Solving Basic Models Using Dynare

Presentation by Valerie Ramey

Overview

- Dynare is a collection of Matlab, Scilab or Gauss routines that solve, simulate, and estimate nonlinear models with forward-looking variables.

- It was written by researchers at CEPREMAP.

- The software and documentation is available free-of-charge at: http://www.cepremap.cnrs.fr/dynare/

- It is currently used in macro graduate courses at a number of institutions, such as MIT and Northwestern.

Suppose we want to analyze the canonical RBC social planner problem:

$$\max E\sum_{t=0}^{\infty}\beta^{t}\left\{\ln c_t+\psi\ln(1-l_t)\right\} \qquad \text{subject to}$$

$$c_t+k_t=k_{t-1}^{\alpha}(e^{z_t}l_t)^{1-\alpha}+(1-\delta)k_{t-1},\ \ \forall t>0$$

$$z_t=\rho z_{t-1}+\varepsilon_t, \qquad \varepsilon_t\sim N(0,\sigma^2)$$

Note: For Dynare, capital must be defined as end of the period of capital. Note different timing in second equation.

As we know, we cannot solve this model analytically. Dynare provides a quick and easy way to analyze these types of models.

Step 1: Derive the first-order conditions for the nonstochastic version of your problem.

$$\frac{1}{c_t} = \beta \frac{1}{c_{t+1}} \left\{ 1 + \alpha k_t^{\alpha-1} e^{z_{t+1}} l_{t+1}^{1-\alpha} - \delta \right\}$$

$$\psi \frac{c_t}{1-n_t} = (1-\alpha) k_{t-1}^{\alpha} (e^{z_t})^{(1-\alpha)} l_t^{-\alpha}$$

Step 2: Write your Dynare Program – call it "filename.mod"

Part I: Defining Variables

```
% Basic RBC Model
%
% Jesus Fernandez-Villaverde
% Philadelphia, March 3, 2005

%----------------------------------------------------------------
% 0. Housekeeping (close all graphic windows)
%----------------------------------------------------------------

close all;

%----------------------------------------------------------------
% 1. Defining variables
%----------------------------------------------------------------

var y c k i l y_l z;
varexo e;
parameters beta psi delta alpha rho;
```

Part II. Calibration

```
%---------------------------------------------------------------------
% 2. Calibration
%---------------------------------------------------------------------
alpha   = 0.33;
beta    = 0.99;
delta   = 0.023;
psi     = 1.75;
rho     = 0.95;
sigma   = (0.007/(1-alpha));
```

Part III. Model

```
%----------------------------------------------------------------
% 3. Model
%----------------------------------------------------------------
model;

  (1/c) = beta*(1/c(+1))*(1+alpha*(k^(alpha-1))*(exp(z(+1))*l(+1))^(1-alpha)-delta);

  psi*c/(1-l) = (1-alpha)*(k(-1)^alpha)*(exp(z)^(1-alpha))*(l^(-alpha));

  c+i = y;

  y = (k(-1)^alpha)*(exp(z)*l)^(1-alpha);

  i = k-(1-delta)*k(-1);

  y_l = y/l;

  z = rho*z(-1)+e;

end;
```

Part IV. Computation

```
%--------------------------------------------------------------
% 4. Computation
%--------------------------------------------------------------

initval;
  k = 9;
  c = 0.76;
  l = 0.3;
  z = 0;
  e = 0;
end;
```

“When the **initval** block is followed by the command steady, it is not necessary to provide exact initialization values for the endogenous variables. steady will use the values provided in the **initval** block as initial guess in the non-linear equation solver and computes exact values for the endogenous variables at the steady state. “

Part IV. Computation - continued

```
shocks;

var e = sigma^2;

end;


steady;

stoch_simul(hp_filter = 1600, order = 1);
```

Some side notes - 1:

- For stochastic simulations, the **shocks** block specifies the non zero elements of the covariance matrix of the shocks.

Example

```
shocks;
var e = 0.000081;
var e,u = phi*0.009*0.009;
var u = 0.000081;
var v;
stderr 0.009;
end;
```

- If you have permanent shocks, solve the model with the proper use of initval and endval.

- If the shocks are temporary, you can just use initval (for calculating the steady state) and shocks.

Some side notes -2:

Method (from Dynare manual):

“stoch_simul computes a (2^{nd} order) Taylor approximation of the decision and transition functions for the model, impulse response functions and various descriptive statistics (moments, variance decomposition, correlation and autocorrelation coefficients). For correlated shocks, the variance decomposition is computed as in the VAR literature through a Cholesky decomposition of the covariance matrix of the exogenous variables. When the shocks are correlated, the variance decomposition depends upon the order of the variables in the varexo command.

The Taylor approximation is computed around the steady state.”

There are many options for the stoch_simul statement.

Part V. Results Statement

```
%--------------------------------------------------------------
% 5. Some Results
%--------------------------------------------------------------
statistic1 = 100*sqrt(diag(oo_.var(1:6,1:6)))./oo_.mean(1:6);
table('Relative standard deviations in %',strvcat('VARIABLE','REL.
S.D.'),lgy_(1:6,:),statistic1,10,8,4);
```

or

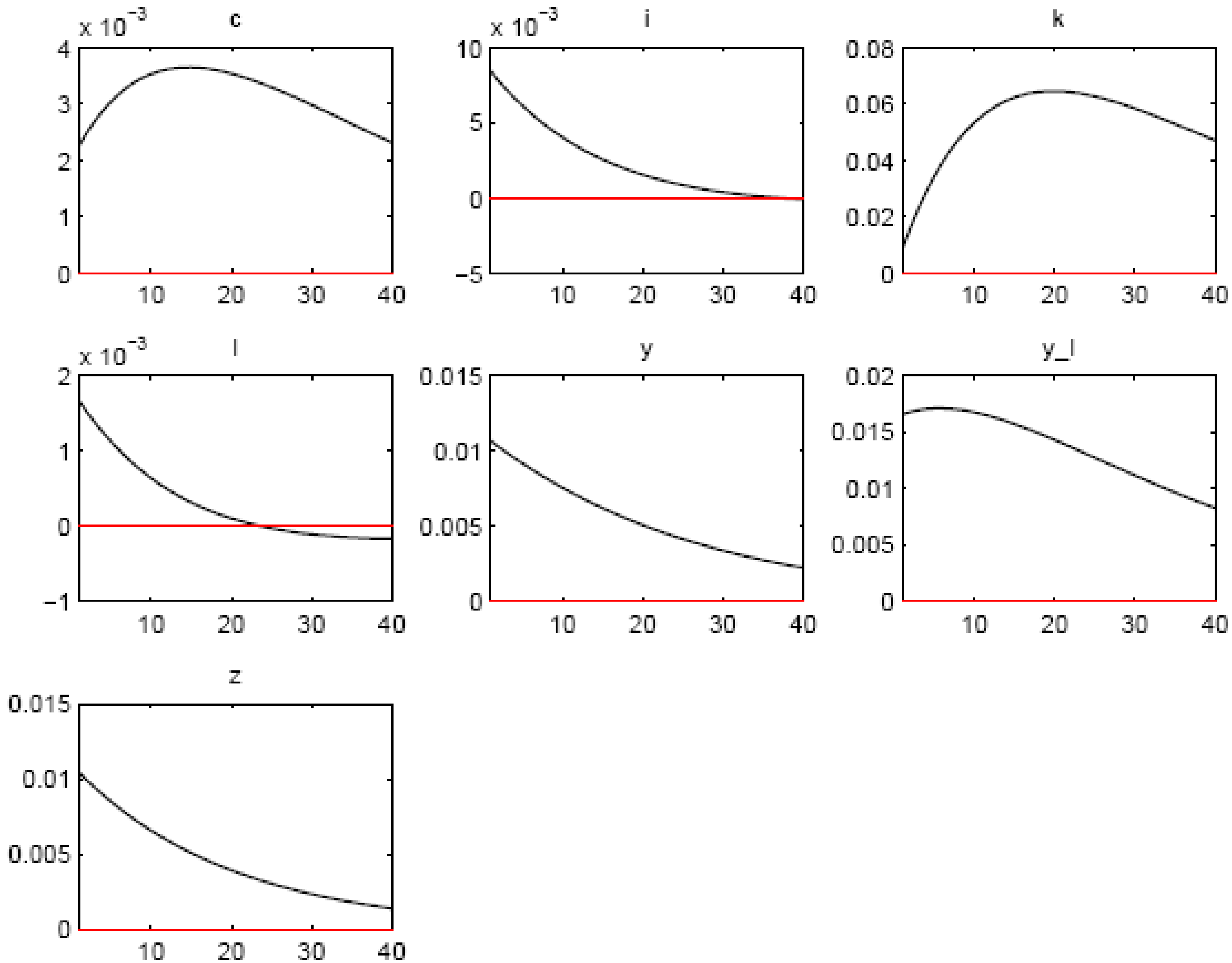
dynatype prints the listed variables in a text file named *FILENAME*. If no *VARIABLE_NAME* is listed, all endogenous variables are printed.

dynatype [(*FILENAME*)] *VARIABLE_NAME* [*VARIABLE_NAME* ...] ;

Part VI. Running the Program

In matlab, simply type: dynare filename.mod

Part VII. Some Output
A. IRFs for Technology Shock with rho = 0.95
(filename_IRF_e.pdf or filename_IRF_e.fig)



Part VII. Some Output
B. from filename.log

STEADY-STATE RESULTS:

c	0.793902
i	0.236201
k	10.2696
l	0.331892
y	1.0301
y_l	3.10373
z	0

MODEL SUMMARY

Number of variables:	7
Number of stochastic shocks:	1
Number of state variables:	2
Number of jumpers:	3
Number of static variables:	3

Part VII. Some Output
B. from filename.log - continued