FNCE 924 — Problem Set 2

Felix Nockher and Patrick Shultz

March 13, 2019

Part I: Asset Prices with Complete Markets

Consider the Lucas (1978) asset pricing model and assume consumers have CRRA preferences $\left(u(c_t) = \frac{c_t^{1-\gamma}}{1-\gamma}\right)$ with risk aversion γ and discount rate β. Suppose that income/dividend/endowment growth obeys

$$\log(y_{t+1}/y_t) = g + \epsilon_{t+1}$$

where ϵ_{t+1} is i.i.d. $N(0, \sigma^2)$

a) Compute the market price of a risk free bond that is purchased in period t and pays exactly one unit of consumption in period $t+1$.

The Lucas asset pricing model implies identical agents that can trade a stock and a bond. Shares in the aggregate endowment are given by $\theta^s(S^t) \in \Theta^s$ and shares of the bond are given by $\theta^b(S^t) \in \Theta^b$. Each household solves the following problem

$$V(\theta^s, \theta^b, S) = \max_{\{c, \theta'^s, \theta'^b\}} \left\{ u(c) + \beta \int_S V(\theta'^s, \theta'^b, S') F(S, dS') \right\}$$

subject to the constraint $c + \theta'^s Q^s + \theta'^b Q^b = [y + Q^s]\,\theta^s + \theta^b$. Letting $\lambda(S)$ denote the Lagrange multiplier in state S, we get the following first order conditions

$$u_c(c(\theta^s, \theta^b, S)) - \lambda(S) = 0$$
$$\beta \int_S V_{\theta^b}(\theta'^s, \theta'^b, S') F(S, dS') - \lambda(S) Q^b(S) = 0$$

The bond price is given by combining the Envelope condition $\left(V_{\theta^b} = \lambda(S) = u_c(c)\right)$ and the FOC

$$\begin{aligned} Q^b(S) &= \beta E_t \left[\frac{u_c(c')}{u_c(c)} \right] \\ &= \beta E_t \left[\left(\frac{c'}{c} \right)^{-\gamma} \right] \\ &= \beta E_t \left[\exp\left\{ -\gamma \log\left(\frac{c'}{c} \right) \right\} \right] \\ &= \beta \exp\left\{ -\gamma g + \frac{1}{2}\gamma^2 \sigma^2 \right\} \end{aligned}$$

where the final equality holds because in equilibrium $c_t = y_t$ and by the properties of log-normally distributed variables.

b) Next, compute the market price, Q^b_j, of a risk free bond that is purchased in period t and pays exactly one unit of consumption in period $t+j$ for any maturity j.

The pricing kernel from time t to time j is given by

$$M_{t,t+j} = \beta^j \frac{u_c(c_{t+j})}{u_c(c_t)}$$

so the price of the bond is given by

$$\begin{aligned}
Q^b_{t+j} &= E_t\left[M_{t+j}\right] \\
&= \beta^j E_t\left[\left(\frac{c_{t+j}}{c_t}\right)^{-\gamma}\right] \\
&= \beta^j E_t\left[\left(\frac{c_{t+j}}{c_{t+j-1}}\right)^{-\gamma} \cdots \left(\frac{c_{t+1}}{c_t}\right)^{-\gamma}\right] \\
&= \beta^j\left(\exp\left(j(-\gamma g + \frac{1}{2}\gamma^2\sigma^2)\right)\right)
\end{aligned}$$

where the final equality holds again by $c_t = y_t$ in equilibrium and properties of log-normal variables.

c) We can make the income/ dividend/ endowment growth process have a time-varying mean.

$$log(y_{t+1}/y_t) \sim N(g_{t,t+1}, \sigma^2)$$

where $g_{t,t+1}$ stands for the growth from t to $t+1$ and could follow an AR(1) process:

$$g_{t,t+1} = (1-\varphi)\bar{g} + \varphi g_t + \sigma_g^2 \epsilon_{t+1} \tag{1}$$

with $\epsilon \overset{iid}{\sim} N(0,1)$ and $\varphi \in (-1,1)$.

Then the time t price of a bond paying 1 unit of consumption for sure at time $t+j$ would be:

$$\begin{aligned}
Q^B_j &= \beta^j E_t\left[\frac{u_c(c_{t+j})}{u_c(c_t)}\right] \\
&= \beta^j E_t\left[\left(\frac{c_{t+j}}{c_t}\right)^{-\gamma}\right] \\
&= \beta^j E_t\left[exp\left\{-\gamma log\left(\frac{c_{t+j}}{c_t}\right)\right\}\right] \\
&= \beta^j exp\left\{E_t\left[-\gamma log\left(\frac{c_{t+j}}{c_t}\right)\right] + \frac{1}{2}Var_t\left[-\gamma log\left(\frac{c_{t+j}}{c_t}\right)\right]\right\} && \text{due to log normality} \\
&= \beta^j exp\left\{E_t\left[-\gamma log\left(\frac{y_{t+j}}{y_t}\right)\right] + \frac{1}{2}Var_t\left[-\gamma log\left(\frac{y_{t+j}}{y_t}\right)\right]\right\} && \text{due to Lucas (1978) resource constraint}
\end{aligned}$$

Since we defined the process of log endowment growth above as the AR(1) process in equation 1 we can then recursively solve for the two terms:

$$
\begin{aligned}
E_t\left[log\left(\frac{y_{t+j}}{y_t}\right)\right] &= E_t\left[g_{t,t+j}\right] \\
&= \sum_{i=0}^{j-1}\varphi^i(1-\varphi)\bar{g}+\varphi^i g_t \\
Var_t\left[log\left(\frac{y_{t+j}}{y_t}\right)\right] &= Var_t\left[g_{t,t+j}\right] \\
&= Var_t\left[\sum_{i=0}^{j-1}\varphi^i\sigma_g\epsilon_{t+1+i}\right] && \text{already omitting constant terms} \\
&= \sigma_g^2\sum_{i=0}^{j-1}\varphi^{2i} && \text{due to iid'ness of } \epsilon
\end{aligned}
$$

Then the time t price of a bond paying 1 unit of consumption for sure at time $t+1$ is

$$
Q_j^B = \beta^j exp\left\{-\gamma\left[\sum_{i=0}^{j-1}\varphi^i(1-\varphi)\bar{g}+\varphi^i g_t\right]+\frac{1}{2}\left[\gamma^2\sigma_g^2\sum_{i=0}^{j-1}\varphi^{2i}\right]\right\} \tag{2}
$$

Consequently, the yield of maturity j is:

$$
R_j^b = (Q_j^B)^{-1/j} = \beta^{-1}exp\left\{-\gamma\left[\sum_{i=0}^{j-1}\varphi^i(1-\varphi)\bar{g}+\varphi^i g_t\right]+\frac{1}{2}\left[\gamma^2\sigma_g^2\sum_{i=0}^{j-1}\varphi^{2i}\right]\right\}^{-1/j} \tag{3}
$$

We see that due to the stationarity of the mean process g_t the further we look into the future ($j \to \infty$), the less the most current observation g_t impacts our outlook and the closer we get to the long-term mean. Thereby the rate of convergence (i.e. the slope of the yield curve) depends on the persistence of the process which is in turn determined by how close φ is to $|1|$. For instance with the following parameters $\gamma = 2,\ \bar{g} = 0.1,\ \sigma_g = 0.3,\ g_t = 0.03$ we get the following yield curves:

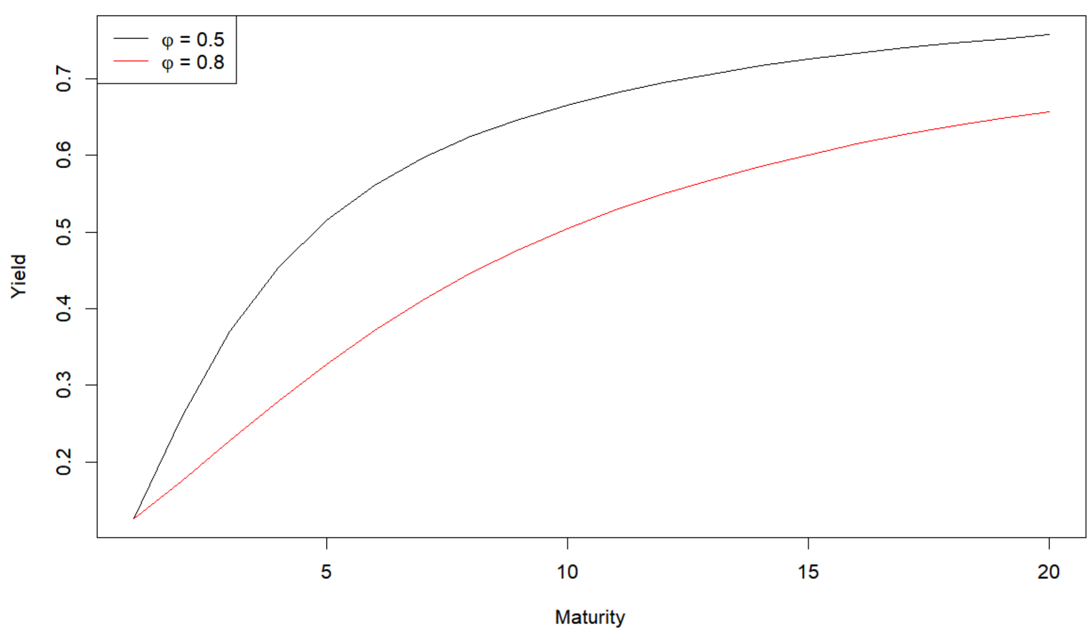

Figure 1: Implied yield curve with time-varying mean of the endowment growth process

Part II: Computing Asset Prices in Incomplete Markets

a) The parameters of the economy in which each of the 5,000 households optimizes are:

$$\gamma = 2$$
$$\beta = 0.97$$
$$k_{t+1} \geq -1$$
$$logy_t = \rho logy_{t-1} + \sigma(1-\rho^2)^{1/2}\epsilon_t \ with \ \rho = 0.6, \ \sigma = 0.35, \ \epsilon \sim^{iid} WN(0,1)$$

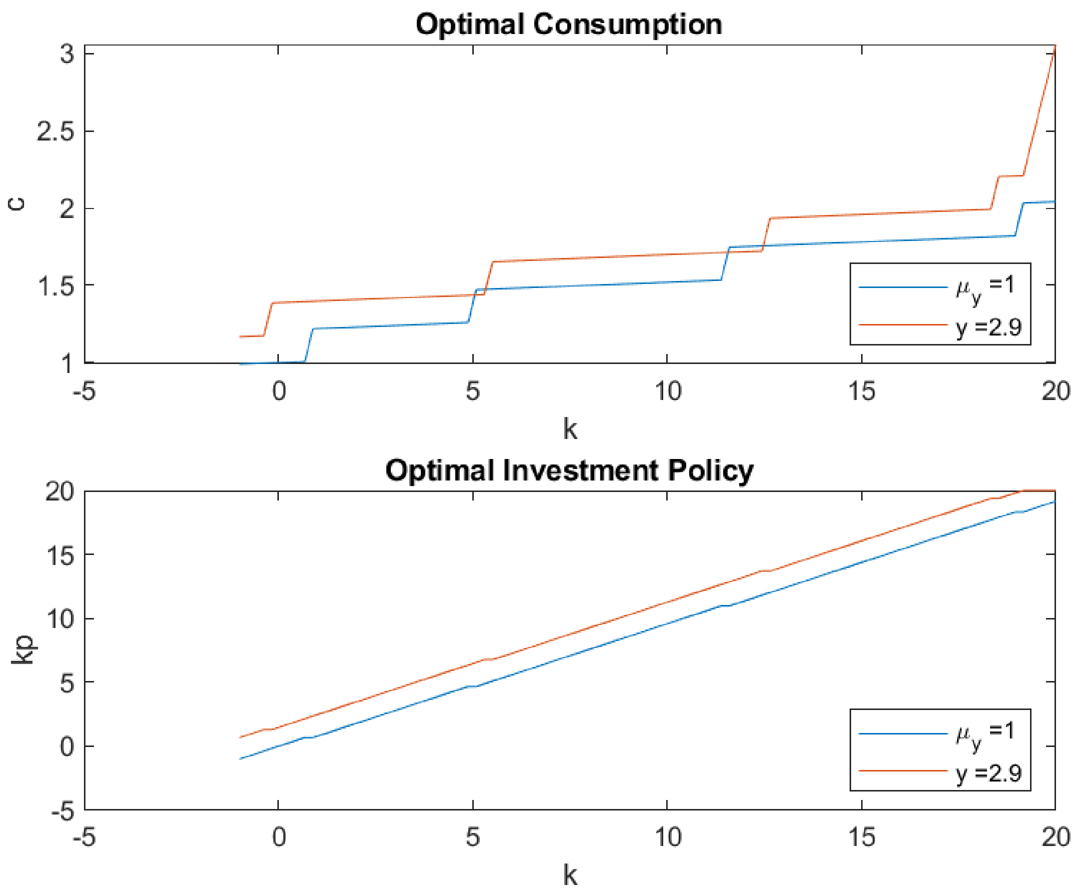

Figure 2: Optimal policies of each household for $R = 1.01$

Figure 2 gives the optimal policies of consumption and investment of a household for the mean/ median income μ_y and for the highest income levels.

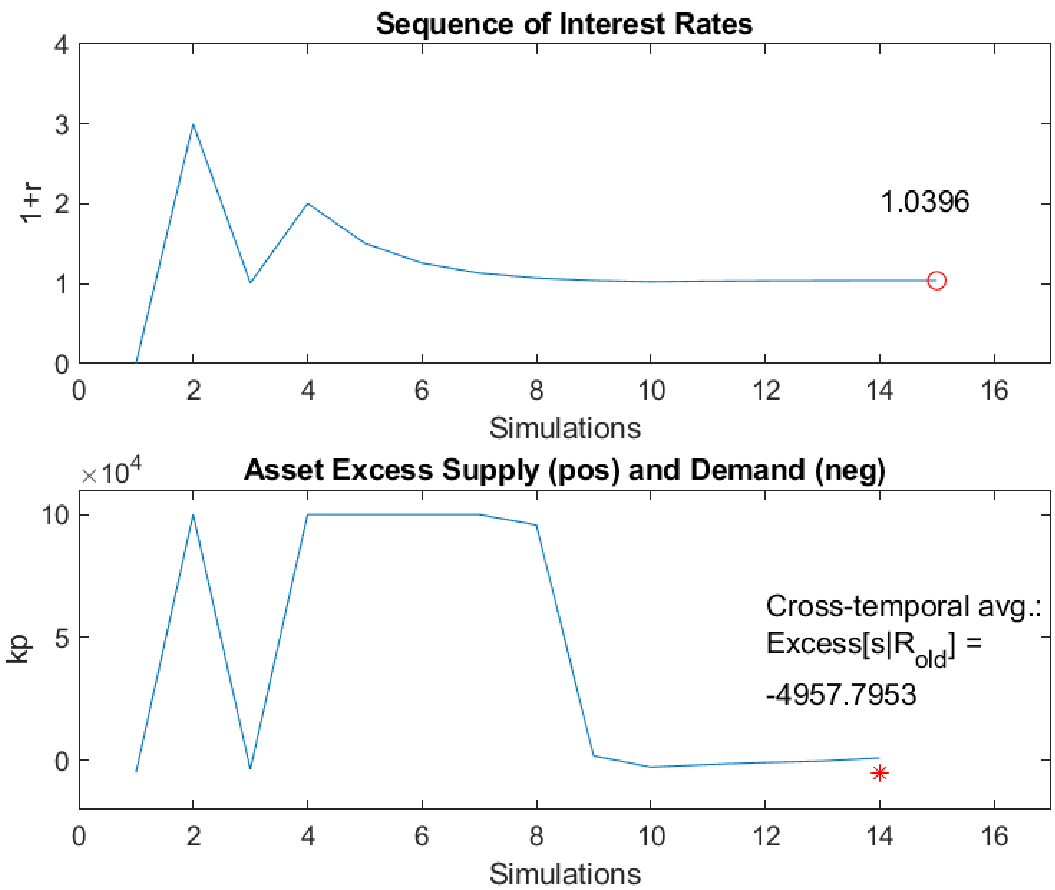

Figure 3: Convergence to Equilibrium with 5000 Households

Firstly, since the (stable) economy comprises of 2000 periods the asset excess demand/ supply in Figure 3 is the following aggregation per simulation s:

$$Excess(s) = \frac{1}{2000} \sum_{t=1}^{2000} supply_{\ t,k'}$$

where

$$supply_{\ t,k'} = \sum_{i=1}^{5000} k'_{t,i}$$

is the economy's aggregate demand across all households i in period t.

Extra Credit – Note the red $\star$ in the lower graph of Figure 3: We see that when we allow agents to borrow more, $k_{min} \geq -2$ rather than the previous lower bound of -1, agents naturally make use of this newly extended constraint. However, since we left $1 + r$ at its previous equilibrium level more people borrow than lend with $Excess$ being clearly negative, i.e. the markets don't clear period-by-period under this price – the asset price r would need to increase for deterring this new excess borrowing and thus for supply and demand to re-balance.

Part III: Investment and Asset Prices

a) The one period gross return, $R^K_{t,t+1}$ to the accumulation of physical capital, k', via the first order condition for k' by the firm:

$$v_0 = E_0 \sum_{t=0}^{\infty} M_{0,t} d_t = E_0 \sum_{t=0}^{\infty} M_{0,t}(\pi_t - I_t - \Phi_t) \text{ s.t. } I_t = K_{t+1} - (1-\delta)K_t$$

$$\frac{\partial v_0}{\partial k_{t+1}} : 0 = E_0 \left\{ \left[-\frac{\partial I_t}{\partial k_{t+1}} - \frac{\partial \Phi_t}{\partial I_t}\frac{\partial I_t}{\partial k_{t+1}} \right] + M_{0,t+1}\left[\frac{\pi_{t+1}}{\partial k_{t+1}} - \frac{\partial I_{t+1}}{\partial k_{t+1}} - \left(\frac{\partial \Phi_{t+1}}{\partial k_{t+1}} + \frac{\partial \Phi_{t+1}}{\partial I_{t+1}}\frac{\partial I_{t+1}}{\partial k_{t+1}} \right) \right] \right\}$$

$$1 = E_0 \left\{ \frac{M_{0,t+1}}{M_{0,t}} \underbrace{\left[\frac{\frac{\pi_{t+1}}{\partial k_{t+1}} - \frac{\partial I_{t+1}}{\partial k_{t+1}} - \left(\frac{\partial \Phi_{t+1}}{\partial k_{t+1}} + \frac{\partial \Phi_{t+1}}{\partial I_{t+1}}\frac{\partial I_{t+1}}{\partial k_{t+1}} \right)}{\frac{\partial I_t}{\partial k_{t+1}} + \frac{\partial \Phi_t}{\partial I_t}\frac{\partial I_t}{\partial k_{t+1}}} \right]}_{\equiv R^K_{t,t+1}} \right\}$$

$$= E_0 \left[M_{t,t+1} \left(\frac{a_{t+1} - \gamma(-\delta + \eta_{t+1} + 1)^2 - \delta + \frac{\gamma k_{t+2}^2}{k_{t+1}^2} + 1}{1 + 2\gamma\left(\frac{i_t}{k_t} - \eta_t\right)} \right) \right] \blacksquare$$

b) First, rewrite the problem with q as the Lagrange multiplier on the investment constraint:

$$\mathcal{L} = E_0 \sum_{t=0}^{\infty} M_{0,t}(\pi_t - I_t + q_t(I_t - k_{t+1}K_t(1-\delta)))$$

Then we can solve the problem w.r.t. I_t: $\Psi_t = max_{I_t}\{-I_t - \Phi_t + q_t I_t\}$ giving us $q_t^\star = 1 + \frac{\partial \Phi_t}{\partial I_t}$. Then using the fact from the lecture that under linear homogeneity $v_t^e = q_t^\star k_{t+1}$ we can rewrite the return to the owner of the firm:

$$R^S_{t,t+1} = \frac{v^e_{t+1} + d_{t+1}}{v^e_t}$$

$$= \frac{(1 + \frac{\partial \Phi_{t+1}}{\partial I_{t+1}})k_{t+2} + \pi_{t+1} - \overbrace{I_{t+1}}^{=\frac{\partial I_{t+1}}{\partial k_{t+1}}k_{t+1} + k_{t+2}} - \Phi_{t+1}}{(1 + \frac{\partial \Phi_t}{\partial I_t})k_{t+1}}$$

$$= \frac{\frac{\partial \pi_{t+1}}{\partial k_{t+1}}k_{t+1} - \frac{\partial I_{t+1}}{\partial k_{t+1}}k_{t+1} \overbrace{-\Phi_{t+1} + \overbrace{\frac{\partial \Phi_{t+1}}{\partial I_{t+1}}}^{=2\gamma\left[\frac{I_{t+1}}{k_{t+1}} - \eta_{t+1}\right]} k_{t+2}}^{=-\frac{d\Phi_{t+1}}{dk_{t+1}}k_{t+1}}}{(1 + \frac{\partial \Phi_t}{\partial I_t})k_{t+1}}$$

$$= \frac{\frac{\pi_{t+1}}{\partial k_{t+1}} - \frac{\partial I_{t+1}}{\partial k_{t+1}} - \left(\frac{\partial \Phi_{t+1}}{\partial k_{t+1}} + \frac{\partial \Phi_{t+1}}{\partial I_{t+1}}\frac{\partial I_{t+1}}{\partial k_{t+1}} \right)}{\frac{\partial I_t}{\partial k_{t+1}} + \frac{\partial \Phi_t}{\partial I_t}\frac{\partial I_t}{\partial k_{t+1}}}$$

$$= \frac{a_{t+1} - \gamma(-\delta + \eta_{t+1} + 1)^2 - \delta + \frac{\gamma k_{t+2}^2}{k_{t+1}^2} + 1}{1 + 2\gamma\left(\frac{i_t}{k_t} - \eta_t\right)}$$

$$= R^K_{t,t+1} \blacksquare$$

Appendix - Matlab Code

```
%%
% *%%% Patrick's & Felix' Code*
%Task (1a)
%% Discretize log(y)-AR process

L = 19;
ny = L; % number of grid points
rho_lny = 0.6; % autocorrelation coefficient
bar_lny = 0/(1-rho_lny); % unconditional mean of lny
sigma_eps = 0.35*sqrt(1-rho_lny^2);

[grid_lny,P,d]=tauchen1(ny,bar_lny,rho_lny,sigma_eps);

%% Household's problem
% VFI

% Set parameter values
r = 1.01;
sigma = 1/2; % = 1/risk aversion
beta = 0.97;

S = 2500; % number simulated periods
H = 5000; % number of households

% Set capital grid parameters
M = 100;
nk = M + 1;
nkp = nk;
kmin = -1; % borrowing limit
kmax = -20*kmin;
kstep = (kmax - kmin)/(nk - 1);

k = [kmin: kstep: kmax]';                          % Grid for current wealth
kp = k;                                            % Grid for next period wealth

y = exp(grid_lny);

rate_seq = [0, 3, r]; %bounds as starting value for interest rate chain
suppl_seq = [H*kmin-1, kmax*H+1]; %bounds for capital suppl-demand chain, in line w/ R=1 and R=10
alpha = 3; % counter for rates evaluated

tic;
toc_seq = 0;
while abs(rate_seq(end)-rate_seq(end-1)) > 0.0005
    % equivalent result for the following while cond. (H*kstep := max dev. dep. on the economy):
    % (abs(suppl_seq(end)-suppl_seq(end-1)) > H*kstep) || (suppl_seq(end)*suppl_seq(end-1)>0)


% Compute momentary utility function
c = zeros(nkp, nk, ny);
U = zeros(nkp, nk, ny);

for j = 1:ny
for i = 1:nk
   c(:,i,j) = max( (r*k(i) + y(j))*ones(size(kp)) - kp, 0.001);
end
end

U = (c.^(1 - 1/sigma) - 1 )/(1 - 1/sigma);

% Initial guess for the value function

V0 = zeros(nk,ny);
```

```
init = repmat(kp,1,size(y,2))+repmat(y,size(kp,1),1);
V1 = (init.^(1 - 1/sigma) - 1 )/(1 - 1/sigma);

kp_opt_ind = zeros(nk,ny);

err = 1;

% Start iterations to determine optimal value function
while err > 0.0001
    for i=1:ny
    [V1(:,i), kp_opt_ind(:,i)] = max(U(:,:,i) + beta*V0(:,:)*P(i,:)'*ones(1,nkp));
    % for every income level y_i pick the max kp along the k grid
    % use transition matrix to find expected next periods value fct. which
    % is effectively a probability weighted average value function (expected value)
    end
    % for each level of y_i we add last round's optimal kp picks (which is
    % V0, so it's mutliplied by beta etc.) which is constant per level of k
    % (thus upscaled in the kp dimension
    temp = (V1 - V0).^2;
    err = sum(temp(:))/numel(V1);
    V0 = V1; %we now have a matrix of kp values on a k*y grid
end

nybar = 0.5*(ny+1);

if r == 1.01
figure(1)
subplot(2,1,1)
plot(k,V1(:,nybar))
title(['\mu_y =' num2str(round(y(nybar),3))])
xlabel('k')
ylabel('V(\cdot)')
subplot(2,1,2)
plot(k,V1(:,end))
title(['y = ' num2str(round(y(end),3))])
xlabel('k')
ylabel('V(\cdot)')
%saveas(figure(1),[pwd '/02_Graphs/II_1a_TwoLevels.png'])
end


%Task (2c)
%% Extract c* and kp*

kp_opt = kp*ones(1,ny);
kp_opt = kp_opt(kp_opt_ind);

c_opt = transpose(max( (ones(ny,1)*r*k' + y'*ones(1,nk)) - kp_opt', 0.001));

if r == 1.01
figure(2)
subplot(2,1,1)
plot(k,c_opt(:,nybar), k,c_opt(:,end))
title('Optimal Consumption')
legend(strcat('\mu_y = ', num2str(round(y(nybar),1))), ...
        strcat('y = ', num2str(round(y(end),1))),'Location','SouthEast')
xlabel('k')
ylabel('c')
subplot(2,1,2)
plot(k,kp_opt(:,nybar), k,kp_opt(:,end))
title('Optimal Investment Policy')
legend(strcat('\mu_y = ', num2str(round(y(nybar),1))), ...
        strcat('y = ', num2str(round(y(end),1))),'Location','SouthEast')
xlabel('k')
ylabel('kp')

saveas(figure(2),[pwd '/02_Graphs/II_a_kpAndcPerK.png'])
end
```

```
%Task (2b)
%% Simulate the economy
chain_length = S;
chain = zeros(H,chain_length);
chain(:,1) = ones(H,1)*(ny+1)/2; % initial state number to start chain
%chain(1:3,1) = [1 4 9]

% preallocate memory to store y realizations chosen by Markov chain
y_markov = zeros(H, chain_length);
grid_lny_H = repmat(grid_lny,H,1);
y_markov(:,1) = grid_lny_H(1,chain(:,1));

for i=2:chain_length
    distribution = P(chain(:,i-1), :);
    cumulative_distribution = cumsum(distribution,2);
    j = rand(H,1);
    chain(:,i) = max(1,L-sum(cumulative_distribution>j,2));
    % L the number of states in y, sum() gives row-wise no of cols in cdf
    % matrix P where j is exceeded, so L-sum() is col-number in P, max is
    % to handle cases where even first row exceeds j - avoiding col=0
    y_markov(:,i) = grid_lny_H(1,chain(:,i));
end
y_chain = exp(y_markov);
k_chain = zeros(H,S+1); % requires one column more as policy concerns kp
k_chain(:,1) = repmat(k(0.5*(nk+1)),H,1);
c_chain = zeros(H,S);

%k_chain(1:3,1) = k(1:3);

for j = 1:S
[lia, loc] = ismember(k_chain(:,j),k);
[lib, locy] = ismember(y_chain(:,j),y);
   k_chain(:,j+1) = diag(kp_opt(loc,locy));

c_chain(:,j) = (max( (r*k_chain(:,j) + y_chain(:,j) ) - k_chain(:,j+1), 0.001));
end
% cut out first 20% of obs for stable economy (0.2*S=500, from 501
% onwards and accounting for kp being shifted by +1 makes +2)

k_stable = k_chain(:,round(0.20*S,0)+2:end);

supply = sum(k_stable,1);
suppl_seq = [suppl_seq mean(supply)];

if suppl_seq(alpha)*suppl_seq(alpha-1)<0 %eq. in the middle of n and n-1
r = 0.5*(rate_seq(alpha)+rate_seq(alpha-1)); %avg
elseif suppl_seq(alpha)*suppl_seq(alpha-2)<0 %eq. in the middle of n and n-2
r = 0.5*(rate_seq(alpha)+rate_seq(alpha-2)); %avg
elseif suppl_seq(alpha)<0 %avg with last opposing sign rate
        r = 0.5*(rate_seq(alpha)+rate_seq(find(suppl_seq>0,1,'last')));
else r = 0.5*(rate_seq(alpha)+rate_seq(find(suppl_seq<0,1,'last')));
end

rate_seq = [rate_seq r];
alpha = alpha +1
toc1 = toc
toc_seq = [toc_seq toc1];
end
eq_rate = r;
X = ['End of loop, target interest rate is ', num2str(round(eq_rate,2))];
disp(X)

%% Change borrowing constraint but not r

kmin = -2; % *NEW* borrowing limit
kstep = (kmax - kmin)/(nk - 1);

```

```
k = [kmin: kstep: kmax]';                               % Grid for current wealth
kp = k;                                                 % Grid for next period wealth


% Compute momentary utility function
c = zeros(nkp, nk, ny);
U = zeros(nkp, nk, ny);

for j = 1:ny
for i = 1:nk
   c(:,i,j) = max( (r*k(i) + y(j))*ones(size(kp)) - kp, 0.001);
end
end

U = (c.^(1 - 1/sigma) - 1 )/(1 - 1/sigma);

% Initial guess for the value function

V0 = zeros(nk,ny);
init = repmat(kp,1,size(y,2))+repmat(y,size(kp,1),1);
V1 = (init.^(1 - 1/sigma) - 1 )/(1 - 1/sigma);

kp_opt_ind = zeros(nk,ny);

err = 1;

% Start iterations to determine optimal value function
while err > 0.0001
    for i=1:ny
    [V1(:,i), kp_opt_ind(:,i)] = max(U(:,:,i) + beta*V0(:,:)*P(i,:)'*ones(1,nkp));
    % for every income level y_i pick the max kp along the k grid
    % use transition matrix to find expected next periods value fct. which
    % is effectively a probability weighted average value function (expected value)
    end
    % for each level of y_i we add last round's optimal kp picks (which is
    % V0, so it's mutliplied by beta etc.) which is constant per level of k
    % (thus upscaled in the kp dimension
    temp = (V1 - V0).^2;
    err = sum(temp(:))/numel(V1);
    V0 = V1; %we now have a matrix of kp values on a k*y grid
end

nybar = 0.5*(ny+1);

%% Extract c* and kp*

kp_opt = kp*ones(1,ny);
kp_opt = kp_opt(kp_opt_ind);

c_opt = transpose(max( (ones(ny,1)*r*k' + y'*ones(1,nk)) - kp_opt', 0.001));

%% Simulate the economy
chain_length = S;
chain = zeros(H,chain_length);
chain(:,1) = ones(H,1)*(ny+1)/2; % initial state number to start chain
%chain(1:3,1) = [1 4 9]

% preallocate memory to store y realizations chosen by Markov chain
y_markov = zeros(H, chain_length);
grid_lny_H = repmat(grid_lny,H,1);
y_markov(:,1) = grid_lny_H(1,chain(:,1));

for i=2:chain_length
    distribution = P(chain(:,i-1), :);
    cumulative_distribution = cumsum(distribution,2);
    j = rand(H,1);
    chain(:,i) = max(1,L-sum(cumulative_distribution>j,2));
    % L the number of states in y, sum() gives row-wise no of cols in cdf
```

```
    % matrix P where j is exceeded, so L-sum() is col-number in P, max is
    % to handle cases where even first row exceeds j - avoiding col=0
    y_markov(:,i) = grid_lny_H(1,chain(:,i));
end
y_chain = exp(y_markov);
k_chain = zeros(H,S+1); % requires one column more as policy concerns kp
k_chain(:,1) = repmat(k(0.5*(nk+1)),H,1);
c_chain = zeros(H,S);

for j = 1:S
[lia, loc] = ismember(k_chain(:,j),k);
[lib, locy] = ismember(y_chain(:,j),y);
   k_chain(:,j+1) = diag(kp_opt(loc,locy));

c_chain(:,j) = (max( (r*k_chain(:,j) + y_chain(:,j) ) - k_chain(:,j+1), 0.001));
end
% cut out first 20% of obs for stable economy (0.2*S=500, from 501
% onwards and accounting for kp being shifted by +1 makes +2)

k_stable = k_chain(:,round(0.20*S,0)+2:end);

supply_new = sum(k_stable,1);
excess_new = mean(supply_new);

%% Plot results
figure(3)
subplot(2,1,1)
plot(rate_seq)
title('Sequence of Interest Rates')
xlabel('Simulations')
ylabel('1+r')
axis([0 alpha+2 0 4])

N = [num2str(round(rate_seq(end),4))]; % If "N" is not cellstr or string datatype, must be column vector
labelinds = [length(rate_seq)];
text(labelinds-1,rate_seq(labelinds)+1,N);
hold on
plot(labelinds,rate_seq(labelinds),'ro')

subplot(2,1,2)
plot(suppl_seq)
title('Asset Excess Supply (pos) and Demand (neg)')
xlabel('Simulations')
ylabel('kp')
axis([0 alpha+2 -0.2*H*kmax 1.1*H*kmax])

N = "Cross-temporal avg.:" + "\n" +"Excess[s|R_{old}] =" + "\n" + num2str(excess_new);
N = compose(N);
labelinds = [length(suppl_seq)];
text(labelinds-2,excess_new+0.5*H*kmax,N);
hold on
plot(labelinds,mean(supply_new),'r*')

saveas(figure(3),[pwd '/02_Graphs/II_d_RateConvergence.png'])

figure(4)
plot(toc_seq/60,'o')
xlabel('Simulations')
ylabel('Minutes')
saveas(figure(4),[pwd '/02_Graphs/II_d_Time.png'])
```