Programa de Pós-Graduação em Economia - FEARP/USP

I Exame de Macroeconomia II

Professores: Jefferson Bertolai & Fábio Gomes - 06 de outubro de 2016
Monitor: Matheus Melo

Problem 1 (**3.0 pontos**). Considere o problema do planejador no modelo de Crescimento Ótimo com um setor apresentado em sala. Conforme discutido durante o curso, tal problema pode ser escrito como

$$v^*(k_0) \equiv \sup_{\{k_{t+1}\}_{t=0}^{\infty}} \left\{ \sum_{t=0}^{\infty} \beta^t U[f(k_t) - k_{t+1}] \right\} \qquad s.t.\ 0 \leq k_{t+1} \leq f(k_t)\ e\ k_0\ dado \tag{1}$$

em que $f(k_t) = F(k_t, 1) + (1 - \delta)k_t$ denota a oferta total de bens no período t. Tal formato foi denomidado **problema sequencial** do planejador (SP). Associou-se ao problema sequencial (1) a equação funcional

$$v(k) = \sup_{0 \leq y \leq f(k)} \{U[f(k) - y] + \beta v(y)\}, \qquad \forall k \in \mathbb{R} \tag{2}$$

a qual define o que denominamos como **problema recursivo** do planejador (FE). A solução do problema (FE) é uma função v tal que $v = Tv$, em que o operador T é tal que

$$(Th)(x) = \max_{y \in \Gamma(x)} U[f(x) - y] + \beta h(y), \qquad \forall x \in \mathbb{X}. \tag{3}$$

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

Suponha $U(c) = \ln(c)$, $f(k) = \sqrt{k}$ e $\beta = 0.9$. Para responder aos itens a seguir, considere a aproximação discreta X (discretização) para o espaço de estados $\mathbb{X}$ tal que $X = \{0.0, 0.1, 0.2, \cdots, 0.9, 1.0\} \subset \mathbb{X}$.
Todos os programas precisam estar em linguaguem Matlab e prontos para rodar.

(a) Utilizando o Método de Discretização e o Método de Iteração da **Função Valor**, escreva um programa que calcule uma aproximação $\hat{v} : X \mapsto \mathbb{R}$ para a solução da equação funcional (2) e a função política associada $\hat{g} : X \mapsto X$.

- calcule $\hat{v}$ como o ponto fixo do Operador de Bellman $\hat{T}$ a seguir

$$(\hat{T}h)(x) = \max_{y \in \Gamma(x) \cap X} U[f(x) - y] + \beta h(y), \qquad \forall x \in X. \tag{4}$$

- a função $\hat{g}$ é tal que $\forall x \in X$

$$\hat{g}(x) = \arg \max_{y \in \Gamma(x) \cap X} U[f(x) - y] + \beta \hat{v}(y) = \{y \in \Gamma(x) \cap X : \hat{v}(x) = U(k, y) + \beta \hat{v}(y)\}.$$

- escreva seu programa como uma função (M-file), chamada `Bellman_Operator()`, que aceita como insumo (*input*) o valor para o parâmetro β e retorna como resultado (*output*) os vetores $\hat{v}$ e $\hat{g}$.

(b) Usando a função `Bellman_Operator()` obtida no item (a) e $\beta = 0.9$, escreva um programa que calcule a sequência $\{\hat{k}_t\}_{t=0}^{200}$ tal que $\hat{k}_0 = 0.1$ e $\hat{k}_{t+1} = \hat{g}(\hat{k}_t)$ para $t = 0, 1, 2, \cdots, 199$.

- escreva seu programa como uma função (M-file), chamada `Seq_Capital()`, que aceita como insumo (*input*) valores para β, $\hat{k}_0$ e T e retorna como resultado (*output*) o vetor $\hat{k} = [\hat{k}_0, \hat{k}_1, \cdots, \hat{k}_T]$.

(c) Usando a função `Seq_Capital()` obtida no item (b), $\beta = 0.9$, $\hat{k}_0 = 0.1$, escreva um programa que calcule o capital estacionário definido por $\hat{g}$, ou seja, que calcule $\bar{k}$ tal que $\bar{k} = \hat{g}(\bar{k})$.

(d) **(Bônus: 0.5 ponto adicional)** Uma forma de acelerar a convergência no método de iteração da função valor iterar o *guess* para a função valor para uma função política fixa (conforme estudado durante o curso no exercício 4.4 de Stokey, Lucas and Prescott (1989)). Reescreva a função `Bellman_Operator()` utilizando agora este artifício para acelerar a convergência.

Problem 2 (5.0 pontos)**.** Seja o operador T em $\mathbb{C}(\mathbb{X})$, espaço das funções limitadas e contínuas em $\mathbb{X}$, tal que

$$(Th)(x) = \max_{y \in \Gamma(x)} U(x, y) + \beta h(y), \qquad \forall x \in \mathbb{X} \text{ e para todo } h \in \mathbb{C}(\mathbb{X}). \tag{5}$$

Tal operador foi denominado Operador de Bellman. O resultado a seguir (discutido em aula) mostra que a solução da equação funcional (2) é única e pode ser calculada como o ponto fixo do Operador de Bellman, ou seja, $v = Tv$.

T.4.6 Seja $\mathbb{X}$, Γ, U e β tais que **(H.4.3)**[1] é **(H.4.4)**[2] sejam válidas. Seja $\mathbb{C}(\mathbb{X})$ o espaço das funções contínuas e limitadas $h : \mathbb{X} \mapsto \mathbb{R}$, com a norma do sup. Então,

(i) o operador T mapeia $\mathbb{C}(\mathbb{X})$ em si: $T : \mathbb{C}(\mathbb{X}) \mapsto \mathbb{C}(\mathbb{X})$.

(ii) T possui um, e somente um, ponto fixo v e $v \in \mathbb{C}(\mathbb{X})$

(iii) $\forall v_0 \in \mathbb{C}(\mathbb{X})$

$$||T^n v_0 - v|| \leq \beta^n ||v_0 - v||, \qquad n = 0, 1, 2, \ldots \tag{6}$$

(iv) dado v, a correspondência política ótima $G : \mathbb{X} \mapsto \mathbb{X}$ definida em (2) é *compact-valued* e uhc.

Para demonstrar os itens **(i)** e **(ii)** acima, responda aos itens a seguir:

(a) Para $x \in \mathbb{X}$ e $h \in \mathbb{C}(\mathbb{X})$ dados, seja $H : \mathbb{X} \mapsto \mathbb{R}$ tal que $H(y) := F(k, y) + h(y)$ para todo $y \in \mathbb{X}$.

- Note que H é uma função contínua, pois é a soma de duas funções contínuas.

Mostre que H é uma função limitada. Argumente então que pode-se concluir $H \in \mathbb{C}(\mathbb{X})$.

(b) Mostre que o máximo em (5) é atingido, ou seja, $\forall x \in \mathbb{X}$ e todo $h \in \mathbb{C}(\mathbb{X})$, $\exists y^* \in \Gamma(x)$ tal que $(Tf)(x) = U(x, y^*) + \beta h(y^*)$.

(c) Mostre que $\forall h \in \mathbb{C}(\mathbb{X})$ tem-se $Th \in \mathbb{C}(\mathbb{X})$ e, portanto, $T : \mathbb{C}(\mathbb{X}) \mapsto \mathbb{C}(\mathbb{X})$. Ou seja, o item **(i)** se verifica.

(d) Verifique que o operador T em (5) satisfaz as Condições Suficientes de Blackwell (monotonicidade e desconto) e, portanto, é uma contração.

[1] $\mathbb{X}$ é um conjunto convexo do $\mathbb{R}^l$ e a correspondência $\Gamma : \mathbb{X} \mapsto \mathbb{X}$ é não vazia, *compact-valued* e contínua.

[2] A função $U : A \mapsto \mathbb{R}$ é limitada e contínua e $\beta \in (0, 1)$.

(e) Mostre que T possui um, e somente um, ponto fixo v.

(f) Usando o fato de que $\mathbb{C}(\mathbb{X})$ é um espaço de Banach (espaço normado completo), mostre que o ponto fixo de T é uma função limitada e contínua, ou seja, $v = Tv \in \mathbb{C}(\mathbb{X})$. Ou seja, o item **(ii)** se verifica.

Problem 3 (3.0 pontos)**.** Considere a seguinte extensão do modelo de Gerações Sobrepostas (OLG) discutido em sala de aula. Os indivíduos vivem por 3 períodos, de forma que em cada data há três gerações de pessoas na economia: jovens, adultos e idosos. A utilidade instantânea do consumo é $u(c) = 1 - 1/c$ e a utilidade total (de toda a vida) do indivíduo é

$$U(c_1, c_2, c_3, h) = u(c_1) + u(c_2) + u(c_3) + h$$

em que c_i denota o consumo do indivíduo na idade $i \in \mathbb{I} := \{1, 2, 3\}$ e h denota a herança deixada para o (único) filho. O tamanho da população é (foi normalizado para) 1.
A restrição orçamentária na idade $i \in \mathbb{I}$ é $c_i + w_{i+1} \leq d + w_i$, em que w_i denota a riqueza na idade i (com $w_4 = h$) e d é um fluxo de dotação constante no tempo.
Suponha que os níveis de riqueza w_i possíveis são aqueles do conjunto $\mathbb{X} = \{0.0, 0.1, 0.2, \cdots, 0.9, 1.0\}$. Sabe-se que na solução ótima vale $c_i = d + w_i - w_{i+1}$ e, portanto, o problema do indivíduo com idade $i = 1$ e riqueza $w_1 = w \in \mathbb{X}$ é

$$v(w) := \max_{\{w_{i+1}\}_{i=1}^{3}} \left\{ \sum_{i=1}^{3} u(d + w_i - w_{i+1}) + w_4 \right\} \tag{ID}$$
$$\text{s.t.} \begin{cases} 0 \leq w_{i+1} \leq w_i, \quad \forall i \in \mathbb{I} \\ w_{i+1} \in \mathbb{X} \\ w_1 = w \text{ e } d > 0 \text{ dados} \end{cases}$$

Todos os programas precisam estar em linguaguem Matlab e prontos para rodar.

(a) Utilizando o Método de Indução Retroativa e $d = .5$, escreva um programa que calcule $v : \mathbb{X} \mapsto \mathbb{R}$ definida em (ID) e a função política associada $g : \mathbb{I} \times \mathbb{X} \mapsto \mathbb{X}$.

- lembre-se de que a função valor do método de indução retroativa $V : \mathbb{I} \times \mathbb{X} \mapsto \mathbb{R}$ é tal que $v(w) = V(1, w)$ para todo $w \in \mathbb{X}$.
- escreva seu programa como uma função (M-file), chamada `Backward_Induction()`, que aceita como insumo (*input*) o valor para o parâmetro d e retorna como resultado (*output*) o vetor v e a matriz g.

(b) Seja $p_t(i, w)$ a quantidade (proporção) de pessoas que no período t possui estado $(i, w) \in \mathbb{I} \times \mathbb{X}$. Ou seja, $p_t : \mathbb{I} \times \mathbb{X} \mapsto [0, 1]$ é a distribuição de pessoas no espaço de estados $\mathbb{I} \times \mathbb{X}$. A matriz de transição para esta economia é a matriz M tal que

$$P_{t+1} = M * P_t, \qquad \text{para todo } t = 0, 1, 2, 3, \cdots$$

em que $*$ denota multiplicação matricial e, para todo t, P_t é o vetor coluna

$$P_t = \begin{bmatrix} p_t(1,0.0) \\ p_t(1,0.1) \\ \vdots \\ p_t(1,1.0) \\ p_t(2,0.0) \\ p_t(2,0.1) \\ \vdots \\ p_t(2,1.0) \\ p_t(3,0.0) \\ p_t(3,0.1) \\ \vdots \\ p_t(3,1.0) \end{bmatrix}$$

Usando a função `Backward_Induction()` obtida no item (a) e $d = 0.5$, escreva um programa que calcule a matriz de transição da economia M.

- escreva seu programa como uma função (M-file), chamada `Markov_Chain()`, que aceita como insumo (*input*) o valor para o parâmetro d e retorna como resultado (*output*) a matriz M.

(c) Usando a função `Markov_Chain()` obtida no item (b) e $d = 0.5$, escreva um programa que calcule a distribuição estacionária de pessoas sobre $\mathbb{I} \times \mathbb{X}$, ou seja, que calcule o vetor P tal que $P = M * P$.

- construa uma sequência de distribuições $\{P_t\}_{t=0}^{T}$, com $P_0(j) = 1/33, \forall j = 1, 2, \cdots, 33$ e $T = 2$. Aumente T até o ponto em que $\sum_{j=1}^{33} |P_T(j) - P_{T-1}(j)| < 0.001$. Com isso, P_T será a (aproximação para a) distribuição estacionária P.
- escreva seu programa como uma função (M-file), chamada `SS_Distribution()`, que aceita como insumo (*input*) o valor para d e retorna como resultado (*output*) o vetor P.

(d) Suponha que a poupança dos indivíduos é feita em unidades de capital. A poupança (capital) agregada (per capita) pode ser usado no início de cada período para a produção de bens de consumo y por meio da função de produção $F(k, 1) = \sqrt{k}$. Usando a função `SS_Distribution()` obtida no item (c) e $d = 0.5$, escreva um programa que calcule a distribuição estacionária P, o nível estacionário de capital per capita $\bar{k}$ e de produto per capita $\bar{y}$ para esta economia.

- escreva seu programa como uma função (M-file), chamada `SS_GDP()`, que aceita como insumo (*input*) o valor para d e retorna como resultado (*output*) o número $\bar{y}$.

(e) **(Bônus: 0.5 ponto adicional)** este nível estacionário de produto $\bar{y}$ pode ser comparado com a dotação per capita d a fim de garantir factibilidade de da dotação. Ou seja, o nível de dotação per capita precisa ser igual ao nível de produto per capita: $d = y$. Usando a função `SS_GDP()` obtida no item (d), escreva um programa que calcule para esta economia: o produto estacionário $\bar{y}$ e a dotação estacionária d tais que $\bar{y} = d$.