B&K Methods

July 21, 2015

This method originally come from Blanchard and Kahn (1980)[1].

1 The Method

1. X_t is $n+m$ column vector in log-linearization term.
$$X_t = \begin{pmatrix} X_{1t} \\ X_{2t} \end{pmatrix}$$
2. $n = \#$ of jump(er) or forward-looking variables, $m = \#$ of state/predetermined
3. Unless something odd with M , you will have $n+m$ unique eigenvalues and vectors.
$$E_t X_{t+1} = M X_t$$
It is easy to see that M is a square matrix with dimension $n+m$.
4. Qualifiers:
   (a) Don't include static or redundant variables which shows up in equilibrium conditions only at time t. for example, $y_t = A_t K_t^{\alpha}$. It is static in the sense that it only shows up explicitly in equilibrium conditions in period t and not in $t+1, t-1$. It is redundant in term of it could be expressed in linear form of state variables. Think about why we can not in term of logic we have below?
   (b) Variables that are already in % units not show in percentage deviation but absolute deviation, such as interest rates, inflation rates etc. Whereas $\tilde{c}_t = \frac{c_t - c}{c}$ in percentage difference form and $\tilde{r}_t = r_t - r$ in absolute deviation. If $r = 4\%$, and $r_t = 5\%$, the absolute deviation is 1% while the percentage deviation is 25%, it does not make any sense.
5. Problem need to know policy function, $X_{1t} = \phi X_{2t}$ which maps the states and predetermined to controls.

[1] Blanchard, O. J. and C. M. Kahn (1980). "The Solution of Linear Difference Models under Rational Expectations." Econometrica 48 (5): pp. 1305-1311.

6. Generalized Schur Decomposition $M = \Gamma\Lambda\Gamma^{-1}$,where Γ is eigenvectors matrix in columns, Λ is the matrix of eigenvalues.

7. Order the eigenvalues/vectors form smallest to the largest in modulus.

$$\Lambda = \begin{pmatrix} \underset{Q\times Q}{\Lambda_1} & 0 \\ 0 & \underset{B\times B}{\Lambda_2} \end{pmatrix}$$

   where Λ_1 is stable with every elements is less than unity $|\Lambda_1| < 1$ and Λ_2 is unstable with $|\Lambda_2| \geq 1$.

8. Partition the eigenvalues matrix into stable $(Q - Q)$ and unstable part $B - B$, where Q and B are sizes of the sub-matrix Λ_1 and Λ_2, the number of eigenvalues less than unity and greater than unity. It is easy to see that

$$Q + B = m + n$$

   where n is the number of jumper and m is the number of state or predetermined variables.

9. Replace $\Gamma^{-1}X_t$ with Z_t to reduce the model into

$$E_t X_{t+1} = MX_{t+1} = \Gamma\Lambda\Gamma^{-1}X_{t+1}$$

$$E_t\Gamma^{-1}X_{t+1} = \Gamma^{-1}X_t \rightarrow E_t Z_{t+1} = \Lambda Z_t$$

$$Z_t = \begin{pmatrix} \underset{Q\times 1}{Z_{1t}} \\ \underset{B\times 1}{Z_{2t}} \end{pmatrix}$$

$$E_t \begin{pmatrix} Z_{1t+1} \\ Z_{2t+1} \end{pmatrix} = \begin{pmatrix} \Lambda_1 & 0 \\ 0 & \Lambda_2 \end{pmatrix} \begin{pmatrix} Z_{1t} \\ Z_{2t} \end{pmatrix} = \begin{pmatrix} \Lambda_1 Z_{1t} \\ \Lambda_2 Z_{2t} \end{pmatrix}$$

10. This means that $E_t Z_{2t+1} = \Lambda_2 Z_{2t} = \Lambda_2^2 Z_{2t-1} = \cdots = \Lambda_2^t Z_{21}$. For any given initial values Z_{21}, this is an explosive path for variables Z_{2t}. This does not make any sense for economic variables and also in-consistence with the so-called transversality condition. Since $|\Lambda_2| \geq 1$, you find $Z_{2t} = 0$, for all t.

11. Partition Γ^{-1} to recover the X_t.

$$\Gamma^{-1} = \begin{pmatrix} Q\times n & Q\times m \\ B\times n & B\times m \end{pmatrix} = \begin{pmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{pmatrix}$$

12. Using $Z_{2t} = 0$ to find the policy function and necessary conditions for the policy function to exist. $B = n$, i.e. **the number of unstable eigenvalues equal to the number of forward-looking variables** (Think about the intuition, you just think the jump variables are jumping around, not stable). When $B < n$, this means that the number of equations are

less than the unknowns (see the following first equation), and in turn this means that you are going to have multiple solutions. When $B > n$, the number of equations is larger than the unknowns, you do not have a solution since the equations are linear in variables. And this is the so-called **BK** condition.

$$\begin{pmatrix} \underset{Q\times 1}{Z_{1t}} \\ \underset{B\times 1}{Z_{2t}} \end{pmatrix} = Z_t = \Gamma^{-1} X_t = \begin{pmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{pmatrix} \begin{pmatrix} X_{1t} \\ X_{2t} \end{pmatrix}$$

$$0 = \underset{B\times 1}{Z_{2t}} = \underset{B\times n}{G_{21}} \underset{n\times 1}{X_{1t}} + \underset{B\times m}{G_{22}} \underset{m\times 1}{X_{2t}}$$

$$X_{1t} = -G_{21}^{-1} G_{22} X_{2t}$$

the policy function $\phi = -G_{21}^{-1} G_{22}$.

13. (Skip) Simple examples: Deterministic examples $A_t = 1$, $\gamma = 1$. Two equations,see the notes. You derives the two log-linearization equations for the students. Tell the students what M is for this model.

    (a) Explain the β,σ to the students: If the time unit is year, $beta = .95$ (real rate is about 5%) and if in quarterly, $\beta = .99$, which mean that real interest rate is roughly 4%, which you can see roughly $4 * (1-\beta) = 4 * 0.01 = 4\%$. σ is the curvature parameter in the utility function. The larger the curvature is, the approximation errors is. So we you have large σ, you have larger approximation error.

    (b) Show the students the Matlab code to reproduce the above logic (not in Dynare).

    (c) First you should have parameters values

    (d) Find the steady states for variables

    (e) Find the matrix M.

    (f) Find the eigenvalues and eigenvectors of M, and order them from smallest to largest (you need to write or find eig_order.m somewhere).

    (g) Construct the inverse of the eigenvectors matrix. And using $Z_{2t} = 0$ to find the policy function.

2 Example

Though the deterministic version is discussed above, we consider a simple stochastic growth model, the social planner problem can be written:

$$max_{c_t, k_{t+1}} \sum_{t=0}^{\infty} \beta^t \frac{c_t^{1-\sigma} - 1}{1-\sigma}$$

$$s.t.$$

$$c_t + I_t \leq y_t$$

where the production function $y_t = a_t k_t^{\alpha}$,where a_t has a log-AR(1) process as follows:

$$loga_t = \rho loga_{t-1} + \epsilon_t$$

where ϵ_t is draw from a white noise process, $0 < \rho < 1$ is a parameter. And the capital accumulation follows

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

The Lagrangian is

$$\mathcal{L} = E_0 \sum_{t=0}^{\infty} \beta^t \left\{ \frac{c_t^{1-\sigma} - 1}{1-\sigma} + \lambda_t \left(a_t k_t^{\alpha} - c_t + (1-\delta) k_t - k_{t+1} \right) \right\}$$

$$\begin{aligned}
\partial c_t : \quad & c_t^{-\sigma} && = \lambda_t \\
\partial \lambda_{t:} \quad & k_{t+1} && = a_t k_t^{\alpha} - c_t + (1-\delta) k_t \\
\partial k_{t+1:} \quad & \lambda_t && = \beta E_t \lambda_{t+1} \left(a_{t+1} \alpha k_{t+1}^{\alpha-1} + 1 - \delta \right)
\end{aligned}$$

and the transversality condition $\beta^t E_t c_t^{-\sigma} k_{t+1} \to 0, t \to +\infty$ which ensure the stable solution. The FOCs as follows, we have three equations: the resource constraint, the Euler equation and the technology shock:

$$k_{t+1} = a_t k_t^{\alpha} - c_t + (1-\delta) k_t \tag{1}$$

$$c_t^{-\sigma} = \beta E_t c_{t+1}^{-\sigma} \left(a_{t+1} \alpha k_{t+1}^{\alpha-1} + 1 - \delta \right) \tag{2}$$

$$loga_t = \rho loga_{t-1} + \epsilon_t$$

Noting that we do not include the two static variables: y_t and I_t in the FOC conditions.

1. We first need to find the steady state values of the three variables: k_t, c_t, a_t. For technology shock, the steady state value usually define as unity, i.e., $a^* = 1$. The reason why we define $a^* = 1$ is simple. Just think the production function. The technology shock is defined as a scale factor for production. When $a_t > 1$, it will expand the production, otherwise it will contract the overall production. In steady state, it does not has any effect to production.From Eq.(2), we have

   $$k^* = \left(\frac{\alpha}{\frac{1}{\beta} - (1-\delta)} \right)^{\frac{1}{1-\alpha}}$$

   From Eq.(1), we have

   $$c^* = (k^*)^{\alpha} - \delta k^*.$$

   If every parameters are estimated or calibrated, we can calculate the steady state values of capital stock and consumption. Let's log-linearize the system.

2. The log-linearized version of the three equation system are (the notation are simplified for simplicity)

$$
\begin{aligned}
\tilde{k}_{t+1} &= -\frac{c}{k}\tilde{c}_t + \frac{1}{\beta}\tilde{k}_t + k^{\alpha-1}\tilde{a}_t \\
\tilde{c}_{t+1} &= \tilde{c}_t + \frac{\beta\left(\alpha-1\right)\alpha k^{\alpha-1}}{\sigma}\tilde{k}_{t+1} + \frac{\beta\alpha k^{\alpha-1}}{\sigma}\tilde{a}_{t+1} \\
\tilde{a}_t &= \rho\tilde{a}_{t-1} + \epsilon_t
\end{aligned}
$$

Then we write above system into the form

$$E_t X_{t+1} = M X_t$$

i.e.,

$$E_t \begin{pmatrix} \tilde{c}_{t+1} \\ \tilde{k}_{t+1} \\ \tilde{a}_{t+1} \end{pmatrix} = \begin{pmatrix} 1 - \frac{c}{k}\frac{\beta r(\alpha-1)}{\sigma} & \frac{r(\alpha-1)}{\sigma} & \frac{\beta r}{\sigma}\left(\rho + r\frac{\alpha-1}{\alpha}\right) \\ -\frac{c}{k} & \frac{1}{\beta} & \frac{r}{\alpha} \\ 0 & 0 & \rho \end{pmatrix} \begin{pmatrix} \tilde{c}_t \\ \tilde{k}_t \\ \tilde{a}_t \end{pmatrix}$$

where $r = \frac{1}{\beta} - (1-\delta)$. Let $\alpha = 0.36, \beta = 0.99, \delta = 0.025, \sigma = 2, \rho = 0.9$, then $k^* = 37.9839, c^* = 2.7543$. Then

$$M = \begin{pmatrix} 1.0008 & -0.0112 & 0.0146 \\ -0.0725 & 1.0101 & 0.0975 \\ 0 & 0 & 0.9000 \end{pmatrix}$$

We do not include output and investment into the system. Just think about why we do not include this static variables into the system. The reason is that it will make the matrix M not invertible. That is to say that M will have a zero eigenvalue. This is because the row in M correspondent to output will be simply the linear combination of other rows and this can be seen after some algebra. That is reason why sometimes we call this variable 'redundant' variable.

3. Let's See how we can solve the system. We will program in Matlab. Before we proceed, we need to know how many jumpers and states in the system. We only have one jumper, that is consumption, hence $n = 1$. The number of state or predetermined variables is two, the technology shock a_t and capital stock k_{t+1} which is predetermined variable, hence $m = 2$. Here our goal is to find the policy function

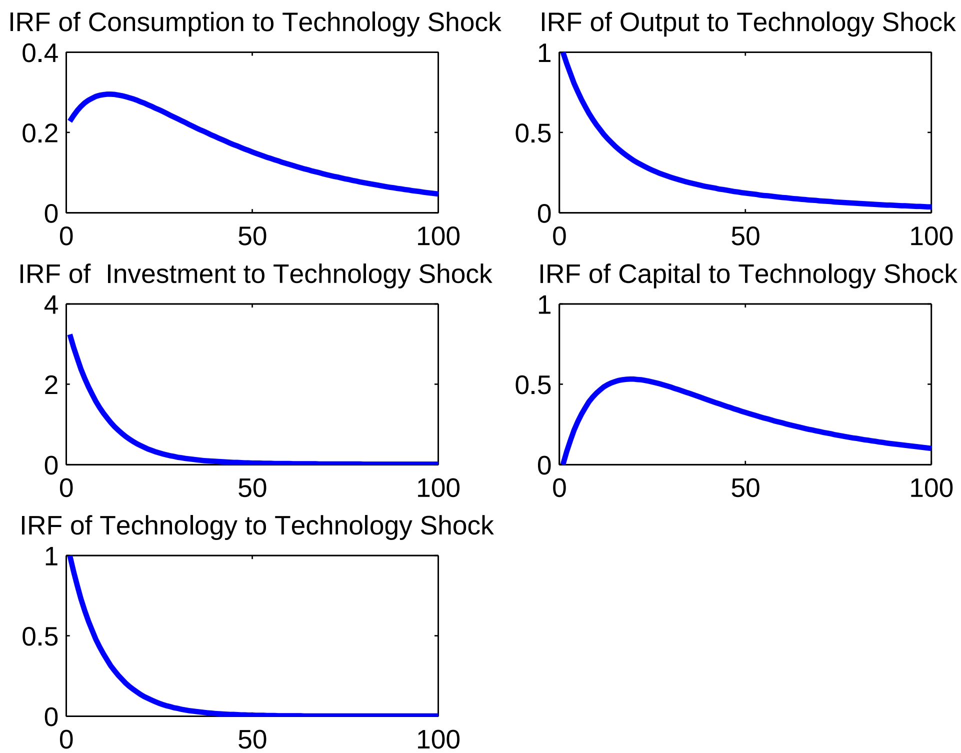
$$\tilde{c}_t = \gamma_1\tilde{k}_t + \gamma_2\tilde{a}_t \tag{3}$$

i.e., to find the coefficient γ_1, γ_2. The Matlab results show the policy function is

$$\tilde{c}_t = .4692\tilde{k}_t + .2276\tilde{a}_t$$

then for output and investment, the so-called 'static' variables. Static variables are defined as variables in the model only show up at date t, not

Figure 1: The IRFs of one unit of technology shock



$t-1$ and $t+1$, i.e., not backward-looking. The production function shows that

$$\tilde{y}_t = \tilde{a}_t + \alpha \tilde{k}_t$$

and the resource constriant shows that

$$\tilde{I}_t = \frac{y^*}{I^*}\tilde{y}_t - \frac{c^*}{I^*}\tilde{c}_t = \frac{y^*}{I^*}(\hat{a}_t + \alpha \tilde{k}_t) - \frac{c^*}{I^*}(\gamma_1 \tilde{k}_t + \gamma_2 \tilde{a}_t$$

Once we know the two states and policy function, we immediately know output and investment. But sometimes, some static variables present some headaches. A example is one from the model where we have variable labor. The best practice is to eliminate the labor after log-linearization.

3 The Codes

```
%This code is written by Xiangyang Li, 2014-1@UND
```


$$\hat{c}_t = \frac{c_t - c^*}{c^*} = \frac{c_t}{c^*} - 1$$

```
clear;
%the parameters and steady states
beta = .99;
alpha = .36;
sigma = 2;
delta = .025;
rho = .9 ;
ks = (alpha/(1/beta - 1 + delta))^(1/(1-alpha)); %steady state of capital stock
cs = ks^alpha - delta*ks; %steady state of consumption;
ck= cs/ks; %consumption capital ratio
R = 1/beta - 1 + delta; %simplifying parameter
ys = ks^alpha; %steady state of proudction
is = ys - cs; %steady state of investment

%preparation for solution
n = 1; %number of jumper
m =2; %number of states
M=zeros(3,3); %the coefficient matrix
M(1,1)= 1- ck*(alpha-1)*beta*R/sigma;
M(1,2)= (alpha-1)*R/sigma;
M(1,3)= beta*R*(rho + (alpha-1)*R/alpha)/sigma;
M(2,1)=-ck;
M(2,2) = 1/beta;
M(2,3) =R/alpha;
M(3,3)=rho;

[vv,lamb] = eig(M); %find the eigenvalues of M;
[lamb_sorted, index] = sort(abs(diag(lamb)));

%sort the eigenvector matirx
for ii = 1:m + n
    vv_sorted(:,ii) = vv(:,index(ii));
end

%find the index of the first eigenvalue whose value >=1
first_unstable_index = find(abs(lamb_sorted)>=1, 1 );

% num of stable eigs
Q = first_unstable_index - 1;
% num of unstable eigs
B = m+n - Q;

G = inv(vv_sorted);
G21 = G(Q+1:Q+B,1:n); % low left, G21,size = B*n;
G22 = G(Q+1:Q+B,n+1:n+m);% lower right, G22, size = B*m;
```

```
%the policy function coefficients
pol= -inv(G21)*G22;

%% verify the state space representation of RBC model
M11 = M(1:n,1:n);
M12 = M(1:n,n+1:n+m);
M21 = M(n+1:n+m,1:n);
M22 = M(n+1:n+m,n+1:n+m);
C = M11*pol +M12;
A = M21 * pol + M22;
% exactly the same;
C == pol*A
%% do the IRF
%number of simulation;
%if you use great number,like H=10000, you will get strange results, this
%about why?
%but for small number it works.
H = 100;

%n+m rows correspondent to the number of variables
IRF = zeros(n+m,H);
%setting up the first period
IRF(n+2,1) = 1; %one unit shock to technology shock.
IRF(n+1,1) = 0; % for capital, starts from steady state

% for the controls, actually only for consumption, n=1, the policy function
IRF(1:n,1) = pol*IRF(n+1:n+m,1);

%
for ii=2:H
    IRF(:,ii) = M*IRF(:,ii-1);
end

%including another two variables
IRF2 = zeros(2,H);
for jj=1:H
    IRF2(1,jj) = IRF(3,jj) + alpha*IRF(2,jj);%for output
    IRF2(2,jj) = ys/is*IRF2(1,jj) - cs/is*IRF(1,jj);%for investment
end

%plot the IRF
figure(1)
subplot(3,2,1)
plot(IRF(1,:),'-b','Linewidth',2)
title('IRF of Consumption to Technology Shock')
```

```
subplot(3,2,2)
plot(IRF2(1,:),'-b','Linewidth',2)
title('IRF of Output to Technology Shock')

subplot(3,2,3)
plot(IRF2(2,:),'-b','Linewidth',2)
title('IRF of  Investment to Technology Shock')

subplot(3,2,4)
plot(IRF(2,:),'-b','Linewidth',2)
title('IRF of Capital to Technology Shock')

subplot(3,2,5)
plot(IRF(3,:),'-b','Linewidth',2)
title('IRF of Technology to Technology Shock')

%% do simulation of the model

%ensure the replication of simulation results
randn('state',1234567);

%number of data to be simulated
T = 1000 + 1;

%sampling from standard normal distribution;
e = 0.1*randn(1,T);

%state sapce, by construction is stable
M1 = M(1:n,:); %the first n rows
M2 = M(n+1:n+m,:); % the last m rows;
M21=M2(:,1:n);
M22=M2(:,n+1:n+m);
A = M21*pol + M22;
B = [0;1]; %by definition, we only have one exogenous shock, technology shock.
C = pol;

% [consumption, output ,investment]

C = [C;alpha 1;(ys/is)*alpha - (cs/is)*pol(1,1) (ys/is)*1 - (cs/is)*pol(1,2)];

s = zeros(2,T); % states, [capital , technology]
c = zeros(3,T); % controls, [consumption, output ,investment]

%s(:,1) = B*e(1,1);

for j = 2:T
```

```
    s(:,j) = A*s(:,j-1) + B*e(1,j); %the state equations
end

for j = 1:T
    c(:,j) = C*s(:,j); %the measurement equations, states map to controls;
end
% coverting to levels
Simss = repmat( [ks 1]',1,T);
s =Simss.* (repmat([1;0],1,T) + s);
%
Sim2ss = repmat([cs ys is]',1,T);
c = Sim2ss.*(1+c);

% do hp filtering
lambda = 1600;

%*t, trend part; *c, cycle part;
[yt,yc] = hp_filter(c(2,:)',lambda);
[ct,cc] = hp_filter(c(1,:)',lambda);
[it,ic] = hp_filter(c(3,:)',lambda);

%find out the simulated moments:standard deviations
SDy = std(yc);
SDc = std(cc);
SDi = std(ic);

%relative volatility
Rc = SDc/SDy;
Ri = SDi/SDy;

% the following simulation will explode up.
%add the shocks to the system
% Sigma = [0; 0 ;1];
% Sim = zeros(n+m,T);
% %Sim(:,1) = Sigma*e(1,1);
% for ii=2:T
%     Sim(:,ii) = M*Sim(:,ii-1) + Sigma*e(1,ii);
% end
% % Y and Investment
% Sim2 = zeros(2,T);
% for jj=1:T
%     Sim2(1,jj) = Sim(3,jj) + alpha*Sim(2,jj);
%     Sim2(2,jj) = ys/is*Sim2(1,jj) - cs/is*Sim(1,jj);
% end
```

4 The IRF

1. Show the IRF for 20 horizons and 10000 horizons to find out what's going on. When H=10000, you find the consumption and capital going to infinity. Rounding errors matters when horizon get large since Matlab inverses matrix. The unstable part in matrix M will take advantage of this and finally let the system goes to explore. How can we fix this problems? That is to say how can we get the dynamics right? We need load the errors into the current state variables by replacing period $t+1$ controls by states using policy function, and the round errors does not matters . And by construction, the matrix A is stable(see Section state space respresentation). Show the pics of IRF for both small and large horizons.

2. Intuition for calculating IRF manually: First, you write the model in State Space Representation; second, you begins your calculation by setting up the initial values for the states (this initial values are usually steady states, and of course you can start from non-steady state. And theoretically speaking you can start from any point you want.) and then iteratively calculates the IRF for states first. For example, if you have two state variables:
$$\begin{pmatrix} K_t \\ A_t \end{pmatrix} = A \begin{pmatrix} K_{t-1} \\ A_{t-1} \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} e_t$$
the capital and technology shock, where e_t is a scalar shock. Third, you use the measurement equation and IRF of states to calculates IRF for control variables. Be sure that the variables are in the form you desired, i.e. in levels, log-deviation,etcs. And this method, by construction, is stable. If you do not write the model in state space representation form, and directly from the reduced form solution, you could get exploded IRF due to the rounding errors in Matlab. Only small models, for example, one controls, two states are appropriate to calculate IRF or simulate the model manually and this will help you build up intuitions, For large models, it is better to use Dynare or other tools to calculate IRF.

5 The Simulation

See the Matlab code for how to manually simulate the model. Here is the results for simulation.

Figure 2: The Simulation of Consumption, output and investment