Replication Code: "What Can Time-Series Regressions Tell Us About Policy Counterfactuals?"

Alisdair McKay & Christian K. Wolf

April 5, 2023

1 Introduction

This document describes the replication files for the paper "What Can Time-Series Regressions Tell Us About Policy Counterfactuals?". The code produces all numbers and figures referred to in the paper. The files are organized into two main folders—one for the theoretical results and one for the empirical analysis.

All codes have been run and tested on Matlab R2021b. To ensure that all codes run, the variable "local"—located near the top of the various m-files—needs to be changed to reflect the local machine. In addition to code that we have produced, the replication material draws on the following files that have been wholly or partially produced by other authors:

- We use a simple HANK model to provide a numerical illustration of our theoretical identification result. Our solution of the HANK model uses: first, several files from the replication codes of the article Ahn et al. (2017); and second, the CompEcon toolbox of Miranda and Fackler, available here: `www4.ncsu.edu/~pfackler/compecon`. The codes closely build on those used by one of the authors in Wolf (2021).
- For plotting purposes we use the file `jbfill.m`, available on Mathworks file exchange. The license is reproduced in the folder `empirics/_auxiliary_functions`.

The rest of this note describes the contents of each folder in detail.

2 Theory

This folder contains all files necessary to produce the figures that we use to illustrate our identification result—i.e., Figure 1 as well as Figure A.1.

- The sub-folder `_auxiliary_functions` contains various functions that are used in our computations.
- We illustrate our identification result in the context of a simple structural HANK model. The file `get_hank_inputs.m` in the sub-folder `_hank_inputs` produces the Jacobian derivative matrices—following Auclert et al. (2021)—used in our numerical solution of the model. The code produces the file `inputs_hank.mat` which is needed for all subsequent computations.
- The file `get_cnfctl.m` in the sub-folder `cnfctl_policy.m` produces Figure 1. The file proceeds as follows: we first specify the environment, then solve the model under a counterfactual policy rule, and finally use policy shocks to the baseline rule to predict the counterfactual, using either our identification result or our generalization of the approach of Sims & Zha (1995). The sub-folder `_results` then stores the figures that make up Figure 1 in the paper.
- The file `get_optpol.m` in the sub-folder `opt_policy.m` produces Figure A.1. The file proceeds similarly to the counterfactual rule case: we first specify the environment, then solve for optimal policy outcomes by solving the policymaker problem, and finally use policy shocks to the baseline rule to construct the optimal policy outcome. The figure is stored in `_results`.

3 Empirics

This folder contains all files necessary to produce the figures that we use to showcase our empirical method—Figures 2-5 in the main text, and Figures C.1-C.4 in the appendix. The code also produces Table C.1.

- The sub-folder `_auxiliary_functions` contains various functions that are used in our computations.

- All data required to run the various VARs are collected in the file `data.csv` located in the sub-folder `_data`. Data sources are discussed in the article, and further details are provided in the accompanying file `construction.txt`.

- The file `plot_emp_illustr.m` in the sub-folder `illustrate_method` produces Figure 3. We stress that the inputs used by this file—i.e., `emp_illustr_inputs.mat`—do *not* come from any empirical or model-based analysis; the figure is purely illustrative.

- The file `run_vars.m` in the sub-folder `var_inputs.m` runs the four VARs that we use as inputs for our various policy counterfactual exercises: the baseline and alternative shock monetary policy shock VARs; the investment shock VAR; and a reduced-form VAR giving us Wold innovation impulse responses. Results are stored in the sub-folder `_results`. The file `plot_var_irfs.m` uses these results to produce the figures that make up Figure C.1 in the paper.

- Our policy counterfactual applications are run in the folder `policy_counterfactuals`.

    1. The file `run_inv_cnfctl.m` produces our Lucas critique-robust counterfactual policy approximation for the investment shock. The beginning of the file contains two options—which monetary shocks to use (baseline or alternative) and which counterfactual policy rule to study (output gap stabilization, Taylor rule, interest rate peg, nominal GDP targeting). By varying those options the file produces Figures 3-4 as well as Figures C.2-C.4.
    2. The file `run_inv_optpol.m` produces our optimal policy counterfactual for the investment shock. When set to use the baseline monetary policy shocks this file produces Figure 5.
    3. The file `run_wold_optpol.m` constructs counterfactual optimal policy second moments for output, inflation, and interest rates. When set to use the baseline monetary policy shocks this file produces Table C.1.

    Results for all experiments are stored in the folder `_results`.

References

Ahn, S., Kaplan, G., Moll, B., Winberry, T., & Wolf, C. K. (2017). When Inequality Matters for Macro and Macro Matters for Inequality. *NBER Macroeconomics Annual*, *32*.

Auclert, A., Bardóczy, B., Rognlie, M., & Straub, L. (2021). Using the Sequence-Space Jacobian to Solve and Estimate Heterogeneous-Agent Models. Working Paper.

Sims, C. A. & Zha, T. (1995). Does monetary policy generate recessions? Working Paper.

Wolf, C. K. (2021). Interest Rate Cuts vs. Stimulus Payments: An Equivalence Result. Working Paper.