

Stochastic Two Period OLG Model

Prof. Lutz Hendricks

Econ821

January 26, 2016

Contents

Introduction	3
Household	4
Computing the Household Problem	7
Stationary Equilibrium	11
Exercise: Pseudo-code for Algorithm	13
Calibration Algorithm	15
Calibration Results	18
References	20

Introduction

We build up towards solving a stochastic, heterogeneous agent OLG model like [Huggett \(1996\)](#).

Building on the deterministic model, we change:

- Households draw random labor endowments at each age: $e^a \in \{e_1^a, \dots, e_{n_w}^a\}$.
- Endowment draws are independent over time: $\Pr(e^a = e_j^a) = P_e^a(j)$.
- Incomes received are then: $y_j^a = e_j^a w^a$.

Young wage rate is determined by firms' marginal product of labor.

Old "wage rate" is set by government as a transfer.

Household

$$\max u(c_t^y) + E_t \beta u(c_{t+1}^o)$$

subject to

$$c_{t+1}^o - y_{t+1}^o = (1 + r_{t+1}) (y_t^y - c_t^y) \quad (1)$$

where

$$E_t u(c_{t+1}^o) = \sum_{j=1}^{n_w} P_e^o(j) u(c_{t+1}^o(y_j^o)) \quad (2)$$

Timing: Realizations of y are known before consumption is chosen.

Lagrangian

$$\Gamma(y_t^y) = \max u(c_t^y) + \beta E_t u \left([1 + r_{t+1}] [y_t^y - c_t^y] + y_{t+1}^o \right)$$

Euler equation

$$u'(c_t^y) = \beta (1 + r_{t+1}) E_t u' \left([1 + r_{t+1}] [y_t^y - c_t^y] + y_{t+1}^o \right) \quad (3)$$

Example: Certainty Equivalence: $u(c) = a c - b c^2$

Euler equation becomes

$$a - b c_t^y = \beta (1 + r_{t+1}) E_t \{a - b c_{t+1}^o\} \quad (4)$$

$$a - b c_t^y = \beta (1 + r_{t+1}) [a - b (1 + r_{t+1}) (y_t^y - c_t^y) - b E_t \{y^o\}] \quad (5)$$

Rearrange

$$c_t^y = \frac{[1 - \beta (1 + r)] a / b + \beta (1 + r)^2 E_t \{W_t\}}{1 + \beta (1 + r)^2}$$

Implications:

- *Certainty equivalence:* Consumption only depends on the expected value of lifetime income, $E_t \{W_t\}$.
- Marginal propensity to consume is the same out of present income and out of expected future income.
- Consumption itself is a martingale.

Computing the Household Problem

For this model, the method we used for the deterministic case still works.
Search over values of s to find the zero of the Euler equation

$$u'(y^y - s) = \beta [1 + r] \sum_j P_e^o(j) u'(y_j^o + [1 + r] s)$$

Result: Saving function $s(y^y)$. This is a $(1 \times n_w)$ vector.

For multi-period models there are two complications:

- the consumption function for $t + 1$ is not known (but can be found by backward induction).
- wealth is an additional, continuous state variable (this is handled by interpolation).

Household Dynamic Program

For more complicated models it is convenient to write the household problem recursively:

$$V^o(y^o, s) = u(y^o + [1 + r] s) \quad (6)$$

$$V^y(y^y) = \max u(y^y - s) + \beta E \{V^o(y^o, s)\} \quad (7)$$

First-order conditions

$$V_s^o(y^o, s) = [1 + r] u'(y^o + [1 + r] s) \quad (8)$$

$$u'(y^y - s) = \beta E \{V_s^o(y^o, s)\} \quad (9)$$

The end result is the same.

Exercise: Pseudo code for Household Problem

Algorithm for Household Problem

Take as given: parameters, prices.

1. Calculate marginal utility when old from the budget constraint:

$$MU^o(y^o, s) = u'(y^o + [1 + r] s)$$

To compute this, approximate on a grid for wealth: $s \in \{s_1, \dots, s_{n_s}\}$.

Then MU^o is a $(n_w \times n_s)$ matrix.

2. Compute expected marginal utility for every wealth level:

$$EMU^o(s_i) = \sum_{j=1}^{n_w^o} P_e^o(j) MU^o(y^o, s_i).$$

EMU^o is a $(1 \times n_s)$ vector.

3. Search for a zero of the Euler equation deviation

$$y^y - s = (u')^{-1}(\beta [1 + r] EMU^o(s)) \quad (10)$$

hh_solve_olg2s.m

Note: The only change from the deterministic case is that marginal utility is replaced by EMU^o .

Stationary Equilibrium

Objects:

- Scalars: G, K, L, r, w .
- Policy functions: $c^y(y^y), s(y^y), c^o(s, y^o)$.
- A distribution of households over types; density $\Lambda(y^y, y^o)$.

Steady state conditions:

In this simple model the density is exogenous: $\Lambda(y_j^y, y_i^o) = P_e^y(j) \cdot P_e^o(i)$.

Policy functions solve the household problem.

Firm:

$$r = (1 - \tau_r) F_K(K, L) - \delta \quad (11)$$

$$w = (1 - \tau_w) F_L(K, L) \quad (12)$$

Market clearing:

$$L = N \sum_j P_e^y(j) e_j^y \quad (13)$$

$$K(1 + n) = N \sum_j P_e^y(j) s(y_j^y) \quad (14)$$

$$G + \sum_j P_e^o(j) y_j^o N / (1 + n) = \tau_w F_L(K, L) L + \tau_r F_K(K, L) K \quad (15)$$

Steady State in Per Capita Terms

Scalars: $g = G/N, k = K/L, l = L/N, r, w$.

Steady state conditions:

$$r = (1 - \tau_r) f'(k) - \delta \quad (16)$$

$$w = (1 - \tau_w) [f(k) - f'(k) k] \quad (17)$$

$$l = \sum_j P_e^y(j) e_j^y \quad (18)$$

$$s(y^y) = y^y - c^y(y^y) \quad (19)$$

$$c^o(s, y^o) = y^o + (1 + r) s \quad (20)$$

$$k l (1 + n) = \sum_{j=1}^{n_e^y} P_e^y(j) s(y_j^y) \quad (21)$$

$$g + (1 + n)^{-1} \sum_{j=1}^{n_e^o} P_e^o(j) y_j^o = \tau_w [f(k) - f'(k) k] l + \tau_r f'(k) k l \quad (22)$$

Note: l is exogenous.

Exercise: Pseudo-code for Algorithm

Guess k .

Compute r, w from steady state conditions.

Solve household problem for policy functions for every e_j^y .

Compute aggregate k from household savings.

Iterate until deviation from capital market clearing is close to zero.

Program: `bg_comp_olg2s.m`.

Computing Aggregate Capital

Direct method: In this simple model, the distribution of savings is known. Aggregate saving can be computed directly as

$$k_l(1+n) = \sum_{j=1}^{n_e^y} P_e^y(j) s(y_j^y)$$

Simulation method: In multi-period models, we don't know the distribution of savings. Then we simulate a large number of households, starting from age 1 forward.

Compute aggregate wealth as: [mean of savings over all simulated households] \times [mass of households].

Calibration Algorithm

Fix k at the value implied by the target level for K/Y : $k = (AK/Y)^{1/(1-\alpha)}$

Compute r, w from marginal products.

Guess β .

Solve household problem for $s(y^y)$.

Iterate until deviation from capital market clearing is small.

Program: cal_comp_olg2s.m.

Parameters

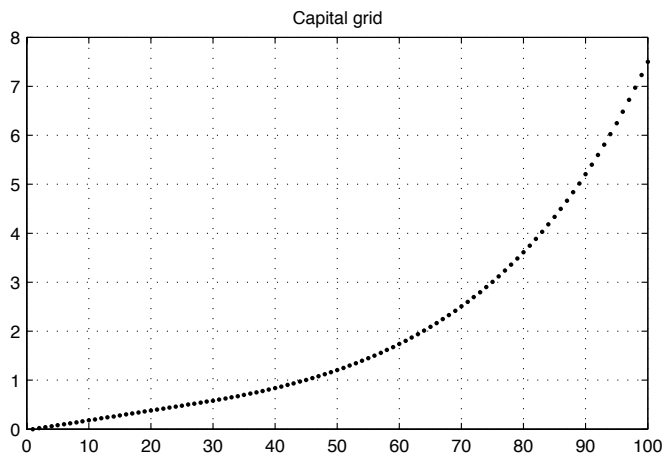
By default, parameters are the same as in the deterministic model.

Labor endowments: Should be calibrated to match earnings processes. For now: set to arbitrary values.

Capital grid: Grid must be tight at low s where marginal utility may be highly nonlinear. Number of grid points must be large enough so that approximation errors are small: $n_k = 100$.

`set_kgrid_olg2s.m` sets a capital grid with constant steps for low s and constant exponential steps for high s .

Capital Grid



Calibration Results

```
>> cal_comp_olg2sS(1, 0, 111);
```

Deviation from calibration targets: 0.000000
beta = 2.284. Annual beta = 1.028
k = 0.151.

Verify that steady state computation returns desired K/Y :

```
>> bg_comp_olg2sS(1, 0, 111);
```

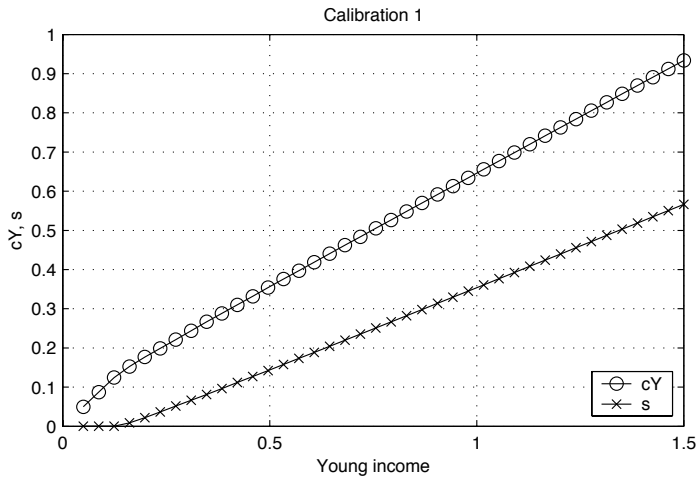
Steady state. calNo = 1. expNo = 0.
k = 0.151. y = 1.563.
r = 3.322. w = 1.000.
cY/wY = 0.645. s/y = 0.227
Deviation = 0.000000
Steady state K/Y = 0.097. Target: 0.097.

Verify that the model replicates the certainty case: calNo = 50.

Consumption Function

Solve household problem for various levels of young income.

`show_cons_fct_olg2s.m`



Consumption function is nearly linear.

References

Heer and Maussner (2008) contains lots of detail on computing OLG models.

Guvenen (2011): a practical guide.

References

- GUVENEN, F. (2011): "Macroeconomics with Heterogeneity: A Practical Guide," *Economic Quarterly-Federal Reserve Bank of Richmond*, 97, 255.
- HEER, B. AND A. MAUSSNER (2008): "Dynamic General Equilibrium Modelling Springer Berlin," *Heidelberg, New York*.
- HUGGETT, M. (1996): "Wealth distribution in life-cycle economies," *Journal of Monetary Economics*, 38, 469–494.