Macroeconomía Internacional Cuantitativa

Francisco Roldán*

October 2021

a entregar no después del 20 de octubre

1. Problema de la torta

En clase vimos cómo resolver el problema asociado con la ecuación de Bellman

$$v(k) = \max_{k'} u\left(k(1+r) - k'\right) + \beta\mathbb{E}[v(k')]$$

1.1 *Funciones de consumo*

1. Para más práctica graficando: crear un objeto de tipo `CakeEating` con los parámetros por default, resolverlo usando `vfi!` o `vfi_itp!`
2. Usar `scatter` y `plot` para crear un gráfico de la función de consumo `ce.gc` como función del capital
   - Este gráfico no debería ser súper informativo (por qué?)
3. Para mostrar mejor el resultado, graficar c/k como función de k, la fracción de torta consumida como proporción de la torta que queda. (ayuda: usar la división lugar a lugar de dos vectores con el operador `x./y`)
4. También mostrar la función de ahorro `ce.gk` igual que la de consumo, dividiendo por el capital inicial.
   - Algo interesante que notar? (cuánto suman c/k y k'/k? esperabas eso?)

1.2 *Simulador de torta*

Escribir un simulador para el problema de la torta. Para esto

1. Elegir un tiempo máximo T, un estado inicial k_0.

*email: *froldan6@gmail.com*

2. Inicializar dos vectores $\mathcal{C}$ y $\mathcal{K}$ para guardar las sucesiones $\{c_t, k_t\}_{t=0}^T$.

3. Inicializar interpoladores para recuperar funciones de consumo y ahorro a partir de los vectores `gc` y `gk`.

4. Para cada $t \in \{0, \dots, T\}$, como ya sabemos k_t, usar la función de consumo para averiguar c_t y la función de ahorro para averiguar k_{t+1}. Guardar c_t y k_{t+1} como los elementos correspondientes de los vectores que preparamos.

5. Usar `scatter` para graficar el consumo a lo largo del tiempo y la torta que va quedando. Se parece al que vimos en clase?

Observación Como antes, es buena idea meter los pasos 2–4 en una función que tome como argumentos un problema `ce` (ya resuelto), T y k_0, y devuelva los vectores $\mathcal{C}$ y $\mathcal{K}$.

2. Fluctuación de ingresos

Al agregar un proceso estocástico para $\{y_t\}_t$, obtenemos un problema de fluctuación de ingresos con ecuación de Bellman

$$v(k, y) = \max_{k'} u(y + k(1+r) - k') + \beta \mathbb{E}\left[v(k', y') \mid y\right]$$

que se puede manejar con el código `ifp.jl`.

2.1 *Funciones de consumo*

Elegir la forma de mostrar $c(k, y)$ en un modelo resuelto. Ideas posibles (pero no exhaustivo): k en el eje x, distintas líneas para distintos valores de y, usando un vector de `scatters`; curvas de nivel como función de k y de y usando `contour`. Opcional: mostrame $c(k, y)/y$. Cómo cambia la propensión al consumo con el nivel de k?

2.2 *Simulador*

Escribir un simulador para el problema de fluctuación de ingresos. Todo es parecido al punto anterior pero tenés que decidir cómo hacer para sacar un ingreso aleatorio en cada período. (podés elegir si querés simular el AR(1) 'de verdad' e interpolar todo o simular la cadena de Markov)

Del simulador van a salir series $\{k_t, y_t, c_t\}_t$. Cuál es la distribución ergódica de c/y y de k? Podés mostrar histogramas y calcular la media y ciertos cuantiles de la distribución ergódica.

2.3 *Robustness – opcional*

Podemos modificar el problema del agente introduciendo preferencias por robustez. En este caso tendríamos la siguiente ecuación de Bellman

$$v(k, y) = \max_{k'} u(y + k(1+r) - k') + \beta \mathbb{T}_\theta \left[v(k', y') \mid y \right]$$

donde como antes $\mathbb{T}_\theta(X) = -\frac{1}{\theta} \log \mathbb{E}\left[\exp(-\theta X)\right]$.

Usando el simulador: Cómo cambia la distribución del capital al aumentar θ? Cómo cambia la distribución de c/y (la propensión promedio al consumo)?

Mirando las reglas de decisión: Cómo cambia la función de consumo en proporción al ingreso $c(k, y)/y$ (la propensión marginal al consumo en equilibrio parcial)?

3. Modelos de default

El primer bloque de la materia trata de modelos de default del tipo

$$\mathcal{V}(b, y) = \mathcal{P}(b, y) v^D((1-\hbar)b, y) + (1 - \mathcal{P}(b, y)) v^R(b, y)$$
$$\mathcal{P}(b, y) = \mathbb{P}\left(\xi < v^D((1-\hbar)b, y) - v^R(b, y)\right)$$

$$v^R(b, y) = \max_{b'} u(c) + \beta \mathbb{E}\left[\mathcal{V}(b', y') \mid y\right]$$
$$\text{sujeto a } c + \kappa b = y + q(b', y)(b' - (1-\rho)b)$$

$$v^D(b, y) = u((1-\Delta)y) + \beta \mathbb{E}\left[\theta \mathcal{V}(b, y') + (1-\theta) v^D(b, y)\right]$$

para ξ con alguna distribución práctica (que tenga media y mediana 0 ayuda; personalmente, me gusta $\xi \sim \mathcal{N}(0, \sigma_\xi)$).[1]

3.1 *Funciones de política*

Con los parámetros por default del tipo `Default`, mostrar la probabilidad de default en equilibrio como función de (b, y).

Comparar con

- El mismo modelo pero resuelto con costos de default más chicos (por ejemplo, $\Delta = 0.08$ en vez de 0.1). Cómo cambia el nivel de deuda al que hay default? Comente

[1] esto hace que (i) la probabilidad de default sea mayor cuanto mayor sea el valor de estar en default respecto de estar en repago, (ii) si $v^D = v^R$ la probabilidad de default sea $1/2$, (iii) la probabilidad de default sea simétrica. Al final, lo que importa es que dadas v^R y v^D, $\mathcal{V}$ pase 'por el medio' y sea lo más suave posible.

- El mismo modelo pero resuelto con haircuts $\hbar$ más altos. Cómo cambia el nivel de deuda al que hay default? Comente
- Finalmente, el mismo modelo pero con un factor de descuento más alto (más paciente). Es obvio para qué lado va a cambiar el equilibrio? Por qué?

3.2 *Simulaciones*

Usando el código `def_simul.jl` que compartí (o escribiendo el tuyo!) vamos a simular series de tiempo de estos modelos. El simulador que compartí devuelve un `Dict{Symbol, Vector{Float64}}` con entradas `:b` (para la deuda), `:y` (para la dotación de producto), `:c` (para el consumo), `:q` (para el precio observado de la deuda), `:spread` (para el precio expresado como spreads).

3.2.1 *Distribución de la deuda*

Para deuda de largo plazo (por ejemplo, $\rho = 0.35, \hbar = 0.5$) y de corto plazo ($\rho = 1, \hbar = 1$), resolver y simular por T períodos (por ejemplo, 100000). Finalmente mostrar en dos histogramas superpuestos cómo cambia la distribución del ratio deuda-producto al alargar la duración de la deuda.

3.2.2 *Correlaciones*

En la simulación del modelo con deuda de largo plazo, calcular momentos: varianza del consumo, varianza del producto, correlación entre producto y spreads. *Opcional*: usando `DataFrames, GLM`, tomar la regresión de spreads contra deuda y producto. Comente.

3.2.3 *Robustness*

Darle robustez al gobierno (deudor) con parámetro θ. Comparado con nuestro benchmark sin robustez, qué le pasa a la deuda/producto en equilibrio (en simulaciones)? Cuánta deuda puede sostener sin default? (mirando las reglas de decisión).

3.3 *Deuda contingente*

En el modelo básico, la deuda promete pagos que sólo varían en el tiempo. Deuda emitida en t paga $\kappa(1-\rho)^{s-1}$ en $t+s$. Esto da lugar a

1. Una restricción de presupuesto para el gobierno (en repago)

$$c + \kappa b = y + q(b', y)(b' - (1-\rho)b)$$

2. Una ecuación de precios para la deuda

$$q(b', y) = \frac{1}{1+r}\mathbb{E}\left[(1 - 1_D(b', y'))R(b'', y') + 1_D(b', y')(1-\hbar)q_D((1-\hbar)b', y')\right]$$

El objetivo de este punto es modificar la estructura de la deuda para indexar los cupones al ingreso de la economía. Para eso, dado $\alpha \in (0, 1]$, vamos a hacer que el cupón sea

$$\kappa(y) = \kappa(1 + \alpha(y - \bar{y}))$$

donde $\bar{y}$ es la media incondicional del proceso para la dotación (que solemos normalizar a 1).

Para resolver el modelo con deuda contingente, para dado α vas a necesitar modificar la estructura de pagos de la deuda (acordate que κ aparece en dos lugares).

Es mejor indexar la deuda al ingreso? Comparando la función de valor del problema con deuda indexada y sin indexar, si el gobierno pudiera elegir el α (por fuera del modelo), elegiría $\alpha = 0$ (deuda no contingente) o un α positivo? (no te pido que encuentres el α óptimo pero podés buscarlo). Cuál dirías que es la ventaja de indexar la deuda en este caso?

3.4 *Robustness – muy opcional*

Algo interesante en este modelo es darles a los *acreedores* preferencias por robustez. Esto va a afectar el precio de la deuda (ya que los acreedores robustos piensan que estados malos para ellos, como el default, son más probables por el temita del pesimismo).

La modificación que hay que hacerle al modelo viene por este lado: el precio de la deuda es ahora

$$q(b', y) = \frac{1}{1+r}\mathbb{E}\left[\frac{\exp(-\theta_L v_L(b', y', d'))}{\mathbb{E}\left[\exp(-\theta_L v_L(b', y', d')) \mid y\right]}\text{el valor del repago en } (b', y', d')\right]$$

La clave está en que el factor de descuento estocástico de los acreedores depende de su propia función de valor descripta con la deuda b que el deudor les debe, el ingreso y, y una indicadora de si la deuda está o no en default. Esta función de valor toma en cuenta las políticas percibidas que va a seguir el país deudor (igual que antes cuando q dependía de b'')

$$\begin{aligned} v_L(b, y, d) &= c_L + \beta_L \frac{-1}{\theta_L} \log \mathbb{E}\left[\exp(-\theta_L v_L(b', y', d'))\right] \\ \text{sujeto a } c_L &= w_L + \kappa b - q(b', y)(b'(b, y) - (1-\rho)b) \quad \text{si la deuda está pagando} \\ c_L &= w_L \quad \text{en default} \end{aligned}$$

donde w_L es un número fijo (e irrelevante, al ser las preferencias neutrales frente al riesgo más allá de la robustez) y (muy importante!) el acreedor toma la estrategia del deudor como dada (en particular cuánta deuda le va a vender).

Siguiendo la lógica del 'estilo teoría de juegos', un algoritmo sugerido es

- Dejando fija la estructura de `mpe!`, reemplazar `q_iter!` por

- Una iteración sobre la función de valor de los acreedores. Esta iteración toma como dadas (*i*) la estrategia de default del deudor, (*ii*) la estrategia de emisión de nueva deuda del deudor, (*iii*) la estrategia de precios de la deuda de hoy

- Una iteración sobre el precio de la deuda en sí. Dada la función de valor en el paso anterior, podemos calcular el núcleo del factor de descuento estocástico $\exp(-\theta_L v_L(b', y', d'))$ en cada estado de mañana (b', y', d') incluyendo el default. Dado ese núcleo podemos calcular $q(b', y)$ básicamente igual que en el modelo base pero multiplicando el repago en cada estado de mañana por el núcleo. Al mismo tiempo, vamos llevando las sumas parciales del núcleo para al final del loop tener $\mathbb{E}\left[\exp(-\theta_L v_L(b', y', d'))\right]$ para poder dividir la suma total.

Una buena referencia (conceptual) para este modelo es Pouzo and Presno (2016). Ojo que ellos usan otro algoritmo para resolver el modelo (una serie de trucos que hacen que el 'estilo equilibrio general' funcione) así que yo no lo leería buscando pistas de cómo estructurar este ejercicio.

References

Pouzo, Demian, and Ignacio Presno. 2016. "Sovereign Default Risk and Uncertainty Premia." *American Economic Journal: Macroeconomics* 8 (3): 230–66. 10.1257/mac.20140337.