

## 4 Lista 4

### 4.0 [X] Definições, Proposições, Lemas e Observações

---

**Teorema de Benveniste-Scheinkman.** Sejam  $X \subseteq \mathbb{R}^n$  convexo,  $V : X \rightarrow \mathbb{R}$  côncava,  $x_0 \in \text{int}(X)$  e  $D$  uma vizinhança de  $x_0$ . Se existe uma função diferenciável  $W : X \rightarrow \mathbb{R}$  com  $W(x_0) = C(x_0)$  e  $W(x) \leq V(x)$  para todo  $x \in D$ , então  $V$  é diferenciável em  $x_0$ , e

$$\partial_i V(x_0) = \partial_i W(x_0), \quad i = 1, 2, \dots, n$$

em que  $\partial_i$  é a derivada parcial em relação à  $i$ -ésima coordenada.

---

**Definição 59 (Krueger, 2017).** Uma distribuição  $\Pi \in \mathbb{R}_+^N$  que satisfaz

$$\Pi = \pi' \Pi \iff \Pi' \pi = \Pi'$$

é chamada de uma distribuição estacionária associada com a matriz de Markov  $\pi$ .

---

**Teorema 60 (Krueger, 2017).** Dada uma matriz de Markov,  $\pi$ , existe pelo menos uma distribuição estacionária.

---

**Teorema 2.2.1 (Ljungqvist & Sargent, 2018).** Seja  $\pi$  uma matriz estocástica, com  $\pi_{i,j} > 0, \forall i, j$ . Então,  $\pi$  possui uma única distribuição estacionária.

---

**Teorema 2.2.2 (Ljungqvist & Sargent, 2018).** Seja  $\pi$  uma matriz estocástica e  $\pi^n = \pi \times \pi \times \dots \times \pi$  ( $n$  vezes). Se  $\pi_{i,j}^n > 0, \forall i, j$ , então  $\pi$  possui uma única distribuição estacionária.

---

#### Modelo de Crescimento Neoclássico Estocástico.

- Tecnologia:  $y_t = e^{z_t} F(k_t, n_t)$ 
  - Choque tecnológico  $z \in Z = \{z_1, z_2, \dots, z_N\}$  com média 0 e  $N$  estados.
  - Seja  $\pi$  a matriz de transição de Markov e  $\Pi$  a distribuição estacionária.

- Preferências:

$$\mathbb{E}_0 \sum_{t=0}^{\infty} \beta^t u(c_t), \quad \beta \in (0, 1)$$

---

#### Formulação Recursiva do Problema com Incerteza.

$$\begin{aligned} v(k, z) &= \max_{0 \leq k' \leq e^z F(k, 1) + (1-\delta)k} \{U(c) + \beta \mathbb{E}_{z'} [v(k', z')]\} \\ &= \max_{0 \leq k' \leq e^z F(k, 1) + (1-\delta)k} \left\{ U(e^z F(k, 1)(1-\delta)k - k') + \beta \sum_{z'} \pi(z'|z) v(k', z') \right\} \end{aligned}$$

em que:

- $n = 1$ : pessoas não valorizam lazer

- $F(k, n) = k^\alpha n^{1-\alpha} \xrightarrow[n=1]{\text{blue}} k^\alpha$
- $c = e^z F(k, n) + (1 - \delta)k - k' \xrightarrow[n=1]{\text{blue}} e^z F(k, 1)(1 - \delta)k - k'$
- $\mathbb{E}_{z'} [v(k', z')] = \sum_{z'} \pi(z'|z) v(k', z')$

### Formulação Recursiva do Problema com Incerteza e Escolha de Trabalho.

$$\begin{aligned}
 v(k, z) &= \max_{\substack{0 \leq k' \leq e^z F(k, n) + (1-\delta)k \\ 0 \leq n \leq 1}} \{U(c, \ell) + \beta \mathbb{E}_{z'} [v(k', z')]\} \\
 &= \max_{\substack{0 \leq k' \leq e^z F(k, n) + (1-\delta)k \\ 0 \leq n \leq 1}} \left\{ U(e^z F(k, 1)(1 - \delta)k - k', \ell) + \beta \sum_{z'} \pi(z'|z) v(k', z') \right\}
 \end{aligned}$$

em que:

- $U(c, \ell) = U(c, 1 - n)$ : pessoas valorizam lazer, então decidem o quanto irão trabalhar

## 4.1 $\boxed{\times}$

Considere um consumidor de vida infinita que tem uma renda constante,  $y$ , a cada período. Este consumidor encara o seguinte problema.

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

$$\text{s.a } c_t + a_{t+1} \leq y + (1 + r_t)a_t \quad \forall t, \quad a_0 > 0 \text{ dado,}$$

em que, para cada período  $t$ ,  $a_t$  representa a riqueza do consumidor,  $c_t$  é o consumo,  $r_t$  é a taxa de juros e  $\beta \in (0, 1)$ . Suponha que o consumidor conhece  $\{r_t\}_{t=0}^{\infty}$  e que  $u(c) = c^{1-\sigma}/(1-\sigma)$  com  $\sigma > 0$  e  $\sigma \neq 1$ .

(a) Escreva o problema recursivo do consumidor.

O problema recursivo é dado por:

$$\begin{aligned} v(a) &= \max_{c, a'} \{u(c) + \beta v(a')\} \\ &= \max_{c, a'} \left\{ \frac{c^{1-\sigma}}{1-\sigma} + \beta v(a') \right\} \\ &= \max_{a'} \left\{ \frac{1}{1-\sigma} [y + (1+r)a - a']^{1-\sigma} + \beta v(a') \right\} \quad (c = y + (1+r)a - a') \end{aligned}$$

(b) Encontre a equação de Euler associada ao problema do consumidor.

Por CPO:

$$\begin{aligned} [a'] : \quad 0 &= \frac{1}{1-\sigma} (1-\sigma) [y + (1+r)a - a']^{-\sigma} (-1) + \beta v'(a') \\ &= -[y + (1+r)a - a']^{-\sigma} + \beta v'(a') \end{aligned} \quad (4.1.1)$$

Usando o Teorema de Benveniste-Scheinkman, obtemos

$$\begin{aligned} v'(a) &= \frac{1}{1-\sigma} (1-\sigma) [y + (1+r)a - a']^{-\sigma} (1+r) \\ &= (1+r) [y + (1+r)a - a']^{-\sigma} \end{aligned} \quad (4.1.2)$$

Usando  $a = a_t$ ,  $a' = a_{t+1}$  e  $r = r_t$ , avançando um período em (4.1.2) e aplicando em (4.1.1):

$$\begin{aligned} 0 &= -[y + (1+r_t)a_t - a_{t+1}]^{-\sigma} + \beta(1+r_{t+1}) [y + (1+r_{t+1})a_{t+1} - a_{t+2}]^{-\sigma} \\ \beta(1+r_{t+1}) &= \left( \frac{y + (1+r_{t+1})a_{t+1} - a_{t+2}}{y + (1+r_t)a_t - a_{t+1}} \right)^{\sigma}, \end{aligned} \quad (4.1.3)$$

e obtemos a equação de Euler. ■

(c) Mostre que se  $\beta = (1 + r_{t+1})^{-1}$ , então  $c_{t+1} = c_t$ .

Aplicando  $\beta = (1 + r_{t+1})^{-1}$  em (4.1.3), segue que:

$$\begin{aligned} (1 + r_{t+1})^{-1}(1 + r_{t+1}) &= \left( \frac{y + (1 + r_{t+1})a_{t+1} - a_{t+2}}{y + (1 + r_t)a_t - a_{t+1}} \right)^\sigma \\ 1^{\frac{1}{\sigma}} &= \left( \frac{y + (1 + r_{t+1})a_{t+1} - a_{t+2}}{y + (1 + r_t)a_t - a_{t+1}} \right)^{\frac{\sigma}{\sigma}} \\ 1 &= \frac{y + (1 + r_{t+1})a_{t+1} - a_{t+2}}{y + (1 + r_t)a_t - a_{t+1}}. \end{aligned}$$

Usando  $c_t = y + (1 + r_t)a_t - a_{t+1}$ , temos

$$1 = \frac{c_{t+1}}{c_t} \iff c_t = c_{t+1},$$

como queríamos demonstrar. ■

**(d) O que ocorre com o consumo se  $\beta > (1 + r_{t+1})^{-1}$ ? e se  $\beta < (1 + r_{t+1})^{-1}$ ?**

- Se  $\beta > (1 + r_{t+1})^{-1}$ , então o lado esquerdo de (4.1.3) será maior do que 1 e, portanto, no lado direito, precisamos que  $c_{t+1} > c_t$ . Logo, o consumo crescerá no período seguinte.
- Se  $\beta < (1 + r_{t+1})^{-1}$ , então o lado esquerdo de (4.1.3) será menor do que 1 e, portanto, no lado direito, precisamos que  $c_{t+1} < c_t$ . Logo, o consumo diminuirá no período seguinte. ■

**(e) Escreva os passos de um algoritmo para computar a solução do problema recursivo do consumidor.**

1. Defina um grid  $\mathbb{A} = \{a_1, a_2, \dots, a_m\}$  para as riquezas.
2. Defina a função utilidade do modelo:  $u(c) = \frac{c^{1-\sigma}}{1-\sigma}$ , com  $\sigma > 0$  e  $\sigma \neq 1$ .
3. Estabeleça os valores para parâmetros:  $\beta, y$  e  $\{r_t\}_{t=0}^\infty$
4. Defina vetores para guardar as funções valor e política:  $V$ ,  $TV$  e  $g_a$ . Note que cada um desses vetores têm dimensão  $m \times 1$ . Coloque valores de zeros em todos os vetores. Usaremos  $V$ , em que  $V(a) = 0, \forall a$ , como chute inicial para nossa iteração.
5. Estabeleça um critério de convergência para a iteração da função valor:  $\varepsilon \sim 0$  e positivo.
6. Para cada  $a \in \mathbb{A}$ , calcule

$$TV(a) = \max_{a' \in \mathbb{A}} \left\{ \frac{c^{1-\sigma}}{1-\sigma} + \beta V(a') \right\}, \quad \text{s.a. } 0 \leq a' \leq y + (1 + r)a$$

Guarde  $g_a(a) = a'$  que resolvem o problema acima

7. Calcule a distância  $D = \max |TV - V|$  e atualize  $V \equiv TV$ .
8. Se  $D > \varepsilon$ , volte para o passo 6. Caso contrário, pare. ■

## 4.2 [X]

Considere uma ilha (economia) com uma única árvore de Lucas.

- Os frutos (chamados de dividendos pelos habitantes da ilha) que crescem na árvore são a única fonte de consumo.
- Esses frutos seguem o seguinte processo estocástico:
  - $d_{t+1} = \gamma d_t$  com probabilidade  $\pi$  ou  $d_{t+1} = d_t$  (com probabilidade  $(1 - \pi)$ ).
  - se em um dado período  $T$  temos que  $d_T = d_{T-1}$  então para todo  $t > T$  vale que  $d_t = d_T$ .

Existe uma massa unitária de pessoas iguais nessa ilha com preferências sobre um fluxo de consumo dadas por

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

em que  $u(c) = c^{1-\sigma}/(1-\sigma)$ . Suponha que  $\sigma > 0$ ,  $\sigma \neq 1$ ,  $0 < \beta < 1$ ,  $\gamma > 1$  e que  $\beta\gamma^{1-\sigma} < 1$ .

(a) Escreva o problema recursivo de um agente representativo.

Além do consumo a cada período,  $c_t$ , o agente escolhe a fração de árvores,  $s_t$ , que deseja comprar. Então, o problema é dado por

$$\max_{\{c_t, s_{t+1}\}} \left\{ \sum_{t=0}^{\infty} \beta^t \mathbb{E}[u(c_t)] \right\}, \quad \text{s.a. } c_t + s_{t+1}p_t \leq (p_t + d_t)s_t, \text{ dado } s_0 \quad (4.2.1)$$

A partir dele, podemos obter o problema na forma recursiva:

$$\max_{s', c} \{u(c) + \beta \mathbb{E}[V(s', d')]\}, \quad \text{s.a. } c + p(d)s' \leq [p(d) + d]s.$$

■

(b) Defina um equilíbrio competitivo recursivo para essa economia.

**Definição (Equilíbrio Competitivo Recursivo):** Um equilíbrio competitivo recursivo para a economia descrita é uma coleção de funções  $\{V, g, p\}$ , em que  $V$  é uma função valor do problema recursivo,  $g$  é função política e  $p$  é a função preço, tais que:

- Dada a função preço  $p$ , a função valor  $V$  e a função política resolvem o problema recursivo de programação dinâmica do consumidor.
- Os mercados se equilibram com  $g(s, d) = 1$ , para todo par  $(s, d)$

Note que, pela condição (ii), temos que  $s = s' = 1$ . Portanto, pela restrição orçamentária de fluxo, temos:

$$c + p(d).1 = [p(d) + d].1 \implies c = d, \quad (4.2.2)$$

ou seja, no equilíbrio competitivo recursivo, todo dividendo é consumido.

■

---

(c) Encontre a condição de primeira ordem do problema do agente representativo e, utilizando a condição de equilíbrio, escreva a equação de apreçamento do ativo.

---

Resolveremos o problema pela forma sequencial (4.2.1). Considerando  $u'' < 0 < u'$  e aplicando  $c_t = (p_t + d_t)s_t - s_{t+1}p_t$ , obtemos:

$$\max_{s_{t+1}} \left\{ \sum_{t=0}^{\infty} \beta^t \mathbb{E}[u((p_t + d_t)s_t - s_{t+1}p_t)] \right\}$$

Note que, no somatório acima, há dois termos que possuem  $s_{t+1}$ :

$$\dots + \beta^t \mathbb{E}[u((p_t + d_t)s_t - s_{t+1}p_t)] + \beta^{t+1} \mathbb{E}[u((p_{t+1} + d_{t+1})s_{t+1} - s_{t+2}p_{t+1})] + \dots$$

Logo, por CPO:

$$\begin{aligned} [s_{t+1}] : \quad 0 &= -\beta^t \mathbb{E}[u'(c_t)p_t] + \beta^{t+1} \mathbb{E}[u'(c_{t+1})(p_{t+1} + d_{t+1})] \\ \beta^t \mathbb{E}[u'(c_t)p_t] &= \beta^{t+1} \mathbb{E}[u'(c_{t+1})(p_{t+1} + d_{t+1})] \\ p_t &= \frac{\beta^{t+1}}{\beta^t} \cdot \frac{\mathbb{E}[u'(c_{t+1})(p_{t+1} + d_{t+1})]}{\mathbb{E}[u'(c_t)]} \\ p_t &= \beta \mathbb{E} \left[ \frac{u'(c_{t+1})}{u'(c_t)} (p_{t+1} + d_{t+1}) \right]. \end{aligned}$$

Pela condição de equilíbrio  $c_t = d_t$  (4.2.2), segue que

$$p_t = \beta \mathbb{E} \left[ \frac{u'(d_{t+1})}{u'(d_t)} (p_{t+1} + d_{t+1}) \right]. \quad (4.2.3)$$

■

---

(d) Escreva os preços de equilíbrio como uma função dos dividendos.

---

Usaremos o método "guess-and-verify". Suponha que a função preço seja uma proporção do dividendo e dependa de seu estado no processo estocástico ( $d_{t+1} = \gamma d_t$  ou  $d_{t+1} = \gamma d_t$ ):

$$p_t = \begin{cases} p_s d_t, & \text{se } d_t = d_{t-1} \\ p_g d_t, & \text{se } d_t = \gamma d_{t-1} \end{cases} \quad (p_s \text{ e } p_g \text{ constantes})$$

Logo, se em um período tivermos  $d_t = d_{t-1}$ , sabemos que  $d_{t+1} = d_t$ . Note que

$$u(d_t) = \frac{d_t^{1-\sigma}}{1-\sigma}$$

e, portanto,

$$u'(d_t) = (1-\sigma) \frac{d_t^{1-\sigma-1}}{1-\sigma} = d_t^{-\sigma}.$$

Assim,

$$\frac{u'(d_{t+1})}{u'(d_t)} = \left( \frac{d_{t+1}}{d_t} \right)^{-\sigma}. \quad (4.2.4)$$

Aplicando (4.2.4) em (4.2.3) e lembrando que este é o caso em que  $d_t = d_{t-1}$ , segue que

$$\begin{aligned}
p_t &= \beta \left[ \left( \frac{d_{t+1}}{d_t} \right)^{-\sigma} (p_{t+1} + d_{t+1}) \right] && (\text{tira } \mathbb{E} \text{ pois caso único } d_t = d_{t+1}) \\
p_t &= \beta [1^{-\sigma} (p_{t+1} + d_{t+1})] && (d_t = d_{t+1}) \\
p_s d_t &= \beta (p_s d_t + d_t) && (p_t = p_s d_t) \\
p_s d_t &= \beta (p_s + 1) d_t \\
p_s &= \beta p_s + \beta \\
p_s (1 - \beta) &= \beta \\
p_s &= \frac{\beta}{1 - \beta}. && (4.2.5)
\end{aligned}$$

Se  $d_t = \gamma d_{t-1}$ , então, no valor esperado, consideraremos possíveis transições para os casos  $d_{t+1} = d_t$  e  $d_{t+1} = \gamma d_t$ . Segue que

$$\begin{aligned}
p_t &= \beta \left[ \underbrace{\pi \left( \frac{u'(d_{t+1})}{u'(d_t)} (p_{t+1} + d_{t+1}) \right)}_{\text{caso } d_{t+1} = \gamma d_t} + \underbrace{(1 - \pi) \left( \frac{u'(d_{t+1})}{u'(d_t)} (p_{t+1} + d_{t+1}) \right)}_{\text{caso } d_{t+1} = d_t} \right] \\
p_g d_t &= \beta \left[ \pi \left( \frac{u'(d_{t+1})}{u'(d_t)} (p_g d_{t+1} + d_{t+1}) \right) + (1 - \pi) \left( \frac{u'(d_{t+1})}{u'(d_t)} (p_s d_t + d_{t+1}) \right) \right] \\
&&& (p_t = p_g d_t \text{ ou } p_t = p_s d_t) \\
p_g d_t &= \beta \left[ \pi \left( \left( \frac{d_{t+1}}{d_t} \right)^{-\sigma} (p_g d_{t+1} + d_{t+1}) \right) + (1 - \pi) \left( \left( \frac{d_{t+1}}{d_t} \right)^{-\sigma} (p_s d_t + d_{t+1}) \right) \right] \\
&&& (\text{usando 4.2.4}) \\
p_g d_t &= \beta \left[ \pi \left( \left( \frac{d_{t+1}}{d_t} \right)^{-\sigma} (p_g d_{t+1} + d_{t+1}) \right) + (1 - \pi) (1^{-\sigma} (p_s d_t + d_t)) \right] \\
&&& (d_{t+1} = d_t) \\
p_g d_t &= \beta \left[ \pi \left( \left( \frac{d_{t+1}}{d_t} \right)^{-\sigma} (p_g d_{t+1} + d_{t+1}) \right) + (1 - \pi) \left( \frac{\beta}{1 - \beta} d_t + d_t \right) \right] \\
&&& (\text{usando 4.2.5}) \\
p_g d_t &= \beta \left[ \pi (\gamma^{-\sigma} (p_g \gamma d_t + \gamma d_t)) + (1 - \pi) \left( \frac{\beta}{1 - \beta} + 1 \right) d_t \right] \\
&&& (d_{t+1} = \gamma d_t \iff \frac{d_{t+1}}{d_t} = \gamma) \\
p_g &= \beta \left[ \pi \gamma^{-\sigma} (p_g \gamma + \gamma) + (1 - \pi) \left( \frac{\beta}{1 - \beta} + 1 \right) \right] \\
p_g &= \beta \pi \gamma^{1-\sigma} (p_g + 1) + \beta (1 - \pi) \left( \frac{\beta + 1 - \beta}{1 - \beta} \right) \\
p_g - (\beta \pi \gamma^{1-\sigma}) p_g &= \beta \pi \gamma^{1-\sigma} + \frac{\beta (1 - \pi)}{1 - \beta} \\
p_g &= \frac{1}{1 - \beta \pi \gamma^{1-\sigma}} \left( \beta \pi \gamma^{1-\sigma} + \frac{\beta (1 - \pi)}{1 - \beta} \right).
\end{aligned}$$

Dado que encontramos as constantes  $p_s$  e  $p_g$ , o palpite da função preço está correta. ■

---

(e) Seja  $T$  o primeiro período tal que  $d_{T-1} = d_T$ . Vale  $p_{T-1} > p_T$ ? Encontre condições para que isto seja verdade. Interprete os resultados.

---

Note que  $p_{T-1}$  é o preço enquanto ainda  $d_t = d_{t-1}$  e a partir de  $p_T$  estamos no caso em que  $d_t = d_{t-1}$ , ou seja, estas funções preço correspondem a  $p_g d_t$  e  $p_s d_t$ , respectivamente. Portanto, basta encontrar as condições para que  $p_g > p_s$ :

$$\begin{aligned}
\frac{1}{1 - \beta\pi\gamma^{1-\sigma}} \left( \beta\pi\gamma^{1-\sigma} + \frac{\beta(1-\pi)}{1-\beta} \right) &> \frac{\beta}{1-\beta} \\
\frac{1}{1 - \beta\pi\gamma^{1-\sigma}} \left( \pi\gamma^{1-\sigma} + \frac{(1-\pi)}{1-\beta} \right) &> \frac{1}{1-\beta} \\
\frac{1}{1 - \beta\pi\gamma^{1-\sigma}} \left( \frac{(1-\beta)\pi\gamma^{1-\sigma} + (1-\pi)}{1-\beta} \right) &> \frac{1}{1-\beta} \\
\frac{(1-\beta)\pi\gamma^{1-\sigma} + (1-\pi)}{1 - \beta\pi\gamma^{1-\sigma}} &> 1 \\
(1-\beta)\pi\gamma^{1-\sigma} + (1-\pi) &> 1 - \beta\pi\gamma^{1-\sigma} \\
\pi\gamma^{1-\sigma} + (1-\pi) &> 1 \\
\gamma^{1-\sigma} &> \frac{1 - (1-\pi)}{\pi} \\
\gamma^{1-\sigma} &> 1 \\
\gamma &> 1.
\end{aligned}$$

Para valer  $p_{T-1} > p_T$ , precisamos de  $\gamma > 1$  (como suposto inicialmente), ou seja, quando vale  $d_{t+1} = \gamma d_t$ , há crescimento dos dividendos. ■



### 4.3 [☒]

Seja a Matriz estocástica

$$M = \begin{pmatrix} 0.20 & 0.30 & 0.40 & 0.10 \\ 0.10 & 0.10 & 0.70 & 0.10 \\ 0.95 & 0.024 & 0.025 & 0.001 \\ 0.25 & 0.25 & 0.25 & 0.25 \end{pmatrix}$$

---

(a) Defina a distância entre duas matrizes como a soma do quadrado das diferenças entre cada entrada das matrizes. No Python, defina uma distribuição inicial

$$P_0 = (0.25, 0.25, 0.25, 0.25),$$

ache o  $k$ , tal que a distancia entre  $P^{k+1}$  e  $P^k$  seja menor do que  $10^{-7}$ .

---

Usando Definição 59 (Krueger, 2017), encontraremos a distribuição de probabilidade  $P$  que satisfaz  $P'\pi = \Pi'$ . Faremos iterações até a distância entre  $P^{k+1}$  e  $P^k$  ser menor do que  $10^{-7}$ .

```
1 # Construindo a matriz de Markov e a distribuição inicial
2 P_it = [np.array([0.25, 0.25, 0.25, 0.25])]
3
4 pi = np.array([[0.200, 0.300, 0.400, 0.100],
5               [0.100, 0.100, 0.700, 0.100],
6               [0.950, 0.024, 0.025, 0.001],
7               [0.250, 0.250, 0.250, 0.250]])
8
9 # Estabelecendo parâmetros
10 tol_norma = 1e-7
11 norma = np.inf # Apenas para entrar no loop
12 it = 0
13
14 # Realizando iterações até distância de P_t e P_{t+1} ser menor do que 10^{-7}
15 while norma > tol_norma:
16     it += 1
17
18     P = np.dot(P_it[it - 1], pi)
19     P_it.append(P)
20
21     norma = np.sum((P - P_it[it - 1]) ** 2)
22     print("Iteração {}: com norma = {:.7f}".format(it, norma))
23
24 print("Distribuição estacionária:", np.round(P_it[it], 4))
25
1 Distribuição estacionária: [0.4266 0.1731 0.3199 0.0804]
```

---

(b) Reporte a distribuição invariante de  $M$ . Há mais do que uma?

---

A distribuição invariante da matriz de Markov  $M$  é dada por

$$P^{ss} = [0.4266 \quad 0.1731 \quad 0.3199 \quad 0.0804]$$

Se a distribuição invariante for única, usando o Teorema 2.2.1 (Ljungqvist & Sargent, 2018), cada elemento da matriz de Markov,  $M_{i,j}, \forall i, j$ , deve ser estritamente maior do que zero.

```

1 # Verificar se existe algum 0 na matriz de Markov
2 qtd_zeros = 0
3 for p in range(len(pi)):
4     for q in range(len(pi[0])):
5         if pi[p, q] == 0:
6             qtd_zeros += 1
7
8 if qtd_zeros == 0:
9     print("A distribuição estacionária é única, pois todos elemento na matriz",
10         "de Markov é estritamente maior do que zero.")
11 else:
12     print("A distribuição estacionária não é única, pois há", qtd_zeros,
13         "elemento(s) igual(is) a zero na matriz de Markov.")

```

1 A distribuição estacionária é única, pois todos elemento na matriz de Markov é estritamente maior do que zero.

Como nenhuma entrada na matriz de Markov  $M$  é 0 e não é necessário verificar em  $M^n$ . Porém, para praticar, faremos a verificação em  $M^n$ , usando o Teorema 2.2.2 (Ljungqvist & Sargent, 2018), em que cada elemento de  $M^n_{i,j} > 0, \forall i, j$ , se a distribuição invariante for única.

```

1 # Calculando a matriz de Markov elevada a n
2 pi_it = [pi]
3 tol_norma = 1e-7
4 norma = np.inf
5 it = 0
6
7 while norma > tol_norma:
8     it += 1
9     pi_novo = np.dot(pi_it[it - 1], pi)
10    pi_it.append(pi_novo)
11    norma = np.sum((pi_it[it] - pi_it[it - 1]) ** 2)
12    print("Iteração {}: com norma = {:.7f}".format(it, norma))
13
14 print(np.round(pi_it[it], 4))

```

```

1 Iteração 1: com norma = 1.6061799
2 ...
3 Iteração 11: com norma = 0.0000000
4
5 [[0.4266 0.1731 0.3199 0.0804]
6  [0.4267 0.1731 0.3199 0.0804]
7  [0.4266 0.1731 0.3199 0.0804]
8  [0.4266 0.1731 0.3199 0.0804]]

```

```

1 # Verificar se existe algum 0 na matriz de Markov iterada
2 qtd_zeros = 0
3 for p in range(len(pi_it[it])):
4     for q in range(len(pi_it[it])):
5         if pi[p, q] == 0:
6             qtd_zeros += 1
7
8 if qtd_zeros == 0:
9     print("A distribuição estacionária é única, pois todos elemento na matriz",
10         "de Markov é estritamente maior do que zero.")
11 else:
12     print("A distribuição estacionária não é única, pois há", qtd_zeros,
13         "elemento(s) igual(is) a zero na matriz de Markov.")

```

1 A distribuição estacionária é única, pois todos elemento na matriz de Markov é estritamente maior do que zero.



#### 4.4 $\boxed{\times}$

Suponha que o planejador deseja maximizar

$$\mathbb{E} \left\{ \sum_{t=0}^{\infty} \beta^t u(c_t, l_t) \right\}, \quad 0 < \beta < 1$$

sujeito a restrição de recursos

$$c_t + k_{t+1} = z_t f(k_t, l_t) + (1 - \delta)k_t, \quad 0 < \delta < 1$$

com condições iniciais  $k_0 > 0$  e  $z_0 > 0$ . A produtividade  $z_t$  evolui de acordo com um processo de Markov com probabilidade de transição  $F(z' | z) = \text{Prob}[z_{t+1} \leq z' | z_t = z]$  e média incondicional  $\bar{z} > 0$ .

Neste problema o planejador escolhe quanto trabalho  $l_t$  ofertar. Assuma que  $u(c_t, l_t)$  é estritamente crescente e estritamente côncava em  $c_t$ , e é estritamente decrescente e estritamente convexa em  $l_t$ . A função de produção  $f(k_t, l_t)$  é estritamente crescente e estritamente côncava em ambos argumentos e tem retornos constantes de escala.

---

(a) Seja  $v(k, z)$  a função valor do planejador. Escreva e explique a equação de Bellman que determina  $v(k, z)$ .

---

Note que, neste modelo, as probabilidades de transição de um estado  $z$  para  $z'$  seguem uma distribuição de probabilidade com a c.d.f.  $F(z'|z)$  (diferente da seção 6.4 de Krueger (2017) que é discretizado). Logo, ao invés calcular o valor esperado pela soma dos valores de  $v(k', z')$  ponderados pelas probabilidades de transição de  $k$  para o estado  $k'$ , integraremos  $v(k', z')$  pela distribuição  $F(z'|z)$ . Portanto,

$$\begin{aligned} v(k, z) &= \max_{\substack{0 \leq k' \leq z f(k, l) + (1-\delta)k \\ 0 \leq l \leq 1}} \{U(c, l) + \beta \mathbb{E}_{z'} [v(k', z')]\} \\ &= \max_{\substack{0 \leq k' \leq z f(k, l) + (1-\delta)k \\ 0 \leq l \leq 1}} \left\{ U(z f(k, l)(1 - \delta)k - k', l) + \beta \int v(k', z') dF(z'|z) \right\}. \end{aligned} \quad (4.4.1)$$

Esta equação de Bellman inclui não apenas o estoque de capital  $k$  que o planejador traz para o período corrente, mas também o estado da tecnologia  $z$ . Ainda, o trabalhador decidirá o quanto irá trabalhar, dado que, além do consumo, valoriza também o lazer. Logo, para maximizar a expressão, o planejador irá escolher ambos capital do período seguinte  $k'$  e quantidade de trabalho  $l$ . ■

---

(b) Derive as condições de otimalidade do problema do planejador.

---

Estratégia da Prova:

- Seção 6.4.1 de Krueger (2017) ou Notas de aula
-

Por CPO, a partir da equação de Bellman (4.4.1), temos

$$[l] : \quad 0 = U_c(c, l)zf_l(k, l) + U_l(c, l) \quad (4.4.2)$$

$$[k'] : \quad 0 = U_c(c, l)(-1) + \beta \int v_k(k', z')dF(z'|z) \quad (4.4.3)$$

em que os subscritos indicam as derivadas parciais. A condição (4.4.2) pode ser reescrita como:

$$-\frac{U_l(c, l)}{U_c(c, l)} = zf_l(k, l). \quad (4.4.4)$$

Esta é a condição de otimalidade intratemporal que postula que, no ótimo, o planejador equaciona a taxa marginal de substituição entre trabalho e consumo em relação ao produto marginal do trabalho (Krueger, 2017, pág. 133).

Usando Teorema de Benveniste-Scheinkman (condição de envelope), segue que

$$v_k(k, z) = U_c(c, l)[zf_k(k, l) + (1 - \delta)]. \quad (4.4.5)$$

Usando os índices temporais  $k'$  e  $z'$  em (4.4.5) e aplicando em (4.4.3), obtemos a equação de Euler intertemporal:

$$U_c(c, l) = \beta \int U_c(c', l')[zf_k(k', l') + (1 - \delta)]dF(z'|z), \quad (4.4.6)$$

tal que a restrição de recursos é dada por

$$c + k' = zf(k, l) + (1 - \delta)k. \quad (4.4.7)$$

Portanto, as condições de otimalidade são dadas por: condição de otimalidade intratemporal (4.4.4), equação de Euler intertemporal (4.4.6) e restrição de recursos (4.4.7), em que o planejador escolhe  $c, l$  e  $k'$ , dado os estados  $k, z$ . ■

### (c) Suponha que

$$u(c, l) = \log c - \frac{l^{1+\varphi}}{1+\varphi}, \quad \varphi > 0$$

e

$$f(k, l) = k^\alpha l^{1-\alpha}, \quad 0 < \alpha < 1.$$

**Encontre os valores de steady state não estocástico de consumo, capital e trabalho em termos dos parâmetros do modelo. Suponha que existe um aumento permanente no nível de produtividade  $\bar{z}$ . Explique como isto muda os valores de estado estacionário do consumo, capital e trabalho. Dê uma intuição econômica para seus resultados.**

### Estratégia da Prova:

- Calcular as derivadas parciais e aplicá-las nas condições de otimalidade encontradas.
- Como as relações de estado estacionário  $c = c' = \bar{c}$ ,  $l = l' = \bar{l}$  e  $k = k' = \bar{k}$ , e, como é não-estocástico,  $z = z' = \bar{z}$ .
- Encontrar as relações  $\bar{k}/\bar{l}$ ,  $\bar{y}/\bar{l}$ ,  $\bar{k}/\bar{y}$ ,  $\bar{c}/\bar{y}$  e  $\bar{l}$ .
- Verificar como um choque em  $\bar{z}$  afeta  $\bar{c}$ ,  $\bar{k}$  e  $\bar{l}$ .

Primeiro, usando as funções  $U(c, l)$  e  $f(k, l)$  dadas, calcularemos as derivadas parciais utilizadas no item (c):

$$U_c(c, l) = 1/c \quad \text{e} \quad U_l(c, l) = (1 + \varphi) \frac{l^{1+\varphi-1}}{(1 + \varphi)} = l^\varphi$$

$$f_l(k, l) = (1 - \alpha)k^\alpha l^{-\alpha} = (1 - \alpha) \left( \frac{k}{l} \right)^\alpha \quad \text{e} \quad f_k(k, l) = \alpha k^{\alpha-1} l^{1-\alpha} = \alpha \left( \frac{k}{l} \right)^{\alpha-1}.$$

Aplicando as derivadas parciais em (4.4.4) e (4.4.6), e aplicando  $f(k, l)$  em (4.4.7), obtemos:

$$-\frac{l^\varphi}{1/c} = z(1 - \alpha) \left( \frac{k}{l} \right)^\alpha \iff cl^\varphi = z(1 - \alpha) \left( \frac{k}{l} \right)^\alpha \quad (4.4.4')$$

$$\frac{1}{c} = \beta \int \frac{1}{c'} \left[ z' \alpha \left( \frac{k'}{l'} \right)^{\alpha-1} + (1 - \delta) \right] dF(z'|z) \quad (4.4.6')$$

$$c + k' = zk^\alpha l^{1-\alpha} + (1 - \delta)k. \quad (4.4.7')$$

No estado estacionário não-estocástico, não há incerteza, então  $z = z' = \bar{z}$ . Além disso, no estado estacionário, temos que  $c = c' = \bar{c}$ ,  $l = l' = \bar{l}$  e  $k = k' = \bar{k}$ . Portanto, as três condições de otimalidade são dadas por:

$$\bar{c}\bar{l}^\varphi = \bar{z}(1 - \alpha) \left( \frac{\bar{k}}{\bar{l}} \right)^\alpha \quad (4.4.4'')$$

$$\frac{1}{\bar{c}} = \beta \int \frac{1}{\bar{c}} \left[ \bar{z} \alpha \left( \frac{\bar{k}}{\bar{l}} \right)^{\alpha-1} + (1 - \delta) \right] dF(\bar{z}|\bar{z}) \iff 1 = \beta \left[ \bar{z} \alpha \left( \frac{\bar{k}}{\bar{l}} \right)^{\alpha-1} + (1 - \delta) \right] \quad (4.4.6'')$$

$$\bar{c} + \bar{k} = \bar{z}\bar{k}^\alpha \bar{l}^{1-\alpha} + (1 - \delta)\bar{k} \iff \bar{c} + \delta\bar{k} = \bar{z}\bar{k}^\alpha \bar{l}^{1-\alpha} = \bar{z}f(\bar{k}, \bar{l}) \equiv \bar{y} \quad (4.4.7'')$$

A partir de (4.4.6''), a razão capital/trabalho é dada por:

$$\begin{aligned} \frac{1}{\beta} - (1 - \delta) &= \bar{z} \alpha \left( \frac{\bar{k}}{\bar{l}} \right)^{\alpha-1} \\ \frac{\frac{1}{\beta} - 1 + \delta}{\bar{z} \alpha} &= \left( \frac{\bar{k}}{\bar{l}} \right)^{\alpha-1} \\ \frac{\bar{z} \alpha}{\frac{1}{\beta} - 1 + \delta} &= \left( \frac{\bar{k}}{\bar{l}} \right)^{1-\alpha} \quad (\text{elevado a } -1) \\ \left( \frac{\bar{z} \alpha}{\rho + \delta} \right)^{\frac{1}{1-\alpha}} &= \frac{\bar{k}}{\bar{l}}. \end{aligned} \quad (4.4.8)$$

em que  $\rho \equiv 1/\beta - 1$ . Logo, aplicando  $\bar{k}/\bar{l}$  em (4.4.7''), obtemos a razão output/trabalho:

$$\begin{aligned} \bar{y} &= \bar{z}\bar{k}^\alpha \bar{l}^{1-\alpha} = \bar{z} \left( \frac{\bar{k}}{\bar{l}} \right)^\alpha \bar{l} \\ \frac{\bar{y}}{\bar{l}} &= \bar{z} \left[ \left( \frac{\bar{z} \alpha}{\rho + \delta} \right)^{\frac{1}{1-\alpha}} \right]^\alpha = \bar{z}^{1+\frac{\alpha}{1-\alpha}} \left( \frac{\alpha}{\rho + \delta} \right)^{\frac{\alpha}{1-\alpha}} \quad (\text{usando 4.4.8}) \\ \frac{\bar{y}}{\bar{l}} &= \bar{z}^{\frac{1}{1-\alpha}} \left( \frac{\alpha}{\rho + \delta} \right)^{\frac{\alpha}{1-\alpha}} \end{aligned} \quad (4.4.9)$$

E a razão capital/output é dada por:

$$\frac{\bar{k}}{\bar{y}} = \frac{\bar{k}/\bar{l}}{\bar{y}/\bar{l}} = \frac{\left(\frac{\bar{z}\alpha}{\rho+\delta}\right)^{\frac{1}{1-\alpha}}}{\bar{z}^{\frac{1}{1-\alpha}} \left(\frac{\alpha}{\rho+\delta}\right)^{\frac{\alpha}{1-\alpha}}} = \frac{\left(\frac{\alpha}{\rho+\delta}\right)^{\frac{1}{1-\alpha}}}{\left(\frac{\alpha}{\rho+\delta}\right)^{\frac{\alpha}{1-\alpha}}} = \left(\frac{\alpha}{\rho+\delta}\right)^{\frac{1-\alpha}{1-\alpha}} = \frac{\alpha}{\rho+\delta}, \quad (4.4.10)$$

que não depende de  $\bar{z}$ . Da restrição de recursos (4.4.7''), obtemos a razão consumo/output:

$$\begin{aligned} \frac{\bar{c} + \delta\bar{k}}{\bar{y}} &= 1 \\ \frac{\bar{c}}{\bar{y}} &= 1 - \frac{\delta\bar{k}}{\bar{y}} = 1 - \delta \left(\frac{\alpha}{\rho+\delta}\right), \end{aligned} \quad (4.4.11)$$

que também não depende de  $\bar{z}$ . De (4.4.4''), segue que

$$\begin{aligned} \bar{c}\bar{l}^\varphi &= \bar{z}(1-\alpha) \left[ \left(\frac{\bar{z}\alpha}{\rho+\delta}\right)^{\frac{1}{1-\alpha}} \right]^\alpha \\ \bar{c}\bar{l}^\varphi &= (1-\alpha)\bar{z}^{1+\frac{\alpha}{1-\alpha}} \left(\frac{\alpha}{\rho+\delta}\right)^{\frac{\alpha}{1-\alpha}} \\ \bar{c}\bar{l}^\varphi &= (1-\alpha)\bar{z}^{\frac{1}{1-\alpha}} \left(\frac{\alpha}{\rho+\delta}\right)^{\frac{\alpha}{1-\alpha}} \\ \bar{c}\bar{l}^\varphi &= (1-\alpha)\frac{\bar{y}}{\bar{l}} \quad (\text{usando (4.4.9)}) \\ \bar{l}^{1+\varphi} &= (1-\alpha)\frac{\bar{y}}{\bar{c}} = \frac{(1-\alpha)}{\bar{c}/\bar{y}} \\ \bar{l}^{1+\varphi} &= \frac{(1-\alpha)}{1 - \delta \left(\frac{\alpha}{\rho+\delta}\right)} \quad (\text{usando (4.4.11)}) \\ \bar{l}^{1+\varphi} &= \frac{(1-\alpha)}{\left(\frac{\rho+\delta-\delta\alpha}{\rho+\delta}\right)} \\ \bar{l}^{1+\varphi} &= \frac{(1-\alpha)(\rho+\delta)}{\rho+\delta(1-\alpha)} \\ \bar{l} &= \left[ \frac{(1-\alpha)(\rho+\delta)}{\rho+\delta(1-\alpha)} \right]^{\frac{1}{1+\varphi}}, \end{aligned} \quad (4.4.12)$$

que é novamente independente de  $\bar{z}$ . Portanto,

- um aumento permanente em  $\bar{z}$  não altera a quantidade de trabalho  $\bar{l}$ ;
- como, em (4.4.9),  $\bar{y}/\bar{l}$  é crescente em  $\bar{z}$ , e  $\bar{l}$  não depende de  $\bar{z}$ , então  $\bar{y}$  é crescente em  $\bar{z}$ ; e
- como, em (4.4.11),  $\bar{c}/\bar{y}$  é independente de  $\bar{z}$ , mas sabemos que  $\bar{y}$  é crescente em  $\bar{z}$ , então  $\bar{c}$  deve crescer na mesma proporção que  $\bar{y} \implies \bar{c}$  é crescente em  $\bar{z}$ .
- como, em (4.4.10),  $\bar{k}/\bar{y}$  é independente de  $\bar{z}$ , mas sabemos que  $\bar{y}$  é crescente em  $\bar{z}$ , então  $\bar{k}$  deve crescer na mesma proporção que  $\bar{y} \implies \bar{k}$  é crescente em  $\bar{z}$ . ■

(d) Suponha que os possíveis valores de  $z_t$  são  $\mathcal{Z} = \{0.8, 1, 1.2\}$  e que a matriz de transição é dada por

$$\pi = \begin{pmatrix} 0.20 & 0.5 & 0.3 \\ 0.10 & 0.6 & 0.3 \\ 0.25 & 0.25 & 0.50 \end{pmatrix}.$$

Sejam  $\alpha = 0.3, \beta = 1/1.05, \delta = 0.05, \varphi = 1$ . Escreva um código Python para calcular a função valor e funções políticas. Use um grid para o capital entre 0 e 100.<sup>3</sup>

```

1 # Definição dos parâmetros
2 k_grid = np.linspace(0.01, 6, 51)
3 n_k = len(k_grid)
4 alpha = 0.3
5 beta = 1 / 1.05
6 delta = 0.05
7 phi = 1
8 n_grid = np.linspace(0, 1, 11)
9 # n_grid = np.array([1/4, 2/4, 3/4, 1])
10 n_n = len(n_grid)
11 z_grid = np.array([0.8, 1.0, 1.2])
12 n_z = len(z_grid)
13 pi = np.array([[0.20, 0.50, 0.30],
14               [0.10, 0.60, 0.30],
15               [0.25, 0.25, 0.50]])
16
17
18 # Criando lista de listas para incluir funções valor e política
19 # Note que, num modelo que considera incerteza, teremos uma função para cada z
20 v_it = [np.zeros((n_k, n_z))]
21 gk_it = [np.zeros((n_k, n_z))]
22 gn_it = [np.zeros((n_k, n_z))]
23
24
25 # Estabelecendo variáveis para realizar os loops
26 tol_norma = 1e-5 # Distância entre funções máxima para considerar convergência
27 tol_it = 500 # Número máximo de iterações (caso não convirja antes)
28 norma = np.inf # Valor apenas para entrar no loop
29 it = 0 # N iterações (antes utilizamos n - agora usado para trabalho)
30
31
32 while norma > tol_norma and it < tol_it:
33     it += 1 # Atualizando o número da iteração
34
35     # Criando objetos para preencher com funções objetivo, valor e política
36     f_obj = np.zeros((n_k, n_z, n_k, n_n)) # Lista com n_k matrizes n_k x z
37     Tv = np.zeros((n_k, n_z))
38     gn = np.zeros((n_k, n_z))
39     gk = np.zeros((n_k, n_z))
40
41     for i_k, k in enumerate(k_grid):
42         for i_z, z in enumerate(z_grid):
43             for i_kk, kk in enumerate(k_grid):
44                 for i_n, n in enumerate(n_grid):
45                     c = z*(k**alpha * n**(1 - alpha)) + (1 - delta)*k - kk
46                     if c > 0:
47                         Ev = np.dot(pi[i_z, :], v_it[it - 1][i_kk, :])
48                         f_obj[i_k, i_z, i_kk, i_n] = log(c) - (n**(1 + phi) / 1 + phi
49                     ) + beta*Ev
50                     else:
51                         f_obj[i_k, i_z, i_kk, i_n] = -np.inf

```

<sup>3</sup>Usar o zero para capital não é uma boa ideia.

```

52     # Preenchida uma matriz kk x n, encontrar elemento que maximiza
53     indice_max = np.argmax(f_obj[i_k, i_z])
54     indice_gk = indice_max // n_n # Divisão Inteira - Índice de k'
55     indice_gn = indice_max % n_n # Resto da Divisão - Índice de n
56
57     Tv[i_k, i_z] = np.max(f_obj[i_k, i_z])
58     gk[i_k, i_z] = indice_gk
59     gn[i_k, i_z] = indice_gn
60
61
62     # Após preencher totalmente gk e gn, trocar índices pelos valores nos grids
63     for p in range(len(gk)):
64         for q in range(len(gk[0])):
65             gk[p, q] = k_grid[int(gk[p, q])]
66             gn[p, q] = n_grid[int(gn[p, q])]
67
68     v_it.append(Tv)
69     gk_it.append(gk)
70     gn_it.append(gn)
71
72     norma = np.max(abs(v_it[it] - v_it[it - 1]))
73     print('A iteração {} terminou com norma igual a {:.5f}'.format(it, norma))

```

```

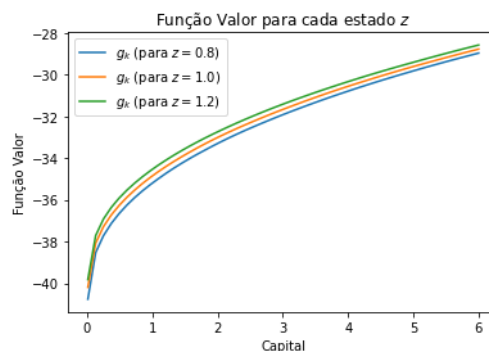
1 A iteração 1 terminou com norma igual a 3.32584
2 A iteração 2 terminou com norma igual a 2.94468
3 (...)
4 A iteração 247 terminou com norma igual a 0.00001
5 A iteração 248 terminou com norma igual a 0.00001

```

```

1 """ Visualização Gráfica da Função Valor """
2 fig, ax = plt.subplots()
3
4 # Inclusão de cada função valor no gráfico
5 ax.plot(k_grid, v_it[it][:,0], label='$g_k$ (para $z = 0.8$)')
6 ax.plot(k_grid, v_it[it][:,1], label='$g_k$ (para $z = 1.0$)')
7 ax.plot(k_grid, v_it[it][:,2], label='$g_k$ (para $z = 1.2$)')
8
9 # Legendas
10 ax.set_xlabel('Capital')
11 ax.set_ylabel('Função Política')
12 ax.set_title('Função Política e $g(k)$ estacionário')
13 ax.legend()

```



```

1 """ Visualização Gráfica da Função Política do Capital """
2 fig, ax = plt.subplots()
3
4 # Inclusão de cada função valor no gráfico
5 ax.plot(k_grid, gk_it[it][:,0], label='$g_k$ (para $z = 0.8$)')
6 ax.plot(k_grid, gk_it[it][:,1], label='$g_k$ (para $z = 1.0$)')
7 ax.plot(k_grid, gk_it[it][:,2], label='$g_k$ (para $z = 1.2$)')
8 ax.plot(k_grid, k_grid, '--', label='45 graus')

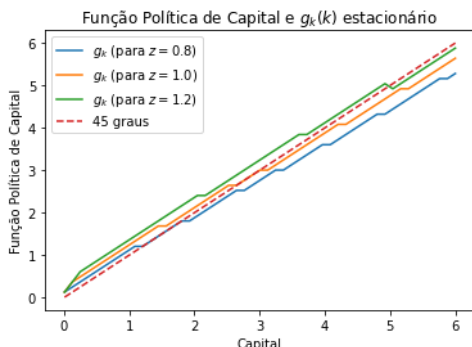
```



```

9
10 # Limites do gráfico
11 # ax.set_ylim([4.5, 6]) # tamanho mínimo e máximo vertical
12 # ax.set_xlim([4.5, 6]) # tamanho mínimo e máximo horizontal
13
14 # Legendas
15 ax.set_xlabel('Capital')
16 ax.set_ylabel('Função Política')
17 ax.set_title('Função Política e  $g(k)$  estacionário')
18 ax.legend()
19
20 plt.show()

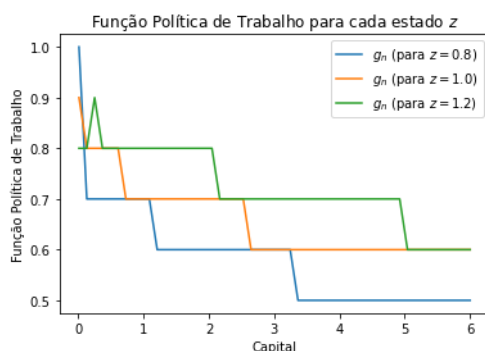
```



```

1 """ Visualização Gráfica da Função Política do Trabalho """
2 fig, ax = plt.subplots()
3
4 # Inclusão de cada função valor no gráfico
5 ax.plot(k_grid, gn_it[it][:,0], label='$g_n$ (para $z = 0.8$)')
6 ax.plot(k_grid, gn_it[it][:,1], label='$g_n$ (para $z = 1.0$)')
7 ax.plot(k_grid, gn_it[it][:,2], label='$g_n$ (para $z = 1.2$)')
8
9 # Legendas
10 ax.set_xlabel('Capital')
11 ax.set_ylabel('Função Política')
12 ax.set_title('Função Política e  $g(k)$  estacionário')
13 ax.legend()

```



```

1 """ Cálculo dos capitais estocásticos """
2 for i_z, z in enumerate(z_grid):
3     i_k = 0
4     loop = 0
5     # Achar índice do capital estacionário
6     while k_grid[i_k] != gk_it[it][i_k, i_z] and loop < 100: # até termos k = k'
7         i_k = np.where(k_grid == gk_it[it][i_k, i_z])[0][0] # Aplica o índice de k'
8         loop += 1 # Inserido por loops infinitos próximo ao k estacionário
9
10    print('O capital estacionário para z = {} é {:.3f}'.format(z, k_grid[i_k]))

```

```
1 0 capital estacionário para  $z = 0.8$  é 1.208
2 0 capital estacionário para  $z = 1.0$  é 2.646
3 0 capital estacionário para  $z = 1.2$  é 5.042
```