# Economic growth
# Resursive deterministic models 2
# Class 6

Prof. George McCandless
UCEMA

Fall 2009

## 1 Recursive deterministic models

Recursive deterministic models

- What is a recursive problem
- Nature of the problem is the same independent of the period
  1. Same maximization problem
  2. Same budget constraints
  3. Initial values can be different
- Example
  1. A household maximizes its discounted utility stream subject to a budget constraint
  2. If each period the utility **function** is the same
  3. The **form** of the budget constraints are the same
     - The values in the budget constraint can change
     - Wealth can be different in different periods

States and controls

- Three types of variables
  1. State variables
  2. Control variables
  3. Other (jump) variables

Policy Function

- The solution we look for is called a **Policy Function**
- A policy function gives
  - The optimizing values for the time $t$ **Controls,**
    - $y_t$ is the vector of controls
  - As a function of the values of the time $t$ **States,**
    - $x_t$ is the vector of states
  - A policy function is of the form

$$y_t = H\left(x_t\right)$$

How the model proceeds

- Normally we can think of the budget constraints as giving
- the time $t+1$ states, $x_{t+1}$
- as a result of the time $t$ states and controls. $x_t$ and $y_t$
- so we have

$$x_{t+1} = G(x_t, y_t)$$

- This means that once we have the policy function
- we can get a process (difference equation) of

$$x_{t+1} = G(x_t, H\left(x_t\right)) = \overline{G}(x_t)$$

Infinite horizon problem with states and controls

- Robinson Crusoe want to maximize the discounted utility

$$\max \sum_{i=0}^{\infty} \beta^i u(c_{t+i}, 1 - h_{t+i}),$$

subject to the budget restrictions,

[5cm]

5cm

$$k_{t+1} = (1-\delta)k_t + i_t,$$

5cm

$$y_t = f(k_t, h_t) = c_t + i_t.$$

- $k_t$ is the state variable

- choices of controls
- $h_t$ and
- $k_{t+1}$ or $c_t$ or $i_t$

.Writing the problem with labor and capital as the control

- Use the budget constraint to write

$$c_t = f(k_t) + (1-\delta)k_t - k_{t+1}$$

- Substitute this into the utility function to get

$$\max \sum_{i=0}^{\infty} \beta^i u \left[ f(k_{t+i}, h_{t+i}) + (1-\delta)k_{t+i} - k_{t+i+1}, 1 - h_{t+i} \right]$$

- In this case, both $h_{t+i}$ and $k_{t+i+1}$ are controls in period $t+i$
  - $k_{t+i}$ is the state in that period
  - $k_{t+i+1}$ will be the state for the next period

.The value function

**Definition 1** *For given values of the state variables at time t, the value function gives the value of the discounted objective function when that objective function is being maximized.*

- The value function is a function of the **state variables**
- The discounted objective function is being optimized
- Example:

$$V(k_t) =$$

$$\max_{\{k_s, h_{s-1}\}_{s=t+1}^{\infty}} \sum_{i=0}^{\infty} \beta^i u(f(k_{t+i}, h_{t+i}) - k_{t+1+i} + (1-\delta)k_{t+i}, 1 - h_{t+i})$$

.Recursive problems

- The time $t$ problem

$$V(k_t) =$$

$$\max_{\{k_s, h_{s-1}\}_{s=t+1}^{\infty}} \sum_{i=0}^{\infty} \beta^i u(f(k_{t+i}, h_{t+i}) - k_{t+1+i} + (1-\delta)k_{t+i}, 1 - h_{t+i})$$

  is recursive

- In time $t+1$, Robinson Crusoe is solving

$$V(k_{t+1}) =$$

$$\max_{\{k_s,h_{s-1}\}_{s=t+2}^{\infty}} \sum_{i=0}^{\infty} \beta^i u(f(k_{t+1+i}, h_{t+1+i}) - k_{t+2+i} + (1-\delta)k_{t+1+i}, 1 - h_{t+1+i})$$

.Writing the recursive problem
The time $t$ problem

$$V(k_t) = \max_{\{k_s,h_{s-1}\}_{s=t+1}^{\infty}} \sum_{i=0}^{\infty} \beta^i u(f(k_{t+i}, h_{t+i}) - k_{t+1+i} + (1-\delta)k_{t+i}, 1-, h_{t+i})$$

can be written as

$$V(k_t) = \max_{k_{t+1},h_t} u(f(k_t) - k_{t+1} + (1-\delta)k_t) +$$

$$\beta \max_{\{k_s,h_{s-1}\}_{s=t+2}^{\infty}} \sum_{i=0}^{\infty} \beta^i u(f(k_{t+1+i}, h_{t+1+i}) - k_{t+2+i} + (1-\delta)k_{t+1+i}, 1 - h_{t+1+i})$$

or as the Bellman's equation

$$V(k_t) = \max_{k_{t+1},h_t} [u(f(k_t, h_t) - k_{t+1} + (1-\delta)k_t) + \beta V(k_{t+1})]$$

.The Bellman equation

- The recursive equation

  $$V(k_t) = \max_{k_{t+1},h_t} [u(f(k_t, h_t) - k_{t+1} + (1-\delta)k_t) + \beta V(k_{t+1})]$$

  or the one for the simplier model where labor is constant

  $$V(k_t) = \max_{k_{t+1}} [u(f(k_t) - k_{t+1} + (1-\delta)k_t) + \beta V(k_{t+1})]$$

  are called Bellman's equations
- It is recursive because $V(k_t)$ dependes on the value of the same function $V(\cdot)$ but evaluated at $k_{t+1}$
- Notice that although in the first case, one maximizes over $k_{t+1}$ and $h_t$, it is only $k_{t+1}$ that is a state variable
- It is a one period problem,
  - One only chooses the value of $k_{t+1}$ (or of $k_{t+1}$ and $h_t$ )
  - Notice that all future utility is captured in $V(k_{t+1})$
  - Choice of $k_{t+1}$ will change the value of $V(k_{t+1})$

– Choice of $h_t$ is simply a problem for time $t$ although the choice of $h_t$ helps determine the possible $k_{t+1}$

Solving a recursive problem (using the simple example)

- Take the derivative of

$$V(k_t) = \max_{k_{t+1}} \left[ u(f(k_t) - k_{t+1} + (1-\delta)k_t) + \beta V(k_{t+1}) \right]$$

with respect to $k_{t+1}$

- Get

$$0 = -u'(f(k_t) - k_{t+1} + (1-\delta)k_t) + \beta V'(k_{t+1})$$

- Problem is that we do not know the function $V(k_{t+1})$ nor its derivative $V'(k_{t+1})$

Benveniste - Scheinkman envelope theorem conditions

- Benveniste - Scheinkman give conditions under which one can find $V'(\cdot)$
- Take derivative of

$$V(k_t) = \max_{k_{t+1}} \left[ u(f(k_t) - k_{t+1} + (1-\delta)k_t) + \beta V(k_{t+1}) \right]$$

with respect to $k_t$

- Get

$$V'(k_t) = u'(f(k_t) - k_{t+1} + (1-\delta)k_t) \left( f'(k_t) + (1-\delta) \right)$$

which we can evaluate at $k_{t+1}$

- The result is called an envelope theorem

.

- First order condition is

$$\begin{aligned} 0 &= -u'(f(k_t) - k_{t+1} + (1-\delta)k_t) + \beta V'(k_{t+1}) \\ &= -u'(c_t) + \beta V'(k_{t+1}) \end{aligned}$$

- B-S envelope condition is

$$V'(k_t) = u'(f(k_t) - k_{t+1} + (1-\delta)k_t) \left( f'(k_t) + (1-\delta) \right)$$

- However, we need it for time $t+1$, so it should be written as

$$\begin{aligned} V'(k_{t+1}) &= u'(f(k_{t+1}) - k_{t+2} + (1-\delta)k_{t+1}) \left( f'(k_{t+1}) + (1-\delta) \right) \\ &= u'(c_{t+1}) \left( f'(k_{t+1}) + (1-\delta) \right) \end{aligned}$$

- We put that in the first order condition to get

$$0 = -u'(c_t) + \beta \left[ u'(c_{t+1}) \left( f'(k_{t+1}) + (1-\delta) \right) \right]$$

.First order and B-S envelope conditions

- That equation can be rearranged to get the Euler equation

$$\frac{u'(c_t)}{u'(c_{t+1})} = \beta \left( f'(k_{t+1}) + (1-\delta) \right).$$

- In a stationary state, where $c_t = c_{t+1}$, this is

$$\frac{1}{\beta} - (1-\delta) = f'(\overline{k}).$$

.General version of problem

- Let $x_t$ be the state variables and $y_t$ the controls
- We want solve

$$V(x_t) = \max_{\{y_s\}_{s=t}^{\infty}} \sum_{s=t}^{\infty} \beta^{s-t} F(x_s, y_s)$$

  subject to the set of budget constraints

$$x_{s+1} = G(x_s, y_s).$$

- The functions, $F(\cdot,\cdot)$ and $G(\cdot,\cdot)$, are the same for all periods
- Both time $t$ state variables and control variables can be in the objective function and the budget constraints at time $t$.
- This can be written as a Bellman equation,

$$V(x_t) = \max_{y_t} \left[ F(x_t, y_t) + \beta V(x_{t+1}) \right],$$

  subject to the budget constraints

$$x_{t+1} = G(x_t, y_t),$$

.General version of problem

- The Bellman equation can be written as

$$V(x_t) = \max_{y_t} \left[ F(x_t, y_t) + \beta V(G(x_t, y_t)) \right]$$

- We solve for a **policy function** of the form

$$y_t = H(x_t)$$

- The time $t$ controls are functions of the time $t$ state variables

- Notice that the problem is a **functional equation** and that the solution is the **function** $y_t = H(x_t)$

.General version of problem: the first order conditions

- Taking the derivative of the Bellman equation gives

$$0 = F_y(x_t, y_t) + \beta V'(G(x_t, y_t))G_y(x_t, y_t)$$

- As before we can find the Benveniste-Scheinkman envelope theorem

$$V'(x_t) = F_x(x_t, y_t) + \beta V'(G(x_t, y_t))G_x(x_t, y_t)$$

  - If $G_x(x_t, y_t) = 0$
  - The envelope condition is simply $V'(x_t) = F_x(x_t, y_t)$ or $V'(x_{t+1}) = F_x(x_{t+1}, y_{t+1})$
  - The solution can be written as the Euler equation

$$0 = F_y(x_t, y_t) + \beta F_x(G(x_t, y_t), y_{t+1})G_y(x_t, y_t)$$

  - $y_{t+1}$ is still a problem
  - If the function, $F_x(G(x_t, y_t), y_{t+1})$, is independent of $y_{t+1}$, the equation can be solved for, $y_t = H(x_t)$
  - Normally, explicit solutions cannot be found

.Approximation of the value function

- One can approximate the value function numerically

- Choose some initial **function** $V_0(x_t)$

  - Most any function will do
  - a good one is $V_0(x_t) = c$
  - where $c$ is a constant (0, for example)

- Find (approximately) the **function** $V_1(x_t)$

$$V_1(x_t) = \max_{y_t} [F(x_t, y_t) + \beta V_0(G(x_t, y_t))]$$

  over a dense set of values from the domain of $x_t$

- One now has the function $V_1(x_t)$

Approximation of the value function (continued)

- Using this function $V_1(x_t)$, find

$$V_2(x_t) = \max_{y_t} [F(x_t, y_t) + \beta V_1(G(x_t, y_t))]$$

  over a dense set of values from the domain of $x_t$

  - one will need to interpolate the function $V_1(x_t)$
  - when the needed $G(x_t, y_t)$ is not part of the dense set of $x_t$
  - linear interpolation is normally good enough

- Using $V_2(x_t)$ repreat the process
- Get a sequence $\{V_i(x_t)\}_{i=0}^{\infty}$
- Bellman showed that $\{V_i(x_t)\}_{i=0}^{\infty} \longrightarrow V(x_t)$
- Once you have $V(x_t)$ finding $y_t = H(x_t)$ is easy
  - Actually, one finds a sequence $\{H_i(x_t)\}_{i=0}^{\infty} \longrightarrow H(x_t)$
  - while finding $\{V_i(x_t)\}_{i=0}^{\infty} \longrightarrow V(x_t)$
- Why does this work? Answer $= \beta$

Problems of dimensionality

- How well do we choose to approximate the function
- How many points in the domain of $x_t$
- If $x_t \in \mathbb{R}^1$ we can choose lots of points, $M$ points
- As dimensionality of $x_t$ grows (say to $\mathbb{R}^N$)
  - number of points needed is $M^N$ which can be very large

Comparing example economy to general problem 1: using B-S

- The objective function is

$$F(x_t, y_t) = u(f(k_t) - k_{t+1} + (1 - \delta)k_t)$$

- The budget constraint is

$$k_{t+1} = x_{t+1} = G(x_t, y_t) = y_t = k_{t+1}$$

  or

$$k_{t+1} = k_{t+1}$$

- The first order condition is

$$\begin{aligned} 0 &= F_y(x_t, y_t) + \beta V'(G(x_t, y_t))G_y(x_t, y_t) \\ &= -u'(f(k_t) - k_{t+1} + (1-\delta)k_t) + \beta V'(G(x_t, y_t)) \cdot 1 \end{aligned}$$

- Because $\partial k_{t+1}/\partial k_t = 0$, the B-S envelope theorem condition is

$$V'(x_t) = F_x(x_t, y_t) = u'(f(k_t) - k_{t+1} + (1-\delta)k_t)\,(f'(k_t) + (1-\delta))$$

.Comparing example economy to general problem 1: using B-S

- Use this $V'(\cdot)$ in the first order conditions to get the Euler equation

$$\begin{aligned} 0 &= -u'(f(k_t) - k_{t+1} + (1-\delta)k_t) \\ &\quad + \beta\left[u'(f(k_{t+1}) - k_{t+2} + (1-\delta)k_{t+1})\,(f'(k_{t+1}) + (1-\delta))\right]. \end{aligned}$$

- Sometimes the Euler equation is all that you need
- we can also find the stationary state where $k_t = k_{t+1} = k_{t+2} = \overline{k}$ as

$$f'(\overline{k}) = \frac{1}{\beta} - (1-\delta)$$

.Approximation of the Value function

- To approximate the value function need explicit functions for $u(c_t)$ and $f(k_t)$
- Let $f(k_t) = k_t^\theta$ and $u(c_t) = \ln(c_t)$
- Let $\delta = .1$, $\theta = .36$, and $\beta = .98$ (consistent with annual data for US)
- The Bellman equation is

$$V(k_t) = \max_{k_{t+1}} \left[\ln(k_t^\theta - k_{t+1} + (1-\delta)k_t) + \beta V(k_{t+1})\right]$$

- Note: stationary state $\overline{k} = 5.537$ (how do you find this?)

.Approximation of the Value function

- Choose $V_0(\cdot) = 0$ (a constant initial guess for value function)
- Find $V_1(\cdot)$ using

$$\begin{aligned} V_1(k_t) &= \max_{k_{t+1}} \left[\ln(k_t^\theta - k_{t+1} + (1-\delta)k_t) + \beta V_0(k_{t+1})\right] \\ &= \max_{k_{t+1}} \left[\ln(k_t^{.36} - k_{t+1} + .9k_t) + .98 \cdot 0\right] \end{aligned}$$

  for a dense set of $k_t$

- Find $V_2(\cdot)$ using

$$V_2(k_t) = \max_{k_{t+1}} \left[\ln(k_t^{.36} - k_{t+1} + .9k_t) + .98 \cdot V_1(k_{t+1})\right]$$

  for a dense set of $k_t$. Use linear interpolation of $V_1(k_{t+1})$ between known points

- Repeat $N$ times. Get approximate $V(k_t)$ function (as close as you want)

.Computer program

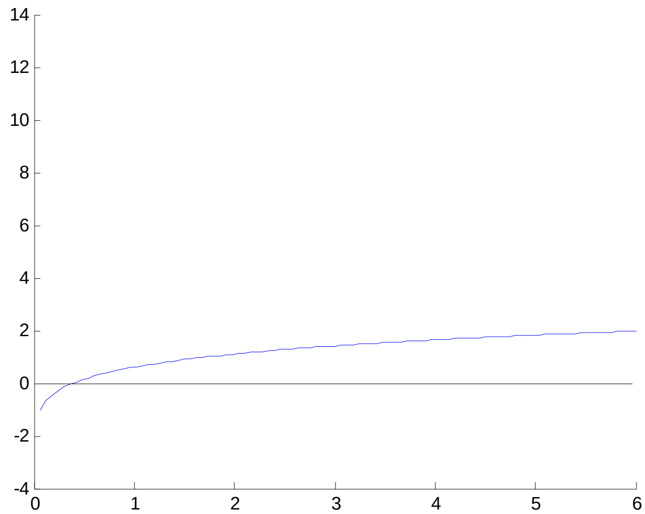
Main program

```
global vlast beta delta theta k0 kt
hold off
hold all
%set initial conditions
vlast=zeros(1,100);
k0=0.06:0.06:6;
beta=.98;
delta=.1;
theta=.36;
numits=240;
%begin the recursive calculations
for k=1:numits
     for j=1:100
         kt=j*.06;
         %find the maximum of the value function
         ktp1=fminbnd(@valfun,0.01,6.2);
         v(j)=-valfun(ktp1);
         kt1(j)=ktp1;
     end
 if k/48==round(k/48)
     %plot the steps in finding the value function
     plot(k0,v)
     drawnow
 end
 vlast=v;
end
hold off
% plot the final policy function
plot(k0,kt1)
```

.Computer program

Subroutine (valfun.m) to calculate value function

```
function val=valfun(k)
global vlast beta delta theta k0 kt
%smooth out the previous value function
g=interp1(k0,vlast,k,'linear');
```



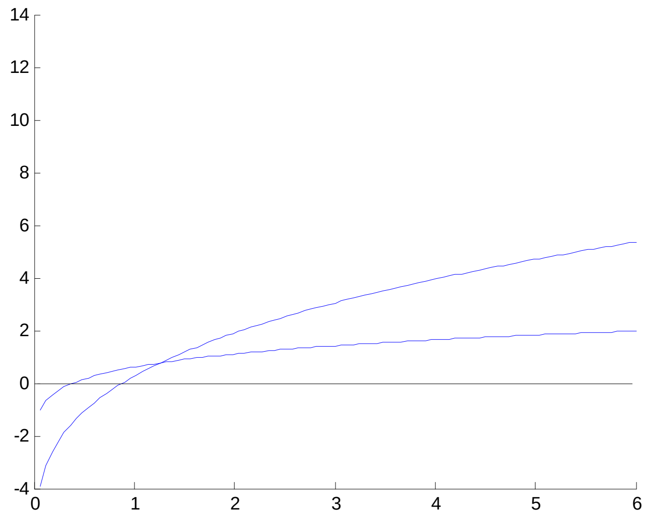

```
%Calculate consumption with given parameters
kk=kt^theta-k+(1-delta)*kt;
if kk <= 0
     %to keep values from going negative
     val=-888-800*abs(kk);
else
     %calculate the value of the value function at k
     val=log(kk)+beta*g;
end
%change value to negative since "fminbnd" finds minimum
val=-val;
```

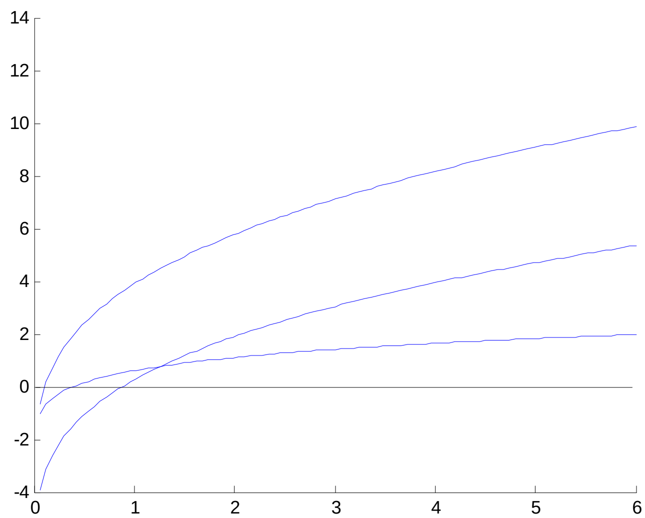
.$V(k_t)$ after one iteration
.$V(k_t)$ after ten iterations
.$V(k_t)$ after 50 iterations
.$V(k_t)$ after 100 iterations
.$V(k_t)$ after 200 iterations
.The policy function after 200 interatioins
.Solving the problem with kt+1 and ht as controls

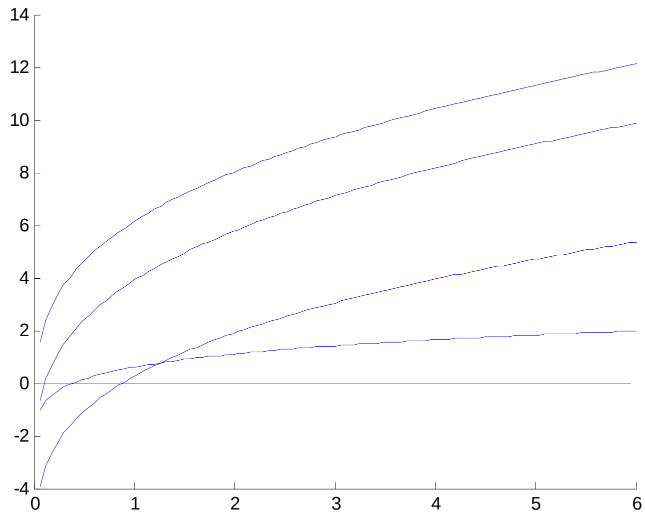
- The Bellmans equation is

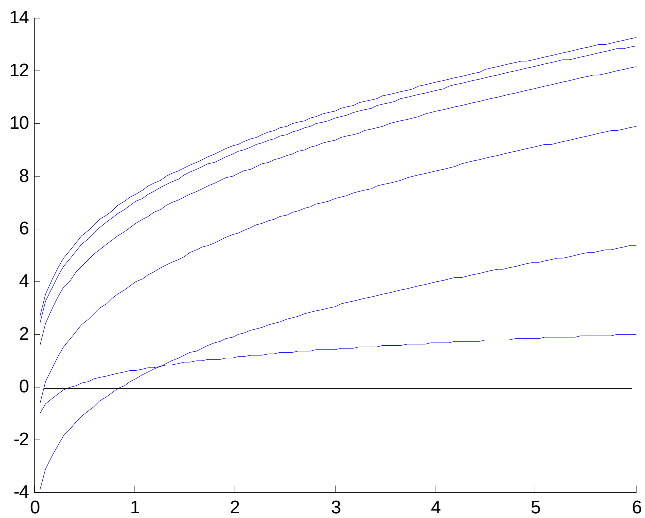
$$V(k_t) = \max_{k_{t+1}, h_t} [u(f(k_t, h_t) - k_{t+1} + (1-\delta)k_t) + \beta V(k_{t+1})]$$

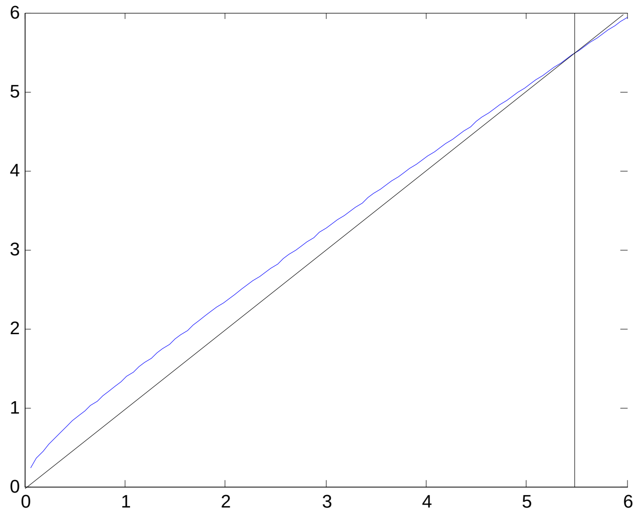
- Start with a guess for $V(k_{t+1})$


14
12
10
8
6
4
2
0
-2
-4
0
1
2
3
4
5
6


14
12
10
8
6
4
2
0
-2
-4
0
1
2
3
4
5
6


14
12
10
8
6
4
2
0
-2
-4
0
1
2
3
4
5
6

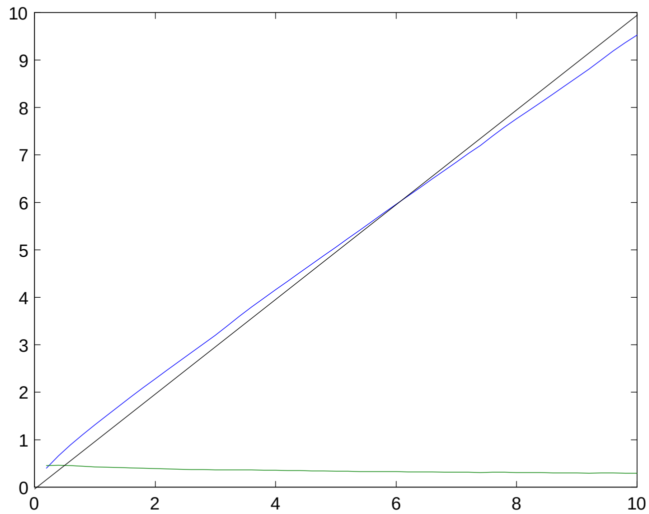

14
12
10
8
6
4
2
0
-2
-4
0
1
2
3
4
5
6




- iterate, maximizing for each $k_t$ in set (here 0 to 10)
- Maximize with respect to $k_{t+1}$ and $h_t$
- use linear interpolation in interations
- Get two policy functions: one for $k_{t+1}$ and one for $h_t$
- However, there is only one value function (since $k_t$ is the only state)

Program

```
global vlast beta delta theta gamma roe k0 kt B C
hold off
hold all
vlast=ones(1,50);
k0=0.2:0.2:10;
kt1=k0;
h=.33*ones(1,50);
xmin=[.21 .01];
xmax=[9.99 .99];
beta=.98;
delta=.1;
theta=.36;
C=2;
roe=.5;
```




```
gamma=1-theta-roe;
B=1.72;
numits=240;
kinit=k0;
for k=1:numits
 for j=1:50
 kt=k0(j);
 z0=[kinit(j),h(j)];
 options = optimset('Display','off','LargeScale','off');
 z=fmincon(@valfun2dext,z0,[],[],[],[],xmin,xmax,[],options);
 v(j)=-valfun2dext(z);
 kt1(j)=z(1);
 h(j)=z(2);
 end
 if k/30==round(k/30)
 plot(k0,v)
 drawnow
 end
 vlast=v;
 kinit=kt1;
end
hold off
plot(k0,kt1,k0,h)
save valvunfile vlast kinit h
function val=valfun2d(x)
```

```
global vlast beta delta theta gamma roe k0 kt B C
k=x(1);
h=x(2);
g=interp1(k0,vlast,k,'linear');
c=kt^theta*h^roe*(kt+C*h)^gamma+(1-delta)*kt-k;
if c < .001
 val=log(.001)-(.001-c)*100000;
else
 val=log(c)+B*log(1-h)+beta*g;
end
val=-val;
```

# 2 Recursive stochastic models

Adding uncertainty (stochastic variable)

- Stochastic variables are state variables
  - One type of state variables are predetermined
    - determined by the control variables in the previous period
  - Another type is stochastic (determined by nature)
- the value for these stochastic variables is chosen by nature
- Need to define the stochastic process that generates them
- A stochastic process is given by the triplet $\{\Omega, F, P\}$ where
  - $\Omega$ is the set of all possible *states of nature*
  - $F$ is the set of *events*: made up of subsets of $\Omega$
  - $P$ is the set of *probability* measures over $F$

An example with uncertainty

- Suppose that technology can take on two values $A^1$ and $A^2$ with probabilities $p^1$ and $p^2 = 1 - p^1$
- For example, these can represent the weather (rainfall) where $A^1$ is the right amount of rainfall and $A^2$ is the case of too little rainfall, so $A^1 > A^2$, the economy is more productive with $A^1$ than with $A^2$
- In period $t$, we know the value of $A_t$, whether it is $A^1$ or $A^2$, we don't know the future values
- However, we know the probabilities of $A^1$ and $A^2$
- Note: it is also possible to have the probabilities as Markov processes where the values of $p_t^i$ depend on the which state occurred in period $t - 1$

.The value function with uncertainty

- A simple growth model
- People want to max

$$\max \sum_{i=0}^{\infty} \beta^i u(c_{t+i}),$$

subject to

$$k_{t+i+1} = (1-\delta)k_{t+i} + i_{t+i},$$

and

$$y_{t+i} = A_{t+i} f(k_{t+i}) = c_{t+i} + i_{t+i}.$$

given $k_t$ and $A_t$

- The value function is

$$\begin{aligned} V(k_t, A_t) &= \max_{k_{t+1}} [u(A_t f(k_t) + (1-\delta)k_t - k_{t+1}) \\ &\quad + p^1 V\left(k_{t+1}, A^1\right) + p^2 V\left(k_{t+1}, A^2\right)] \end{aligned}$$

.The value function with uncertainty

- Notice that

$$\begin{aligned} V(k_t, A_t) &= \max_{k_{t+1}} [u(A_t f(k_t) + (1-\delta)k_t - k_{t+1}) \\ &\quad + p^1 V\left(k_{t+1}, A^1\right) + p^2 V\left(k_{t+1}, A^2\right)] \end{aligned}$$

can be written as

$$V(k_t, A_t) = \max_{k_{t+1}} [u(A_t f(k_t) + (1-\delta)k_t - k_{t+1}) + E_t V(k_{t+1}, A_{t+1})]$$

where

$$E_t V(k_{t+1}, A_{t+1}) = p^1 V\left(k_{t+1}, A^1\right) + p^2 V\left(k_{t+1}, A^2\right)$$

.The value function with uncertainty

- To solve this problem, one needs to find two value functions $V\left(k_t, A^1\right)$ and $V\left(k_t, A^2\right)$
- The problem for the first is

$$V\left(k_t, A^1\right) = \max_{k_{t+1}} \left[u(A^1 f(k_t) + (1-\delta)k_t - k_{t+1}) + E_t V(k_{t+1}, A_{t+1})\right]$$

and for the second

$$V\left(k_t, A^2\right) = \max_{k_{t+1}} \left[u(A^2 f(k_t) + (1-\delta)k_t - k_{t+1}) + E_t V(k_{t+1}, A_{t+1})\right]$$

- To do this, begin with some guesses for the two functions $V_0\left(k_t, A^1\right)$ and $V_0\left(k_t, A^2\right)$
- Itterate to find $V_n\left(k_t, A^1\right)$ and $V_n\left(k_t, A^2\right)$ for $n$ as large as one wants or needs

.The value function with uncertainty

- What happens if the set of events is larger or continuous?
- If larger and finite, need more $V_n\left(k_t, A^i\right)$ functions
- If continuous, need to partition and choose a set of points to interpolate between
- Problems of dimensions can grow very rapidly
- The problem has two components
  - how long the calculations will take
  - how much precision you can get out of your computer
- In chapter 5 of the ABCs of RBCs you can find an example