

Partner assignments

Blaise Albis-Burdige	Jakob Shimer
Todd Bieschke	Bijay Rai
Emanuel Gordis	Jasmine Peng
Alex Hsu	Thomas Malthouse
Yessica Ji	Sarah Kumar
Pratik Kafle	Brian Zilles
Olek Wojcik	Owen Young

Project overview

In this project you will use a numerical software package called Dynare to simulate Romer's real-business-cycle model of Chapter 5. In particular, you will construct your own versions of Figure 5.2 through Figure 5.7, and compare the correlations and autocorrelations of the variables in stochastic simulations to those of the actual U.S. economy.

In order to facilitate solution in Dynare, we must simplify the model by expressing it in terms of log-deviations from the model's steady-state values, then calculating a linear approximation of the model's equations near the steady state in terms of these log-deviations. Parts of this process are in coursebook Chapter 6 (Section D); the others are part of the assignment, which may be done on paper.

Once you have achieved the log-linearization, you will use Dynare (operating within the commercial numerical package MATLAB or the open-source substitute Octave) to perform various deterministic and stochastic simulations. This part is done on a computer (either your own or a Reed computer) on which Dynare and MATLAB/Octave have been installed and should be submitted electronically in a Word or pdf file. (Both Dynare and Octave can be downloaded without cost.) *This project uses software that is not very familiar to the instructor, so if you encounter problems with the software, please ask ASAP so that some combination of me and CUS can solve them for you and for others. Kristin Bott of CUS has worked with some students successfully in past year.*

In the following pages, the tasks that you are to perform are listed in sans-serif text with three bold lines on the left margin:

||| This is how an assigned task will look.

Preparing the model for simulation in Dynare

You should begin by reading Section D of coursebook Chapter 6 carefully. This guides you through the preparatory procedures. On page 6 - 22, the steady-state value of A/K_{-1} is derived for you. You should use a similar process to derive the expressions for the steady-state values of Y/K_{-1} and C/Y .

- ||| 1. Show that the steady-state value of Y/K_{t-1} is $(\rho + \gamma + \delta)/\alpha$ and that the steady-state value of C/Y_t is $1 - (G/Y)^* - \alpha(\gamma + \delta)/(\rho + \gamma + \delta)$.

Remember that you may turn in the algebraic sections in on paper if you wish.

The remainder of section D of the coursebook chapter is devoted to log-linearization of some of the equations of the RBC model as deviations from the steady state.

- ||| 2. Using the methods shown in coursebook Section D to derive equations (13) and (17), derive: (a) an expression corresponding to equation (3) in terms of \tilde{w}_t , \tilde{k}_{t-1} , \tilde{a}_t , and \tilde{l}_t ; (b) an expression for equation (4) in \tilde{r}_t , \tilde{k}_{t-1} , \tilde{a}_t , and \tilde{l}_t ; and (c) an expression for equation (6) in \tilde{c}_t , \tilde{l}_t , and \tilde{w}_t . (All equation numbers refer to the coursebook chapter.)

Simulating the model

Introduction to Dynare

Dynare (<http://www.dynare.org/>) was developed as a software tool to facilitate the simulation of dynamic, stochastic, general-equilibrium (DSGE) models. It runs inside the computation platform MATLAB, or the open-source alternative Octave (<http://www.gnu.org/software/octave/>). Both Dynare and Octave are available for free download, the former running under MATLAB or Octave on all platforms and the latter in separate versions for Windows, Mac, or Linux. CUS has installed Dynare and MATLAB on all public-access computers in the IRCs, PPW, and elsewhere on campus. The current version of Dynare is 4.6.0, which was released last week. Version 4.5.7 is installed on Reed lab computers (ETC, Library, etc.) and definitely runs. The new version has not been tested locally, but it *should* work. The Web site warns that 4.5.7 will not run under Octave on the Mac platform, though it will run under MATLAB. It will run with either Octave or MATLAB on Windows. ***It is strongly recommended that you use Reed computers with MATLAB. This has been tested with the version 4.5.7 of Dynare and MATLAB.***

You input your model, parameter values, and instructions to Dynare through a text file with a .mod suffix. The Quick Start guide, Tutorial, User Guide, and Manual available at the Dynare Web site (link above) provide details if you want them. We will provide a template for the .mod file you will use for this assignment below, so you won't need to learn the details unless you want to. Note that the .mod file *must be in plain text format*. You can use a simple text editor to create and edit such a file. **Do not use Microsoft Word or another advanced word processor to prepare your program.** These word processors save formatting information along with the actual text, which will create errors in Dynare. I recommend Notepad++ on Windows or TextWrangler on Mac. Also, copying and pasting from the pdf version of the template program below may insert unwanted end-of-line characters that will confuse Dynare. A text version of the template is linked on the assignment page of the class Web site.

The .mod file consists of five parts:

- The **preamble** section defines your variables and parameters.
- The **model** section contains the equations of your model in a simple algebraic notation similar to standard programming languages.
- The **steady state** section gives the steady-state (initial) values of the model solution.
- The **shocks** section, in which you tell Dynare about the deterministic or random shocks that you want to simulate.
- The **computation** section that tells Dynare to simulate the model and report the results.

A template Dynare program

The easiest way to describe the various sections of the .mod file is by example. The code below is the beginning of a file you can use to run your simulations and is linked in text-file form from the assignments page. Lines preceded by // are comments.

```
// ***Preamble section***
// Define the endogenous variables
var k a l c y w r g;
// Define the exogenous variables (shocks)
varexo epsg epsa;
// Define the parameters of the model
parameters alpha rhoa rhog rho lstar gamma delta gystar;
// Set the values of the parameters to match Romer's calibration
alpha = 1/3;
rhoa = .95;
rhog = .95;
rho = 0.01;
lstar = 1/3;
gamma = 0.005;
delta = 0.025;
gystar = 0.2;

// ***Model section***
```

```

model;

// Equation (19) from above. Expected future value of x is x(+1)
-c = r(+1) - c(+1);

// Equation (17) from above. Lags of x denoted by x(-1)
(1+gamma) * k = (rho + gamma + delta)/alpha * y
+ (1-delta) * k(-1)
+ (gamma + delta - (rho + gamma + delta)*(1-gystar)/alpha) * c
- gystar*(rho+gamma+delta)/alpha * g;

// Equation (13) from above.
y = alpha * k(-1) + (1-alpha) * (a + l);

// YOU WILL NEED TO ADD THREE EQUATIONS HERE

// Equations (7) and (8) from above
a = rhoa * a(-1) + epsa;
g = rhog * g(-1) + epsg;

end;

// *** Steady-state section *** In the steady state all deviations are zero
initval;
c = 0;
l = 0;
k = 0;
y = 0;
g = 0;
a = 0;
w = 0;
r = 0;
end;

steady;
check;

// *** Shocks section ***
shocks;
var epsa;
periods 1;
values .01;
end;

// *** Computation section ***

```

```

simul( periods=200);

// *** Some MATLAB plot commands similar to those below
subplot(3,3,1);
plot(k);
title('K');
axis([0 40 -0.002 .01]);
subplot(3,3,2);
plot(a);
title('A');
axis([0 40 -0.002 .01]);
subplot(3,3,3);
plot(l);
title('L');
axis([0 40 -0.002 .01]);
subplot(3,3,4);
plot(c);
title('C');
axis([0 40 -0.002 .01]);
subplot(3,3,5);
plot(y);
title('Y');
axis([0 40 -0.002 .01]);
subplot(3,3,6);
plot(w);
title('W');
axis([0 40 -0.002 .01]);
subplot(3,3,7);
plot(r);
title('R');
axis([0 40 -0.002 .01]);
subplot(3,3,8);
plot(g);
title('G');
axis([0 40 -0.002 .01]);

```

Before we get into details, note several features of the Dynare program:

- Variables are case-sensitive.
- All commands end with a semi-colon.
- Blocks of commands such as model equations, steady-state (initial) values, and shocks end with the `end;` statement.
- Options for any command are entered in parentheses immediate after the name of the command.

Preamble section

The definitions of the variables are straightforward. I've used lower-case letters corresponding to the notation in the text (but without tildes). Similarly, the parameters are named in ways that correspond to the equations of the model. The parameter values correspond to those at the beginning of Romer's Section 5.7.

The preamble section should not need to be altered from the above in order to run your simulations.

Model section

The equations are entered in standard computer-algebraic notation, with +, -, *, /, and ^ as the principal operators and nested sets of parentheses used to indicate the order of operation. Note that it is not necessary to "solve" the individual equations to isolate an endogenous variable alone on the left-hand side. Suffix parentheses are used to indicate leads and lags, with $x(-1)$ denoting last period's value of x and $x(+1)$ indicating today's expectation of next year's value of x . Equations can span more than one line of the program, but be sure that each equation ends with a semi-colon.

You will need to add Dynare code for the three linearized log-deviation equations that you derive in Problem 2 above based on equations (3), (4), and (6).

Steady-state section

The variables of our Dynare model are all log-deviations from the steady state, thus their steady-state values are simply zero: when the economy is in the steady state there are no deviations. This makes the steady-state section trivial and it need not be changed for your simulations. The steady and check commands simply direct Dynare to verify that these are indeed a steady state and that certain dynamic-stability conditions hold.

Shocks section

This section defines the shocks that you want to simulate. There are two kinds of simulations that you can run in Dynare: deterministic and stochastic. In a deterministic simulation, you set all shocks to zero except those specified in the shocks section and you use the simul command in the computation section. The file shown above is set up to run a deterministic simulation in which the value of epsa (ϵ_A) is set to +0.01 (one percent) in period one. All other shocks (the government spending shock) and values of epsa in periods other than period one are set to zero. This command should generate simulations that match those shown in Romer's Figures 5.2 through 5.4. To perform simulations of the government spending shock, you would change epsa to epsg in the var command of the shocks section.

Stochastic simulations involve drawing a random value for each shock in each period from a normal probability distribution with specified standard deviations and correlations for the shocks. For example, to simulate the model with random shocks to both epsa and epsg each having a standard deviation of 0.005 (one-half percent) and no correlation between them, you would enter the shocks section as follows:

```
|// *** Shocks section for stochastic simulation***
```

```
shocks;  
var epsa; stderr 0.005;  
var epsg; stderr 0.005;  
end;
```

The fact that you have not entered a correlation value means that Dynare sets it to zero. In fact, any standard deviation or correlation that is not explicitly set in the shocks section is assumed to be zero, so if you left out the `var epsg` line, you would get a simulation with the technology shock varying randomly but the government-spending shock set to zero throughout (because a random variable with a standard deviation of zero is just a constant).

Computation section

The computation section directs Dynare to simulate the model and create specific output elements that you want. As with the shocks section, the computation section will depend on whether you are doing a deterministic or stochastic simulation.

The program above shows a deterministic simulation using the `simul` command. The `periods=200` option tells Dynare to assume that the model will be back in (or sufficiently close to) the steady-state after 200 periods (quarters). Setting this option to a value that is too small will cause the simulation to be incorrect. Examine your results closely to see if there is a “jump” at the end of the simulations. If there is, then you haven’t set this option high enough. There are many other options for the `simul` command that are described in the Dynare documentation.

If you are doing a stochastic simulation, you will use the `stoch_simul` command. Like the `simul` command, it takes a `periods` option that determines the number of periods in the simulation. In a stochastic simulation, you may want to set the number of periods to be much larger (2000?) so that the summary statistics of the simulation can be averaged over a larger number of periods.

The stochastic simulation automatically generates useful output on your screen. This includes the basic summary statistics of your endogenous variables over the periods of the simulation, the pairwise correlations between them, and the autocorrelations of each variable. This output may be inconvenient to copy and paste from the screen window, but it is replicated in a text file with the suffix `.log` following the same name that your `.mod` file has. Dynare creates this `.log` and places it in the same folder as your `.mod` file. You can easily open this file in a text editor and copy results into your report. The stochastic simulation also generates impulse-response graphs for each shock, which should be similar to the output of the deterministic simulations of the shocks.

Producing graphic output

Dynare and Octave/MATLAB are capable of graphing your impulse-response functions (from either a deterministic or stochastic simulation) and your simulated time series from a stochastic simulation, but these are commands that are sometimes platform dependent. Running Dynare under Octave on the Macintosh platform has sometimes proved troublesome, so if you run into trouble there are two practical choices for performing the

computations on this project. If you have a Windows computer, you can download both Dynare and Octave onto it and run that way. Or you can use one of the Reed IRC/PPW computers on which MATLAB and Dynare are installed.

Running Dynare under Macintosh / MATLAB

In order to run your Dynare program on Reed's public computers, you can use MATLAB, which is available in the Application folder. After starting MATLAB, you will get a command prompt. Type in the (case-sensitive) command

```
addpath /Applications/Dynare/4.5.7/matlab
```

This tells MATLAB where to look for the Dynare program. It is possible that you will be using a version other than 4.5.7; if so, you may need to adjust the directory name here.

Next you tell MATLAB where to find your .mod file (and where to put its output files). The easiest option is to put your files in an identifiable (i.e., including your name) folder on the Mac desktop. (Note that on a public Reed computer this folder might be erased during the overnight re-initiation of the computer. Be sure to keep a copy someplace secure.) If you use the name "Jane314" for this folder, then you would type

```
cd ~/Desktop/Jane314/
```

to change MATLAB's default directory.

Now you are ready to tell MATLAB to run your Dynare mod file. If your file (inside the Jane314 folder) is called mymodfile.mod, then you type

```
dynare mymodfile.mod
```

If you have computer-related problems running Dynare on the campus lab computers, contact Kristin Bott in Instructional Technology Services at bottk@reed.edu. Obviously, macro-related problems should come to me.

Running Dynare under Windows / Octave (not recommended!)

Once you have created your .mod file, you can execute it with Dynare. To do so requires three commands. If you are running Octave under Windows, for example, you first need to start Octave, then, at the Octave command prompt, type

```
addpath c:\dynare\4.5.7\matlab
```

This command tells Octave where to find the Dynare execution file. It is possible that you will be using a version other than 4.5.7; if so, you may need to adjust the directory name here. (If you put the Dynare software in a different folder, you would need to change this command to point to that folder.)

You next need to tell Octave where to find your .mod file. The easiest way to do this is to make it your default Octave directory. If you put your .mod file in a folder called mymacroasgn inside the c:\dynare folder, you would say

```
cd c:\dynare\mymacroasgn
```

Now you should be ready to run your Dynare program by typing into Octave:

```
dynare mymodfile.mod
```

where mymodfile.mod is the name of your .mod file, which is located in the folder pointed to by your cd command above.

Your results

If Dynare runs your program successfully, you will see the graphic output you have requested in new windows and (for stochastic simulations) tables of results in your MATLAB window. The results are also saved in a file called `mymacroasgn.log` (where `mymacroasgn` is the name of your `.mod` file) in the same folder as your `.mod` file. To get these results into your report document (or into Excel or another program for further processing) it is easier to copy and paste from the `.log` file than from the MATLAB screen itself.

Dynare diagnostics are largely useless, so if your program does not execute successfully you are probably going to be on your own to figure out why. Common errors to watch for include misspelling variable names, mismatched parentheses in expressions, and leaving off the semi-colon at the end of a command.

3. Replicate Romer's Figures 5.2 through 5.7 for a deterministic simulation first with A, then with G, shocked by 1% (0.01 shock to the log of the variable). Interpret the results. Are they the same as in the textbook? Note that the government spending paths will be more readable if you change the scale on the vertical axis. (Please include a listing of your Dynare code with your report.)
4. Some critics have argued that the only reason this model shows considerable persistence is that the rho parameters are set to 0.95. Rerun the simulations of the previous part with both rho values set to 0.50. How different do the results look? Are the effects larger or smaller? Why? Are they more or less persistent?
5. Perform stochastic simulations of the basic RBC model for each of the following values of the standard deviations of the shocks (again, $0.01 = 1\%$):
 - (a) Only productivity shocks: $sd(\varepsilon_A) = 0.01$, $sd(\varepsilon_G) = 0$;
 - (b) Both productivity and government spending shocks: $sd(\varepsilon_A) = 0.005$, $sd(\varepsilon_G) = 0.005$;
 - (c) Only government spending shock: $sd(\varepsilon_A) = 0$, $sd(\varepsilon_G) = 0.01$.

If you graph the results, be sure to scale them in a way that makes them understandable. Create a table to summarize the results of each part. Each table should contain (in a format similar to the one below) the following results captured from your screen or the log file: (i) the standard deviations of log GDP, log consumption, log employment, log real wages, the real interest rate, and log government spending; (ii) the correlation coefficients between log GDP and each of the other variables; (iii) the first-order autocorrelations of all variables. The table below describes these measures for quarterly cyclical fluctuations in the U.S. from 1985 to 2009. Which measures correspond well to the U.S. economy and which do not? (Compare relative sizes, not absolute levels.) Summarize your results briefly with an overall assessment of your RBC simulation results.

Variable	Standard deviation	Correlation with GDP	1st-order autocorrelation
log GDP	0.0113	(1.000)	0.9110
log consumption	0.0099	0.9302	0.9239
log employment	0.0108	0.8369	0.9182
log real wage	0.0048	0.0133	0.9339
real interest rate	0.0025	0.5516	0.9250
log gov't spending	0.0025	-0.4268	0.9059