Replication code and data for: "Bayesian Inference on Structural Impulse Response Functions"

Mikkel Plagborg-Møller*

July 4, 2018

1 Introduction and acknowledgments

This document describes the replication files for my paper "Bayesian Inference on Structural Impulse Response Functions" (Plagborg-Møller, 2018) and its Online Appendix.[1] The code produces all numbers and figures referred to in the paper, up to simulation noise. The code is extensively commented, and it is hopefully easy to apply to alternative datasets and prior specifications. Please refer to Plagborg-Møller (2018) and Hoffman & Gelman (2014) for detailed descriptions of the posterior simulation algorithm.

The files are organized into three main folders:

1. `data`: Data from Fernald (2014) and the St. Louis Fed's FRED database.
2. `matlab`: Matlab code for producing the numbers and figures in the main paper and Online Appendix, as well as Dynare code for the news shock application.
3. `stata`: Stata files that manipulate the data.

In addition to code I have produced, the replication material draws on the following files that have been wholly or partially produced by other authors:

- The spreadsheet `quarterly_tfp.xls` contains data from Fernald (2014). The spreadsheet has been downloaded from John Fernald's website: http://www.frbsf.org/economic-research/economists/john-fernald/.

*Email: mikkelpm@princeton.edu. I would be grateful for any comments or error reports.

[1] The Online Appendix is available on my website: http://scholar.princeton.edu/mikkelpm/publications/irf_bayes

- The files in the folder `fred` were downloaded from the St. Louis Fed's FRED database.
- The Matlab function `nuts_da_svma.m` is a modified version of the NUTS code available on Matt Hoffman's website: http://matthewdhoffman.com.
- The Matlab function `subtightplot.m` is due to Felipe G. Nievinski, Pekka Kumpulainen, and Nikolay S. I downloaded it from Matlab Central: http://www.mathworks.com/matlabcentral/fileexchange/39664-subtightplot.
- The Dynare file `news.mod`, which contains the news shock DSGE model equations from Sims (2012), was written by me from scratch, but I compared it extensively to Eric Sims' Dynare code. Any errors are my own.

2 Data and Stata files

In the `stata` folder, the Stata do-file `create.do` generates the CSV file `data_q_final.csv` with quarterly data for the news shock application. In turn, the CSV file is loaded by the Matlab routines described below. I include the CSV file in the replication material, so it is not necessary to execute the Stata file before running the Matlab code. The Stata file also creates a few auxiliary Stata databases and a log file.

3 Dynare files

The folder `matlab/dynare_news` contains files for generating impulse responses from the model in Sims (2012). The basic Dynare model file is `news.mod`. Relative to the model in Sims (2012), I have slightly extended the model equations to include rule-of-thumb price setters, as in Galí & Gertler (1999); however, setting the parameter ω equal to zero reverts back to Sims' model, which is what I do for the results in the main paper.

The Matlab file `run_news.m` sets parameters and passes the model file to Dynare. Parameter settings correspond to the baseline calibration in Sims (2012). The script generates a series of plots of impulse response functions, and it stores the most important impulse response functions in the data file `news_irf_grid_horiz3.mat`. This file is read by the Matlab routine that defines the prior for the news shock application, see below.

Dynare must be installed before executing `run_news.m`. However, because I include `news_irf_grid_horiz3.mat` in the replication material, `run_news.m` need not be executed prior to running the posterior simulation algorithm for the news shock application.

4 Posterior simulation files

The folder `matlab/hmc_svma` contains several files that produce all the figures in the main paper and Online Appendix. The main executable Matlab file is `run_svma.m`. Other files in the `matlab/hmc_svma` root folder set the prior, data, and settings for the application and simulations. The files contained in the various subfolders execute the Hamiltonian Monte Carlo (or more specifically, NUTS) algorithm, compute likelihoods, simulate data, produce plots, etc. I describe the files below.

The code is at present set up to work with the multivariate Gaussian prior family introduced in Section 2.5 of the main paper. I discuss below which files should be altered to use other types of priors.

4.1 `invert_identif_example.m`

This file produces Figure 2 in the main paper. It is otherwise unrelated to the posterior simulation files, except that it also uses the root-flipping routines.

4.2 `plot_rhos.m`

This file produces Figure 4 in the main paper, using the procedures for plotting prior IRF draws described below.

4.3 Prior, data, and settings files

The files whose names start with `prior_` define the multivariate Gaussian prior for each application. The Matlab structure "prior" contains all prior settings. These include variable and shock names (for plotting), the prior means "mu" and "mu_sigma", the prior standard deviations "tau" and "tau_sigma", and the prior smoothness parameter "rho". Finally, the normalization of the impact impulse responses is determined by the vector "norm_var", whose elements are $(i_1, i_2, \dots, i_n)$, in the notation of the main paper. If the reader wishes to use a prior that is not of the multivariate Gaussian type in Section 2.5 of the main paper, I describe the necessary steps below.

The files whose names start with `data_` define the datasets (actual or simulated) for the applications. In particular, each file defines a $T \times n$ data matrix "Y", either by reading CSV-formatted data or by simulating artificial data. Optionally, the files set true values for Θ and σ (for plotting and simulation), if these exist.

The files whose names start with `settings_` define settings for running the NUTS algorithm, including the number of MCMC steps, the target acceptance rate, etc. Please refer to the settings files for a detailed description of the various options. For other applications than the ones in my paper, it may be necessary to tweak the settings to optimize performance.

4.4 `run_svma.m`

The main executable Matlab file is `run_svma.m`. This file (with appropriate settings) produces all figures in the main paper and Online Appendix, except Figures 2 and 4.

The top part of the code sets a number of settings. First, the application number is set. Nine applications are pre-defined:

1. The baseline simulation with $\rho_{ij} = 0.9$ in Section C.1.2 of the Online Appendix.
2. The alternative simulation with $\rho_{ij} = 0.3$ in Section C.1.2 of the Online Appendix.
3. The "misspecified persistence" simulation in Section C.1.3 of the Online Appendix.
4. The "misspecified cross-correlations" simulation in Section C.1.3 of the Online Appendix.
5. The "misspecified persistence and cross-correlations with larger prior variance" simulation in Section C.1.3 of the Online Appendix.
6. The "grossly misspecified prior" simulation in Section C.1.3 of the Online Appendix.
7. The news shock application with actual data in Section 4 of the main paper.
8. The news shock application with actual data and larger prior variance in Section C.2.5 of the Online Appendix.
9. The news shock simulation with simulated data in Section C.2.8 of the Online Appendix.

Each of these applications loads its own combination of data, prior, and settings files.

To use the code with a new application, simply create new prior, data, and settings files (see above), and set the three variables "prior_filename", "data_filename", and "settings_filename" in `run_svma.m` accordingly. Finally, the "results_filename" variable must be set. Results from the posterior simulation algorithm will be stored here (if posterior plots are requested but the NUTS algorithm has not yet been run, the code will attempt to load results from this file).

The lines of code below the comment “Decide what to do” determine which overall parts of the algorithm should be run. These parts include plotting the prior, loading and/or plotting the data, running the NUTS algorithm on the loaded data, storing the results of the NUTS algorithm in a Matlab data file, and plotting the results of the NUTS algorithm. If the NUTS algorithm has not been run, but the code is set to plot results, it will attempt to load an already-stored Matlab data file. The replication material includes nine such data files in the `matlab/hmc_svma/results` subfolder, containing the simulation output used for producing the plots in the main paper.

The rest of the settings after the “Decide what to do” code lines switch on or off various detailed plotting options. Simply switch all of these on to produce all figures. The various types of plots produced are described in detail in the code comments. Some of the plot types produced by the code are not included in the main paper.

If a non-Gaussian prior is desired, it is necessary to change some of the calls of the plotting functions mid-way through the code. These calls currently plot prior confidence bands that are based on percentiles of the Gaussian distributions. However, these calls only affect prior plots, not the posterior inference.

4.5 `data_sim` folder

This folder contains routines for detrending data using the biweight kernel smoother, for simulating Gaussian SVMA data, and for simulation smoothing of the structural shocks (Durbin & Koopman, 2012, Ch. 4.9).

4.6 `densities` folder

This folder contains code for computing the Kalman filter, the Whittle likelihood, the prior, and the posterior. The Kalman filter is only used for doing inference on structural shocks, as well as for computing relative likelihood weights, which may be used in the optional reweighting procedure described in the Online Appendix (reweighting is not used for any of the results reported in the main paper or Online Appendix). The file `prior_svma.m` returns the log prior density and its gradient, under the assumption of a multivariate Gaussian prior as in Section 2.5 of the main paper. Modify this file if other types of priors are desired.

4.7 `identif_set` folder

This folder contains files for computing the identified set of the SVMA model. These files are used by `invert_identif_example.m` and for finding an initial approximation to the mode of the posterior before the NUTS algorithm is run, see the Online Appendix.

4.8 `nuts` folder

This folder contains files that implement the posterior simulation algorithm, including computing the initial value of the chain and running the NUTS algorithm. The main NUTS file is `nuts_da_svma.m`, which is a modified version of Matt Hoffman's generic NUTS code.

4.9 `reporting` folder

This folder contains several routines that report various features of the prior or posterior distributions, mostly through plots. The wrapper function for summarizing the posterior is `results_svma.m`. Other functions compute and/or plot IRFs, long-run IRFs, shock standard deviations, the FEVD, shocks, time series, information criteria, autocorrelations, various measures of invertibility, closest invertible IRFs, prior/posterior predictive checks, and the Müller (2012) prior sensitivity measure.

The files `results_svma.m` and `plot_longrun_hist.m` rely on percentiles of the Gaussian distribution when plotting features of the prior, and so these should be modified if a non-Gaussian prior is assumed.

4.10 `results` folder

This folder contains Matlab data files with posterior simulation output from the application and simulations. These files contain the particular posterior draws that were used to construct the reported figures and numbers in the main paper and Online Appendix. Running the NUTS algorithm again should reproduce these data files, up to simulation noise.[2]

[2]The random number generator seed set in `run_svma.m` is the one used to produce the plots in the paper, but the effect of the seed appears to depend on operating system and possibly Matlab version.

References

Durbin, J. & Koopman, S. (2012). *Time Series Analysis by State Space Methods* (2nd ed.). Oxford Statistical Science Series. Oxford University Press.

Fernald, J. (2014). A Quarterly, Utilization-Adjusted Series on Total Factor Productivity. Federal Reserve Bank of San Francisco Working Paper No. 2012-19.

Galí, J. & Gertler, M. (1999). Inflation dynamics: A structural econometric analysis. *Journal of Monetary Economics* 44(2), 195–222.

Hoffman, M. D. & Gelman, A. (2014). The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research* 15, 1593–1623.

Müller, U. K. (2012). Measuring prior sensitivity and prior informativeness in large Bayesian models. *Journal of Monetary Economics* 59(6), 581–597.

Plagborg-Møller, M. (2018). Bayesian Inference on Structural Impulse Response Functions. Working paper. http://scholar.princeton.edu/sites/default/files/mikkelpm/files/irf_bayes.pdf.

Sims, E. R. (2012). News, Non-Invertibility, and Structural VARs. In N. Balke, F. Canova, F. Milani, & M. A. Wynne (Eds.), *DSGE Models in Macroeconomics: Estimation, Evaluation, and New Developments*, Vol. 28 of *Advances in Econometrics*, 81–135. Emerald Group Publishing.