

ECON 712B - Problem Set 3

Alex von Hafften*

12/3/2020

1. An endowment economy consists of two types of consumers indexed by $i = 1, 2$. There is one nonstorable consumption good. Consumer 1 has initial endowment stream $e_0^1 = 1$ while consumer 2 has initial endowment $e_0^2 = 0$. At some random date $s > 0$ the endowments will switch so $e_t^1 = 0$ and $e_t^2 = 1$ for all $t \geq s$. Suppose the switch happens with probability δ each period until it is realized. Both consumers have preferences ordered by $E_0 \sum_{t=0}^{\infty} \beta^t \log c_t^i$. For this problem you can either consider an Arrow-Debreu market structure or a recursive equilibrium with a sequential market structure. For the Arrow-Debreu market, consider a single market at date zero with claims to consumption in any future date and any state. For the recursive structure, consider a market each period in claims to the two "trees," and use backward induction to find the values after and then before the switch of the endowments.
 - (a) Define a competitive equilibrium for this economy for your choice of market structure, being explicit about the objects which make up the equilibrium and the conditions they must satisfy.
 - (b) Compute a competitive equilibrium, finding the consumption allocation and the prices (as of date zero) of claims to each consumer's endowment process.
 - (c) What is price of a claim to the aggregate endowment? What is the risk-free interest rate?
 - (d) Now suppose that $\lambda \in (0, 1)$ is the Pareto weight on consumer 1 with $1 - \lambda$ the weight on consumer 2. Formulate the social planner's problem and characterize the solution of the Pareto optimal allocation for a given λ .
 - (e) Relate the competitive equilibrium that you found earlier with the Pareto optima. Is the competitive equilibrium optimal? (Can you find a particular λ so that they are equivalent?) For a given λ , can you find a way to decentralize the Pareto optimum as a competitive equilibrium?

*I worked on this problem set with a study group of Michael Nattinger, Andrew Smith, and Ryan Mather. I also discussed problems with Emily Case, Sarah Bass, and Danny Edgel.

2. Suppose that a representative agent has preferences: $E \sum_{t=1}^{\infty} \beta^t u(c_t)$ over the single nonstorable consumption good (“fruit”), where $u(c) = c^{1-\gamma}/(1-\gamma)$ for $\gamma > 0$. Her endowment of the good is governed by a Markov process with transition function $F(x', x)$.
 - (a) Define a recursive competitive equilibrium with a market in claims to the endowment process (“trees”).
 - (b) When the consumer has logarithmic utility, $u(c) = \log c$, what is the equilibrium price/dividend ratio of a claim to the entire consumption stream? How does it depend on the distribution of consumption growth?
 - (c) Suppose there is news at time t that future consumption will be higher. How will prices respond to this news? How does this depend on the consumer’s preferences (which are CRRA, but not necessarily log)? Interpret your results.
 - (d) Consider an option which is bought or sold in period t and entitles the current owner to exercise the right to buy one “tree” in period $t + 1$ at the fixed price \bar{p} specified at date t . (The buyer may choose not to exercise this option.) Find a formula for the price of this option in terms of the parameters describing preferences and endowments.

3. Suppose consumers have preferences given by: $E \sum_{t=0}^{\infty} \beta^t \frac{c_t^{1-\gamma}}{1-\gamma}$ with $\beta \in (0, 1)$ and $\gamma > 1$. The consumer's budget constraint is: $c_t + a_{t+1} = wl_t + (1+r)a_t$ where l_t is the labor endowment, a_t is the amount of assets, $c_t \geq 0$ is consumption, $w > 0$ is the (constant) wage, and $r > 0$ is the (constant) real interest rate. The consumer also faces the borrowing constraint $a_t \geq 0$. Suppose that the labor endowment follows a finite-state Markov chain with transition matrix Q .

(a) Write down the consumer's Bellman equation, and state the optimality conditions.

$$V(a, l) = \max_{(c, a') \in \Gamma(a, l)} \left\{ u(c) + \beta \int V(a', l') Q(l, dl') \right\}$$

$$\implies V(a, l) = \max \left\{ u(wl + (1+r)a - a') + \beta \int V(a', l') Q(l, dl') \right\}$$

FOC:

$$0 = (-1)u'(wl + (1+r)a - a') + \beta \int V_a(a', l') Q(l, dl')$$

$$\implies u'(c) = \beta \int V_a(a', l') Q(l, dl')$$

Envelope condition:

$$V_a(a, l) = u'(wl + (1+r)a - a')(1+r)$$

$$\implies V_a(a, l) = (1+r)u'(c)$$

FOC and envelope condition implies a stochastic Euler equation:

$$u'(c) = \beta(1+r) \int u'(c', l') Q(l, dl')$$

$$\implies u'(c_t) = \beta(1+r) E_t u'(c_{t+1})$$

Plugging in the functional form for the utility function:

$$c_t^{-\gamma} = \beta(1+r) E_t c_{t+1}^{-\gamma}$$

- (b) The rest of the question will numerically solve for the policy functions and the stationary distribution. Suppose that l_t takes on the following values: $l_t \in \{0.7, 1.1\}$, with the transition matrix given by: $Q = \begin{bmatrix} 0.85 & 0.15 \\ 0.05 & 0.95 \end{bmatrix}$, where $Q(i, j) = P(l_{t+1} = l_j | l_t = l_i)$. Find the stationary distribution and the unconditional mean of the labor endowment.

The stationary distribution of a Markov Chain is π such that $\pi Q = \pi$. Thus we find the eigenvalues and eigenvectors of Q . The characteristic polynomial of Q is

$$(0.85 - \lambda)(0.95 - \lambda) - (0.15)(0.05) = \lambda^2 - 1.8\lambda + 0.8 = (\lambda - 1)(\lambda - 0.8)$$

So the eigenvalues are $\lambda_1 = 1$ and $\lambda_2 = 0.8$. For λ_1 ,

$$0.85x + 0.15y = x$$

$$0.95x + 0.05y = y$$

These equations imply that $(0.5, 0.5)'$ is a coresponding eigenvector. For λ_1 ,

$$0.85x + 0.15y = 0.8x$$

$$0.95x + 0.05y = 0.8y$$

These equations imply that $(0.25, 0.75)'$ is the coresponding eigenvector. Double-checking, we see that $(0.25, 0.75)'$ is the stationary distribution:

$$\begin{bmatrix} 0.25 \\ 0.75 \end{bmatrix} \begin{bmatrix} 0.85 & 0.15 \\ 0.05 & 0.95 \end{bmatrix} = \begin{bmatrix} 0.25 \\ 0.75 \end{bmatrix}$$

The unconditional mean of labor endowment is $(0.25)(0.7) + (0.75)(1.1) = 1$.

```
Q <- matrix(c(0.85, 0.05, 0.15, 0.95), nrow = 2)
```

```
# Numerical estimation
```

```
x <- c(0.5, 0.5)
```

```
for (i in 1:100) x <- x %*% Q
```

```
x
```

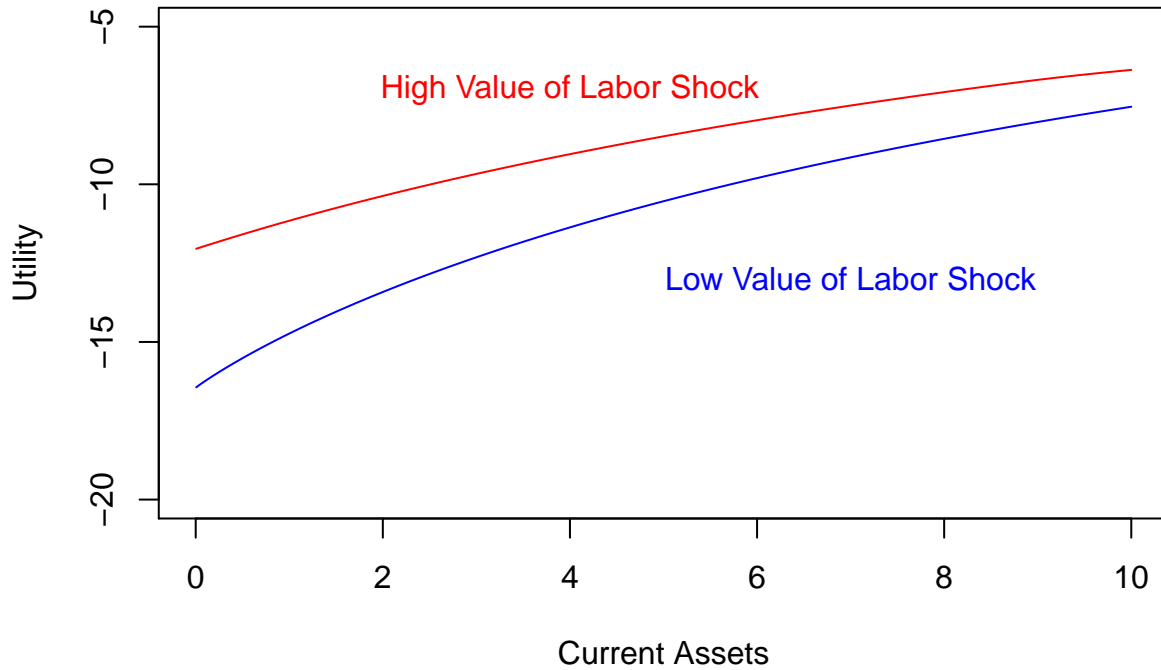
```
##      [,1] [,2]
```

```
## [1,] 0.25 0.75
```

- (c) Take the following as parameters: $\beta = 0.95, \gamma = 3, r = 0.03, w = 1.1$. Discretize the asset choice into a finite grid. Note that this directly restricts the assets to lie in a compact set. Numerically solve for the value functions by iterating on the Bellman equation.
- i. Plot the value functions for the high and low values of the labor shocks. Do they appear to have the properties that we have assumed?

Yes, the value function for the high labor shocks is larger than the value function for low labor shocks. The difference decreases as the level of current assets increases.

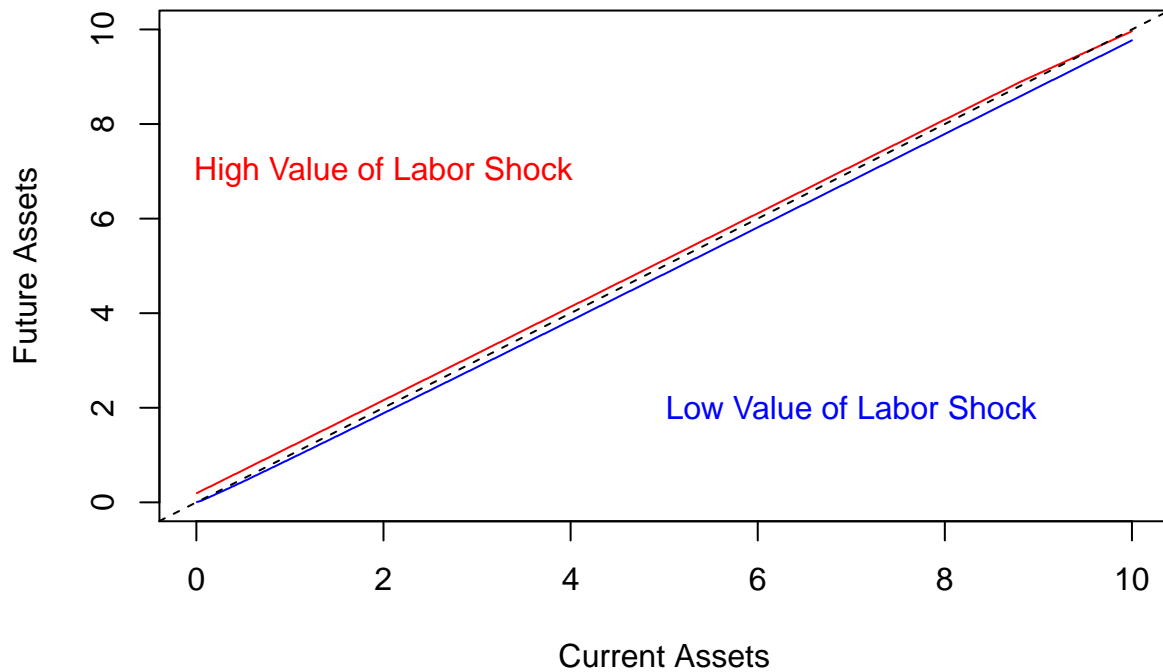
Value Function



- ii. Plot the asset holdings decision rules $a'(a, l_i)$ for the two labor shocks. Is there a value \bar{a} such that $a'(a, l_2) < a$ for all $a \geq \bar{a}$? What do the features of the decision rules suggest about the long-run behavior of assets a_t ?

```
## # A tibble: 1 x 3
##       a a_prime   c
##   <dbl> <dbl> <dbl>
## 1  0.01   0.01 0.770

## # A tibble: 10 x 3
##       a a_prime   c
##   <dbl> <dbl> <dbl>
## 1  9.62   9.62  1.50
## 2  9.62   9.62  1.50
## 3  9.63   9.63  1.50
## 4  9.63   9.63  1.50
## 5  9.64   9.64  1.50
## 6  9.64   9.64  1.50
## 7  9.65   9.65  1.50
## 8  9.65   9.65  1.50
## 9  9.66   9.66  1.50
## 10 9.66   9.66  1.50
```



- (d) To find the stationary distribution over assets and labor, first construct a (large) transition matrix over (a, l) . Then starting from an arbitrary initial distribution, iterate until convergence on the mapping from the current distribution into the distribution in the next period. What is the mean of the stationary asset distribution? Plot the (marginal) asset distribution.

Appendix

```
# parameters
beta <- .95
gamma <- 3
r <- 0.03
w <- 1.1
l_low <- 0.7
l_high <- 1.1

# assets grid
min_a <- 0.005
max_a <- 10
inc_a <- 0.005
a <- seq(min_a, max_a, by = inc_a)
n <- length(a)

ones <- rep(1, times=n)

# initialize value grid and decision rule grid
v <- rep(0, times=n)
decision <- rep(0, times=n)

# create consumption and utility matrixes for high and low labor shocks
# (columns are different values of a; rows are values of a')
c_matrix_low <- w * l_low + (1 + r)*ones %*% t(a) - a %*% t(ones)
c_matrix_high <- w * l_high + (1 + r)*ones %*% t(a) - a %*% t(ones)

utility_matrix_low <- (c_matrix_low^(1-gamma))/(1-gamma)
utility_matrix_high <- (c_matrix_high^(1-gamma))/(1-gamma)

utility_matrix_low[c_matrix_low < 0|is.finite(utility_matrix_low)] <- -1000
utility_matrix_high[c_matrix_high < 0|is.finite(utility_matrix_high)] <- -1000

# solve bellman for low labor shocks
test <- 10
iter <- 1
while (test != 0) {
  # create value matrix for all different a and a' values
  value_matrix <- utility_matrix_low + beta * v %*% t(ones)

  # find column max with max utility
  tv <- apply(value_matrix, 2, max)
  tdecision <- apply(value_matrix, 2, which.max)

  # loop ends if no changes from previous iteration
  test <- max(tdecision - decision)

  # update value grid and decision rule
  decision <- tdecision
  v <- tv
  iter <- iter + 1
}
```



```

# capital and consumption decision rule
policy_function_low <- tibble(a = a, a_prime = min_a + decision*inc_a) %>%
  mutate(c = w * l_low + (1 + r)*a - a_prime)

v_low <- v

# solve bellman for high labor shocks
test <- 10
iter <- 1
while (test != 0) {
  # create value matrix for all different a and a' values
  value_matrix <- utility_matrix_high + beta * v %*% t(ones)

  # find column max with max utility
  tv <- apply(value_matrix, 2, max)
  tdecision <- apply(value_matrix, 2, which.max)

  # loop ends if no changes from previous iteration
  test <- max(tdecision - decision)

  # update value grid and decision rule
  decision <- tdecision
  v <- tv
  iter <- iter + 1
}

# capital and consumption decision rule
policy_function_high <- tibble(a = a, a_prime = min_a + decision*inc_a) %>%
  mutate(c = w * l_high + (1 + r)*a - a_prime)

v_high <- v

```