

Tutorial: Estimating parameters in macroeconomic models

by Kevin Kotzé

The objective of this tutorial is to provide an introduction to the basic functionality that is required to estimate models in Dynare with the use of Bayesian techniques. Data for the South African economy has been used for this purpose, to identify the stochastic shocks and parameter values.¹ The model that is used for this purpose, takes the form of a closed-economy New-Keynesian variety, which is presented in Galí (2008) and Galí (2015).

In addition the specification of the model is consistent with one that does not contain trends, and as such, all of the trends in the observed variables (within the dataset) have been removed, during the data transformation process. One of the reasons for doing so, is that this model is solved for variables that were log-linearised. When completing this calculation, we can make use of the method that was developed by Uhlig (1999), or a Taylor series approximation around a localised steady-state.

To then ensure that the observed variables have a well-defined steady-state, we would often remove the trend to allow for them to display stationary characteristics.² Therefore, such a model would describe the behaviour of the economy around the Balanced Growth Path (BGP), which may be termed the business cycle, while abstracting from the movement of the actual BGP.³

Then finally, as this section is practically-oriented and largely serves as an introduction to these tools, you may wish to refer to the more comprehensive discussion of these techniques in DeJong and Dave (2011), while a more complete discussion the practical tools that are used to implement these techniques is contained in Adjemian et al. (2011).⁴

1 Simulating the Model

Before estimating the parameters in a model, it is usually always a good idea to first simulate the model with calibrated parameters. This serves as a very useful step, as it will consider whether the Blanchard-Khan conditions are satisfied. In addition, it will also highlight any typos or other input errors that may be contained in the first few blocks of your model file.⁵ During this stage you can also determine whether the impulse response functions and other results appear to be reasonable. If you cannot get your model to run correctly with the aid of a calibrated simulation, it is highly unlikely that the estimation of the model will improve matters.

The file `NK_simul.mod` seeks to accomplish this objective, where you will note that we have included a number of additional equations that calculate annual rates from the quarterly counterparts, since most of the results in Galí (2008) are quoted in annual terms.

```
% Dynare file for a closed-economy New Keynesian Model

% See chapter 3 of Gali (2008), "Monetary Policy, Inflation,
% and the Business Cycle" for details of the model

% The model is simulated in order to describe the evolution
% of deviations from the steady-state of endogenous
% variables following a shock

var pi          % inflation
    y_gap       % output gap
    y_ss        % steady-state of output
    y           % output
    r_nat       % natural interest rate
    r_real      % real interest rate
    in          % nominal interest rate
    n           % hours worked
    m_grow_ann  % money growth
    nu          % AR(1) monetary policy shock process
    a           % AR(1) technology shock process
    r_real_ann  % annualized real interest rate
    i_ann       % annualized nominal interest rate
    r_nat_ann   % annualized natural interest rate
    pi_ann;     % annualized inflation rate

varexo eps_a    % technology shock
    eps_nu;     % monetary policy shock

parameters sigma phi phi_pi phi_y theta rho_nu rho_a betta eta alpha epsilon
Omega psi_n_ya lambda kappa;

sigma = 1;      % log utility
phi = 1;        % unitary Frisch elasticity
phi_pi = 1.5;   % inflation feedback Taylor Rule
phi_y = .5/4;   % output feedback Taylor Rule
theta = 2/3;    % Calvo parameter
rho_nu = 0.5;   % autocorrelation monetary policy shock
rho_a = 0.9;    % autocorrelation technology shock
betta = 0.99;   % discount factor
eta = 4;        % semi-elasticity of money demand
alpha = 1/3;    % capital share
epsilon = 6;    % demand elasticity
Omega = (1-alpha)/(1-alpha+alpha*epsilon);      % see pg 47
psi_n_ya = (1+phi)/(sigma*(1-alpha)+phi+alpha); % see pg 48
lambda = (1-theta)*(1-betta*theta)/theta*Omega; % see pg 47
kappa = lambda*(sigma+(phi+alpha)/(1-alpha));    % see pg 49

model(linear);
pi=betta*pi(+1)+kappa*y_gap;      % 1 NK Phillips Curve (eq. 21)
y_gap=-1/sigma*(in-pi(+1)-r_nat)+y_gap(+1); % 2 Dynamic IS Curve (eq. 22)
in=phi_pi*pi+phi_y*y_gap+nu;     % 3 Interest Rate Rule (eq. 25)
y=a+(1-alpha)*n;                % 4 Production function (eq. 13)
```

```

m_grow_ann=4*(y-y(-1)-eta*(in-in(-1))+pi); % 5 Money growth (eq. 4)
r_nat=sigma*psi_n_ya*(a(+1)-a); % 6 Natural interest (eq. 23)
r_real=in-pi(+1); % 7 Real interest rate
y_ss=phi*a; % 8 Steady-state of output
y_gap=y-y_ss; % 9 Output gap
i_ann=4*in; % 10 Annual nom interest rate
r_real_ann=4*r_real; % 11 Annual real interest rate
r_nat_ann=4*r_nat; % 14 Annual nat interest rate
pi_ann=4*pi; % 15 Annual inflation
nu=rho_nu*nu(-1)+eps_nu; % 16 Monetary policy shock
a=rho_a*a(-1)+eps_a; % 17 Technology shock
end;

shocks;
var eps_nu = 0.25^2; % 1 std dev shock of 25 points (i.e. 1% annual)
var eps_a = 1^2; % 1 std dev shock to technology
end;

steady; % should all be zero (dev. from zero steady-state)
check;

stoch_simul(order = 1,irf=15) y_gap pi_ann y n i_ann r_real_ann m_grow_ann nu
a;

```

In this case you would note that the impulse response functions are exactly equivalent to those that are provided in figures (3.1) and (3.2) of Galí (2015). Hence, we have a fair degree of confidence in the fact that the structure of the model is sound and we can proceed with the estimation of the parameters in the model.

2 Observed data and stochastic error terms

To estimate certain parameters in this model we could consider the extent of price rigidities in the market, or the central banks reaction to output fluctuations. From your courses in econometrics, you would appreciate that each observed variable is made up of an explained part and an unexplained stochastic part. Therefore, we will need as many stochastic terms as there are observed variables. Or to use the common expression, "... we use the observed variables to identify each of the shocks in model."⁶ Of course, one may include more stochastic terms than observed variables, but in this case you need to ensure that the shocks are correctly identified, by paying careful attention to the results of the impulse response functions and other results.

In the `NK_simul.mod` model we could include data for the interest rate and for technology (which could be derived from production data) to identify the monetary policy and technology shocks. To allow for the dataset to incorporate a number of other interested variables, we can extend the model to incorporate a cost-push shock, which could be identified by inflation, as well as an aggregate demand shock, which could be identified by the output gap. This would give us the file `NK_stoch.mod`, which includes;

```

var ...
    pssi          % AR(1) cost-push process
    gamma;        % AR(1) aggregate demand process

varexo ...
    eps_c         % cost-push shock
    eps_d;        % aggregate demand shock

parameters ... rho_c rho_d;

...
rho_c = 0.5;      % autocorrelation cost-push shock
rho_d = 0.5;      % autocorrelation demand shock

model(linear);
pi=beta*pi(+1)+kappa*y_gap+pssi;    % 1 NK Phillips Curve
y_gap=-1/sigma*(in-pi(+1)-r_nat)+y_gap(+1)+gamma; % 2 Dynamic IS Curve
...
pssi=rho_c*pssi(-1)+eps_c;          % 18 Cost-push shock
gamma=rho_d*gamma(-1)+eps_d;        % 19 Aggregate demand shock
end;

shocks;
...
var eps_c = 0.5^2;      % 1/2 std dev shock to demand shock
var eps_d = 0.5^2;      % 1/2 std dev shock to cost-push shock
end;

```

2.1 Declaring observed variables and priors for estimation

while the first parts of this model are the same as those of the stochastic simulation, we are now going to include data for an observed interest rate (initially). Since the declaration of the model variables, parameters, and model equations are equivalent to that which was included in the stochastic simulation, we begin our new file `NK_est1.mod`, with the same blocks for these elements. In addition, you will also note that the declaration of the model is also as per the stochastic simulation.

2.2 Calibrating those parameters that are not going to be estimated

It is not usually a good idea to estimate all the parameters at once, when estimating the model for the first time, as you will not know where the problem arises should it not find a solution. After performing a stochastic simulation, start with estimating a few of the parameters with relatively tight priors. Note that we have commented out a number of parameters, which were previously calibrated, since this is the parameter that we want to estimate. Commenting out these calibrated parameters is not always necessary, since if you specify that you want to estimate the parameter, the calibrated value will just act as a starting value. Note that where there are parameters that rely on each other, you will need to calibrate at least one of them if the solution is to be tractable.

```

sigma = 1;      % log utility
phi = 1;        % unitary Frisch elasticity
%phi_pi = 1.5;  % inflation feedback Taylor Rule
%phi_y = .5/4;  % output feedback Taylor Rule
theta = 2/3;    % Calvo parameter
%rho_nu = 0.5;  % autocorrelation monetary policy shock
rho_a = 0.9;    % autocorrelation technology shock
%rho_c = 0.5;   % autocorrelation cost-push shock
%rho_d = 0.5;   % autocorrelation demand shock
betta = 0.99;   % discount factor
eta = 4;        % semi-elasticity of money demand
alphha = 1/3;   % capital share
epsilon = 6;    % demand elasticity
Omega = (1-alphha)/(1-alphha+alphha*epsilon); % see pg 47
psi_n_ya = (1+phi)/(sigma*(1-alphha)+phi+alphha); % see pg 48
lambda = (1-theta)*(1-betta*theta)/theta*Omega; % see pg 47
kappa = lambda*(sigma+(phi+alphha)/(1-alphha)); % see pg 49

```

2.3 Declaring observable variables

This should not come as a surprise. Dynare must know which variables are observable for the estimation procedure. Note, these variables must be available in the data file, as explained in section *Launching the estimation* below. For the moment, we write:

```
varobs y_gap pi in;
```

2.4 Declaring priors

Priors play an important role in Bayesian estimation and consequently deserve a central role in the specification of the `NK_est1.mod` file. Priors, in Bayesian estimation, are declared as a distribution. The general syntax to introduce priors in Dynare is the following:

```

estimated_params ;
PARAMETER_NAME, PRIOR_SHAPE, PRIOR_MEAN, PRIOR_STANDARD_ERROR [, PRIOR_3RD_PARAMETER] [, PRIOR_4TH_PARAMETER];
end;

```

where the following table defines each term more clearly

PRIOR_SHAPE	Corresponding distribution	Range
normal_pdf	$N(\mu, \sigma)$	\mathbb{R}
gamma_pdf	$G_2(\mu, \sigma, p_3)$	$[p_3, +\infty)$
beta_pdf	$B(\mu, \sigma, p_3, p_4)$	$[p_3, p_4]$
inv_gamma_pdf	$IG_1(\mu, \sigma)$	\mathbb{R}^+
uniform_pdf	$U(p_3, p_4)$	$[p_3, p_4]$

where μ is the PRIOR_MEAN, σ is the PRIOR_STANDARD_ERROR, p_3 is the PRIOR_3RD_PARAMETER (whose default is 0) and p_4 is the PRIOR_4RD_PARAMETER (whose default is 1).

Note that when specifying a uniform (flat) distribution between 0 and 1 as a prior for a parameter, say α , you should put two empty spaces for parameters μ and σ , and then specify parameters p_3 and p_4 , since the uniform distribution only takes p_3 and p_4 as arguments. For instance, you would write `alpha, uniform_pdf, , , 0,1; .`

For a more complete review of all possible options for declaring priors, as well as the syntax to declare priors for maximum likelihood estimation (not Bayesian), see the *Dynare Reference Manual*, which may be located in the `C:\dynare\4.5.6\doc\dynare.pdf`.

Choosing the appropriate prior for your parameters can be a tricky, yet very important endeavour. It is worth spending time on your choice of priors and to test the robustness of your results to your priors. Some considerations may prove helpful.

First, think about the domain of your prior over each parameter. Should it be bounded? Should it be opened on either or both sides? Remember also that if you specify a probability of zero over a certain domain in your prior, you will necessarily also find a probability of zero in your posterior distribution. Then, think about the shape of your prior distribution. Should it be symmetric? skewed? If so, on which side? You may also go one step further and build a distribution for each of your parameters in your mind. Ask yourself, for instance, what is the probability that your parameter is bigger than a certain value, and repeat the exercise by incrementally decreasing that value. You can then pick the standard distribution that best fits your perceived distribution.

Finally, instead of describing here the shapes and properties of each standard distribution available in Dynare, you are instead encouraged to visualize these distributions yourself, either in a statistics book or on the Web. In addition, you should also look at what the literature suggests. For the South African economy you may want to consult Alpanda, Kotzé, and Woglom (2010b), Alpanda, Kotzé, and Woglom (2010a), and Alpanda, Kotzé, and Woglom (2011).

In this example those parameters that may take on positive real numbers are assigned gamma distributions, while those that are bounded to be between zero and one have beta distributions. The mean for this parameters largely correspond with the calibrated values, but the size of the autoregressive terms have been decreased slightly, as these should not be too close to one.⁷ In this instance we have not employed bounds, however, the second moment for some of the parameters is rather small (i.e. 0.01) to avoid it from moving too far from the expected value. For the standard deviations of the shocks, we make use of an inverse-gamma distribution as is standard practice, with a small mean and an infinite standard deviation.

```

estimated_params;
phi, gamma_pdf, 1, 0.05;
theta, beta_pdf, 0.66, 0.1;
rho_nu, gamma_pdf, 0.5, 0.05;
rho_c, gamma_pdf, 0.5, 0.1;
rho_d, gamma_pdf, 0.5, 0.1;
phi_y, gamma_pdf, 0.25, 0.05;
phi_pi, gamma_pdf, 1.5, 0.05;
stderr eps_nu, inv_gamma_pdf, 0.005, inf;
stderr eps_c, inv_gamma_pdf, 0.005, inf;
stderr eps_d, inv_gamma_pdf, 0.005, inf;
end;

```

Since we have not included data to identify the technology shock we calibrate the variance of this shock as before:

```

shocks;
var eps_a = 1^2;          % 1 std dev shock to technology
end;

```

3 Launching the estimation

To estimate a model in Dynare, all that is necessary is to add the command `estimation` at the end of the `NK_est1.mod` file. In addition, you also need to specify the name of the file in which the data is stored. Easy enough, but you should be aware of some of the options available for the command (to be entered in parentheses and sequentially, separated by commas, after the command). Before providing a list of some of the most common and useful options, I'd like to encourage you to view the *Dynare Reference Manual* for a complete list of features.

- `datafile = FILENAME`: the datafile (a `.m` file, a `.mat` file, or an `.xls` file). Note that the variables do not need to show up in any order, but vectors of observations need to be named with the same names as those in `varobs`. In Excel files, for instance, observations could be ordered in columns, and variable names would show up in the first cell of each column.
- `nobs = INTEGER`: the number of observations to be used (default: all observations in the file)
- `first_obs = INTEGER`: the number of the first observation to be used (default = 1). This is useful when running loops, or instance, to divide the observations into sub-periods.
- `prefilter = 1`: the estimation procedure demeans the data (default=0, no prefiltering). This is useful if model variables are in deviations from steady state, for instance, and therefore have zero mean. Demeaning the observations would also impose a zero mean (steady-state) on the observed variables.

- `mh_replic = INTEGER`: number of replication for Metropolis Hasting algorithm. For the time being, `mh_replic` should be larger than 1200 (default = 20000)
- `mh_nblocks = INTEGER`: number of parallel chains for Metropolis-Hasting algorithm (default = 2). Despite this low default value, it is advisable to work with a higher value, such as 5 or more. This improves the computation of between group variance of the parameter means, one of the key criteria to evaluate the efficiency of the Metropolis-Hastings to evaluate the posterior distribution. However, as this often takes a great deal of time to run, most researchers use 2 chains, at most.
- `mh_drop = DOUBLE`: the fraction of initially generated parameter vectors to be dropped before using posterior simulations (default = 0.5; this means that the first half of the draws from the Metropolis-Hastings are discarded).
- `mh_jscale = DOUBLE`: the scale to be used for the jumping distribution in MH algorithm. The default value is rarely satisfactory in large models. This option must be tuned to obtain, ideally, an acceptance rate of 25% in the Metropolis-Hastings algorithm (default = 0.2). The idea is not to reject or accept too often a candidate parameter; the literature has settled on a value of between 0.2 and 0.4. If the acceptance rate were too high, your Metropolis-Hastings iterations would never visit the tails of a distribution, while if it were too low, the iterations would get stuck in a subspace of the parameter range. Note that the acceptance rate drops if you increase the scale used in the jumping distribution and vice a versa.
- `mh_init scale = DOUBLE`: the scale to be used for drawing the initial value of the Metropolis-Hastings chain (default=2*`mh_jscale`). The idea here is to draw initial values from a stretched out distribution in order to maximize the chances of these values not being too close together, which would defeat the purpose of running several blocks of Metropolis-Hastings chains.
- `mode_compute=INTEGER`: specifies the optimizer for the mode computation.

```
- 0: the mode isn't computed. `mode file` must be specified
- 1: uses Matlab fmincon (see the *Dynare Reference Manual* for additional options).
- 2: uses Lester Ingber's Adaptive Simulated Annealing.
- 3: uses Matlab `fminunc`.
- 4 (default): uses Chris Sims `csminwel`.
```

There are currently many more algorithms available (and they are likely to grow).

- `mode_check`: when `mode_check` is set, Dynare plots the minus of the posterior density for values around the computed mode for each estimated parameter in

turn. This is helpful to diagnose problems with the optimizer. A clear indication of a problem would be that the mode is not at the trough (bottom of the minus) of the posterior distribution.

- `bayesian_irf`: triggers the computation of the posterior distribution of impulse response functions (IRFs). The length of the IRFs are controlled by the `irf` option. To build the posterior distribution of the IRFs, Dynare pulls parameter and shock values from the corresponding estimated distributions and, for each set of draws, generates an IRF. Repeating this process often enough generates a distribution of IRFs. Note that if you stop the estimation procedure after calculating the posterior mode, or carry out maximum likelihood estimation, only the corresponding parameter estimates will be used to generate the IRFs. If you instead carry out a full Metropolis-Hastings estimation, on the other hand, the IRFs will use the parameters the posterior distributions, including the variance of the shocks.

All options available for `stoch_simul` can simply be added to the above options, separated by commas. To view a list of these options, either see the *Dynare Reference Manual*.

- `forecast = INTEGER`: computes the posterior distribution of a forecast on `INTEGER` periods after the end of the sample used in estimation. The corresponding graph includes one confidence interval describing uncertainty due to parameters and one confidence interval describing uncertainty due to parameters and future shocks. Note that Dynare cannot forecast out of the posterior mode. You need to run Metropolis-Hastings iterations before being able to run forecasts on an estimated model. Finally, running a forecast is very similar to an IRF, as in `bayesian_irf`, except that the forecast does not begin at a steady state, but simply at the point corresponding to the last set of observations. The goal of undertaking a forecast is to see how the system returns to steady state from this starting point. Of course, as observation do not exist for all variables, those necessary are reconstructed by sampling out of the posterior distribution of parameters. Again, repeating this step often enough yields a posterior distribution of the forecast.

Since the model has an steady-state of zero, it is necessary to make use of the `prefilter=1` if the data has not been previously demeaned. You then need to create an excel spreadsheet, or a Matlab file (`.mat` or `.m`) with the data in it. This file needs to employ the same naming convention as that which is provided in the `varobs` declaration. In this example, `NK_est1.mod`, we could make use of the following options:⁸

```
estimation(datafile = SAdata, prefilter = 1)
```

Once again we would possible like to inspect the impulse response functions and as such we would make use of the stochastic simulation command, where we look at the effect of a shock over twelve periods (quarters):

```
stoch_simul(order = 1, irf=15) y_gap pi_ann y n i_ann r_real_ann m_grow_ann nu
a;
```

4 Interpreting output

As in the case of model solution and simulation, Dynare returns both tabular and graphical output. On the basis of the options entered in the `NK_est1.mod` file, Dynare will display the following results.

4.1 Tabular results

After Dynare has loaded the data it starts making use of numerical iterations to find the posterior mode. When the improvement from one iteration to the next approximates zero, Dynare give the value of the objective function (the posterior Kernel) at the mode and displays two important table summarizing results from posterior maximization.

```
Improvement on iteration 33 =          0.000000095
improvement < crit termination
Objective function at mode: -538.512563}
```

The first table then summarizes results for parameter values. It includes: prior means, posterior mode, standard deviation of the mode (based on the assumption of a Normal, probably erroneous when undertaking Bayesian estimation, as opposed to standard maximum likelihood), as well as the prior distribution and standard deviation (pstdev).

```
RESULTS FROM POSTERIOR MAXIMIZATION

parameters
  prior   mean    mode    s.d. prior pstdev
phi       1.000   0.9975  0.0499 gamm  0.0500
theta     0.660   0.6765  0.1061 beta  0.1000
rho_nu    0.500   0.5576  0.0597 gamm  0.0500
rho_c     0.500   0.4371  0.0702 gamm  0.1000
rho_d     0.500   0.4789  0.0974 gamm  0.1000
phi_y     0.250   0.6098  0.0870 gamm  0.0500
phi_pi    1.500   1.4992  0.0492 gamm  0.0500

standard deviation of shocks
  prior mean    mode    s.d. prior pstdev
eps_nu  0.005   0.0132  0.0012 invg      Inf
eps_c   0.005   0.0050  0.0006 invg      Inf
eps_d   0.005   0.0023  0.0009 invg      Inf
```

It is followed by a second table summarizing the same results for the shocks (if these are estimated). It may be entirely possible that you get an infinite value for a standard deviation, this is simply the limit case of the inverse gamma distribution and it implies that the parameter has not been accurately identified by the data.

5 Graphical results

Figure 1 comes up soon after launching Dynare as little computation is necessary to generate it. The figure returns a graphical representation of the priors for each parameter of interest. Note that where the standard deviation of the prior is relatively small, the distribution is more concentrated around a particular value. Of course after evaluating the likelihood, the distribution for the posterior parameter estimate may occur outside of this distribution, but it will be less likely than the chance of occurring within it.

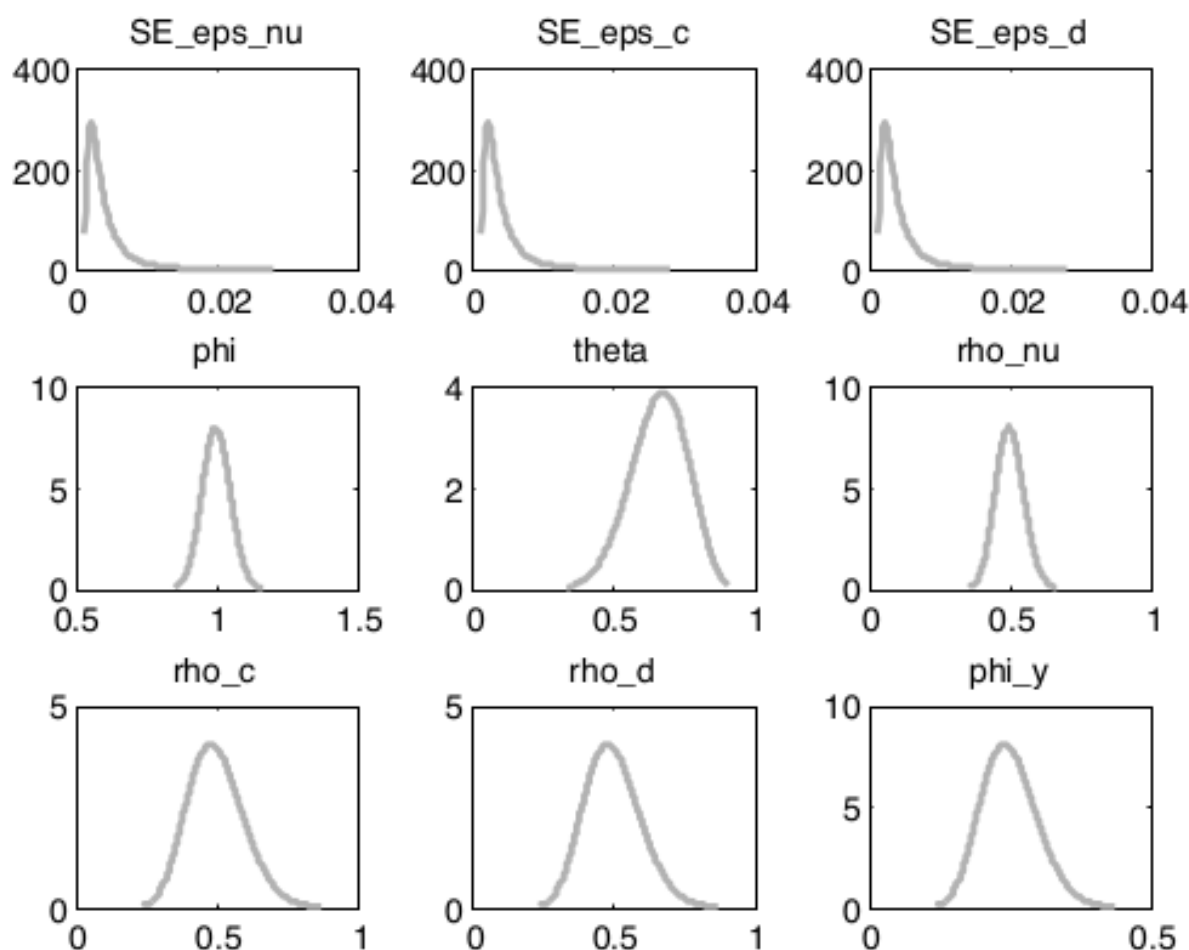
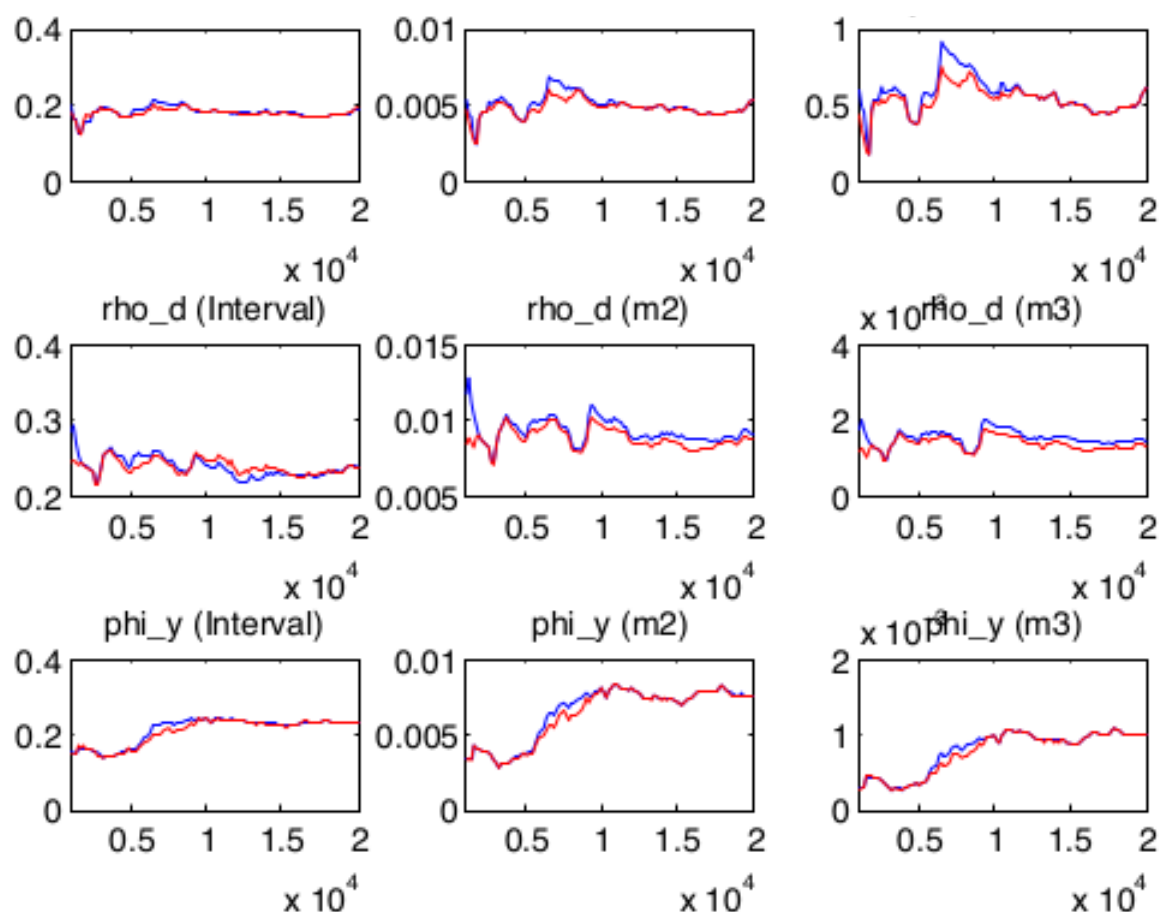


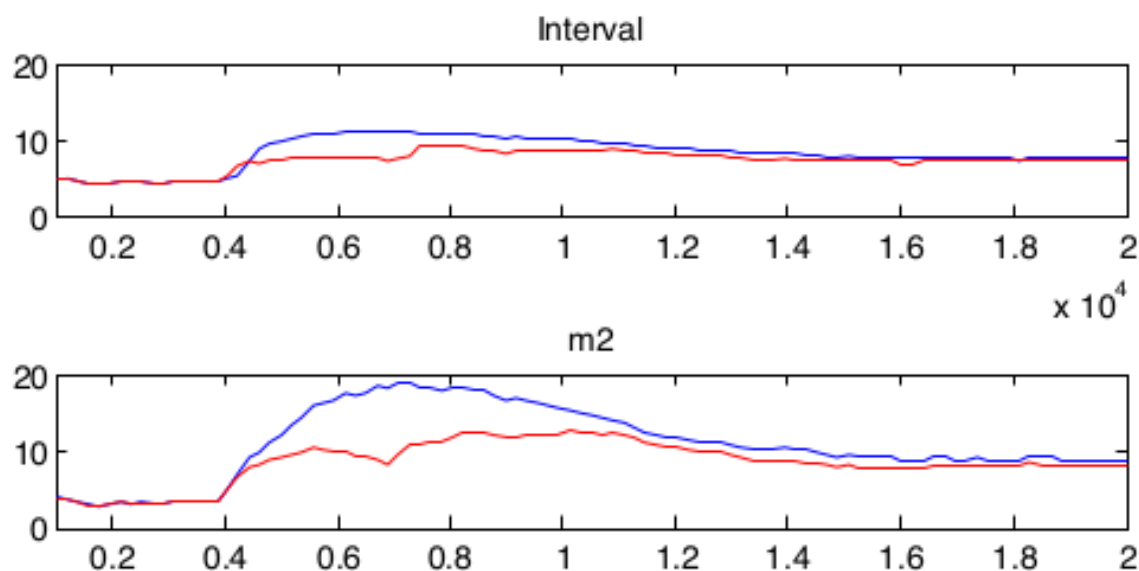
Figure 1: **Priors**

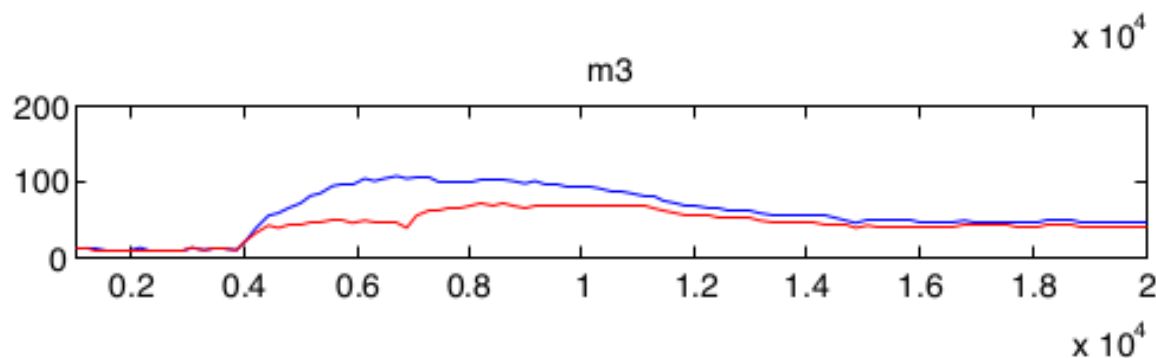
If you select more than one Monte Carlo Markov Chain then the second set of graphs display MCMC univariate diagnostics as depicted in figure 2. This is the main source of feedback to gain confidence, or spot a problem, with the results. Recall that Dynare completes several runs of Metropolis-Hastings simulations (as many as determined in the option `mh_nblocks`), where the algorithm will start each time from a different initial value. If the results from are reasonable and the optimizer did not get stuck in an odd area of the parameter space, one should be able to observe that the results within from the Metropolis-Hastings simulation are similar and the various chains should be close (or converge relatively quickly when the length of the chain is set at a relatively large number). In the graphs that follow, this is the case, particularly as the chains are fairly short.

rho_c (Interval) rho_c (m2) x 10⁶ rho_c (m3)

Figure 2: **MCMC univariate diagnostics**

More specifically, the red and blue lines on the charts represent specific measures of the parameter vectors both within and between chains. For the results to be sensible, these should be relatively constant (although there will always be some variation) and they should converge. Dynare reports three measures: “interval”, being constructed from an 80% confidence interval around the parameter mean, “m2”, being a measure of the variance and “m3” based on third moments. In each case, Dynare reports both the within and the between chains measures. The graph entitled “multivariate diagnostic” presents in figure 3 results of the same nature, except that they reflect an aggregate measure based on the eigenvalues of the variance-covariance matrix of each parameter.

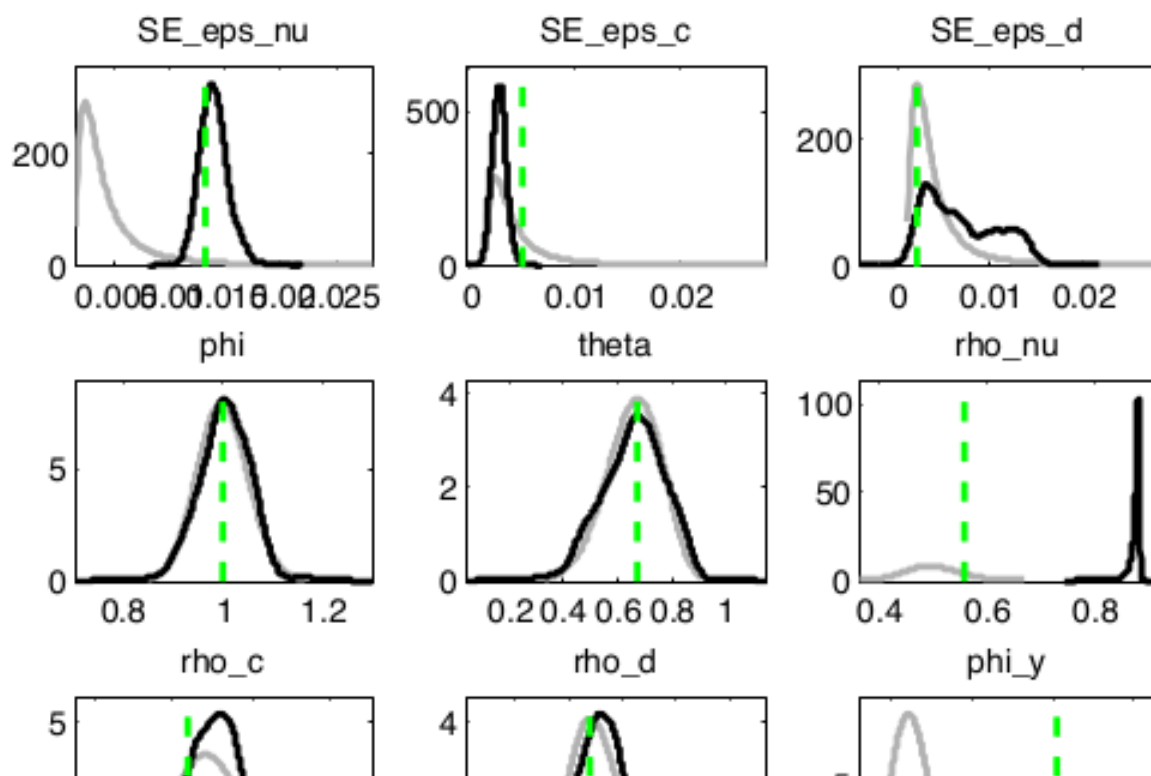


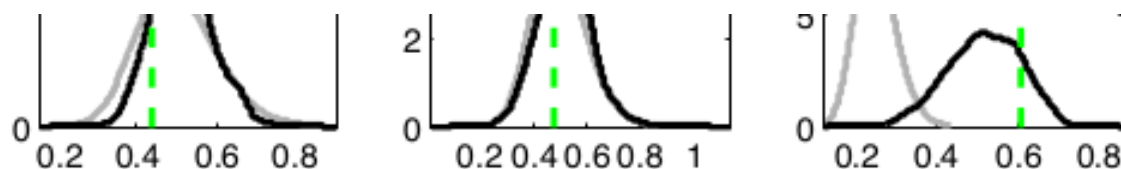
Figure 3: **MCMC multivariate diagnostics**

Note that the horizontal axis represents the number of Metropolis-Hastings iterations that have been undertaken, and the vertical axis the measure of the parameter moments, with the first, corresponding to the measure at the initial value of the Metropolis-Hastings iterations.

If the plotted moments are highly unstable or do not converge, you may have a problem of poor priors. It is advisable to redo the estimation with different priors. If you have trouble coming up with a new prior, try starting with a uniform and relatively wide prior and see where the data leads the posterior distribution. Another approach is to undertake a greater number of Metropolis-Hastings simulations.

The next graph displays more interesting results in figure 4, towards which most of the computations undertaken in this section are directed - the posterior distribution. In fact, the figure compares the posterior to the prior distribution (black vs. grey lines). In addition, on the posterior distribution, Dynare plots a green line which represents the posterior mode. These allow you to make statements about your data other than simply concerning the mean and variance of the parameters; you can also discuss the probability that your parameter is larger or smaller than a certain value.



Figure 4: **Posterior Parameter Estimate**

These graphs are of course especially relevant and present key results, but they can also serve as tools to detect problems or the lack thereof. First, the prior and the posterior distributions should not be excessively different in most cases, or else your data is telling you that your initial assertions were horribly incorrect. Second, the posterior distributions should be close to normal, or at least not display a shape that is clearly non-normal (i.e. multi-modal or excessively skewed), unless there is a specific reason for this. Third, the green mode (calculated from the numerical optimization of the posterior kernel) should not be too far away from the mode of the posterior distribution. If not, it is advisable to undertake a greater number of Metropolis-Hastings simulations.

The last set of graphs in figure 5 provide the smoothed estimated shocks in a useful illustration to eye-ball the plausibility of the size and frequency of the shocks. The horizontal axis, in this case, represents the number of periods in the sample. One thing to check is that the shocks should be centred around zero. That is indeed the case in our example. Similarly, we can also view the smoothed estimated variable in figure 6 which should follow the actual variable to a certain degree.

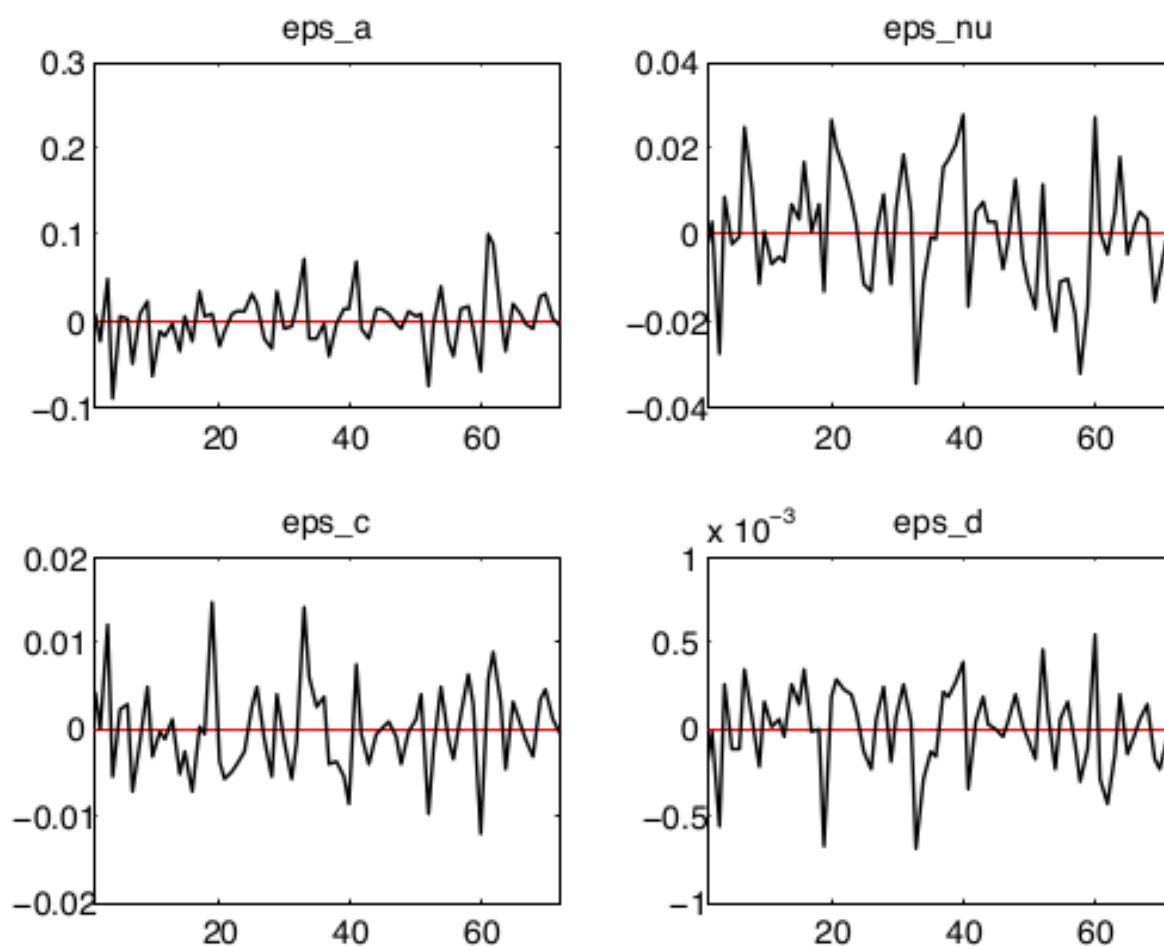
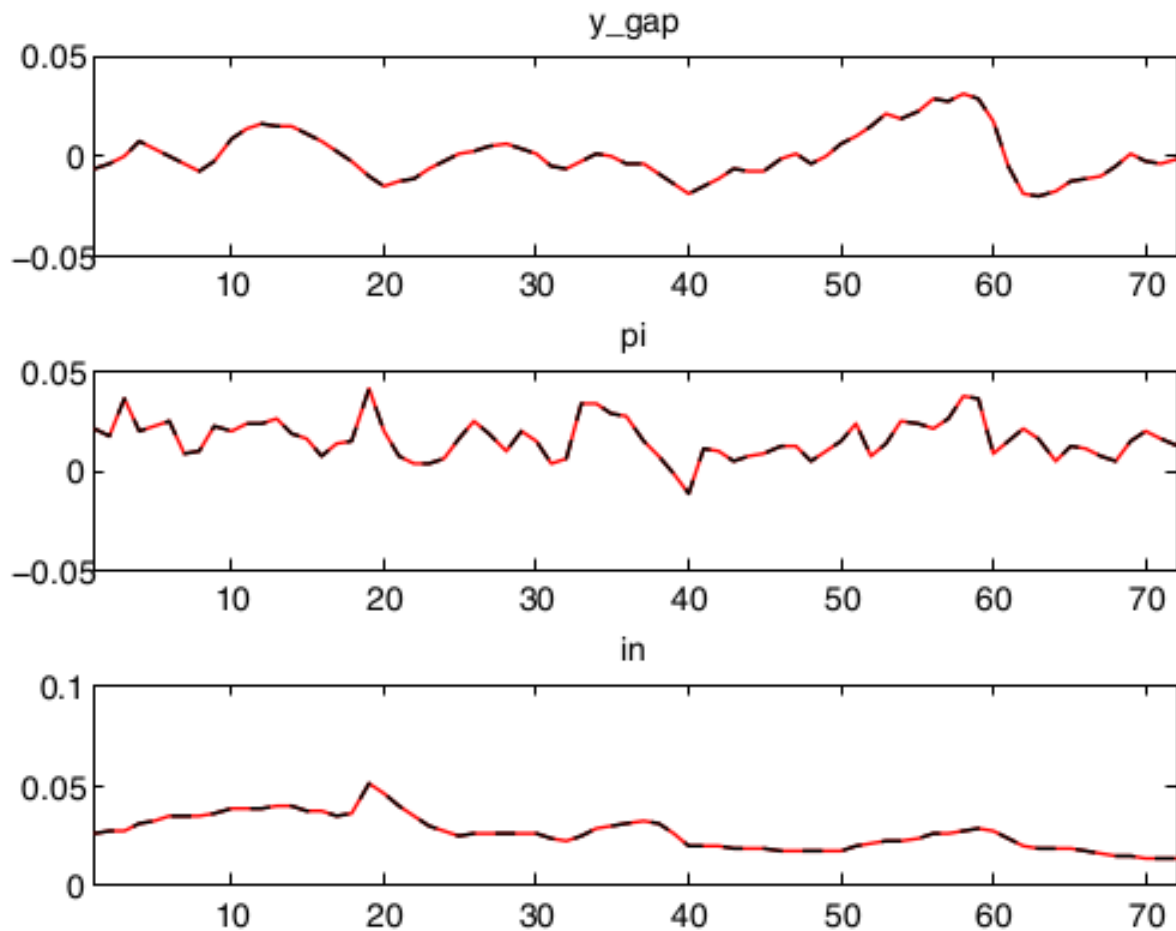
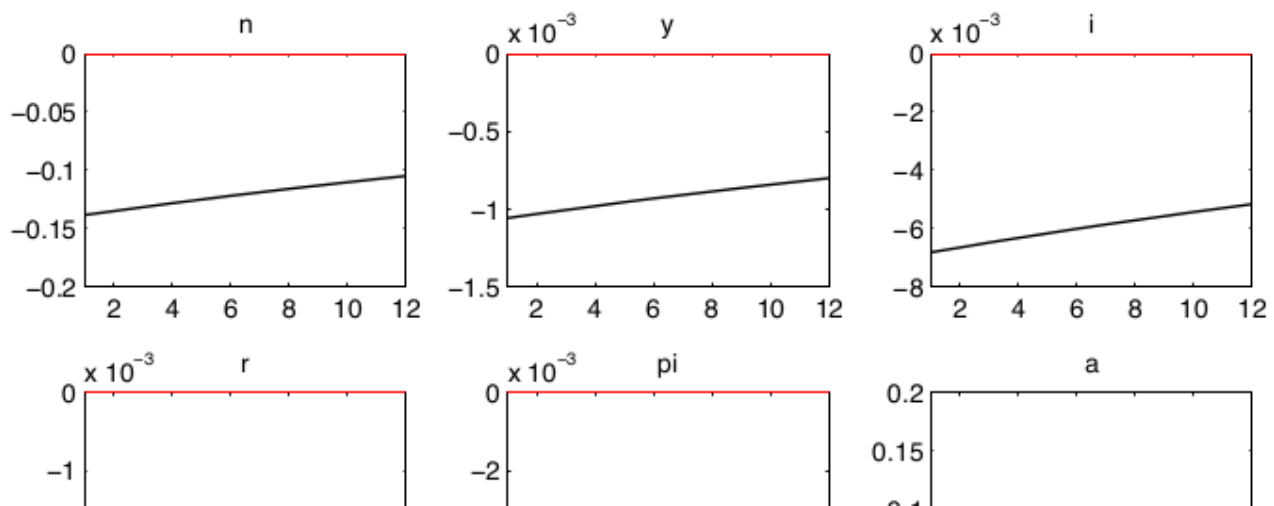
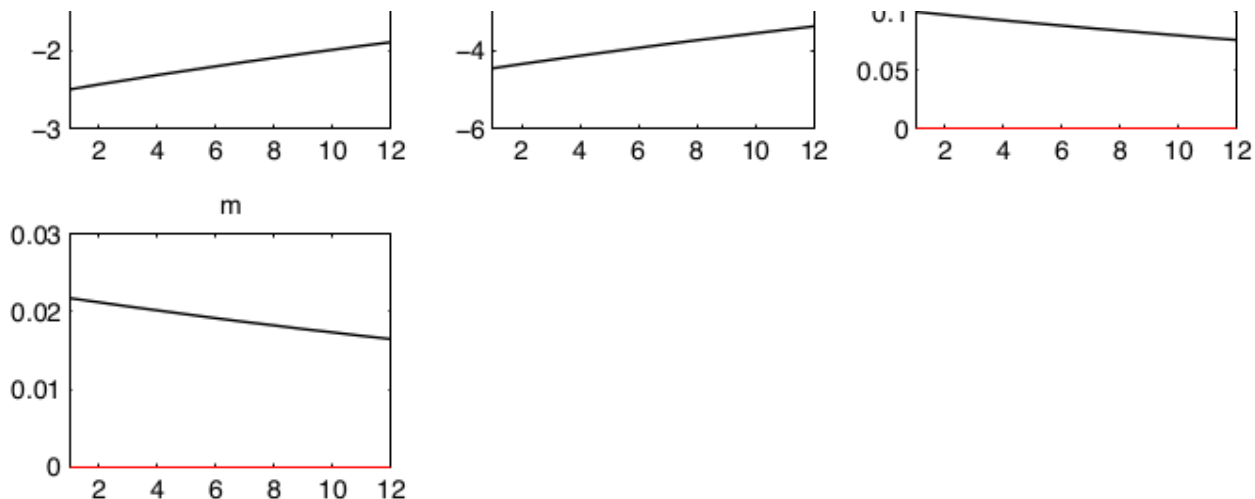
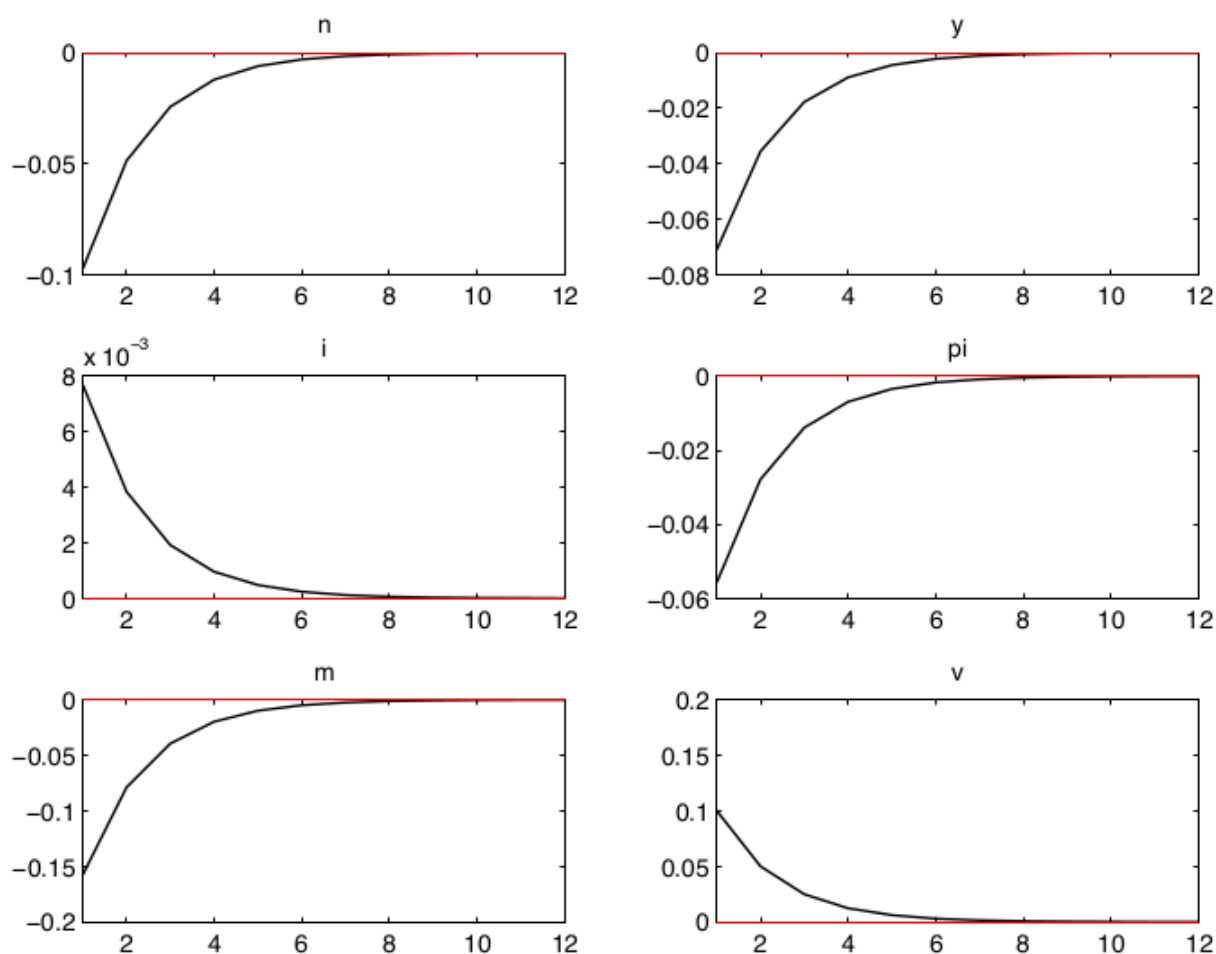


Figure 5: **Smoothed Estimated Shocks**Figure 6: **Smoothed Estimated Variables**

6 Stochastic Simulation Results

If one elects to perform a stochastic simulation, then Dynare will provide you with a number of impulse response functions, which show how the variables in the model are affected (over time) by an increase in the value of the shock. Hence, where x_t represents the variable and ϵ_t represents the shock, the impulse response function will show the value of $\frac{\partial x_t}{\partial \epsilon_t}$. The second point on the impulse response function will be $\frac{\partial x_{t+1}}{\partial \epsilon_t}$ and the third will be $\frac{\partial x_{t+2}}{\partial \epsilon_t}$. And so on.



Figure 7: **Impulse response function to a monetary policy shock**Figure 8: **Impulse response function to a technology shock**

It is important to note that in figure 7, the monetary policy shock has temporary impacts on the variables in the model (as the values of the variables will return to their steady state values after 8 periods). If this were not the case then the most likely cause is that the variables are not stationary (i.e. the effect of a disturbance is permanent).

When we look at the case of the technology shock in figure 9, we note that it takes a long time for the variables in the model to return to their steady state values. In this case you may wish

to change the calibrated value of the parameter that influences the persistence of technology shock, which is set at a relatively large number.

A more detailed description of the role of these shocks in the impulse response functions is provided in Alpanda, Kotzé, and Woglom (2010b), which considers the effects of several shocks in an open-economy model. This description could be used as example of what should be expected for most small open economies.

After completing this procedure, one could then try to estimate additional parameters, include additional observed variables, or increase the standard deviation of the shocks to allow for the data to explain as much as possible.

7 Measurement Error

You may appreciate that some of the data that we work with is subject to measurement error. Indeed, Van Walbeek (2006) has shown that revisions to South African data can be fairly large at times. We could include such measurement error in these models, where we include a very small calibrated shock to each of the observed variables. In `NK_est2.mod` an example of such a model is provided, where we include measurement errors for output and inflation. Note that one should not include measurement errors for interest rates (or exchange rates for that matter) as this variable is not usually subject to such error.⁹

The easiest way of including a measurement error is to specify a shock for the observed variable, such as;

```
shocks;
...
var y_gap = 0.1^2;      % measurement error to output gap
var pi = 0.1^2;        % measurement error to output gap
end;
```

Following the events that transpired with the Global Financial Crisis, including the subsequent Quantitative Easing, it has been noticed that it is becoming more difficult to estimate these models without any measurement errors.

8 Conclusion

In this practical we make use of a closed economy New Keynesian framework to estimate a number of parameters in the model using Bayesian techniques. The essential characteristics of these results are then discussed. With the aid of this note, students should be able to estimate their own dynamic stochastic general equilibrium models.

9 References

Adjemian, S., H. Bastani, M. Juillard, F. Karame, F. Mihoubi, G. Perendia, J. Pfeifer, M. Ratto, and S. Villemot. 2011. “Dynare: Reference Manual, Version 4.” CEPREMAP: Dynare Working Papers, 1.

Alpanda, Sami, Kevin Kotzé, and Geoffrey Woglom. 2010a. “Should Central Banks of Small

Open Economies Respond to Exchange Rate Fluctuations? The Case of South Africa.” *ERSA Working Paper* No. 174.

———. 2010b. “The Role of the Exchange Rate in a New Keynesian DSGE Model for the South African Economy.” *South African Journal of Economics* 78 (2): 170–91.

———. 2011. “Forecasting Performance of an Estimated DSGE Model for the South African Economy.” *South African Journal of Economics* 79 (1): 50–67.

DeJong, David N., and Chetan Dave. 2011. *Structural Macroeconometrics*. 2nd. Princeton: Princeton University Press.

Edge, Rochelle M., Michael T. Kiley, and Jean-Philippe Laforte. 2010. “A Comparison of Forecast Performance Between Federal Reserve Staff Forecasts, Simple Reduced-Form Models, and a Dsge Model.” *Journal of Applied Econometrics* 25 (4): 720–54.

Galí, Jordi. 2008. *Monetary Policy, Inflation, and the Business Cycle: An Introduction to the New Keynesian Framework*. Princeton: Princeton University Press.

———. 2015. *Monetary Policy, Inflation, and the Business Cycle: An Introduction to the New Keynesian Framework*. Second. Princeton: Princeton University Press.

Pfeifer, Johannes. 2013. “A Guide to Specifying Observation Equations for the Estimation of Dsge Models.” University of Mannheim.

Smets, Frank, and Rafael Wouters. 2007. “Shocks and Frictions in Us Business Cycles: A Bayesian DSGE Approach.” *American Economic Review* 97 (3): 586–606.

Uhlig, Harald. 1999. “A Toolkit for Analyzing Nonlinear Dynamic Stochastic Models Easily.” Edited by Ramon Marimon and Andrew Scott. Oxford University Press.

Van Walbeek, C. P. 2006. “Official Revisions to South African National Accounts Data: Magnitudes and Implications.” *South African Journal of Economics* 74(4): 745–65.

-
1. When we include data for a particular variable, we term it an observed variable.↩
 2. Stationary variables have a well defined steady state, as shocks to stationary variables are temporary.↩
 3. It is possible to make use of a model that incorporates features that describe movements of the BGP, as well as movements from the BGP. Doing so would usually involve incorporating a model-consistent description of how the BGP evolves, as in Smets and Wouters (2007). For a practical introduction to such procedures, see Pfeifer (2013).↩
 4. It is also worth noting that the superior fit and forecasting power of these models is usually only apparent when we consider a model with many shocks and observed variables, for which the derivations would take up too much time in class. The interest reader is referred to Smets and Wouters (2007), Edge,

Kiley, and Laforte (2010) and Alpanda, Kotzé, and Woglom (2011).↵

5. For example, you may well have omitted a critical relationship that is required to find the steady-state.↵
6. If you include more variables than shocks, then you are committing an error of stochastic singularity.↵
7. If any of these parameters equals one then the shock would be described by a random walk.↵
8. To ensure that you don't wait too long for the results, you could make use of a single Monte Carlo Markov Chain of a relatively short length for illustrative purpose (i.e. `mh_replic=2500` , `mh_nblocks=1` , `mh_drop=0.45`).↵
9. Further details are available in Pfeifer (2013) and the Dynare Manual.↵