Christiano

Tutorial on Forecasting, Output Gap Estimation, DSGE Model Estimation and the MCMC Algorithm Using Dynare

1. Clarida-Gali-Gertler Model

Following are the equations of the Clarida-Gali-Gertler model.

$$\begin{aligned}
\pi_t &= \beta E_t \pi_{t+1} + \kappa x_t \text{ (Calvo pricing equation)} \\
x_t &= -\left[r_t - E_t \pi_{t+1} - r_t^*\right] + E_t x_{t+1} \text{ (intertemporal equation)} \\
r_t &= \alpha r_{t-1} + (1-\alpha)\left[\phi_\pi \pi_t + \phi_x x_t\right] + u_t \text{ (policy rule)} \\
r_t^* &= \rho \Delta a_t + \frac{1}{1+\varphi}(1-\lambda)\tau_t \text{ (natural rate)} \\
y_t^* &= a_t - \frac{1}{1+\varphi}\tau_t \text{ (natural output)} \\
x_t &= y_t - y_t^* \text{ (output gap)}
\end{aligned}$$

The above equations represent the equilibrium conditions of an economy, linearized about its steady state. In the economy, household preferences are given by:

$$E_0 \sum_{t=0}^{\infty}\left(\log C_t - \exp(\tau_t)\frac{N_t^{1+\varphi}}{1+\varphi}\right),\ \tau_t = \lambda \tau_{t-1} + \varepsilon_t^\tau,\ \varepsilon_t^\tau \tilde{} iid,$$

where C_t denotes consumption, τ_t is a time t preference shock and N_t denotes employment. The budget constraint of the household is:

$$P_t C_t + B_{t+1} \leq W_t N_t + R_{t-1} B_t + T_t,$$

where T_t denotes (lump sum) taxes and profits, P_t is the price level, W_t denotes the nominal wage rate and B_{t+1} denotes bonds purchased at time t which deliver a non-state-contingent rate of return, R_t, in period $t+1$.

Competitive firms produce a homogeneous output good, Y_t, using the following technology:

$$Y_t = \left[\int_0^1 Y_{i,t}^{\frac{\varepsilon-1}{\varepsilon}} di\right]^{\frac{\varepsilon}{\varepsilon-1}},\ \varepsilon > 1,$$

where $Y_{i,t}$ denotes the i^{th} intermediate good, $i \in (0,1)$. The competitive firms takes the price of the final output good, P_t, and the prices of the intermediate goods, $P_{i,t}$, as given and chooses Y_t and Y_{it} to maximize profits. This results in the following first order condition:

$$Y_{i,t} = Y_t\left(\frac{P_t}{P_{i,t}}\right)^{\varepsilon}.$$

The producer of Y_{it} is a monopolist which takes the above equation as its demand curve. Note that if this demand curve is substituted back into the production function,

$$Y_t = \left[\int_0^1 Y_{i,t}^{\frac{\varepsilon-1}{\varepsilon}} di \right]^{\frac{\varepsilon}{\varepsilon-1}} = Y_t P_t^{\varepsilon} \left[\int_0^1 \left(P_{i,t}^{-\varepsilon} \right)^{\frac{\varepsilon-1}{\varepsilon}} di \right]^{\frac{\varepsilon}{\varepsilon-1}} = Y_t P_t^{\varepsilon} \left[\int_0^1 P_{i,t}^{(1-\varepsilon)} di \right]^{\frac{\varepsilon}{\varepsilon-1}},$$

or, after cancelling Y_t and rearranging,

$$\begin{aligned} P_t^{-\varepsilon} &= \left[\int_0^1 P_{i,t}^{(1-\varepsilon)} di \right]^{\frac{\varepsilon}{\varepsilon-1}} \\ P_t &= \left[\int_0^1 P_{i,t}^{(1-\varepsilon)} di \right]^{\frac{1}{1-\varepsilon}}. \end{aligned}$$

Thus, we get a simple expression relating the price of the aggregate good back to the individual prices.

The i^{th} intermediate good firm uses labor, $N_{i,t}$, to produce output using the following production function:

$$Y_{i,t} = \exp(a_t) N_{i,t}, \ \Delta a_t = \rho \Delta a_{t-1} + \varepsilon_t^a,$$

where Δ is the first difference operator and ε_t^a is an iid shock. We refer to the time series representation of a_t as a 'unit root' representation. The i^{th} firm sets prices subject to Calvo frictions. In particular,

$$P_{i,t} = \begin{cases} \tilde{P}_t & \text{with probability } 1-\theta \\ P_{i,t-1} & \text{with probability } \theta \end{cases},$$

where $\tilde{P}_t$ denotes the price chosen by the $1-\theta$ firms that can reoptimize their price at time t. The i^{th} producer is competitive in labor markets, where it pays $W_t(1-\nu)$ for one unit of labor. Here, ν represents a subsidy which has the effect of eliminating the monopoly distortion on labor in the steady state. That is, $1-\nu = (\varepsilon - 1)/\varepsilon$.

At this point it is interesting to observe that if the household and government satisfy their budget constraints and markets clear, then the resource constraint is satisfied (Walras' law). Optimization leads the households to satisfy their budget constraint as a strict equality:

$$\begin{aligned} P_t C_t + B_{t+1} &= W_t N_t + R_{t-1} B_t + T_t \\ &= W_t N_t + R_{t-1} B_t + \overbrace{\int_0^1 P_{i,t} Y_{i,t} - (1-\nu) W_t \int_0^1 N_{i,t} di}^{\text{profits}} - T_t^g, \end{aligned}$$

where T_t^g denotes lump sum taxes raised by the government (profits from the final good firms need not be considered, because they are zero). The government budget constraint is

$$\nu W_t N_t + B_{t+1}^g = T_t^g + R_{t-1} B_t^g,$$

where B_{t+1}^g denotes government purchases of bonds (i.e., 'lending', if positive and 'borrowing' if negative). Note that, clearing in the labor market implies

$$\int_0^1 N_{i,t} di = N_t.$$

By the fact that final good firms make zero profits,

$$\int_0^1 P_{i,t}Y_{i,t} = P_tY_t.$$

Substituting the government budget constraint and the expressions for profits (using labor market clearing) back into the budget constraint:

$$\begin{aligned}
P_tC_t + B_{t+1} &= W_tN_t + R_{t-1}B_t + T_t \\
&= W_tN_t + R_{t-1}B_t + \overbrace{P_tY_t - (1-\nu)\,W_tN_t - \overbrace{\left[-R_{t-1}B_t^g + \nu W_tN_t + B_{t+1}^g\right]}^{=T_t^g}}^{T_t=\text{profits, net of taxes}} \\
&= W_tN_t + R_{t-1}B_t + P_tY_t - (1-\nu)\,W_tN_t + R_{t-1}B_t^g - \nu W_tN_t - B_{t+1}^g \\
&= R_{t-1}B_t + P_tY_t + R_{t-1}B_t^g - B_{t+1}^g
\end{aligned}$$

or,

$$P_tC_t + \left(B_{t+1} + B_{t+1}^g\right) = R_{t-1}\left(B_t + B_t^g\right) + P_tY_t.$$

But, clearing in the bond market requires

$$B_{t+1} + B_{t+1}^g = 0 \text{ for all } t.$$

So,

$$C_t = Y_t,$$

and the resource constraint is satisfied. Incidentally, in this model with lump sum taxes, the equilibrium allocations are independent of the time pattern of government debt. So, for convenience, we just set $B_t^g = 0$ and so market clearing requires $B_t = 0$. Of course, we could have B_t not equal to zero, so that there is positive volume in the debt market. However, this would not be an interesting theory of why there is debt and so we don't do this.

The Ramsey equilibrium for the model is the equilibrium associated with the optimal monetary policy. It can be shown that the Ramsey equilibrium is characterized by zero inflation, $\pi_t = 0$, at each date and for each realization of a_t and τ_t and that consumption and employment in the Ramsey equilibrium corresponds to their first best levels.[1] That is, C_t and N_t satisfy the resource constraint

$$C_t = \exp(a_t)\,N_t,$$

and the condition that the marginal rate of substitution between consumption and labor equals the marginal product of labor

$$\frac{\text{marginal utility of leisure}}{\text{marginal utility of consumption}} = C_t \exp(\tau_t)\,N_t^{\varphi} = \exp(a_t)\,.$$

Solving for N_t :

$$\log(N_t^*) = -\frac{\tau_t}{1+\varphi},\ \ \log(C_t^*) = a_t - \frac{\tau_t}{1+\varphi},$$

[1]For a discussion, see http://faculty.wcas.northwestern.edu/~lchrist/course/optimalpolicyhandout.pdf

where $*$ indicates that the variable corresponds to the Ramsey equilibrium. In the description of the model above, y_t denotes log output and y_t^* denotes log output in the Ramsey equilibrium, i.e., $\log(C_t^*)$. The gross interest rate in the Ramsey equilibrium, R_t^*, satisfies the intertemporal household first order condition,

$$1 = \beta E_t \frac{u_{c,t+1}^*}{u_{c,t}^*} \frac{R_t^*}{1+\pi_{t+1}^*},$$

where $u_{c,t}^*$ indicates the marginal utility of consumption in the Ramsey equilibrium. Also, $\pi_t^* = 0$. With our utility function:

$$1 = \beta E_t \frac{C_t^*}{C_{t+1}^*} R_t^* = \beta E_t \frac{R_t^*}{\exp\left[\Delta a_{t+1} - \frac{\tau_{t+1}-\tau_t}{1+\varphi}\right]} = \beta E_t \exp\left[\log(R_t^*) - \Delta a_{t+1} + \frac{\tau_{t+1}-\tau_t}{1+\varphi}\right],$$

Approximately, one can 'push' the expectation operator into the power of the exponential. Doing so and taking the log of both sides, one obtains:

$$0 = \log\beta + \log(R_t^*) - E_t \Delta a_{t+1} + E_t \frac{\tau_{t+1}-\tau_t}{1+\varphi},$$

or,

$$r_t^* = E_t \Delta a_{t+1} - E_t \frac{\tau_{t+1}-\tau_t}{1+\varphi},$$

where $r_t^* \equiv \log(R_t^* \beta)$, the log deviation of R_t^* from its value in the non-stochastic steady state. The variable, r_t^*, corresponds to the 'natural rate of interest' and y_t^* corresponds to the 'natural rate of output'.

2. Computer Exercises

You will need the Dynare files, cggsim.mod and cggest.mod, as well as the MATLAB m files, plots.m, analyzegap.m, suptitle.m and HPFAST.m, to do this assignment (you can see answers in cggsimans.mod and cggestans.mod).

The HP filter is defined as follows:

$$\min_{\left\{y_t^T\right\}_{t=1}} \sum_{t=1}^{T} \left(y_t - y_t^T\right)^2 + \lambda \sum_{t=2}^{T-1} \left[\left(y_{t+1}^T - y_t^T\right) - \left(y_t^T - y_{t-1}^T\right)\right]^2$$

The parameter, λ, controls how 'smooth' y_t^T is. If $\lambda = 0$, then $y_t = y_t^T$. If $\lambda = \infty$, then y_t^T is a time trend (i.e., a line whose second derivative is zero). In business cycle analysis, it is customary to use $\lambda = 1600$ in studying quarterly. The MATLAB m-file, [y_hp,y_hptrend]=HPFAST(y,lambda) takes y as input and puts out y_hp=$y_t - y_t^T$, y_hptrend=y_t^T.

This assignment explores four things: (i) the estimation of the output gap using the HP filter and a model (ii) estimation, by Bayesian and maximum likelihood methods, of a model, and (iii) the MCMC algorithm as a device for approximating a posterior distribution (iv) basic economic properties of the model.

1. Before turning to the econometric part of the assignment, it is useful to study the economics of the CGG model, by seeing how the CGG economy responds to a shock. Consider the following parameterization:

$$\begin{aligned}
\beta &= 0.97,\ \phi_x = 0,\ \phi_\pi = 1.5,\ \alpha = 0,\ \rho = 0.2,\ \lambda = 0.5,\ \delta = 0.2, \\
\varphi &= 1,\ \theta = 0.75,\ \sigma_a = \sigma_\tau = 0.02,\ \sigma_u = 0.
\end{aligned}$$

   1. In the case of the technology and preference shocks, use Dynare to compute the impulse response functions of the variables to each shock. The m file, plots.m, can be used for this purpose.
      1. Consider the response of the economy to a technology shock and a preference shock. In each case, indicate whether the economy over- or under- responds to the shock, relative to their 'natural' responses. What is the economic intuition in each case?
      2. Replace the time series representation of a_t with

$$a_t = \rho a_{t-1} + \varepsilon_t^a.$$

         How does the response of the economy to ε_t^a with this representation compare to the response to ε_t^a with the unit root representation?
   2. Do the calculations with $\phi_\pi = 0.99$. What sort of message does Dynare generate, and can you provide the economic intuition for it? (In this case, there is 'indeterminacy', which means a type of multiplicity of equilibria...this happens whenever $\phi_\pi < 1$.) Provide intuition for this result.

3. Return to the parameterization, $\phi_\pi = 1.5$. Now, insert r_t into the Cavlo pricing equation. Redo the calculations and note how Dynare reports indeterminacy again. Provide economic intuition for your result.

4. Explain why it is that when the monetary policy rule is replaced by the $r_t = r_t^*$, the natural equilibrium (i.e., Ramsey) is a solution to the equilibrium conditions. Explain why the natural equilibrium is not the only solution to the equilibrium conditions (i.e., the indicated policy rule does not support the natural equilibrium uniquely). Verify this result computationally in Dynare.

5. Now replace the monetary policy rule with

$$r_t = r_t^* + \alpha \left(r_{t-1} - r_{t-1}^* \right) + (1 - \alpha) \left[\phi_\pi \pi_t + \phi_x x_t \right].$$

   Explain why the natural equilibrium is a solution to the equilibrium conditions with this policy. Verify computationally that this policy rule uniquely supports the natural equilibrium (in the sense of satisfying determinacy), as long as ϕ_π is large enough. Provide intuition. Conclude that the Taylor rule uniquely supports the natural equilibrium if the natural rate of interest is included in the rule.

6. Consider the following alternative representation for the technology shock:

$$a_t = \rho a_{t-1} + \xi_t^0 + \xi_{t-1}^1,$$

   where both shocks are iid, so that the sum is iid too. Here, we assume agents see ξ_t^0 at time t and they see ξ_{t-1}^1 at $t-1$. Thus, agents have advance information (or, 'news') about the future realization of a shock. Introduce this change into the code and set $\rho = 0.2$. Verify that when there is a shock to ξ_t^1, inflation falls contemporaneously and the output gap jumps. Provide intuition for this apparently contradictory result. What happens when the natural rate of interest is introduced in the policy rule?

2. We now explore the MCMC algorithm and the Laplace approximation in a simple example. Technical details about both these objects are discussed in lecture notes.[2] One practical consideration not mentioned in the notes is relevant for the case in which the pdf of interest is of a non-negative random variable. Since the jump distribution is Normal, a negative candidate, x, is possible (see the notes for a detailed discussion of x and the 'jump distribution'). As a result, we should assign a zero value to the density of a Weibull over negative random variables when implementing the MCMC algorithm.

   Hopefully, it is apparent that the MCMC algorithm is quite simple, and can be programmed by anyone with a relatively small exposure to MATLAB. A useful exercise to understand how the algorithm works, is to use it to see how well it approximates a simple known function. Thus, consider the Weibull probability distribution function (pdf),

$$ba^{-b}\theta^{b-1}e^{-\left(\frac{\theta}{a}\right)^b}, \ \theta \geq 0,$$

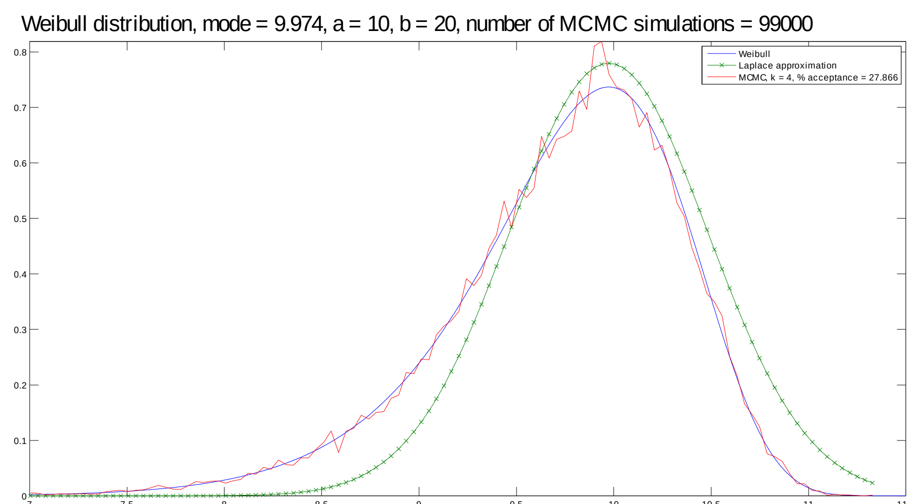
[2] See http://faculty.wcas.northwestern.edu/~lchrist/course/Gerzensee_2013/estimationhandout.pdf

where a, b are parameters. (For an explanation of this pdf, see the MATLAB documentation for $wblpdf(\theta, a, b)$.) Consider $a = 10$, $b = 20$. Graph this pdf over the grid, $[7, 11.5]$, with intervals 0.001 (i.e., graph g on the vertical axis, where $g = wblpdf(x, 10, 20)$, and x on the horizontal axis, where $x = 7 : .001 : 11.5$). Compute the mode of this pdf by finding the element in your grid with the highest value of g. Let f denote the log of the Weibull density function and compute the second derivative of f at the mode point numerically, using the formula,

$$f''(x) = \frac{f(x + 2\varepsilon) - 2f(x) + f(x - 2\varepsilon)}{4\varepsilon^2},$$

for ε small (for example, you could set $\varepsilon = 0.000001$.) Here, x denotes θ^* and f denotes the log of the output of the MATLAB function, $wblpdf$. Set $V = -f''(\theta^*)^{-1}$.[3]

Set $M = 1,000$ (a very small number!) and try $k = 2, 4, 6$. Which implies an acceptance rate closer to the recommended value of around 0.23? Choose the value of k that gets closest to that acceptance rate and note that the MCMC estimate of the distribution is quite volatile. Change M to 10,000. If you have time (now, the simulations takes time) try $M = 100,000$. Note how the MCMC estimate of the distribution is starting to smooth out. When I set $M = 100,000$ and $k = 4$, I obtained (see the MATLAB code MCMC.m, with the parameter iw set to unity) the following result:



[3]The strategy for computing the mode of the Weibull and f'' in the text are meant to resemble what is done in practice, when the form of the density function is unknown. In the case of the Weibull, these objects are straightforward to compute analytically. In particular,

$$f'(\theta) = \frac{b-1}{\theta} - b\left(\frac{\theta}{a}\right)^{b-1}\frac{1}{a},\ f''(\theta) = -\frac{b-1}{\theta^2} - (b-1)\,b\left(\frac{\theta}{a}\right)^{b-2}\frac{1}{a^2}.$$

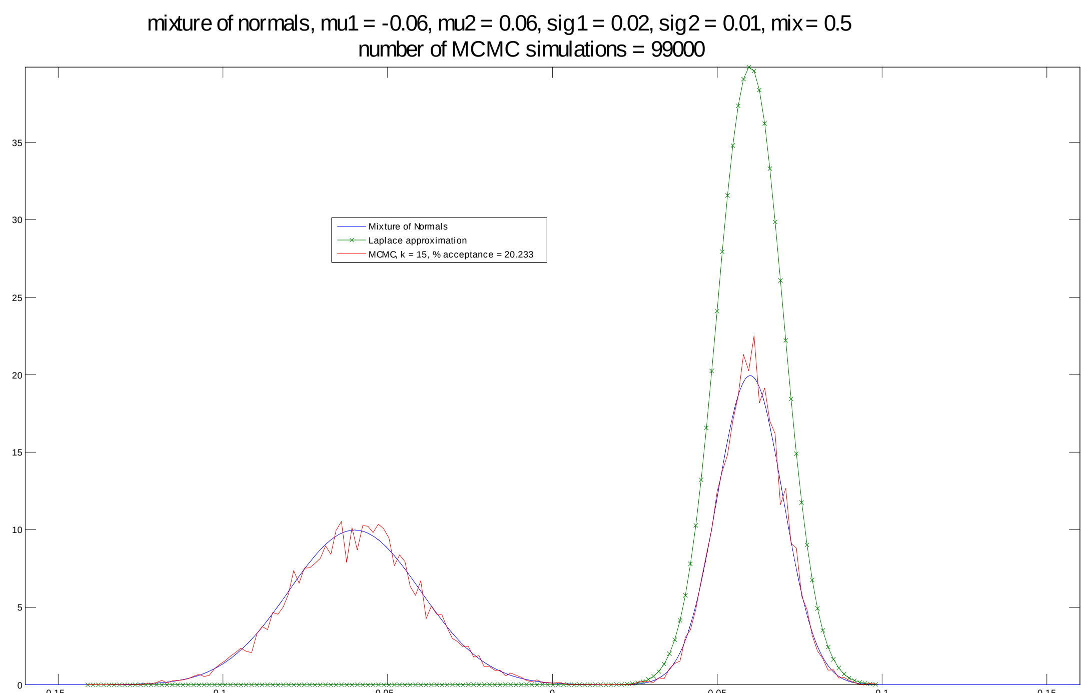
and the mode of f is $\theta^* = ((b-1)/b)^{1/b}\,a$.

Note how well the MCMC approximation works. The Laplace approximation assigns too much density near the mode, and lacks the skewness of the Weibull. Still, for practical purposes the Laplace may be workable, at least as a first approximation in the initial stages of a research project. This could be verified in the early stages of the project by doing a run using the MCMC algorithm and comparing the results with those of the Laplace approximation. In practice, posterior distributions may not be as skewed as the Weibull is.

We subject the MCMC algorithm to a much tougher test if we posit that the true distribution is bimodal, as in the case of a mixture of two Normals. Suppose the i^{th} Normal has mean and variance, μ_i and σ_i^2, respectively, $i = 1, 2$. Suppose also that the $i = 1$ Normal is selected with probability, π, and the $i = 2$ normal is selected with probability $1 - \pi$. In addition, suppose

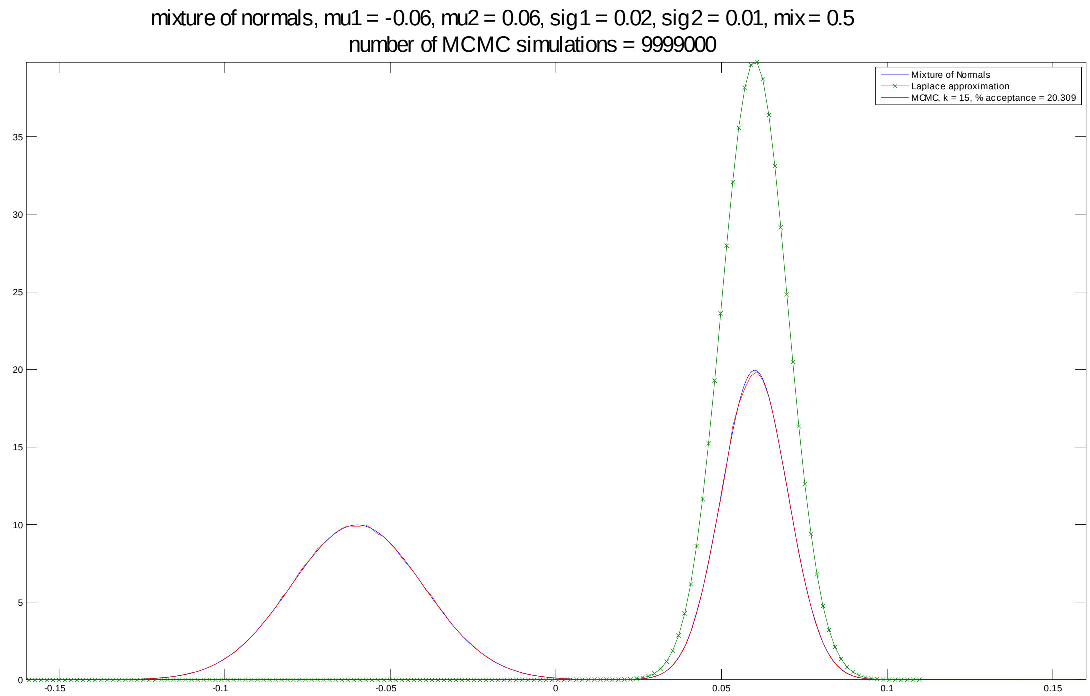
$$\mu_1 = -0.06,\ \mu_2 = 0.06,\ \sigma_1 = 0.02,\ \sigma_2 = 0.01,\ \pi = 1/2.$$

The mode of this distribution is the mode of the Normal with $i = 2$. If we apply exactly the same MCMC algorithm applied above, with $M = 100,000$ and $k = 15$, we obtain the following result:



These results (produced by running MCMC.m with iw set to zero) are comparable in accuracy to what was reported for the Weibull distribution. Taken together these sets of results suggest that the MCMC algorithm is quite good. It is not surprising that the Laplace approximation does poorly in this second example. It does a Normal approximation around the mode on the right. Because it 'thinks' that all the density is around that right mode and that density must integrate to unity, it follows that the

Laplace approximation must rise up much higher than the right mode. To verify that the MCMC distribution in fact is converging to the right answer, the MCMC was run a second time with $M = 10,000,000$. The results are displayed in the following figure. Note that it is almost impossible to distinguish between the actual and the MCMC-generated distributions, so that the MCMC algorithm has roughly converged to the right answer. It is hard to say whether this bimodal example is empirically realistic. These kind of posterior distributions have not been reported in the literature. Of course, this may simply be that the MCMC has failed to find them even though they do exist.[4]



3. From here on, consider the following alternative parameterization, which is more appealing than the one in question 1 from an empirical point of view:

$$\begin{aligned} \beta &= 0.97,\ \phi_x = 0.15,\ \phi_\pi = 1.5,\ \alpha = 0.8,\ \rho = 0.9,\ \lambda = 0.5,\ \delta = 0.2, \\ \varphi &= 1,\ \theta = 0.75,\ \sigma_a = \sigma_\tau = 0.02,\ \sigma_u = 0. \end{aligned}$$

Generate $T = 200$ artificial observations on the 'endogenous' (in the sense of Dynare) variables of the model. These are the variables in the 'var' list. The mod file provided, cggsim.mod, has 7 variables. Before doing the simulation, you should add the growth rate of output to the equations of the model and to the var list (call it 'dy'.) That way,

[4] An early paper by Thomas Sargent suggests that bimodality may be generic in dynamic macroeconomic models. He displays an example in which a parameterization in which persistence reflects the effects of endogenous mechanisms is hard to distinguish econometrically from a parameterization in which persistence reflects the persistence of shocks. See, Sagent, 1978, "Estimation of Dynamic Labor Demand Schedules under Rational Expectations," Journal of Political Economy, Vol. 86, No. 6, Dec., pp. 1009-1044.

Dynare will also simulate output growth. The variables simulated by Dynare are placed in the $n \times T$ matrix, oo_.endo_simul.[5] The n rows of oo_.endo_simul correspond to the $n = 8$ variables in var, *listed in the order in which you have listed them in the var statement* from the first to the last row. To verify the order that Dynare puts the variables in, see how they are ordered in M_.endo_names in the Dynare-created file, cggsim.m.

Retrieve output growth from oo_.endo_simul and get the log level of output, y, using y =cumsum($dysim$), where $dysim$ is the name I arbitrarily assigned to the row of oo_.endo_simul corresponding to output growth. Also, retrieve x from the appropriate row of oo_.endo_simul and create natural output from the relation, $y^* = y - x$.

1. Compute the HP filter of y with $\lambda = 1$ and display a graph with y and y^T. Do this also for $\lambda = 1600$ and for λ =160,000,000. Do the results accord with what you would expect, given the formula for the HP filter above?
2. Graph the HP filter trend, y^T, ($\lambda = 1600$) along with y and y^*. Note how actual output is somewhat more volatile than potential or natural output (recall, the economy overreacts to technology shocks). As a result, the HP filter with $\lambda =$ 1600 over smooths the data. Graph $y_t - y_t^T$ and the true output gap, x_t, as well as y, y^T and y^*. Compute the correlation between $y_t - y_t^T$ and x_t. Also compute the correlation for the case where technology shocks are dominant (i.e., $\sigma_a = 2$, $\sigma_\tau = 0.02$) and for the case where preference shocks are dominant (i.e., $\sigma_a = 0.02$, $\sigma_\tau = 2$). Interpret the results. The MATLAB command for computing the correlation between two variables, w_t and u_t, is corrcoef(w,u). The result of this calculation is a 2×2 matrix with unity on the diagonal and the correlation on the off-diagonal.

The model of this question lies close to the heart of the main paradigm underlying the current view about the monetary transmission mechanism. Note that in the case of this model, the hp-filter is not terrible as a guide to the output gap. This is because the technology shock is the important shock in the dynamics of the data, and the actual data overreact to the technology shock. That is, the natural rate of output is a smooth version of the data. Of course, this is only an example, and is something worth pursuing more carefully using a DSGE model that has more solid empirical foundations.

4. Now we will do some estimation. First, we generate artificial data from the baseline parameterization of the model. Place the simulated data, oo_.endo_simul, in the matrix, sim. Then, save these data to a MATLAB file, data, using the instruction, save data sim. Also, set periods = 5000 in the stoch_simul command. Run the mod file using Dynare. This saves the simulated data. Second, open cggest.mod.

   1. First, do maximum likelihood estimation. Use 4,000 observations to verify that everything is working properly. Consistency of maximum likelihood implies that with this many observations, the probability that the estimates are far from the

[5] Here, endo_simul is the matrix, which is a 'field' in the structure, oo_.

true parameter values is low. Try doing the estimation when you start far from the true parameter value, say with rho=lambda=0.9. Despite the bad initial guess about the parameter values, you should end up roughly at the true values.

2. Redo (a), but now with 30 observations, and you should see that maximum likelihood still works well. Note that although the point estimates look quite good, the standard error on lambda is rather large.

5. Now do Bayesian estimation, using the inverted gamma distribution as the prior on the two standard deviations and the beta distribution as the prior on the two autocorrelations.

   1. Set the mean of the priors over the parameters to the corresponding true values. Set the standard deviation of the inverted gamma to 10 and of the beta to 0.04. (It's hard to interpret these standard deviations directly, but you will see graphs of the priors, which are easier to interpret.) Use 30 observations in the estimation. Adjust the value of k, so that you get a reasonable acceptance rate. I found that $k = 1.2$ works well. Have a look at the posteriors, and notice how, with one exception, they are much tighter than the priors. The exception is lambda, where the posterior and prior are very similar. This is evidence that there is little information in the data about lambda.

   2. Redo (a), but set the mean and standard deviation of the prior on lambda equal to 0.95 and 0.04, respectively. Note how the prior and posterior are again very similar. There is not much information in the data about the value of lambda!

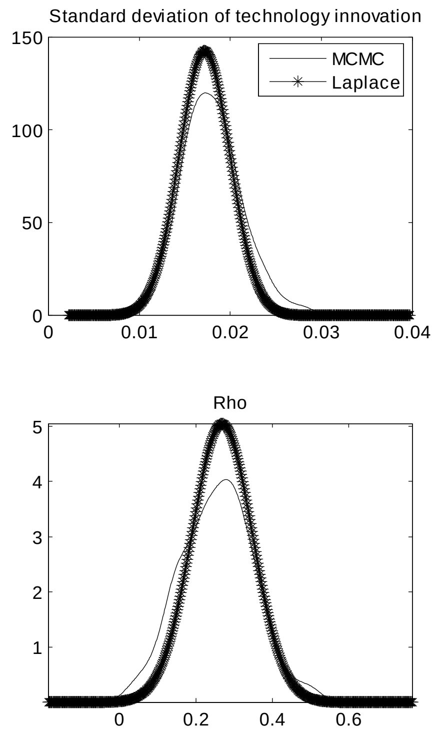
   3. Note how the priors on σ_a and ρ have faint 'shoulders' on the right side. Redo (a), with $M = 4,000$ (M is mh_replic, which controls the number of MCMC replications). Note that the posteriors are now smoother. Actually, $M = 4,000$ is a small number of replications to use in practice.

   4. Now set the mean of the priors on the standard deviations to 0.1, far from the truth. Set the prior standard deviation on the inverted gamma distributions to 1. Keep the observations at 30, and see how the posteriors compare with the priors. (Reset $M = 1,000$ so that the computations go quickly.) Note that the posteriors move sharply back into the neighborhood of 0.02. Evidently, there is a lot of information in the data about these parameters.

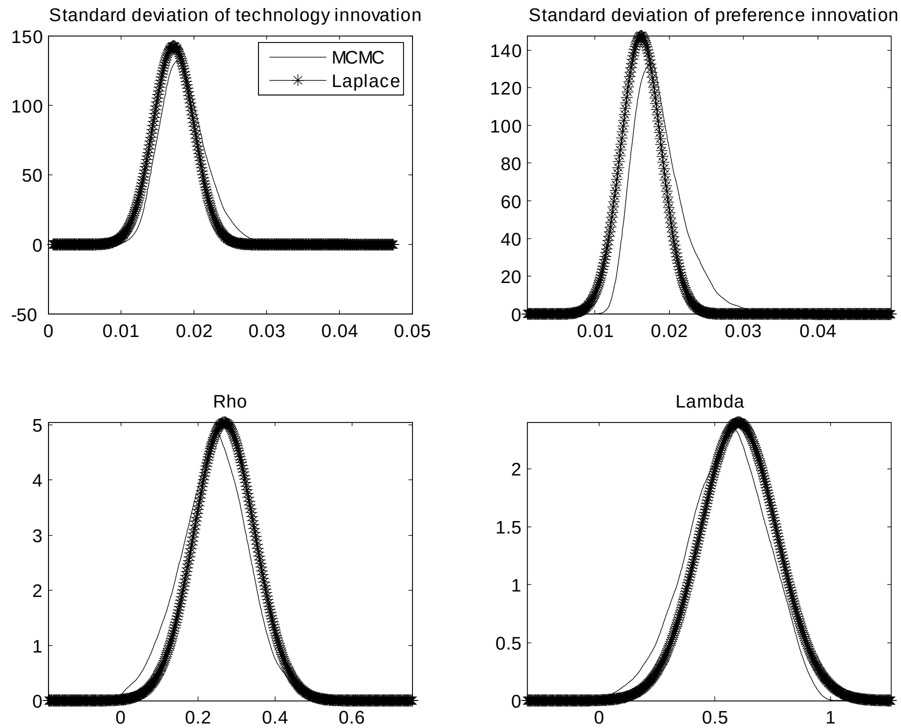
   5. Repeat (a) with 4,000 observations. Compare the priors and posteriors. Note how, with one exception, the posteriors are 'spikes'. The exception, of course, is lambda. Still, the difference between the prior and posterior in this case indicates there is information in the data about lambda.

6. It is of interest to compare the posterior densities approximated by the MCMC algorithm with the Laplace approximation. Consider the setup in 5 (a). You can recover all the information you need for these calculations from the structure, oo_. The posterior distributions of the parameters and shock standard errors are in the structure oo_.posterior_density. Posterior modes are in oo_.posterior_mode. Posterior standard deviations (taken from the relevant diagonal parts of the inverse of the hessian

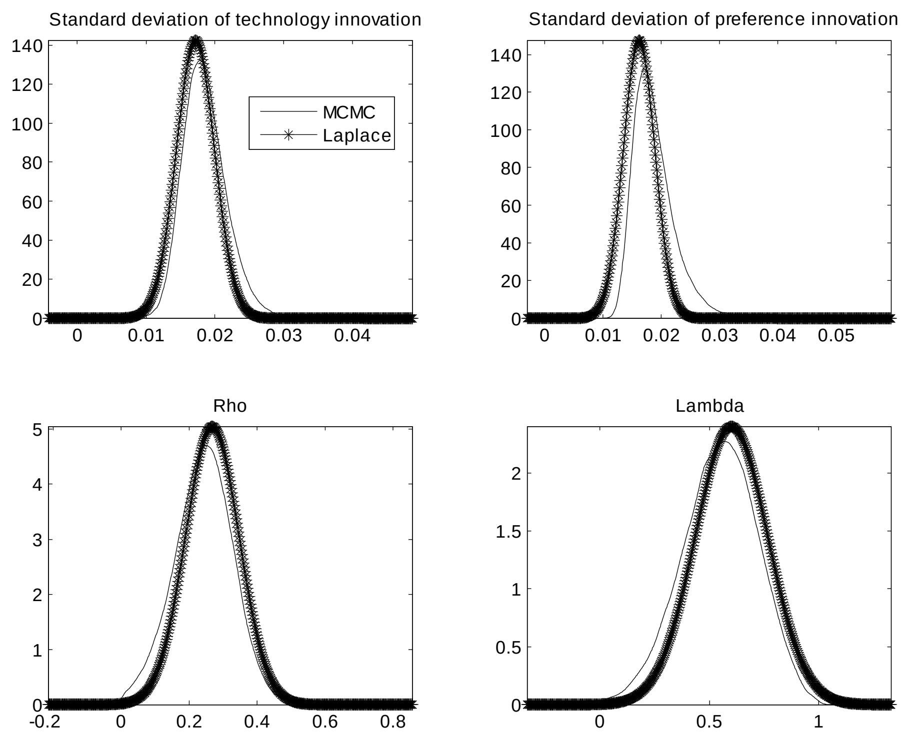
of the log criterion) appear in oo_.posterior_variance (my code for recovering these objects is compareMCMCLaplace.m. Setting $M = 10,000$, I found



When I set $M = 100,000$, the MCMC posteriors became smoother:



Note how much more similar the MCMC and Laplace posteriors are. The tail areas of the MCMC posteriors have thinned out and now resemble more closely the Laplace. Next, I set $M = 1,000,000$ and obtained virtually the same result as with $M = 100,000$:



Thus, in this example it seems that the MCMC algorithm has roughly converged for $M = 100,000$. In addition the Laplace and MCMC approximations deliver very similar results, consistent with the conclusion that the Laplace approach can used at the start and middle of a research project, while the MCMC can be done later on. Note that in any particular project, you can 'test' this proposition doing comparison of the posterior distribution obtained by the Laplace approximation with the posterior distribution obtained by MCMC.

7. The output gap is not in the dataset used in the econometric estimation. However, it is possible to use the Kalman filter to estimate the output gap (actually, all the variables in the var and varexo commands in Dynare) from the available data. There are two ways to do this: 'smoothing' uses all observations on the variables in the dataset (i.e., all the variables in the varobs command) and 'filtering' only uses the part of the dataset prior to the date for which the estimate of the gap is formed (thus, filtered data are one-step-ahead forecasts). To activate the Kalman smoother in Dynare, include the argument, smoother, in the estimation argument list. The smoothed estimates will then be placed in a MATLAB structure oo_.SmoothedVariables. This structure can be accessed either directly from the command line. Alternatively (at least, in MATLAB R2013a) it can be accessed from the 'Home' tab in MATLAB. In the 'variable' portion of that tab, press the drop down arrow next to 'Open Variable'. Then, you will see all the variables in the MATLAB memory. Select oo_ and you will see the contents of that structure. Some of the objects in that structure are simply numbers (they are indicated by cube with four boxes inside) and some of the objects in the structure are themselves structures. Select 'SmoothedVariables' and you will see a

number of subcategories with output related to the Bayesian estimation (for example, oo_.SmoothedVariables.Median.x displays the median smoothed estimate of the output gap, x). To see how well the Dynare-estimated version of the model does at producing a good guess of the output gap, include the code, analyzegap.m, at the end of your mod file. This shows you how to recover the smoothed output gap from Dynare, and allows you to compare it with the actual output gap, as well as with the hp-filtered estimate of the output gap.

8. Dynare also reports confidence intervals for the smoothed variables (e.g., oo_.SmoothedVariables.HPI contains the lower bound of the 95 percent confidence interval for x, in case you set conf_sig =0.95 in the Dynare estimation command). These reflect parameter uncertainty, as well as the difficulty of recovering these variables from the observed data when they are not in the data set. If you run analyzegap.m down to line 42, you will see what this confidence interval looks like, by comparison with the actual gap. Note that occasionally, the actual gap lies outside the confidence interval, as is to be expected.

9. The analysis in the previous question suggests that the output gap can be estimated reliably using the estimated DSGE model. However, in practice one needs the output gap in real time. For this, the smoothed estimates of the output gap are not a reliable indicator. Instead, it is useful to look at the filtered estimates. These are found by running analyzegap.m to line 57, and the filtered data are found in oo_.FilteredVariables. Note that there is a systematic phase shift between the estimated and actual gaps. This is as expected. Turning points are hard to 'see' in real time. They become evident only after the fact. (Dynare also reports 'updated' variables in oo_.UpdatedVariables. These are forecasts of the variables in the var command based on current and past observations on the variables in the varobs command. Not surprisingly, the updated 'estimates' of variables that to be in the econometrician's data set (i.e., appear in the varobs command) coincide with their true values. This is obviously not so for filtered variables.

10. It is interesting to see how the HP filter works in real time. By running analyze.m down to line 76 one obtains an estimate of this. Note that the HP filter does not exhibit the same phase shift as the filtered data. This is because for date t I have computed the HP filter using data up to and including date t. Also graphed are the updated variables, described above. These also do not display a phase shift relative to the data because these estimates of the date t gap include information at date t and earlier in the econometrician's dataset.

11. Dynare will also do forecasting. For this, one includes the argument, forecast=xx, where xx indicates how many periods in the future you want to forecast. (Put in xx=12.) To obtain the forecasts, as well as forecast uncertainty, execute the rest of analyzegap.m. You can see from the analyzegap.m code where in oo_.PointForecast the forecasts as well as the forecast uncertainty is stored.