Formulating and Estimating DSGE Models: A Practical Guide

Christian KEUSCHNIGG, Isabella MAASSEN,

Mara KRITZINGER, Julian JOHS*

This version: HandbookDSGE.tex, October 11, 2021

*Email: Christian.Keuschnigg@unisg.ch.

Contents

—

1 Introduction

Bayesian estimation of model parameters has become increasingly popular in economic research over the last decades. This increase in popularity could be seen for a variety of reasons. On the one hand, there are certain theoretical advantages in the estimation of economic models such as being system based, where all the dynamics can be fit to a vector of aggregate time series instead of focusing on one equilibrium equation of a time. On the other hand, developments in the available software packages made the implementation and interpretation of Bayesian estimates more accessible.

This handbook is supposed to serve as an entry point to the methodology and give step-by-step instructions for implementing a DSGE model for estimation with dynare, a free and open-source software, and MATLAB. It includes references to further helpful resources that can be consulted for a deeper look into certain aspects of this procedure.

We will start by an introduction to Bayesian Estimation of DSGE models to present the main definitions and theoretical foundations. It includes strategies for defining parameters to estimate, as well as gathering and preparing data series for estimation. Afterwards, we proceed with a section with a step-by-step guide on using the software dynare on MATLAB. There is also a recommended file structure that is described in detail.

2 Bayesian Estimation of DSGE Models

The first two parts of this section, on Bayesian Estimation, Bayes' Theorem (2.1) and and the Metropolis-Hastings Algorithm (2.2) were originally developed as part of a Master's thesis[1]. Bayesian Estimation is based on Bayesian updating of beliefs. One of the great advantages of it is, that prior information on the parameters of interest can be used in their estimation. Summarizing, it has the following characteristics as described in An and Schorfheide (2006) as well as Fernández-Villaverde (2010).

- **System-based**: instead of focusing on one equilibrium equation at a time, e.g. the Euler equation, it fits the solved DSGE model to a vector of aggregate time series.
- **Likelihood function:** the likelihood function will be based directly on the DSGE model examined.
- **Prior information:** with Bayesian estimation, prior knowledge on parameter distributions is taken into account.

We will proceed with a brief description of Bayes' Theorem and how it is used in estimating parameter values. The challenges of introducing real-world data are discussed thereafter. Finally, we will conclude this section with some comments on interpreting results and their robustness.

2.1 Bayes' Theorem

An detailed description of Bayesian Inference can be found in Herbst, Schorfheide (2015). Some main statements of their work will be summarized in this section and serve as a starting point for further examination of Bayesian estimation.
Given a vector of model parameters θ, and prior beliefs on their distributions $p(\theta)$,

[1] by Isabella Maassen

Bayesian estimation methods will determine updated beliefs of the parameter distributions given observed data O (*Note: in their work, they use Y to denote the observed data. In order to avoid confusion as Y commonly denotes output, this is changed to O here.*). Bayesian updating takes place via:

$$p(\theta|O) = \frac{p(O|\theta)p(\theta)}{p(O)} \tag{1}$$

The left-hand side (lhs) of this equation is the updated belief, or posterior distribution, on the parameter vector θ. On the right-hand side (rhs), we can see how this is composed. The prior belief on θ is multiplied by the likelihood function $p(O|\theta)$ and divided by the marginal likelihood $p(O)$ of O. In the following, we will take a closer look at each of these components and shine light on their interpretation.

Likelihood Function $\boldsymbol{p(O|\theta)}$: in an easy interpretation, this can be understood as the likelihood to observe vector O given the prior distribution of the parameter θ. In Bayesian Estimation of DSGE models, this is formed directly from the Model.

Marginal Likelihood $\boldsymbol{p(O) = \int p(O|\theta)p(\theta)d\theta}$ can be described as the 'overall' likelihood of observing O. It helps to scale the posterior to integrate to 1.

Example Using Bayesian updating of beliefs, we can employ a simple example from basic probability theory. Imagine there are two producers of light bulbs A and B, with different levels of product quality. While 90% of A's production functions perfectly, only 80% of B does. In a sample of light bulbs, where half of them is from each producer, we want to determine the probability that a certain one we pick stems from each factory. Our **prior** in this case are the probabilities of a randomly drawn light bulb to be from each factory. As half of the elements in the sample are supplies by A and the other half by B, these are 0.5 each:

$$p(A) = 0.5 \quad p(B) = 0.5$$

Moreover, we know that picking a functioning (F) light bulb from producer A is 0.9, from B it is 0.8, and so on:

$$p(F|A) = 0.9 \quad p(\bar{F}|A) = 0.1$$
$$p(F|B) = 0.8 \quad p(\bar{F}|B) = 0.2$$

Lastly, we know the overall probability of drawing a functioning light bulb from the sample:

$$p(F) = p(F|A)p(A) + p(F|B)p(B) = 0.9 * 0.5 + 0.8 * 0.5 = 0.85$$

Applying Bayes' Theorem to this problem will deliver the desired probability, or posterior belief, that the functioning light bulb came from either factory.

$$p(A|F) = \frac{p(F|A)p(A)}{p(F)} = \frac{0.9 * 0.5}{0.85} = \frac{9}{17} > 0.5 = p(A)$$

Hence, observing that a light bulb is working makes it more likely that it came from producer A. The prior $p(A) = 0.5$ has been updated with this additional observation F, and the posterior belief is $p(A|F) = 9/17$.

Applying Bayesian estimation to DSGE models requires us to define a prior $p(\theta)$ and a likelihood function $p(O|\theta)$. We also add a series of observations, for example GDP, and have an algorithm calculate a posterior distribution numerically by drawing randomly from the sample of observations. Some suggest using a slightly different notation: they use π for beliefs and p for probabilities, see for example Fernández-Villaverde (2010).

2.2 Implementation: Algorithm

The algorithm commonly employed in dynare to implement Bayesian estimation is based on Markov Chain Monte Carlo (MCMC) sampling. It originates from contributions of Metropolis et al. (1953) and a generalization developed by Hastings (1970). It is hence commonly designated as the "Metropolis-Hastings" (MH) algorithm. A description

MH Algorithm: Starting with a prior θ_0, the algorithm will randomly draw λ from a distribution of possible θ_t. It will then compare whether this draw if a better fit to the data than the previous parameter vector θ_{t-1}. This is repeated N times.
Starting from $t = 1$ up to N, with θ_0 as prior, repeat:

1. Draw λ from density function $q(\lambda|\theta_{t-1})$

2. Check whether this draw is more likely than the prior via:

$$\rho(\lambda|\theta_{t-1}) = \min\left\{1, \frac{p(O|\lambda)p(\lambda)/q(\lambda|\theta_{t-1})}{p(O|\theta_{t-1})p(\theta_{t-1})/q(\theta_{t-1}|\lambda)}\right\} \tag{2}$$

3. Set $\theta_t = \lambda$ with probability $\rho(\lambda|\theta_{t-1})$, otherwise $\theta_t = \theta_{t-1}$

Basically, the algorithm compares the relative probabilities of either candidate, θ_{t-1} and λ, of predicting the observed data O. If the one of the new draw is larger, it will become the "new" prior in the next repetition. If it is smaller, it only replaces the prior θ_{t-1} with a probability < 1.

2.3 Priors

The first step of Bayesian estimation should be to carefully select parameters of interest to be estimated. The number of parameters estimated is virtually not limited by the number of time series observed. However, it is a good start to take some of the more well-researched parameters as given in the literature, if there are also more obscure parameters that are generally difficult to classify. Which parameters are the most interesting in an application will depend on the respective model and the research question.
In any case, once the parameters to be estimated have been defined, one needs to specify prior distributions. They can be pictured as educated guesses at the true distribution of a parameter. This requires determining the type of distribution (e.g. normal, gamma,

etc...), its mean and standard deviation. By scanning the existing literature for parameter values and calibration of a steady state, this can be achieved. [2]

2.3.1 General considerations[3]

We now turn to the selection of prior distributions. The main problem in selecting priors is that the prior is either uninformative or the posterior outcome is hyper-sensitive to the selection of the prior. A prior can either be a subjective measure of an expert's belief leading to a clearly defined shape (subjective prior) or an agnostic gauge such as a uniform guess (objective or flat prior). In economics, most publications rely on subjective priors that are informed by convention or empirical estimates. Objective prior selection is very rare. This is because in the context of DSGE models prior distributions are designed to reflect empirical observations that are not included in the likelihood.

In order to estimate DSGE models in Dynare we need to select the shape of the prior as well as the prior mean and standard deviation for each of the parameters that we want to estimate. The first step in selecting the prior distribution for a given parameter is to consider a priori plausible ranges for this parameter. The second step after determining a priori plausible ranges for the model parameters is to represent the (subjective) a priori beliefs by probability distributions. In practice, the established procedure is to start by choosing a family of probability distribution for each of the model parameters. Usually, the distribution is chosen based on the domain of the given parameter, as not all probability density functions have the same support (see Section 2.3.2 below for details). The parametrization of each prior distribution is typically chosen such that plausible range for the given model parameter corresponds to a credible interval of 90% or 95%.

Prior means for parameters that affect the steady state, $\theta_{(ss)}$ are often chosen based on

[2]This chapter was contributed by Johs, Julian and Kritzinger, Mara (2021), who have formulated a similar chapter as part of their work for WPZ research

[3]This section is based on Del Negro and Schorfheide (2008), Herbst and Schorfheide (2016), and Fernández-Villaverde, Rubio-Ramírez and Schorfheide (2016).

(pre-)sample or long-run averages[4]. These averages correspond to the parameter values that we use for calibration, e.g. the parameters representing GDP shares are set equal to the average value over a certain time period. Prior means for parameters which control the endogenous propagation mechanism (but do not affect the steady state), $\theta_{(endo)}$, are usually based on existing empirical evidence[5]. Also these correspond to the values we use in calibration. Priors on exogenous variables, $\theta_{(exo)}$, are often difficult to specify due to the fact that exogenous processes are latent. However, it is possible to map beliefs about the persistence and volatility of observed time series into beliefs about the exogenous shock parameters. A good practice is to generate draws from the prior distribution of the model parameters and compute steady-state ratios as well as simulate artificial observations from the DSGE model and compute the implied sample moments to understand the implications of the specified prior. In case these appear to be implausible given the actual data one could respecify the prior distributions of the exogenous shock parameters in an informal iterative procedure up to the point where the implied prior for important parameter transformations (e.g. steady-state ratios) and the simulated sample moments have the desired features[6]. With regard to the AR parameters often Beta distributions centered around 0.8 or 0.9 are used.

So, (pre-)sample/long-run averages and empirical evidence determine the location (mean) of the prior distribution. The last thing we need to specify is the prior standard deviation. This reflects partly data uncertainty and partly estimation uncertainty[7].

[4]When opting for (pre-)sample averages instead of long-run means, it is common to use pre-sample averages for data included in the estimation sample and averages that are current to the estimation period for data not included in the estimation sample.

[5]It is not always necessary to have empirical evidence for the country of interest, but also evidence/data from other countries that are somehow similar can be used. For instance, some priors for the Euro Area could be specified based on US data.

[6]The formal procedure can be found in Del Negro and Schorfheide (2008).

[7]In many cases estimates are sensitive to the choice of time series, e.g. estimates of average real interest rates vary with the chosen interest rate time series.

2.3.2 Distributions

In economic research, the choice set of priors predominantly consists of five common distributions: Beta (β), Gamma and inverse Gamma (γ and γ^{-1}), normal ($\mathcal{N}$), and uniform ($\mathcal{U}$). We recall the main properties of these distributions:

- **Uniform:** A family of symmetric probability distributions with two parameters, a and b, denoting the endpoints of an interval. a and b can take any number on the real line. Since all values within the interval (a, b) are equally likely the uniform distribution has no mode and its probability density function, $f(x)$, is

$$f(x) = \frac{1}{b-a}$$

  with the first two moments

$$\mu = \mathrm{E}[X] = \frac{1}{2}(a+b) \qquad \text{and} \qquad \sigma^2 = \mathrm{Var}[X] = \frac{1}{12}(b-a)^2.$$

  As already mentioned, we choose a uniform distribution if we roughly know the support but do not want to impose any other assumptions on the shape of the distribution. This is related to classical maximum likelihood-based inference and its special cases such as OLS in the sense that no prior knowledge influences the estimation (apart from the bounds of the interval that the uniform distribution is defined on). Note, however, that Bayesian methods generally use the distribution mean.

- **Normal:** Continuous distribution on $\mathbb{R}$ with mean μ and variance σ^2 that has probability density function, $f(x)$

$$f(x) = \left(\sigma\sqrt{2\pi}\right)^{-1} \exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right].$$

  We typically resort to a normal distribution for variables with continuous support but no further information other than the mean and variance (so no information on kurtosis or skewness). For instance, Adolfson et al. (2007) deploy a normal

distribution as a prior for the growth of the Solow residual ("technological progress") where the mean is the average growth rate of the U.S. economy (2%) and the standard deviation comes from business cycle fluctuations.

Sometimes, we might want to truncate the normal distribution such that its support is no longer $\mathbb{R}$ but some closed or half-closed subset $\mathcal{S} \subseteq \mathbb{R}$. Considering the growth factor as an example, we might truncate the distribution just below 1 to allow slightly negative growth rates, and above 1.2 to allow growth rates of up to 20%.

- **Beta:** Two-parameter family of continuous probability distributions on support $[0, 1]$ with shape parameters $a > 0$ and $b > 0$. The probability density function, $f(x)$, of the Beta distribution is

$$f(x; a, b) = \frac{x^{a-1}(1-x)^{b-1}}{\beta(a, b)} \qquad \text{where} \qquad \beta(a, b) = \frac{\gamma(a)\gamma(b)}{\gamma(a+b)}$$

  Higher values of a move the density closer to one while higher values of b move it closer to zero (see Fig. 1). The first two moments are

$$\mu = \mathrm{E}[X] = \frac{a}{a+b} \qquad \text{and} \qquad \sigma^2 = \mathrm{Var}[X] = \frac{ab}{(a+b)^2\,(1+a+b)}.$$

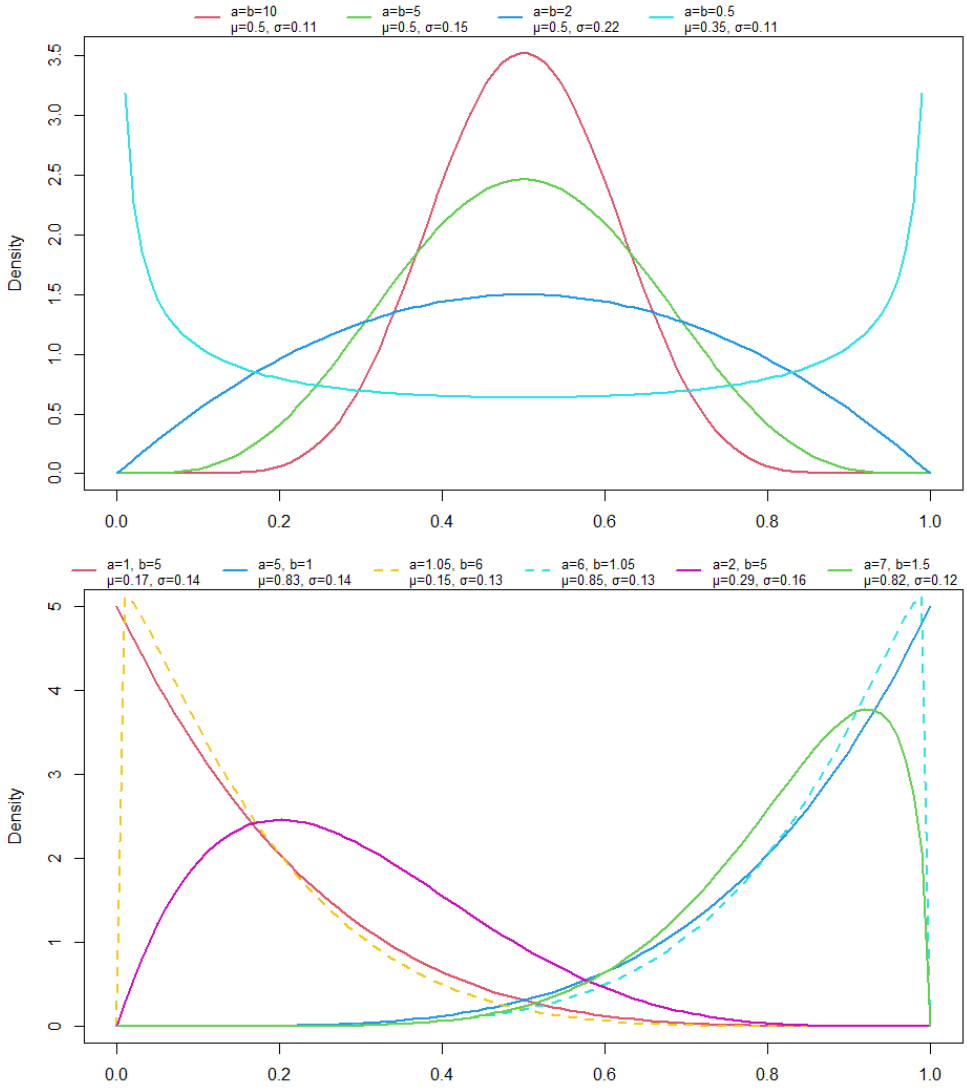
  Note that the Beta(1, 1) distribution is the same as the Uniform(0, 1) distribution. Because the Beta distribution is offering a very flexible family of distributions for random variables taking values between 0 and 1 it can often be used to model percentage or fractional quantities (or any other parameter that lies in this interval, e.g. Cobb Douglas elasticities). In changing parameters a and b, we can both shift the distribution towards low or high values and define whether it is highly concentrated or spread out (see Fig. 1).

- **Gamma:** Two-parameter family of continuous probability distributions on support $\mathbb{R}_{++}$ (excluding zero) with shape parameter[8] k and scale parameter[9] θ. Alternatively

[8]Note that the shape parameter is not a location parameter. It affects the shape of the distribution rather than simply shifting it (as the location parameter does) or stretching/shrinking it (as the scale parameter does).

[9]The larger the scale parameter the more spread out the distribution.

Figure 1: Beta distribution with different parameterizations



one can use the shape parameter α ($\alpha = k$) and the rate parameter β (inverse scale parameter, $\beta = 1/\theta$). These parameters must satisfy $k > 0,\ \theta > 0,\ \alpha > 0,\ \beta > 0$. The distribution's probability density function, $f(x)$, is

$$f(x; k, \theta) = \frac{x^{k-1}}{\Gamma(k)\,\theta^k} \exp\left(-\frac{x}{\theta}\right).$$

The first two moments are

$$\mu = \mathrm{E}[X] = k\theta \quad \text{and} \quad \sigma^2 = \mathrm{Var}[X] = k\theta^2.$$

Gamma is bounded below by zero but is unbounded above. The distribution is useful

in estimating parameters that are positive but are not necessarily restricted to an interval, for instance, output elasticities. By adjusting shape and scale parameters, we can move the mean and mode away from zero. We may choose k and θ to generate desirable moments, i.e., mean and variance (see Fig. 2).

A special case of the Gamma distribution is the exponential distribution, i.e. the family of Gamma distributions with shape parameter $k = 1$. As a result, the standard deviation of the exponential distribution is equal to its mean, and the density gets the classical, continuously decreasing form (see Fig. 2). The exponential distribution is especially useful since it provides any desired strength in parameter shrinkage towards zero, without artificially inflating parameters in some cases such as the Gamma distribution with shape parameters above one, or the inverse-Gamma family.

- **Inverse Gamma:** Two-parameter family of probability distributions with shape parameter α and scale parameter β ($\alpha > 0,\ \beta > 0$) on the positive real line (excluding zero). It corresponds to a Gamma distribution of $1/x$, i.e., the inverse of a variable is Gamma-distributed. All properties carry over. The inverse Gamma distribution's probability density function, $f(x)$, is

$$f(x;\alpha,\beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{-\alpha-1} \exp\left(-\frac{\beta}{x}\right).$$

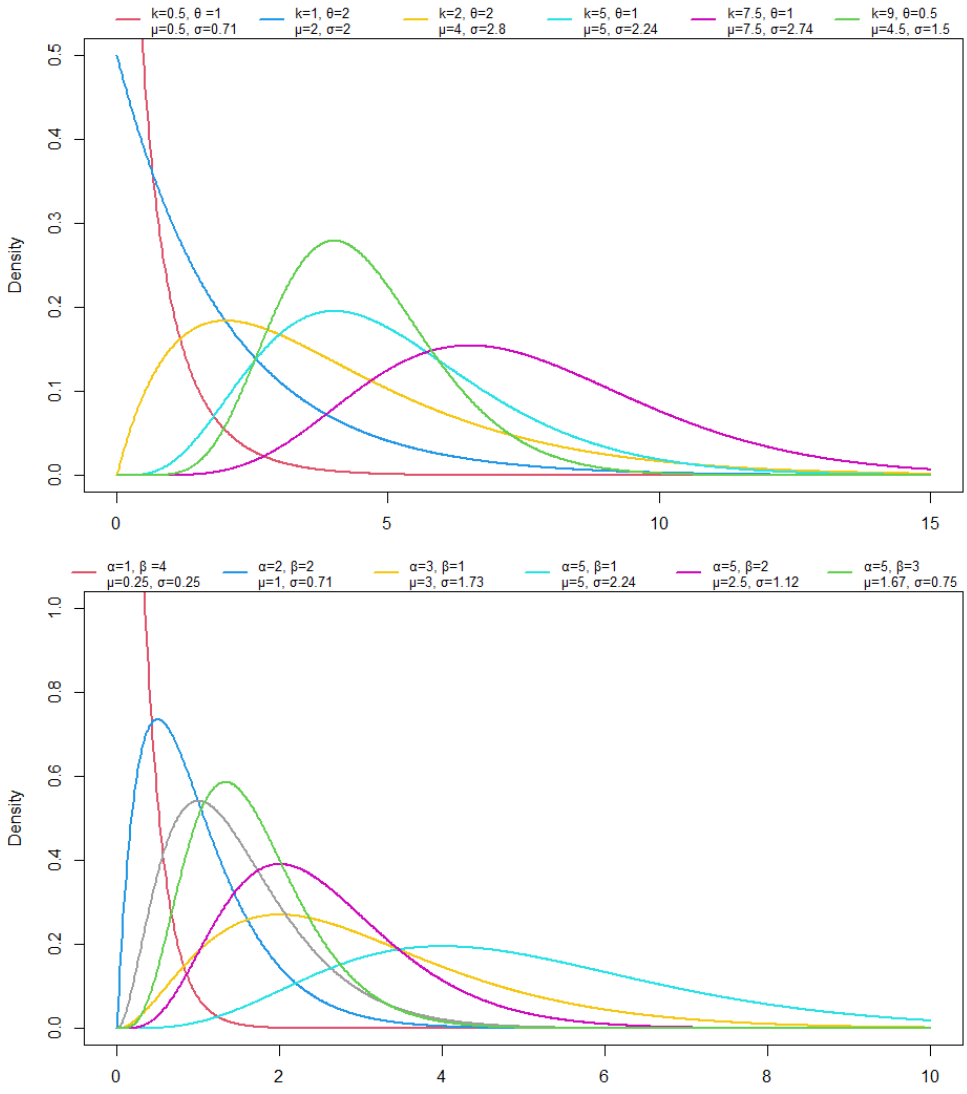
The first two moments are

$$\mu = \mathrm{E}[X] = \frac{\beta}{\alpha - 1} \quad \text{and} \quad \sigma^2 = \mathrm{Var}[X] = \frac{\beta^2}{(\alpha-1)^2(\alpha-2)}.$$

By adjusting shape and scale parameters, we can move the mean and mode away or toward zero. We may again choose α and β to generate desirable moments, i.e., mean and variance (see Fig. 3). Inverse Gamma priors are often used on the size of standard errors since the inverse Gamma distribution is the conjugate prior[10] for the variance parameter of the normal distribution.

[10]Conjugate priors are a kind of natural priors for some kinds of distributions in the sense that the

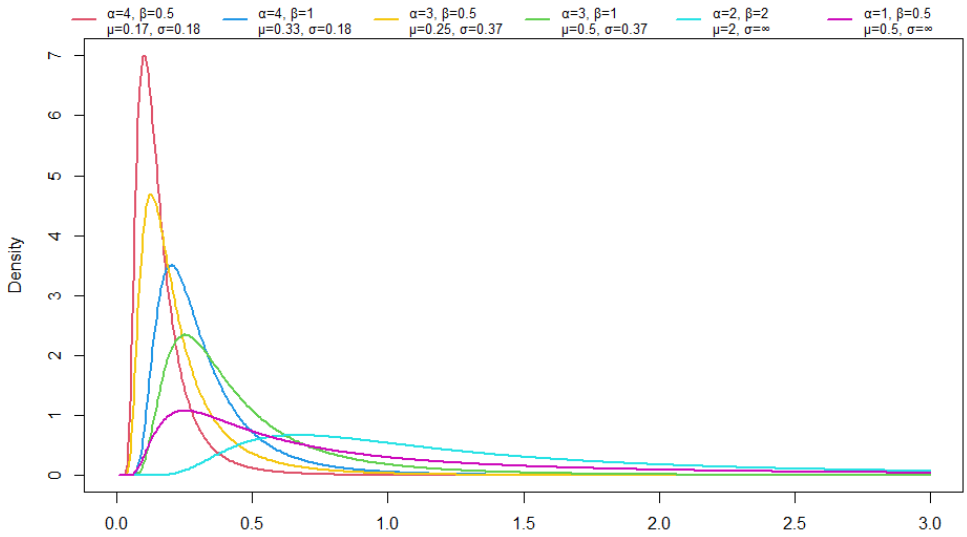
Figure 2: Gamma distribution with different parametrization



Note that it is also feasible to have truncated distributions. Truncation is possible from above and/or below. This is used in cases where a variable must lie within a certain range (within the distribution's support) to have determinancy or when it is necessary to restrict the range in order to obtain the desired results (e.g. in order to have (strictly) convex decreasing returns the corresponding elasticity must be larger than 1).

posterior distribution belongs to the same family of distributions as the prior. For more information, see the Wikipedia page (`https://en.wikipedia.org/wiki/Conjugate_prior`).

Figure 3: Inverse Gamma distribution with different parametrization



Remark 1 *Multivariate versions of the above-stated prior shapes exist. The Dirichlet distribution is a generalization of the Beta distribution; the Weibull and Inverse-Weibull distributions are generalizations of the Gamma and inverse Gamma distributions, respectively; the multivariate normal and uniform distributions also exist. These are, however, not directly implemented in Dynare. Theoretically, one could construct a covariance matrix of errors by estimating all correlations and variances.*

2.4 Data Series in Bayesian Estimation

In estimating DSGE Models with Bayesian methods, one is generally required to introduce observed time series data to the model. A number of dimensions and potential challenges have to be considered in doing so. These are characterized and examined in the following.

1. **Relevance:** it is good practice to select observed data series that are closely related to the parameters of interest as they can be more successful in delivering reasonable posterior distributions.

2. **Frequency:** first, we need to consider the frequency at which data is available. Whenever economic cycles are concerned, it seems reasonable to observe variance

Table 1: Prior distributions: summary

Density	**Support**	**Description**
Uniform	$(-\infty, \infty)$	two-parameter family of symmetric distributions with constant density within the specified interval
Normal	$(-\infty, \infty)$	symmetric distribution with mean μ and standard deviation σ for real valued random variables
Beta	[0,1]	very flexible two-parameter family of distributions for random variables with values between 0 and 1
Gamma	$(0, \infty)$	two-parameter family of distributions of positive random variables with many possible shapes
Inverse Gamma	$(0, \infty)$	two-parameter family of distributions of positive random variables with many possible shapes

at the quarterly level. In any case, the priors as well as actual data series must be defined on the same basis. For example, when defining a prior value of the interest rate, the mean of the annual one needs to be divided by 4 to achieve a specification relevant to a more frequent development.

3. **Seasonality:** most DSGE models do not contain terms to produce seasonal fluctuations. Therefore, data series such as the GDP should be seasonally adjusted as to take this into account.

4. **Number of Data Series:** contrary to the parameters estimated, the number of data series is limited. There can only be as many observed series as there are shock (perturbation) processes. This is because of collinearity concerns. For example, if consumption and output are both introduced into the model, they will likely be closely related in the model. In reality however, these often do not necessarily move together. Therefore, if they are both introduced, their differences should be described by shock processes. However, there are simple ways to introduce such additional shocks, which will be described below in the section.

5. **Stationarity and Trends:** many DSGE models do not feature long-run growth in output as they focus on business cycle movements. However, it is a widely accepted stylized fact that most economies do exhibit such an increase in the per-capita GDP level over time. Therefore, it is an important step in Bayesian estimation to determine whether a model features growth or mere fluctuations around such a trend. This potential stationarity has to be examined. Mostly, this is done by checking whether the expected value of variables that are exposed to shocks is constant, refer to section 2.4.1. If this is the case, the observed time series need to be de-trended. A common procedure for decomposing time series data is the HP-filter, further discussed in section 2.4.2.

6. **Population Growth:** if the DSGE model does not feature population growth, some of the time series need to be adjusted by the size of the population at the time by introducing them at a per-capita level.

7. **Coherence Observations and Model Specification:** the time series need to be introduced to the model in a way that is coherent with the model specification. In some instances, this can mean that only the change in a variable is used as an observation. Other times, it needs to be transformed to fit with the definitions of variables in the theoretical foundations. This is mostly done via so-called observation equations that "translate" observed data into the requirements of the model. This aspect is addresses further in section 2.4.4.

2.4.1 Determining Stationarity

In order to determine the stationarity of a DSGE model, its shock processes need to be examined. This section contains a simple example that will demonstrate this step in the case of stationarity. We will also discuss under which circumstances the model exhibits long-run (positive or negative) growth trends. As most changes in output in DSGE models is driven by a change in productivity, one has to determine the expected value of this shock

process. This will allow for making statements on long-run stationarity.

Example: the productivity level is determined by $a_t = (1-\rho)\,\bar{a} + \rho a_{t-1} + \varepsilon_t^a$ with $\varepsilon_t^a \sim N(0, \sigma_a^2)$ and $\bar{a}$ is the "starting" level of productivity. The expected value of productivity is given by:

$$E[a_t] = E\left[(1-\rho)\,\bar{a} + \rho a_{t-1} + \varepsilon_t^a\right] \tag{3}$$

$$= (1-\rho)\bar{a} + \rho E\left[a_{t-1}\right] + E\left[\epsilon_t^a\right] \tag{4}$$

$$= \sum_{i=0}^{t-1}(1-\rho)\rho^i\bar{a} + \rho^t E[\bar{a}] + \sum_{i=0}^{t-1}\rho^i E[\epsilon_t^a] \tag{5}$$

$$= (1-\rho)\bar{a}\sum_{i=0}^{t-1}\rho^i + \rho^t\bar{a} + E[\epsilon_t^a]\sum_{i=0}^{t-1}\rho^i \tag{6}$$

$$= (1-\rho)\bar{a}\frac{(1-\rho^t)}{(1-\rho)} + \rho^t\bar{a} + E[\epsilon_t^a]\frac{(1-\rho^t)}{(1-\rho)} \tag{7}$$

$$= \bar{a} + E[\epsilon_t^a]\frac{(1-\rho^t)}{(1-\rho)} = \bar{a} \tag{8}$$

Therefore, the expected productivity level of this economy is equal to a predetermined base level $\bar{a}$. This implies that the model does not feature a growth trend. Note that there are two simple changes we could introduce to introduce growth. If the mean of the productivity shock is different from zero, $\varepsilon \sim N(\mu, \sigma_a^2)$, where $\mu \neq 0$, expected utility is $E[a_t] = \bar{a} + \mu\frac{(1-\rho^t)}{(1-\rho)}$. This increases or decreases over time, depending on whether μ is positive or negative.

Note that even when a model is not stationarity and offers a growth trend, this does not necessarily translate well to real-world data and you might still have to detrend your empirical observations.

2.4.2 Detrending

One important step in preparing data for introduction into a Bayesian Estimation with many DSGE models is decomposing it into a trend and a cyclical component. This is usually achieved with a Hodrick-Prescott (HP) filter, developed by Hodrick and Prescott (1997). It has long be the standard approach to detrending time series data. We will

briefly describe its assumptions and estimation approach. Please note that there is a number of other popular detrending methods, which we will address briefly and discuss their pros and cons after a short formal description of the HP-filter.

The Hodrick-Prescott filter: The HP filter decomposes time series observations y_t into a trend τ_t and a cyclical component c_t according to the following objective function, where capital T stands for the full set of data:

$$\{\hat{\tau}_1, ..., \hat{\tau_T}\} = \arg\min_{\tau_1,...,\tau_T} \left(\sum_{s=1}^{T} (y_s - \tau_s)^2 + \lambda \sum_{s=2}^{T-1} [(\tau_{s+1} - 2\tau_t + \tau_{s-1})]^2 \right) \tag{9}$$

As λ affects how much weight the change in the growth rate of the trend component has in the minimization problem. Therefore, the higher this parameter is, the smaller will be the volatility of the trend identified. It is therefore known as the smoothing parameter and should generally be larger for more granular time series data. The filter can be implemented in MATLAB with the hpfilter which is contained in the MATLAB econometrics package. A reasonable value for the smoothing parameter depends on the frequency of observations, for quarterly data it is usually set to $\lambda = 1600$.

However, one can easily see from the objective function in (9) that future values are taken into account in determining the cyclical and trend components. However, most DSGE models do not feature this, as decisions are taken based on the current state of the economy. Thus, in order to have a coherent use of methods, there is a different approach, called the one-sided HP-filter. This filter only takes past and current data points into consideration. The objective function here is

$$\hat{\tau}_t = \arg\min_{\tau_t} \left(\min_{\tau_1,...,\tau_{t-1}} \left(\sum_{s=1}^{t} (y_s - \tau_s)^2 + \lambda \sum_{s=2}^{t-1} [(\tau_{s+1} - 2\tau_s + \tau_{s-1})]^2 \right) \right) \tag{10}$$

A recent research paper by Wolf et al. (2020) compared the two methods and concluded the optimal smoothing parameter λ has to be adjusted when used in the one-sided setting. Then, the extracted cyclical component is scaled by a factor of κ. Their findings are summarized in table 2.

Data type and frequency	Two-sided HP filter	One-sided HP filter	
	λ	λ^*	κ
Yearly, business cycle data	6.25	2.45	1.7962
Quarterly, business cycle data	1,600	650	1.1513
Quarterly, financial cycles	400,000	163,101	1.0360

Table 2: Smoothing Parameter Adjustment for One-Sided HP Filters

Criticism of the HP filter and alternatives: In a paper by Hamilton (2016), he lays out the reasons why he believes the HP filter should be approached more sceptically. He argues that it artificially creates predictive power from past and future values, that imply non-existent correlations, especially when data series are best approximated by a random-walk process. In that case, any observed correlations are rather a description of the filter than the underlying data. He suggests a different approach, that we discuss briefly in section 2.4.3. Additional alternatives to the one-sided HP filter commonly applied are (from Pfeifer, (2020)):

1. Linear-(quadratic) trend: the trend component is approximated with either a linear or quadratic regression, similar to: $Y_t = \beta_0 + \beta_1 t + \epsilon_t$ (linear) and $Y_t = \beta_0 + \beta_1 t + \beta_2 t^2 + \epsilon_t$ (quadratic). Afterwards, the difference between the fitted value and the actual observation denotes the cyclical component.

2. First-difference filter: taking first differences, we can check that the differences in the time series are not auto-correlated. If they are completely random, then the data is best described by a random walk.

2.4.3 Hamilton Filter[11]

Hamilton (2016) proposes a simple autoregressive approach, predicting the trend value of the data using leads and lags. Formally, a Hamilton filter takes the form:

$$y_t^{trend} = \beta_0 + \beta_1 y_{t-h} + \beta_2 y_{t-h-1} + \ldots + \beta_p y_{t-h-(p-1)} + y_t^{cyclic} \tag{11}$$

Where h is the number of leads, p is the number of lags, and y_t^{cyclic} are the residuals of the regression. For quarterly data, Hamilton strongly recommends $h = 8$ and $p = 4$. For example, this means that we predict trend output in 2020 Q1 using the observed output values in 2018 Q1 and 2017 Q4, Q3 and Q2.

Hamilton argues that this method offers a number of advantages over the HP-filter, we will here note only two. Firstly, this filter is one-sided in the sense that the prediction for 2020 Q1 depends only on prior data. This makes it especially useful for making decisions in the present; the HP-filter can be unreliable because the present-day predictions are less robust than earlier ones. This property can be very important for policymakers such as central banks, but is less important for our purposes.

Secondly, the Hamilton filter removes the seasonal trend along with the cyclical trend. This can be very important when working with data that has not been seasonally adjusted, since failing to account for seasonal variation will introduce artificial volatility into our estimates of y_t^{cyclic}. Of course, there are ways to adjust the HP-filter to account for seasonal variation instead.

From our perspective, the primary drawback is that we lose h periods of data at the start of our sample. If the series is very large, this will not be an issue. Even with 20 years of data however, we would be throwing at as much 10% of our sample using the Hamilton filter compared with the HP-filter. This will necessarily lead to less precision in our estimated parameters.

[11] This section contributed by Jacob Stevens

2.4.4 Observation Equations[12]

Often, the levels of data series are not necessarily in line with the specification of business cycle DSGE models. And sometimes, there is no exact counterpart in the theoretical set-up. We thus need to define so-called observation equations to incorporate data series into a Bayesian estimation. It is common practice to denote the raw empirical observations as Y^{data}, whereas lowercase y^{obs} is the detrended, cyclical component after all the preparation steps. It represents the data in the form it enters the estimation. An in-depth discussion can be found in Pfeifer (2020), who offers a great overview on the topic of observation equations and data series in Bayesian estimation in general. In the section on implementation in dynare, we will address this topic again and include a step-by-step description of how to add observation equations to existing dynare code.

Example: Output

We observe output as the GDP in the national accounts. After decomposing the logarithmic trend and cycle components, the cycle component is interpreted as the percentage change from the previous period's level of output. In a DSGE model, however, there is usually no such variable. We therefore need to 'translate' the observed values y^{data} into terms that connect to the theory. The observation equation could be formed as follows, where Y_t is output as defined in the model:

$$y^{obs} = \log Y_t - \log Y_{t-1} \tag{12}$$

Moreover, we need to consider whether there might be measurement inaccuracies in the data series observed. For GDP, this can reasonable be assumed and so we can introduce an error term to allow for noise:

$$y^{obs} = \log Y_t - \log Y_{t-1} + \varepsilon_t^y \tag{13}$$

This has another interesting feature, as it allows for the introduction of an additional data series. Dynare requires that we have at least one shock term per observation equation to

[12] Jacob Stevens also contributed to this section

prevent "stochastic singularity", and measurement errors are a convenient way of introducing a new shock with each series of observed values we introduced (Pfeifer 2020). The distribution (standard deviation) of ε_t^y (along with other measurement shocks) should be added to the list of estimated parameters.

More generally, we can introduce series for other variables such as net exports and investment. Care should be taken to make sure the data is comparable to the model output; it is often convenient to transform the data to a percentage of GDP, and then adjust the relevant observation equation in Dynare accordingly. All series must be the same length, and available together in a single .mat or .csv file, see Pfeifer (2020).

2.5 Interpreting Results and Robustness

In this section, we will discuss how to interpret the outputs of dynare after running a Bayesian estimation. First, it is always a good idea to look at the impulse response funtions (IRFs) as a sanity check to make sure the model is implemented correctly. The graphs supplied by the program will look similar to the example in figure 4. They show how each variable will react to a certain shock and how they develop afterwards. The red line is the development of the variable without a shock. If these seem to match the specification of the model, that is generally a good sign.
A second type of output will be graphs of the prior distributions. They should look like the distributions planned as priors. An example of this graph can be seen in figure 5.

While these first two outputs are important and one should always take a close look at them, they should not offer any surprises if the model is implemented correctly. Now, we will now turn to the probably more interesting part of the estimation output: the posterior distribution. Figure 6 shows the prior (light grey) and posterior distributions (black) together in one graph for each estimated parameter. In the example used here, we can see that the distributions for rho and theta have a more narrow spread. The posterior distribution of tau on the other hand does not seem to differ greatly from the

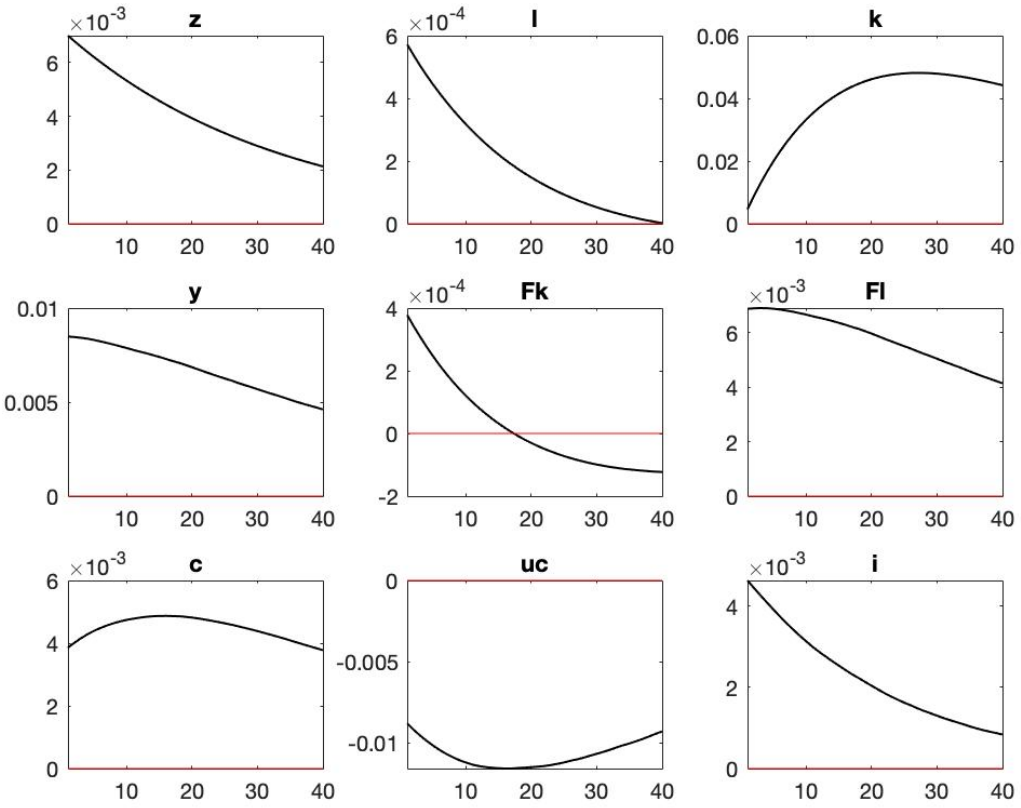


Figure 4: Impulse Response Functions

prior distribution. As a result of the estimation, we also get a table of prior and posterior distribution values as in table 3.

Parameter	prior mean	post. mean	conf.	interval	prior	prior std
rho	0.950	0.9694	0.9630	0.9753	beta	0.0200
theta	0.790	0.7978	0.7844	0.8103	norm	0.0500
tau	2.100	2.0559	1.6169	2.5012	norm	0.3000

Table 3: Estimation Results - Prior and Posterior Distributions

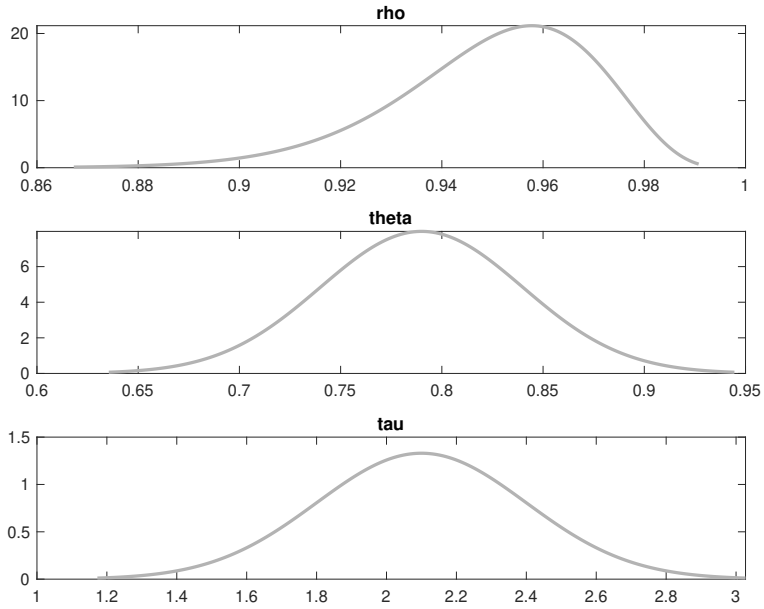


Figure 5: Prior Distribution

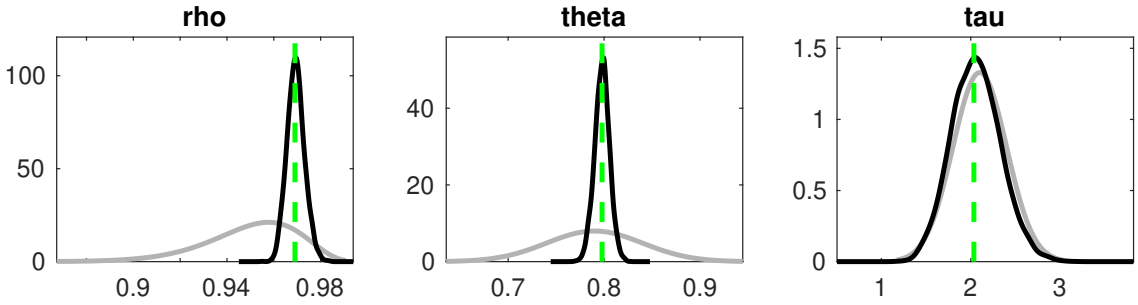


Figure 6: Estimation Results - Prior and Posterior Distributions

2.6 Evaluating the Model

Bayesian methods were initially introduced to improve results compared to fully calibrated models and maximum likelihood (ML) estimators. Calibration is great to explore the properties of a theoretical model because it provides a numerical representation of the model. However, it was admitted that calibration is not an estimate of the model. The improvement offered by Bayesian techniques is that the results are a combination of prior

beliefs and the data instead of fully relying of prior beliefs[13]. Regarding ML estimation Bayesian methods were thought to reduce the bias of the ML estimator (see below).

Evaluating the parameter estimates from any form of estimation procedure is important, as there is always the risk of biases (e.g.model misspecification, etc.). However, in the case of Bayesian estimation the evaluation of results is especially important. Three major issues motivate this. First, it is increasingly common to estimate macroeconomic models using Bayesian techniques but without testing the results. Second, Le et al. (2011) strongly reject the very famous and popular model of Smets and Wouters (2007). Third and maybe most important, because of the peculiarity of specifying priors Bayesian estimation is especially prone to potential biases, in particular in the case of subjective priors that do not reflect the value of the parameters of the model generating the data. This results in misleading parameter estimates as too much weight is placed on prior information compared to observed data. Therefore, "the danger, of course, is that it will be inferred that the model is correct no matter how flawed it may be" (Meenagh, Minford and Wickens, 2020, p.6).

A

2.6.1 Priors as source of bias

How the selection of prior distributions can produce biased results will be illustrated by means of the ML estimator. Recall that the ML estimation can be interpreted as Bayesian estimation with uniform (uninformative) priors. Consider a classical estimation with observed data O. Then θ is chosen such that it maximizes the log-likelihood function $\ln(p(O|\theta))$, i.e.

$$\arg\max_{\theta} \ln(p(O|\theta))$$

[13]Calibrated parameters also represent a priori beliefs (priors), but they are treated as exact since they are not "updated" by the data.

The ML estimator $\hat{\theta}$ is obtained by solving

$$\left.\frac{\partial \ln\left(p(O|\theta)\right)}{\partial \theta}\right|_{\theta=\hat{\theta}} = 0$$

In Bayesian estimation θ is estimated using either the mean or the mode of the posterior distribution. Because it is computationally easier to find the mode we use the mode instead of the mean for illustration[14]. The mode $\bar{\theta}$ is obtained by maximizing the posterior distribution, i.e.

$$\arg\max_{\theta} p(\theta|O) \equiv \arg\max_{\theta} \ln\left(p(\theta|O)\right)$$

This gives (see Bayes Theorem)

$$\ln\left(p(\theta|O)\right) = \ln\left(p(O|\theta)\right) + \ln\left(p(\theta)\right) - \ln\left(p(O)\right).$$

Again, because the last term does not contain θ, it can be ignored. Therefore, we get

$$\arg\max_{\theta} \ln\left(p(\theta|O)\right) \equiv \arg\max_{\theta} \left[\ln\left(p(O|\theta)\right) + \ln\left(p(\theta)\right)\right].$$

We obtain the posterior mode from

$$\left[\frac{\partial \ln\left(p(O|\theta)\right)}{\partial \theta} + \frac{\partial \ln\left(p(\theta)\right)}{\partial \theta}\right]_{\theta=\hat{\theta}} = 0 \tag{14}$$

Solving the first term $\frac{\partial \ln\left(p(O|\theta)\right)}{\partial \theta} = 0$ for θ gives the mode of the likelihood function (ML estimator), and solving the second term $\frac{\partial \ln\left(p(\theta)\right)}{\partial \theta} = 0$ for θ gives the mode of the prior distribution. By solving the sum of the two the posterior mode is obtained. If $\ln\left(p(O|\theta)\right)$ is flat, meaning that the data is uninformative about θ, then $\frac{\partial \ln\left(p(O|\theta)\right)}{\partial \theta}$ is (close to) zero for a range of values of θ. Consequently, the Bayesian estimator is dominated by the prior. If $p(\theta)$ is flat (uniform distribution) then $\frac{\partial \ln\left(p(\theta)\right)}{\partial \theta} = \frac{1}{p(\theta)}\frac{\partial p(\theta)}{\partial \theta} = 0$. It follows, that the Bayesian estimator is dominated by the data (i.e. equal to the ML estimator). How much weight is put on the two terms in Eq.(14) depends on the precision of the ML estimator and the variance of the prior distribution. The more precise the ML estimator the more weight is put on the likelihood, the less precise it is the more weight is put on the prior. Putting a lot of weight on the prior when the prior does not reflect the value of the parameters of the model generating the data, then this results in biased estimates.

[14] In case of a symmetric posterior distribution the mean and the mode coincide.

2.6.2 How to evaluate results and model performance

The results of a model can be evaluated with respect to other models (relative performance) or with respect to the data (absolute performance). While there exist quantitative measures of relative model performance, there does not exist any clear-cut quantitative assessment of the absolute model performance but only indirect inference is possible.

2.6.3 Relative performance – model comparison

Bayesian model comparison based on Bayes factors (BFs) is a method of model selection in a Bayesian framework. BFs are indices of relative evidence of one "model" over another, which can be used in the Bayesian framework as alternatives to classical (frequentist) hypothesis testing indices (such as p-values).

Recall Bayes theorem

$$p(\theta|O) = \frac{p(O|\theta)p(\theta)}{p(O)} \propto p(O|\theta)p(\theta)$$

where the marginal likelihood $p(O) = \int p(O|\theta)p(\theta)d\theta$ is the 'overall' likelihood of observing O. Suppose there are m_i alternative models with $i = 1, 2, \ldots, n$ depending on parameters θ_i. So, we have

$$p(\theta_i|O, m_i) = \frac{p(O|\theta_i, m_i)p(\theta_i|m_i)}{p(O|m_i)}, \qquad p(O|m_i) = \int p(O|\theta, m_i)p(\theta|m_i)d\theta$$

where $p(O|m_i)$ is the data density for model m_i given the parameter vector θ, i.e., the marginal likelihood associated with model m_i, and $p(\theta|m_i)$ denotes the prior density for model m_i.

We can now express the posterior model probability, i.e., whether a model is correct or not, using Bayes rule

$$p(m_i|O) = \frac{p(O|m_i)p(m_i)}{p(O)} \propto p(O|m_i)p(m_i).$$

This shows that the posterior model probability $p(m_i|O)$ is a function of the prior model probability (how likely we believe m_i to be correct before seeing the data) $p(m_i)$ and the marginal likelihood of the model $p(O|m_i)$.

To compare models, say m_1 and m_2 we can compute the posterior odds ratio (PO) which is the ratio of the two model probabilities:

$$PO_{1,2} = \frac{p(m_1|O)}{p(m_2|O)} = \frac{p(O|m_1)p(m_1)}{p(O|m_2)p(m_2)}.$$

When setting the prior odds ratio $\frac{p(m_1)}{p(m_2)}$ to unity we obtain the BF which only depends on the log-likelihood $\ln(p(O|m_i))$ and is thus much easier to compute.

$$BF_{1,2} = \frac{p(O|m_1)}{p(O|m_2)} = \frac{\exp(\ln(p(O|m_1)))}{\exp(\ln(p(O|m_2)))} \tag{15}$$

In general, the model with the highest log-likelihood "wins the likelihood race", i.e. is the preferred one. If, for instance, the marginal data density of m_1 is greater than the the one of m_2 we can conclude that m_1 fits the data better than m_2. This comparison of the log-likelihoods is useful when we want to compare different models or the same model using different parameter values.

Dynare computes the marginal data density of the model using the Geweke (1999) modified harmonic means estimator. The result is displayed in the Dynare output in the estimation results[15]:

```
ESTIMATION RESULTS
Log data density is -41.874154
```

However, by just comparing the log-likelihoods of two or more models we cannot say anything about whether the difference in the log-likelihoods is significant, i.e. whether we can reject the null hypothesis in favor of the alternative hypothesis. BFs, in contrast, can be interpreted. The common interpretation was first established by Harrods (1998) and

[15]After the mode finding another log data density is displayed (first stage result computed via Laplace approximation): `Log data density [Laplace approximation] is -56.672820`

slightly modified by Lee and Wagenmakers (2014). Suppose we test model m_2 against model m_1, i.e. whether m_1 or m_2 is more likely to have produced the observed data:

$BF_{1,2}$	Interpretation
> 100	Extreme evidence for m_1
30 – 100	Very strong evidence for m_1
10 – 30	Strong evidence for m_1
3 – 10	Moderate evidence for m_1
1 – 3	Anecdotal evidence for m_1
1	No evidence
1/3 – 1	Anecdotal evidence for m_2
1/3 – 1/10	Moderate evidence for m_2
1/10 – 1/30	Strong evidence for m_2
1/30 – 1/100	Very strong evidence for m_2
$< 1/100$	Extreme evidence for m_2

Given BFs model probabilities $p_1, p_2, \ldots, p_n$ can be obtained for n models. These show the probability of model m_i being the correct one among all n models. Since $\sum_{i=1}^{n} p_i = 1$ we have

$$\frac{1}{p_1} = \sum_{i=2}^{n} BF_{i,1} \quad \Rightarrow \quad p_i = p_1 BF_{i,1}$$

A limitation of BFs is that the assessment of model fit is only relative to the competing models. The best of all n models may still be poor (potentially misspecified) in capturing important dynamics in the data. Therefore, it is necessary to also evaluate the absolute performance of a particular model against the data (see below).

2.6.4 Absolute performance – model validation

As outlined above, measures of relative performance do not tell anything about absolute model performance as the best among all competing models may still be poor in capturing

important dynamics in the data. This makes it necessary to evaluate the absolute performance of the model, i.e. against the data. However, there does not exist *the* measure of absolute performance, but there are different methods that are rather approximate.

MATCHING MOMENTS Summary statistics have been standard measures of indirect inference since the beginning of DSGE models. The central question when comparing moments is: can the models correctly predict population moments such as e.g. the volatility and correlations of variables.

Following Schorfheide (2000), let O^{sim} be a sample of potential observations. One can derive the sampling distribution of O^{sim} given the current state of knowledge (prior), using Bayes theorem: $p(O^{sim}|O) = \int p(O^{sim}|\theta)p(\theta|O)d\theta$. Let $T(O)$ be a test quantity that reflects an aspect of the data (moment) that one wants to check. In order to assess whether the estimated model can replicate population moments, one sequentially generates draws from the posterior distribution, $p(O|\theta)$, and the predictive distribution $p(O^{sim}|O)$ so that the predictive $T(O^{sim})$ can be computed.

The `stoch_simul` command in Dynare generates these moments of all the endogenous variables. As Dynare computes the path of the variables over the specified period by solving all the equations for every point in time, (empirical) moments of the simulated variables O^{sim} can be calculated[16].

Commonly three sets of moments are used

- Volatility - Standard deviations
- Co-movements – Cross correlations (most often with output)
- Persistence – Autocorrelations

Regarding persistence not only first-order autocorrelations are considered but also autocorrelations up to higher orders. These are usually plotted. Whether or not the

[16]Dynare makes some specific assumptions when simulating and computing moments. First, shocks are assumed to follow a normal distribution and second, future shocks are expected to be zero.

simulated model moments are close (enough) to the actual data moments is a subjective judgment. Matching moments is a form (maybe the most common one) of indirect testing (see below). Fig. 7 shows some examples from the literature.

Figure 7: Examples from the literature: Matching moments

(A) Table 2, Anzoategui et al. (2019), Online Appendix

Table 2: Comparison of Standard Deviations

Variable	Data	Model
Output Growth	0.55	0.63
Consumption Growth	0.51	0.71
Investment Growth	1.54	1.52
Inflation	0.23	0.36
Nominal R	0.60	0.55
Hours (level)	1.82	1.53
R&D Expenditure Growth	4.00	6.83

(B) Table 1 and 3, King and Rebelo (1999)

Table 1

Business Cycle Statistics for the U.S. Economy

	Standard Deviation	Relative Standard Deviation	First Order Auto-correlation	Contemporaneous Correlation with Output
Y	1.81	1.00	0.84	1.00
C	1.35	0.74	0.80	0.88
I	5.30	2.93	0.87	0.80
N	1.79	0.99	0.88	0.88
Y/N	1.02	0.56	0.74	0.55
w	0.68	0.38	0.66	0.12
r	0.30	0.16	0.60	-0.35
A	0.98	0.54	0.74	0.78

Table 3

Business Cycle Statistics for Basic RBC Model[35]

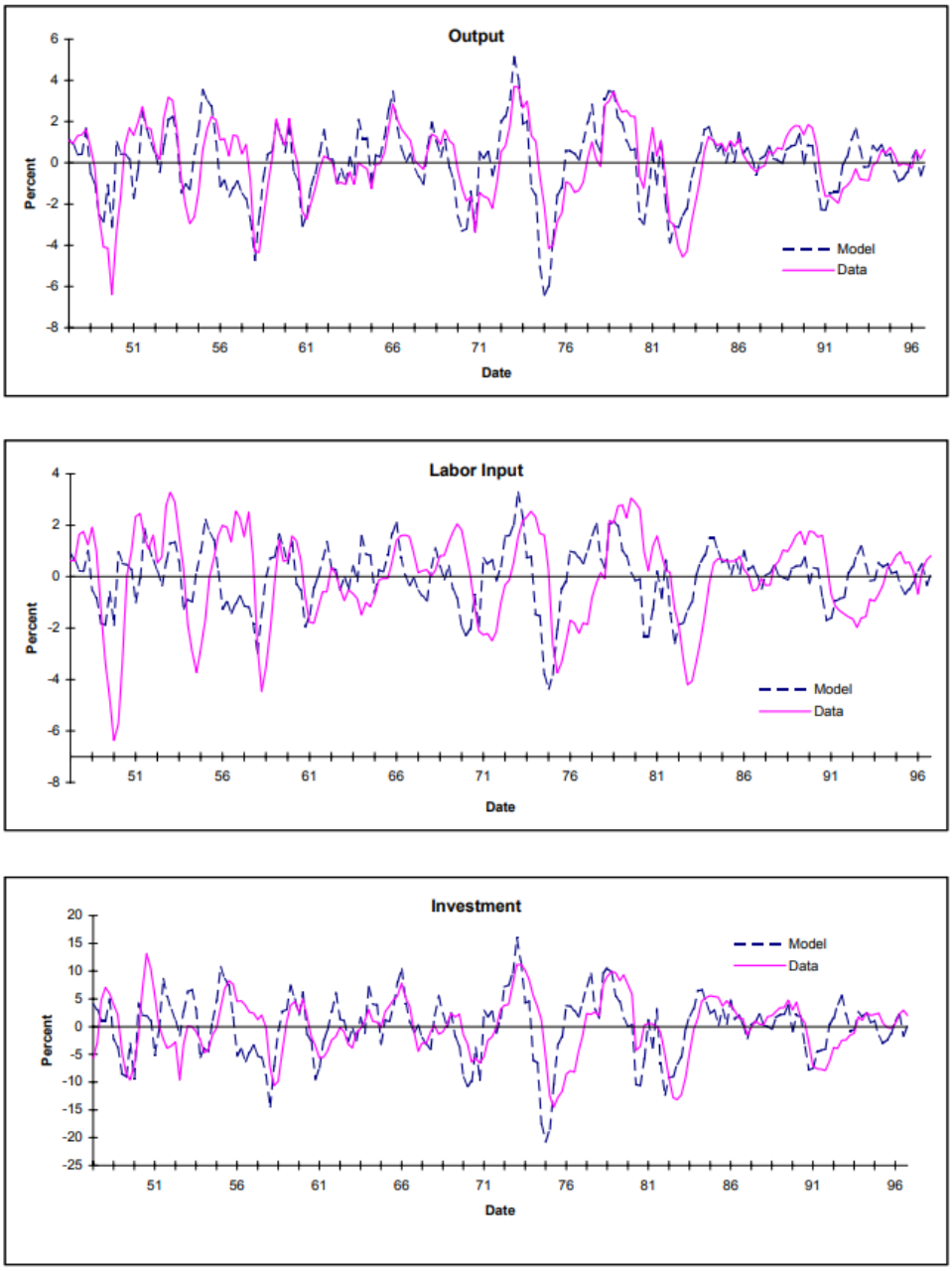
	Standard Deviation	Relative Standard Deviation	First Order Auto-correlation	Contemporaneous Correlation with Output
Y	1.39	1.00	0.72	1.00
C	0.61	0.44	0.79	0.94
I	4.09	2.95	0.71	0.99
N	0.67	0.48	0.71	0.97
Y/N	0.75	0.54	0.76	0.98
w	0.75	0.54	0.76	0.98
r	0.05	0.04	0.71	0.95
A	0.94	0.68	0.72	1.00

COMPARING TIME SERIES An even more intuitive way of assessing whether the model is able to capture important dynamics in the data is to directly compare simulated

data with actual data, i.e. plot both together. Also with this method the judgement of the model performance is subjective. Fig. 8 shows some examples from the literature.

Figure 8: Examples from the literature: Comparing time series

(A) Part of Figure 7, King and Rebelo (1999)



(B) Figure 1, Adolfson et al. (2007)

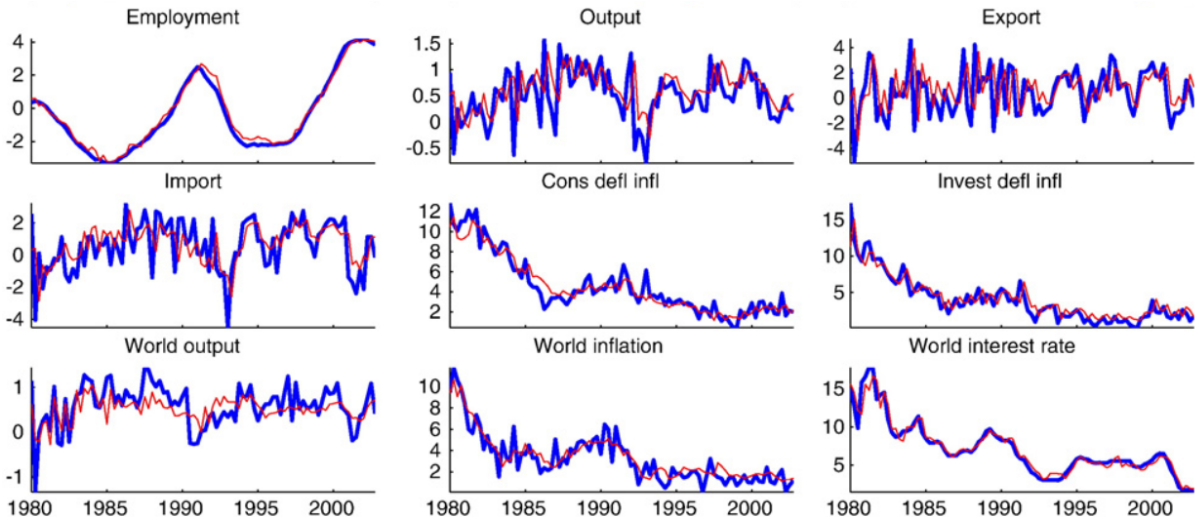


Fig. 1. Data (thick) and one-sided predicted values from the model (thin).

(C) Figure 1, Keuschnigg et al. (2020)

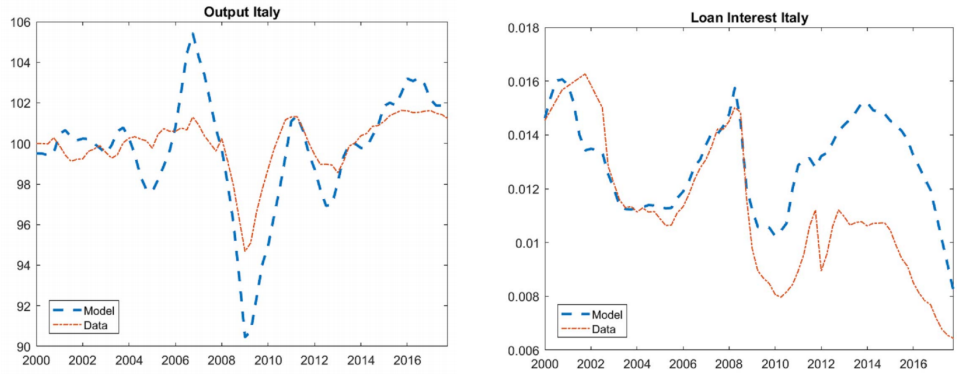


Figure 1: Simulated and Actual Time Series

FORECAST TESTS When assessing the model performance via its forecasting ability the first step is to split the data sample into an in-sample period, which is used to estimate the model and an out-of-sample period. The model, estimated based on the in-sample period, is used to generate forecasts for the out-of-sample period. Comparing the out-of-sample forecasts with the actually observed data in that period evaluates the forecasting performance. If the forecast is able to approximately track the actual data, the model and its parameters can be considered as "good", meaning that parameter estimates are unlikely to be biased, as it has a good ability in capturing the cyclical features of the

data. However, even though out of sample forecasts of DSGE models are often poor, this does not automatically mean that the model is bad. If the in-sample period and the out-of-sample forecast period are very different (e.g. pre-crisi period vs. crisis period) then the bad out-of-sample forecast performance is both expected and explicable. Therefore, and because data availability often does not allow to split the data, the methods outlined above are more common.

INDIRECT TESTING If choosing prior distributions is perceived as a drawback of Bayesian estimation one could use different estimation techniques and compare the results. Meenagh, Minford and Wickens (2020) propose an alternative, namely indirect estimation.

Indirect estimation involves several steps:

1. Simulate the structural model for given parameter values.
2. Estimate an auxiliary model using the simulated data. The role of the auxiliary model is to represent the characteristics of the data. Examples of auxiliary models are sample moments, derivatives of the likelihood function and unrestricted VAR models.
3. Estimate the same auxiliary model but this time using the actually observed data.
4. Compare the estimates of the auxiliary model obtained from simulated data with these obtained from actual data.
5. The values of the structural parameters in step 1 are revised until the estimates of step 2 converge to those of step 3[17].

According to Meenagh, Minford and Wickens (2020) VAR (or VARMA) models are reasonable auxiliary models because first they capture more characteristics of the data then

[17]By using hypothesis tests like the Wald test, Lagrange multiplier, or likelihood-ratio test the equivalence of parameters can be tested.

other potential auxiliary models and second VAR (or VARMA) models with coefficient restrictions are the solution to linearized DSGE models.

Another way would be not to aim for aligning DSGE parameters to the VAR parameters but to adjust DSGE parameters until impulse response functions from both DSGE and VAR models are aligned. Fig. 9 shows some examples from the literature.

Figure 9: Examples from the literature: Indirect testing

(A) Part of Table 2, Smets and Wouters (2003)

TABLE 2. ESTIMATION STATISTICS

Summary of the model statistics: VAR—BVAR—DSGE				
	VAR(3)	VAR(2)	VAR(1)	DSGE-model
In sample RMSE (80:2–99:4)				
Y	0.42	0.44	0.50	0.54
π	0.20	0.21	0.23	0.21
R	0.12	0.12	0.13	0.12
E	0.19	0.20	0.22	0.21
w	0.48	0.51	0.54	0.57
C	0.42	0.44	0.48	0.60
I	1.03	1.08	1.17	1.26

3 Step-by-Step Guide: Dynare Implementation

3.1 Software

MATLAB:

1. **License**: you can get a 30 day trial or buy a license.
2. **Download**: the MATLAB software is availabe for download on the mathworks download page. You will need to create an account first and then log in to it.
3. **Installation:** run the installer, accept the licensing agreement, connect your license via the activation key. A detailed description is available online.

Dynare

1. **License**: Dynare is free and open-source software. You do not need to buy a license.

2. **Download:** Go to the download page and choose the version compatible with your operating system.

3. **Installation:** Open the installer and follow the installation instructions.
   (*Note for Apple users: The computer may block you from opening the installation package because it is not issued by a "verified developer". If the package was downloaded from the official Dynare website, this is nothing to be worried about. Simply go to: "Settings" → "Security" → "General". There, you will find a button to open the package despite the warning.*)

3.2 Document Logic: MATLAB and Model Files

The following section will describe the recommended file structure and explain the code within each of those.

3.2.1 Run.m

This is the central file that tells MATLAB which code to run at what point in the program. When running a simulation or estimation, this is the file to open and execute. The following list contains and explains all the commands typically contained in this code.

1. Adds the required dynare code to a MATLAB program. The path might differ and to find it, one has to search for the dynare folder and copy the path.

   ```
   addpath /Applications/Dynare/4.6.0/matlab;
   ```

2. This clears all variables and figures from the working space and is required before the program is rerun to ensure is can be executed smoothly.

```
clear all; clc;
```

3. Calls the Calib.m file, where the values for parameters and variables are calibrated. Refer to the detailed description of this file below.

   ```
   Calib;
   ```

4. Calls the Variables.m file, where variables are defined. Refer to the detailed description of this file below.

   ```
   Variables;
   ```

5. Calls the SaveInitvals.m file, that saves the calibrated values of the variables as starting values for a simulation. Refer to the detailed description of this file below.

   ```
   SaveInitvals;
   ```

6. Calls the saveParams.m file, that saves the calibrated values of parameters in a file called params.mat. Refer to the detailed description of this file below.

   ```
   SaveParams;
   ```

7. Calls the PrepareEndvals.m file, that solves for a new steady state when a variable is slightly changed. This is a check for the stability of the model implemented. Refer to the detailed description of this file below.

   ```
   PrepareEndvals;
   ```

8. This command instructs MATLAB to run dynare commands on the model contained in the Model.mod file and based on the scenario defined in Scenarios.m. Refer to the detailed description of these files below.

```
dynare Model.mod;
```

3.2.2 Calib.m

The Calib.m file calibrates the steady state, meaning the variables and parameters get assigned their (initial) values.

1. Defines global variables, need to be in the same order as in the file Declarations.m - moreover, artificial values are required for endogenous variables used as a check, and time *t*.

```
global z l k y Fk Fl c uc i...;
```

2. Defines global parameters, need to be in the same order as in the file Declarations.m.

```
global rho s zbar alpha theta beta delta tau...;
```

3. The vector xvec is used to prepare the solutions and check the steady state. The vector needs to be as long as the number of artificial variables without counting *t*. Then, the function *SS(xvec)* is called. This function contains all the dynamic equations in the steady state. The initial value of the artificial variable(s) is used to calculate the steady state values of all the endogenous variables without shocks.

```
t       = 1;
xvec    = ones(1,1);
```

```
ED      = SS(xvec);
disp([' ']);
disp(['CHECK calibrated ISS     ',num2str(sum(abs(ED)))]);
```

3.2.3 SS.m

Contains a function that prepares the steady state values for all the endogenous variables. The equations are a result of the analytical calibration process. At the beginning, global variables and parameters need to be defined, in the same order as in Declarations.m. Moreover, the same artificial variables as in Calib.m including the time variable t are required.

3.2.4 Declarations.m

Defines the endogenous and exogenous variables of the model, as well as the parameters and how they are saved within the program.

1. Endogenous variables:

   ```
   var z l k y Fk Fl c uc i...;
   ```

2. Exogenous variables/shocks.

   ```
   varexo e...;
   ```

3. Parameters of the model.

   ```
   parameters rho s zbar alpha theta beta delta tau...;
   ```

4. Parameter values are 'imported' from the values calibrated in Calib.m and saved.

```
load params;
set\_param\_value('rho',rho);
set\_param\_value('s',s);
set\_param\_value('zbar',zbar);
set\_param\_value('alpha',alpha);
set\_param\_value('theta',theta);
set\_param\_value('beta',beta);
set\_param\_value('delta',delta);
set\_param\_value('tau',tau);...
```

3.2.5 Variables.m

The variable vector "vars" is initialized. Again, variables should be listed in the same order as in Declarations.m:

```
vars = {'z', 'l', 'k', 'y', 'Fk', 'Fl', 'c', 'uc', 'i'...};
```

3.2.6 SaveInitvals.m

The initial calibrated values of the variables are saved. They will all be available with a 0 attached to the respective variable name:

```
for index = 1:length(vars)
    eval([vars{index} '0' '=' vars{index} ';'])
end

save initvals *0;
```

3.2.7 SaveParams.m

Saves the parameters in the file params.mat, use the same order as in Delarations.m: save params.mat rho s zbar alpha theta beta delta tau...;

3.2.8 PrepareEndvals.m

Prepares a steady state after a slight change in the artificial variable(s) and saves the result as end values for all the variables.

```
disp([' ']);
disp(['COMPUTE NEW FSS']);
xvec0   = 1+randn(1,1)/20;
xvec    = fsolve(@SS,xvec0);
ED   = SS(xvec);
disp(['CHECK FSS:       ']);
[xvec0,xvec,abs(ED')]

for index = 1:length(vars)
    eval([vars{index} '\_end' '=' vars{index} ';'])
end

save endvals *end;
```

3.2.9 Model.mod

1. Includes the file Declarations.m that contains the list of endogenous and exogenous variables as well as parameters.

   ```
   @#include "Declarations.m"
   ```

2. Defines the model implemented, meaning all the dynamic equations are listed.

```
model;
z       = rho*z(-1)+s*e;
y       = exp(z)*zbar*l^(1-alpha)*k(-1)^alpha;
Fk      = alpha*y/k(-1);
Fl      = (1-alpha)*y/l;
Fl      = (1-theta)/theta*c/(1-l);
uc      = beta*(Fk(+1)+1-delta)*uc(+1);
c       = theta*(c^theta*(1-l)^(1-theta))^(1-tau)/uc;
i       = y-c;
k       = i + (1-delta)*k(-1);
end;
```

3. Includes initial and ending values of the variables as well as the file Scenarios.m, which contains the specific commands of what dynare is supposed to carry out regarding the model at hand. More detailed information can be found in the description of the Scenarios.m file.

```
@#include "Initvals.m"
@#include "Endvals.m"
@#include "Scenarios.m"
```

3.2.10 Scenarios.m

Contains the commands for dynare, depending on the intended action. A couple of possible scenarios are listed in the following as examples.

1. Temporary Productivity Shock and Perfect Foresight: Solves the model for a number of periods (in this case, 400) with a temporary productivity shock from periods 1 to 10, and a maximum number of iterations of 15.

```
shocks;
var e;
    periods 1:10;
    values (0.01*z0);
end;
perfect_foresight_setup(periods=400);
perfect_foresight_solver(maxit=15);
```

2. Stochastic Simulation: Solves a stochastic version of the model where shocks e happen according to a given distribution. Will simulate all endogenous variables in a set number of periods (here 1000). Finally, the simulated data series are saved in the file 'simudata.mat'. They can be used to show how variables will evolve over time with the given setup or to estimate certain parameters of the model.

```
shocks;
var e; stderr 1;
end;
stoch_simul(periods=1000); save('simudata.mat');
```

3. Bayesian Estimation: In Bayesian estimation, we require prior beliefs on parameter values and their distributions. In the example below, parameters ρ, θ and τ are estimated, the prior belief regarding their distributions is given by the second element of the respective command line. The last two values are the prior mean and standard deviation. The shock process needs to be set up, which happens in

the second block. Finally, we have to define the observed variables (in this case c) and include the estimation command (more on selecting observed variables and estimation strategies in section **??**).

```
estimated_params;
rho, beta_pdf, 0.93, 0.02;
theta, normal_pdf, 0.3, 0.05;
tau, normal_pdf, 2.1, 0.3;
end;
estimated_params_init(use_calibration);
end;

shocks;
var e; stderr 1;
end;

varobs c;
estimation(datafile=simudata,mode_compute=6);
```

3.3 Implementing a Model

1. Variables and Parameters: Declarations.m, Variables.m, Calib.m, SaveParams.m, Model.mod
2. Steady State and xvec
3. Calibration
4. Scenarios

5. Distribution of Priors: beta_pdf, gamma_pdf, normal_pdf, uniform_pdf, inv_gamma_pdf, inv_gamma1_pdf, inv_gamma2_pdf and weibull_pdf. Note that inv_gamma_pdf is equivalent to inv_gamma1_pdf

6. Observation equations, see Pfeifer (2013)

3.4 Simulation

It is necessary to distinguish whether the model at hand is stochastic or deterministic as the solution methods differ. The distinction hinges on whether the occurrence of future shocks is known.

- Deterministic

  In deterministic models, the occurrence of all future shocks is exactly known, hence there is no uncertainty around shocks. If, for example, a shock only occurs in period 1, agents will form their decisions based on the knowledge that future shocks will be zero. However, a shock can also occur in the future, in which case it would be expected by the agents with perfect foresight. At the time of the decision, agents are able to determine optimal actions for all future periods due to full information and perfect foresight. Deterministic models are usually used to study the impact of a policy regime change, for example an introduction of a new tax regime.

- Stochastic

  In stochastic models, only the distribution of shocks is known, thus there is uncertainty around the occurrence of shocks. If, for example, a shock occurs in period 1, agents will base their decisions on the knowledge that the realizations of future shocks are random but will have a mean of zero. In this stochastic setting, agents specify decisions, policy or feedback rules contingent on the possible realizations of shocks.

In a deterministic setting, an accurate model solution can be found by numerical methods. In a stochastic setting, however, the goal is to find a function satisfying the model's optimality conditions. The latter may be non-linear and needs to be approximated then.

As an example for the specification in Dynare, we assume to have a linear production function

$$Y_t = z_t L_{t-1}$$

where z_t denotes productivity and L_{t-1} labor input. Furthermore, productivity is assumed to follow an AR(1) process

$$z_t = \rho z_{t-1} + e_t$$

where ρ measures the persistence of productivity and $e_t \sim N(0, \sigma)$.
In the deterministic setting, we need to specify z_t directly as an exogenous variable in the 'Declarations.m' file:

varexo z;

To simulate the deterministic model, we need to set the following command in the 'Scenarios.m' file:

simul(OPTIONS);
The latter command is equivalent to:
perfect_foresight_setup(OPTIONS);
perfect_foresight_solver(OPTIONS);

In the stochastic setting, we need to specify z_t as an endogenous variable and the productivity shock e_t as an exogenous variable in the 'Declarations.m' file:

var Y L z;

varexo e;

To simulate the stochastic model, we need to set the following command in the 'Scenarios.m' file:

stoch_simul(OPTIONS);

For a comprehensive description of the available options, please see the Dynare Reference Manual.

3.5 Estimation

- Observation equations
- Estimated parameters
- Estimation commands

3.6 Troubleshooting[18]

3.6.1 Priors

- Poor starting values

  It is fairly common to see a Dynare error:

  > dynare_estimation_init:: The steady state at the initial parameters cannot be computed.
  > Impossible to find the steady state (the sum of square residuals of the static equations is 24794.2760). Either the model doesn't have a steady

[18] Jacob Stevens also contributed to this section

state, there are an infinity of steady states, or the guess values are too far from the solution.

The two most common explanations are that (i) you have misspecified an equation in the calibration/steady state sections, meaning the system has no solution is general, or (ii) you have given an implausible starting value for a parameter, meaning the system has no solution for the given parameters.

- Unbounded density
- Two modes
- Wrongly scaled observations

  Different error messages can also implicitly result from wrongly specified data points in the observed time series. A frequent misspecification is the scaling of the observations, e.g. for interest rates. Most financial databases will provide interest rate data in percentage points, i.e. an interest rate of 1% will be reported as 1.00 and not 0.01. This is easily overlooked, you should therefore closely compare your time series with the steady state values of the model to ensure they are in a similar range.

3.6.2 Estimation Performance

- Acceptance Ratio

  You should aim for an acceptance ratio of about 0.33 by varying the scale parameter.

- Scale Parameter

  The scale parameter is defined in the estimation command by *mh_jscale*. Increasing the scale parameter lowers the acceptance ratio, while decreasing the scale parameter increases it.

References

[1] Adolfson, Malin, Laséen, Stefan, Lindé, Jesper, and Villani, Mattias (2007). Bayesian estimation of an open economy DSGE model with incomplete pass-through. *Journal of International Economics*, 72(2), 481–511

[2] An, S., & Schorfheide, F. (2006). Bayesian Analysis of DSGE Models.

[3] Anzoategui, Diego, Diego Comin, Mark Gertler, and Joseba Martinez (2019), Endogenous technology adoption and R&D as sources of business cycle persistence, *American Economic Journal: Macroeconomics* 11(3), 67–110

[4] Del Negro, Marco and Schorfheide, Frank (2008). Forming priors for DSGE models (and how it affects the assessment of nominal rigidities). *Journal of Monetary Economics* 55, 1191–1208

[5] Fernández-Villaverde, J. (2010). The econometrics of DSGE models. SERIEs, 1(1-2), 3-49.

[6] Fernández-Villaverde, Jesús, Rubio-Ramírez, Juan and Schorfheide, Frank (2016). Solution and Estimation Methods for DSGE Models. In Handbook of macroeconomics (Vol. 2, pp. 527-724). Elsevier.

[7] Geweke, John (1999). Computational experiments and reality. University of Minnesota and Federal Reserve Bank of Minneapolis.

[8] Hamilton, J. D. (2018). Why you should never use the Hodrick-Prescott filter. Review of Economics and Statistics, 100(5), 831-843.

[9] Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications.

[10] Herbst, E. P., & Schorfheide, F. (2016). Bayesian estimation of DSGE models. Princeton University Press.

[11] Jeffreys, Harold (1998). *The theory of probability* (3rd ed.). Oxford University Press

[12] King, Roberto, and Rebelo, Sergio (1999). Resuscitating real business cycles. *Handbook of Macroeconomics*, 1, 927–1007

[13] Keuschnigg, Christian, Kirschner, Linda, Kogler, Michael and Winterberg, Hannah (2020). Italy in the Eurozone. CEPR Discussion Paper No. DP15040

[14] Le, Mai, Meenagh, David, Minford, Patrick, and Wickens, Michael (2011). How much nominal rigidity is there in the US economy? Testing a new Keynesian DSGE model using indirect inference. *Journal of Economic Dynamics and Control*, 35(12), 2078–2104

[15] Lee, Michael, Wagenmakers, Eric-Jan (2014). Bayesian cognitive modeling: A practical course. Cambridge University Press

[16] Meenagh, David, Minford, Patrick and Wickens, Michael (2020). Estimating macro models and the potentially misleading nature of Bayesian estimation. CEPR Discussion Paper (No. DP15684)

[17] Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). Equation of state calculations by fast computing machines. The journal of chemical physics, 21(6), 1087-1092.

[18] Pfeifer, J. (2020). A guide to specifying observation equations for the estimation of DSGE models. Research series, 1-150.

[19] Schorfheide, Frank (2000). Loss function-based evaluation of DSGE models. *Journal of Applied Econometrics*, 15(6), 645–670

[20] Smets, Frank, and Wouters, Rafael (2003). An estimated dynamic stochastic general equilibrium model of the euro area. *Journal of the European Economic Association*, 1(5), 1123–1175

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

[22] Wolf, E., Mokinski, F., & Schüler, Y. S. (2020). On adjusting the one-sided Hodrick-Prescott filter.

END