

SVAR-DPMM

Replication files for "The importance of supply and demand for oil prices: evidence from non-Gaussianity" (Braun, 2023)

The code has been tested using Matlab 2020b and runtimes were recorded based on a standard Windows 10 laptop with an Intel(R) Core(TM) i5-6300U CPU @ 2.40GHz Processor. To speed up computations, the code makes use of two mex functions, namely "dcoovstatC.mexw64" and "sampleK_mex.mexw64". These functions are used for the Matteson & Tsay (2017) test statistic and MCMC step updating the mixture allocation using Algorithm 3 of Neal (2000). These mex functions are based on C++ code that was compiled with a MinGW64 Compiler. If you encounter problems with these mex files, you can use the source C++ functions provided in the "mex_files" folder to recompile them on your personal computer. Alternatively, you can use the Matlab function "sampleK_matlab.m" as a substitute for the "sampleK_mex" function. Lastly, please note that the source code of the mex file underlying the Matteson & Tsay test statistic comes from the R Package "steady ICA".

To replicate the results presented in the empirical part of the paper, the following code must be executed sequentially:

1. "runme_baseline_BH19_NG.m" (~ 1.5 hours for 100'000 draws). This code estimates the "baseline" non-Gaussian oil market model and stores the estimates in the folder "results".
2. "runme_baseline_BH19.m" (~25 mins for 100'000 draws). This code estimates the "baseline" Gaussian oil market model and stores the estimates in the folder "results".
3. "Tables_and_Figures_Baseline.m". This code uses the output from the two baseline models and computes the main empirical results presented in the paper:
 - Figure 3: posterior predictive densities of the oil market shocks
 - Table 2: Skewness and Kurtosis of the oil market shocks
 - Figure 4: Posterior distribution of the test statistic proposed by Montiel-Olea et al. (2022) and Matteson & Tsay (2017)
 - Figure 5: posterior density of key structural parameters (including figure D.2 for the remaining parameters shown in Appendix D)
 - Figure 6: Posterior median IRFs with 90% credible intervals (shaded areas)
 - Table 3: Forecast Error Variance Decomposition (FEVD) of the real oil price growth
 - Figure B.7 (Appendix): Markov Chain Monte Carlo output for each element of A, including Geweke's Relative Numerical Efficiency statistics
4. "runme_robustness_models.m" (~6-7 hours). This code estimates six robustness models at once. This includes the non-Gaussian (1) and Gaussian (2) models under a truncated prior for the demand elasticity labeled as R1, the non-Gaussian (3) and Gaussian (4) models using a shorter subsample starting in January 1985 labelled as R2. Finally, two additional non-Gaussian models (5 & 6) are estimated with modified error-distributions including parametric student-t errors labelled as R3 and DPMM errors with a strong prior around one component labelled as R4. Given the amount of robustness models, it will take considerable time to run this code.
5. "Tables_and_Figures_Robustness.m": This code uses the output from the robustness models and computes the following Tables and Figures presented in the paper:

- Table 4: Robustness analysis for the main empirical findings based on models R1 and R2
- Table F.1 (Appendix): Posterior distribution of the degrees of freedom (model R3)
- Figure F.1 (Appendix): posterior predictive densities of the oil market shocks (model R4)
- Table F.2 (Appendix): Further Robustness analysis for the main empirical findings based on models R3 and R4

Finally, the code provided in "runme_small_working_example.m" accompanies the exercises outlined in Appendix B and C. If you plan to use the methodology of the paper, this code will be a more useful starting point as it abstracts from the measurement error required for the oil market model. Furthermore, it illustrates the Marginal Likelihood (ML) estimator described in section 2.5., a useful tool for model comparison.

- First, data is simulated from a bivariate non-Gaussian SVAR(0) described in Appendix B and C
- Second, the SVAR-DPMM model is estimated on the simulated data, illustrating how it can recover the structural parameters
- Third, the code computes the log marginal likelihood (ML) based on the cross entropy method described in the paper, and compares it to a ML estimate for a restricted model where the supply elasticity is restricted to be zero. It then computes the Bayes Factor.