$ . Pelo Teorema de Taylor temos que

$$f(x) \simeq f(x^0) + J(x^0)(x - x^0)$$

$$f(x) = 0 \Rightarrow x^1 = x^0 - J(x^0)^{-1}f(x^0),$$

em que  $x^1$  é nosso candidato a solução.

# Método de Newton multivariado

## Algoritmo

Vamos utilizar a ideia acima para construir um método para encontrar  $x^*$ . Dado um ponto inicial  $x^0$ , defina a sequência

$$x^{k+1} = x^k - J(x^k)^{-1}f(x^k), \forall k \geq 0.$$

Passos:

1. Escolha  $x^0$  e faça  $k = 0$ .
2. Defina  $A_k = J(x^k)$  e resolva para  $s_k$  o seguinte sistema linear  $A_k s_k = -f(x^k)$  e faça  $x^{k+1} = x^k + s_k$ .
3. Se  $\|x^k - x^{k+1}\| > \varepsilon$ , faça  $k = k + 1$  e volte ao passo anterior.
4. Se  $\|f(x^{k+1})\| \leq \varepsilon$  reporte sucesso, caso contrário reporte falha.

# Lucas's signal-extraction model

## Modelo

### Ambiente:

- ▶ tempo discreto e infinito:  $t = 0, 1, 2, \dots$
- ▶ gerações sobrepostas: cada agente vive dois períodos e maximiza:  $E[u(c) + v(c')]$ 
  - ▶  $c$ : lazer quando jovem; e  $c'$  consumo quando velho.
  - ▶  $u'' < 0 < u'$ ,  $v'' < 0 < v'$ ,  $u'(0) = v'(0) = \infty$ .
  - ▶  $g' > 0$  em que  $g(x) = xv'(x)$
- ▶ não há tecnologia de estocagem
- ▶ dotação de horas (lazer):  $w > 0$  somente quando jovem.
  - ▶ dotação pode ser transformada em uma unidade de consumo do período através de tecnologia (função de produção) Um-para-Um.

# Lucas's signal-extraction model

## Choques

tamanho da geração  $t$ :  $N_t$

- ▶  $P(\ln(N_t) = \beta + \Delta i \equiv \ln N^i) = \pi_i$
- ▶ onde  $i \in \{1, 2, \dots, I\}$ .

taxa crescimento da moeda:  $x_t = \frac{M_t}{M_{t-1}}$

- ▶  $M_t - 1$ : quantidade de moeda que a geração  $t - 1$  leva para  $t$ .
- ▶  $P(\ln(x_t) = \lambda + \Delta j \equiv \ln x^j) = \theta_j$ ,
- ▶ onde  $j \in \{1, 2, \dots, J\}$



# Lucas's signal-extraction model

## Informação

**Hipótese:** agentes não observam diretamente  $x_t$  e  $N_t$  em  $t$ , mas conhecem toda história anterior de  $x$  e  $N$ , além de  $M_0$ .

### Definition

Uma história é um vetor

$q_t = (M_0, x_1, \dots, x_t, N_1, \dots, N_t) \in Q_t$ , onde  $Q_t$  é a coleção de histórias possíveis até  $t$ .

Informação incompleta: somente  $z_t = \frac{x_t}{N_t}$  é observável em  $t$

- ▶  $P(\ln(z_t) = \lambda - \beta + \Delta k \equiv \ln z^k) = \phi_k \equiv \sum_{j=1}^J \theta_j \pi_{j-k}$
- ▶ onde  $k = j - i$ ,  $k \in \{1 - I, 2 - I, \dots, J - 1\}$  e  $\pi_i = 0$  para  $i \notin \{1, 2, \dots, I\}$

# Lucas's signal-extraction model

## Preços

- ▶ cada unidade de consumo é vendida ao preço  $p_t$  em  $t$
- ▶ agente jovem poupa sua dotação de lazer em  $y_t$  (= unidades produzidas do bem de consumo corrente) e vende por moeda, recebendo  $m_t = y_t \cdot p_t$  unidades montárias.
- ▶ carrega  $m_t$  para  $t + 1$
- ▶ consumirá  $c'_{t+1} = m_t / p_{t+1}$ .
- ▶ agente não conhece  $p_{t+1}$  em  $t$ , impossibilitando *risk-sharing* perfeito.
- ▶ **Extração de Sinal:** de onde surge as variações de preço?
  - ▶ aumento  $N_t \Rightarrow$  diminuição de  $p_t$  (efeito real da oferta de bens)
  - ▶ aumento de  $x_t \Rightarrow$  aumento de  $p_t$  (efeito nominal de emissão monetária)

# Lucas's signal-extraction model

## Problema do indivíduo jovem

- restrições:

$$\begin{aligned}c_t &\leq w - y_t \\ m_t &\leq p_t y_t \\ p_{t+1} c'_{t+1} &\leq m_t\end{aligned}$$

- escolha de  $y_t$  para maximizar

$$W(y_t) = u(w - y_t) + E \left[ v \left( \frac{y_t p_t}{p_{t+1}} \right) ; p_t, q_{t-1} \right]$$

em que  $q_t \equiv (M_0, x_1, x_2, \dots, x_t, N_1, N_2, \dots, N_t)$

# Lucas's signal-extraction model

## Equilíbrio

### Definition

Um equilíbrio em expectativas racionais do modelo são as sequências de funções  $\{p_t\} : Q_t \rightarrow \mathbb{R}$  e  $\{y_t\} : Q_t \rightarrow \mathbb{R}$  tal que para todo  $t \geq 1$  e todo  $q_t \in Q_t$ ,

- ▶  $y_t = y_t(q_t)$  maximiza  $W(y_t)$  quando  $p_t = p_t(q_t)$  e quando a esperança é tomada com respeito a distribuição de  $p_{t+1} = p_{t+1}(q_{t+1})$  condicional a  $(q_{t-1}, p_t)$ ,
- ▶ Market Clearing:  $N_t y_t(q_t) = \frac{M_t}{p_t(q_t)}$ .

# Lucas's signal-extraction model

## Equilíbrio

Olhamos para uma família menor de equilíbrios:

$$y_t = y_t \left( \frac{x_t}{N_t} \right)$$

- ▶ *guess and verify*
- ▶ *guess*: equilíbrio é estacionário ( $y_t$  depende de  $q_t$  somente por  $z_t$  e a dependencia é tal que  $z_t/y_t$  é estritamente crescente em  $z_t$ )

## Theorem

*Existe um e somente um equilíbrio satisfazendo*

*$y_t(q_t) = y(z_t) > 0$  e  $z_t/y(z_t)$  é estritamente crescente.*

# Lucas's signal-extraction model

## Equilíbrio

### Observação

Como  $z_t$  assume no máximo  $I + J - 1$  valores, então  $y(\cdot)$  é um vetor do  $\mathbb{R}^{I+J-1}$ .

- ▶ CPO do problema do indivíduo:  $W'(y_t) = 0$ ,

$$y_t u'(w - y_t) = E \left[ v' \left( \frac{y_t p_t}{p_{t+1}} \right) \frac{y_t p_t}{p_{t+1}} \middle| p_t, q_{t-1} \right]$$

- ▶ definindo  $f(x) = x u'(x)$ , temos

$$f(y_t) = E[g(y_t p_t / p_{t+1}) | p_t, q_{t-1}]$$

.

- ▶ Pelo MC:

$$\frac{p_t}{p_{t+1}} = \frac{M_t / N_t y_t}{M_{t+1} / N_{t+1} y_{t+1}} = \frac{N_{t+1} y_{t+1}}{x_{t+1} N_t y_t} = \frac{y_{t+1}}{z_{t+1} y_t N_t}$$

# Lucas's signal-extraction model

Equilíbrio

Ficamos com

$$f(y_t) = E \left[ g \left( \frac{y_{t+1}}{z_{t+1} N_t} \right) | p_t, q_{t-1} \right]$$

- ▶ *guess*:  $y_t = y(q_t) = y(z_t)$ .
- ▶ equivalência  $(p_t, q_{t-1})$  com  $(q_{t-1}, z_t)$ .

▶ Daí

$$f(y(z_t)) = E \left[ g \left( \frac{y(z_{t+1})}{z_{t+1} N_t} \right) | z_t \right]$$

- ▶ usando condições de equilíbrio, propriedades dos choques e  $\phi_{ik} \equiv \frac{\pi_i \theta_{i+k}}{\phi_k}$

$$f(y^k) = \sum_i \phi_{ik} \left[ \sum_h \phi_h g(y^h / z^h N^i) \right]$$

# Lucas's signal-extraction model

## Equilíbrio

Então,

$$y^k = f^{-1} \left( \sum_i \phi_{ik} \left[ \sum_h \phi_{h i} g(y^h / z^h N^i) \right] \right). \quad (8)$$

É possível mostrar que o operador acima defini uma contração, logo possui um único ponto fixo.

### Theorem

*Existe um e apenas um equilíbrio tal que  $y_t(q_t) = y(z_t) > 0$  e  $z_t/y(z_t)$  é estritamente crescente.*



# Lucas's signal-extraction model

## Algoritmo

Passos do algoritmo:

1. Carregar os parâmetros.
2. Definir as probabilidades.
3. Chute inicial para o vetor  $y$ .
4. Iterar o operador definidor por (8) até o ponto fixo ser atingido.

# Imrohoroglu's Model

## Environment

*Imrohoroglu (1992) The welfare cost of inflation under imperfect insurance - JEDC*

- ▶ tempo discreto e horizonte infinito
- ▶ contínuo de indivíduos (massa 1)
- ▶ indivíduo possui dois estados: emprego e desemprego
- ▶  $s_i \in E := \{y, \theta y\}$ ,  $\theta < 1$
- ▶ transição markoviana:

$$\pi(s'|s) = \Pr(s_{t+1} = s' | s_t = s) > 0$$

- ▶ preferências representáveis por

$$E \left[ \sum_{t=0}^{\infty} \beta^t u(c_t) \right] \quad \beta \in (0, 1)$$

$$u(c_t) = \frac{c_t^{1-\sigma}}{1-\sigma} \quad \sigma > 1$$

# Imrohoroglu's Model

## Environment

- ▶ Não há comprometimento com ações futuras
- ▶ As pessoas começam na data  $t$  com encaixes monetários  $\tilde{m}_{t-1}$
- ▶ transferências positivas  $T = \tau \tilde{M}_{t-1}$  (lump-sum)
- ▶  $\tau$  é a taxa de crescimento da base monetária:  $\tilde{M}_t = (1 + \tau) \tilde{M}_{t-1}$
- ▶  $p_t$  denota o preço da moeda em unidades do bem de consumo da data  $t$

# Imrohoroglu's Model

## Restrição orçamentária

$$c_t + p_t \tilde{m}_t \leq p_t \tilde{m}_{t-1} + y_t + \tau p_t \tilde{M}_{t-1}$$

em que

$$y_t = \begin{cases} y & \text{se está empregado} \\ \theta y & \text{se não está empregado} \end{cases}$$

Definindo  $m_t = p_t \tilde{m}_t$ ,  $M_t = p_t \tilde{M}_t$  e  $1/(1 + \pi_t) = p_t/p_{t-1}$ , temos

$$c_t + m_t \leq 1/(1 + \pi_t)m_{t-1} + y_t + \tau/(1 + \pi_t)M_{t-1} \quad (9)$$

Monitoria 1

Macroeconomia III

Modelo de Search  
no Mercado de  
Trabalho

Modelo Clássico de  
Crescimento

Árvore de Lucas

Equity Premium  
Puzzle

Sistemas lineares

Método da  
Bisseção

Método de  
Newton-Raphson

Lucas's  
signal-extraction  
model

Imrohoroglu

Huggett's Model

Aiyagari's Model

# Imrohoroglu's Model

## Problema Recursivo

### Equação de Bellman

$$V(s, m) = \max_{m', c} \{ u(c) + \beta \mathbb{E}_{s'} [V(s', m')] \}$$

sa (9)

Monitoria 1

Macroeconomia III

Modelo de Search  
no Mercado de  
Trabalho

Modelo Clássico de  
Crescimento

Árvore de Lucas

Equity Premium  
Puzzle

Sistemas lineares

Método da  
Bisseção

Método de  
Newton-Raphson

Lucas's  
signal-extraction  
model

**Imrohoroglu**

Huggett's Model

Aiyagari's Model

# Imrohoroglu's Model

## Equilíbrio

### Definition

Um equilíbrio recursivo estacionário é um preço  $p$ , uma função valor  $v(s, m)$ , uma função política  $m'(s, m)$ , uma distribuição estacionária  $\lambda(s, m)$  e oferta de moeda  $M$  tais que

(i)  $v(s, m)$  e  $m'(s, m)$  resolvem o problema do indivíduo

(ii) Market clearing no mercado de bens e moeda

$$\sum_{m,s} \lambda(s, m) c(s, m) = \sum_{m,s} \lambda(s, m) y(s)$$

$$\sum_{m,s} \lambda(s, m) m(s, m) = M$$

(iii)  $\lambda$  é uma distribuição estacionária

# Imrohoroglu's Model

## Algoritmo

1. carregar os parâmetros ( $\tau = \pi$ )
2. fixe  $M$  a oferta de moeda
3. Resolva o problema do consumidor para encontrar  $m'(s, m)$
4. Construa a matriz de transição em  $(s, m)$  e encontre  $\lambda$
5. Calcule a demanda por moeda  
$$\sum_{m,s} \lambda(s, m) m(s, m) = D$$
  - ▶ se  $\|D - M\| < \epsilon$  solução encontrada
  - ▶ caso contrário, atualize  $M$  e volte ao passo 3

# Huggett's Model

## Environment

*Huggett (1993) Journal of Economic Dynamics and Control*

- ▶ tempo discreto e horizonte infinito
- ▶ economia de trocas puras (não há produção)
- ▶ contínuo de indivíduos (massa 1)
- ▶  $e_i \in E := \{e_l, e_h\}$ : dotação do agente
- ▶ transição markoviana:

$$\pi(e'|e) = \Pr(e_{t+1} = e' | e_t = e) > 0$$

- ▶ preferências representáveis por

$$E \left[ \sum_{t=0}^{\infty} \beta^t u(c_t) \right] \quad \beta \in (0, 1)$$

$$u(c_t) = \frac{c_t^{1-\sigma}}{1-\sigma} \quad \sigma > 1$$



# Huggett's Model

## Environment

- ▶ há somente um ativo:
  - ▶ vendido ao preço  $q$
  - ▶ paga uma unidade em  $t + 1$
- ▶ restrição de endividamento:  $a \geq \underline{a}$
- ▶ restrição orçamentária

$$c + a'q \leq a + e$$

- ▶  $x = (a, e) \in X$ : estado do indivíduo
  - ▶  $X = A \times E$ : espaço estado
  - ▶  $A = [\underline{a}, \infty)$
  - ▶  $E = \{e_l, e_h\}$  com  $e_h > e_l$

# Huggett's Model

## Problema Recursivo

$$v(x; q) = \max_{(c, a') \in \Gamma(x; q)} \left\{ u(c) + \beta \sum_{e'} \pi(e'|e) v(x'; q) \right\}$$

em que

$$\Gamma(x; q) = \{(c, a') : c + a'q \leq a + e; c \geq 0; a' \geq \underline{a}\}$$

ou ainda,

$$v(x; q) = \max_{a'} \left\{ u(a + e - a'q) + \beta \sum_{e'} \pi(e'|e) v(x'; q) \right\}$$

sujeito a  $a' \in \left[ \underline{a}, \frac{a+e}{q} \right]$

- Seja  $g : X \rightarrow A$  a função política correspondente

# Huggett's Model

## Mudança de estado

### Evolução da Riqueza (ativos e dotação)

$$\lambda_{t+1}(a', e') = \sum_a \sum_e \lambda_t(a, e) \pi(e'|e) I(a', a, e)$$

em que  $I(a', a, e) = 1$  se  $g(a, e) = a'$  e nula cc.

Usando a notação  $x = (a, e)$ ,

$$\lambda_{t+1}(x') = \sum_x \lambda_t(x) \pi(e'|e) I(a', x)$$

# Huggett's Model

## Mudança de estado

- ▶ seja  $\bar{x}$  o vetor de estados dado por

$$[(e_l, a_1), \dots, (e_l, a_n), (e_h, a_1), \dots, (e_h, a_n)]$$

- ▶ Lei de movimento da distribuição:

$$\lambda'(\bar{x}) = \lambda(\bar{x})M$$

em que

$$M_{ij} = \pi(e' | e) I(a', x)$$

se  $x_i = (a, e)$  e  $x_j = (a', e')$

- ▶ uma distribuição  $\bar{\lambda}$  é dita estacionária se

$$\bar{\lambda}(\bar{x}) = \bar{\lambda}(\bar{x})M$$

# Huggett's Model

## Equilíbrio

### Definition

Dado  $\underline{a}$ , um equilíbrio recursivo estacionário é um preço  $q$ , uma função valor  $v(a, e)$ , uma função política  $g(a, e)$  e uma distribuição estacionária  $\lambda(a, e)$  tais que

- (i)  $v(a, e)$  e  $g(a, e)$  resolvem o problema do indivíduo
- (ii)  $\lambda$  é induzida por  $\Pi$  e  $g(a, e)$  (via  $M$ )
- (iii) oferta e demanda se igualam no mercado de crédito

$$G(q) = \sum_x \lambda(x)g(x) = 0$$

# Huggett's Model

## Algoritmo

1. fixe um preço  $q > 0$
2. calcule  $v(x)$  e  $g(x)$  que resolvem o problema do indivíduo
3. calcule  $M$  usando  $\Pi$  e  $g(x)$
4. encontre a distribuição invariante  $\lambda$
5. calcule o excesso de demanda de crédito  $G(q)$ 
  - ▶ se  $G(q) < 0$ , há excesso de oferta:

$$q = \alpha q \quad \alpha < 1$$

e retorne ao passo 2 usando novo preço

- ▶ se  $G(q) > 0$ , há excesso de demanda:

$$q = \alpha q \quad \alpha > 1$$

e retorne ao passo 2 usando novo preço

- ▶ se  $G(q) = 0$ , o mercado está em equilíbrio:  
 $q$  é o preço de equilíbrio

# Aiyagari's Model

## Environment

*Aiyagari (1994) Quarterly Journal of Economics*

- ▶ tempo discreto e horizonte infinito
- ▶ contínuo de indivíduos (massa 1)
- ▶  $n_i \in N := \{n_l, n_h\}$ : produtividade do agente
- ▶ oferta inelástica de trabalho por um salário  $wn_i$
- ▶ transição markoviana *iid*:

$$\pi(n'|n) = \Pr(n_{t+1} = n' | n_t = n) > 0$$

- ▶ preferências representáveis por

$$E \left[ \sum_{t=0}^{\infty} \beta^t u(c_t) \right] \quad \beta \in (0, 1)$$

# Aiyagari's Model

## Environment

- ▶ os agentes acumulam capital
  - ▶ rende uma taxa de juros  $r$
  - ▶ deprecia a uma taxa  $\delta$
- ▶ restrição de endividamento:  $k \geq 0$
- ▶ restrição orçamentária

$$c + k' \leq (1 + r - \delta)k + wn$$

- ▶  $x = (k, n) \in X$ : estado do indivíduo
  - ▶  $X = K \times N$ : espaço estado
  - ▶  $\mathbb{K} = [0, k_{max}]$



# Aiyagari's Model

## Problema Recursivo

$$v(x) = \max_{(c, k') \in \Gamma(x)} \left\{ u(c) + \beta \sum_{n'} \pi(n'|n) v(x') \right\}$$

em que

$$\Gamma(x) = \{(c, k') : c + k' \leq (1 + r - \delta)k + wn; c \geq 0; k' \geq 0\}$$

- Seja  $g : X \rightarrow A$  a função política correspondente

# Aiyagari's Model

## Matriz de Markov Endógena

A partir: de  $\Pi, g$  determinamos a matriz de Markov  $M$  no espaço  $\mathbb{K} \times N$ .

A partir de  $M$  encontramos a distribuição invariante  $\lambda(k, n)$ ,

$$\lambda(k, n) = Pr(k_t = k, n_t = n).$$

# Aiyagari's Model

## Agregação

Vamos nos concentrar em um equilíbrio estacionário. Então:

$$K = \sum_{k \in \mathbb{K}} \sum_{n \in N} g(k, n) \lambda(k, n)$$

e

$$L = \sum_{n \in N} n \bar{\pi}(n)$$

# Aiyagari's Model

## Problema da Firma

Maximizar

$$F(K, L) - rK - wL$$

CPO's

$$\frac{\partial F(K, L)}{\partial K} = r$$

$$\frac{\partial F(K, L)}{\partial L} = w$$

# Aiyagari's Model

## Equilíbrio

### Definition

Um equilíbrio recursivo estacionário é: preços  $(w, r)$ , uma função valor  $v(k, n)$ , uma função política  $g(k, n)$ , uma distribuição estacionária  $\lambda(k, n)$ , capital e emprego agregados  $(K, L)$  tais que

- (i)  $v(k, n)$  e  $g(k, n)$  resolvem o problema do indivíduo
- (ii)  $(K, L)$  resolvem o problema da firma
- (iii)  $\lambda$  é induzida por  $\Pi$  e  $g(k, n)$  (via  $M$ )
- (iv)

$$K = \sum_{k \in \mathbb{K}} \sum_{n \in N} g(k, n) \lambda(k, n) \text{ e } L = \sum_{n \in N} n \bar{\pi}(n)$$

# Aiyagari's Model

## Algoritmo

1. Calcule  $L$  via  $\Pi$ .
2. Chute  $K$
3. Com  $(K, L)$  encontre  $(w, r)$
4. resolva o problema do consumidor e obtenha  $g(k, n)$
5. calcule  $M$  usando  $\Pi$  e  $g(k, n)$
6. encontre a distribuição invariante  $\lambda$
7. calcule  $K' = \sum_{k \in \mathbb{K}} \sum_{n \in N} g(k, n) \lambda(k, n)$ 
  - ▶ se  $|K - K'| \neq 0$ , atualize  $K$  e retorne ao passo 3 usando o novo  $K$
  - ▶ se  $|K - K'| = 0$ , pare, encontramos o equilíbrio.