Replication package for „Forecasting with Shadow-Rate VARs"

- Andrea Carriero (Queen Mary University of London)
- Todd Clark (Federal Reserve Bank of Cleveland)
- Massimiliano Marcellino (Department of Economics, Baffi, Bidsa, IGIER, Bocconi University and CEPR)
- Elmar Mertens (Deutsche Bundesbank; em@elmarmertens.com) *corresponding author*

The usual disclaimers apply: the views and results conveyed in our research are solely those of the authors and do not necessarily reflect the views of the Federal Reserve Bank of Cleveland, the Federal Reserve System, the Eurosystem, or the Deutsche Bundesbank.

A manuscript of our paper and a supplementary online appendix are available here: https://www.elmarmertens.com/research/workingpapers#h.nqkg0ogemt4s

This replication package is maintained at https://github.com/elmarmertens/CCMMshadowrateVAR-code.

When reading this README, please note that links to file names work when viewing the markdown version of this README.md in the main folder of the repository.

Overview

The code in this replication package estimates all models discussed in the paper and produces all figures and tables shown in paper and appendix using MATLAB. There are **2** main files, called **runbatch.m**, and **runtablesandfigures.m** to perform estimation and construct figures and tables, respectively. In addition, the code can be used to reconstruct or update our data inputs from FRED-MD source data, using a single script. The replicator should expect the code to run for about 35 hours on a machine with 32 workers (or cores). The code also works with as few as a single worker, but runtime should be expected to lengthen correspondingly.

This package also contains a csv file with our shadow rate estimates named CCMMshadowrateEstimates.csv, with further details described below.

Data Availability and Provenance Statements

- ☐ This paper does not involve analysis of external data (i.e., no data are used or the only data are generated by the authors via simulation in their code).

Statement about Rights

- ☑ I certify that the author(s) of the manuscript have legitimate access to and permission to use the data used in this manuscript.
- ☑ I certify that the author(s) of the manuscript have documented permission to redistribute/publish the data contained within this replication package. Appropriate permission are documented in the LICENSE.txt file.

Summary of Availability

- ☑ All data **are** publicly available.
- ☐ Some data **cannot be made** publicly available.
- ☐ **No data can be made** publicly available.
- ☐ Confidential data used in this paper and not provided as part of the public replication package will be preserved for ___ years after publication, in accordance with journal policies.

Details on each Data Source

Data.Name	Data.Files	Location	Provided	Citation
"FRED-MD"	2022-09.csv	data/	TRUE	McCracken and Ng (2016)
"Krippner Shadow-rate estimates"	SSR_Estimates_2022March.xlsx	data/	TRUE	Krippner (2015)

Data.Name	Data.Files	Location	Provided	Citation
"Wu-Xia Shadow-rate estimates"	shadowrate_US.xls	data/	TRUE	Wu and Xia (2016)

Dataset list

Our analysis is based on a single data file obtained from FRED-MD. In addition, for comparison plots of our shadow-rate estimates against (updated) estimates produced by Krippner (2015) Wu and Xia (2016) we use two additional data files.

FRED-MD data were downloaded from the Federal Reserve Bank of St. Louis. We use the 2022 September vintage. A copy of the data is provided as part of this archive. The data are in the public domain.

- Datafile: data/2022-09.csv

Updated shadow rate estimates from Krippner (2015) and Wu and Xia (2016) were downloaded from the authors websites. These data series are only used for comparison plots, and not in our estimation. The data are publicly available, and copies are provided with this replication package (including copies in non-proprietary format). Estimates from Krippner can be downloaded from https://www.ljkmfa.com/visitors/ and are subject to his disclaimer and copyright statements (https://www.ljkmfa.com/disclaimer-copyright/). We use his estimates from March 2022. Updated estimates from Wu and Xia can be downloaded from https://sites.google.com/view/jingcynthiawu/shadow-rates. We use a copy downloaded on September 10, 2022.

- Datafile for Krippner estimates: data/SSR_Estimates_2022March.xlsx. A copy of the relevant workbook sheet, "D. Monthly average SSR series," in non-proprietary format is provided as well in data/SSR_Estimates_2022March_MonthlyAverageSSRseries.csv
- Datafile for Wu-Xia estimates: data/shadowrate_US.xls. A copy in non-proprietary format is provided as well in data/shadowrate_US.csv

Computational requirements

Software Requirements

- ☑ The replication package contains one or more programs to install all dependencies and set up the necessary directory structure.

(The replication package provides all dependencies and each script sets up the necessary path variables.)

- MATLAB including Statistics and Machine Learning Toolbox and Parallel Computing Toolbox (code was run with MATLAB Release 2023b, and 2024b)
- Toolbox files maintained by Elmar Mertens (one of the authors), included as part of this package. Also available at https://github.com/elmarmertens/em-matlabbox (Code was run with commit from Dec 30 2024, hash: 84bcdf4)
- Optional: TeX Live to compile tables and figures produced by our MATLAB code into PDF files.
- Optional: Bash or Zsh shell environments in Linux or macOS (henceforth "Bash") to run helper scripts that are provided as part of this replication package. These scripts serve merely to regenerate MATLAB driver files that are provided as part of this replication package, and their use is not necessary to replicate our results. (The scripts can also be used to generate our MATLAB driver files under different parametrizations than used for the paper.)

Controlled Randomness

- ☑ Random seed is set at line **34 in each of the following programs:** goVAR.m, goVAR200811.m, goVARshadowrateBlockNonstructural.m, goVARshadowrateNonstructural.m, goVARshadowrateGeneral.m, doVARshadowrateBlockNonstructural.m, and doVARshadowrateNonstructural.m
- ☐ No Pseudo random generator is used in the analysis described here.

Except for goVAR200811.m, each of the programs listed above executes code in parallel workers using substreams of the global random-generator stream, which is seeded as described above. The substreams are assigned at the following code lines:

- goVAR.m:260
- goVAR200811.m:257
- goVARshadowrateBlockNonstructural.m:276
- goVARshadowrateGeneral.m:280

- goVARshadowrateNonstructural.m:275

From the programs listed above, we also derive MATLAB driver files named `bot*.m` (by using the shell script preparebatch.sh as detailed below) and these inherit the above-described settings for the random seed.

Memory, Runtime, Storage Requirements

Summary

Approximate time needed to reproduce the analyses on a standard (2024) desktop machine:

- ☐ <10 minutes
- ☐ 10-60 minutes
- ☐ 1-2 hours
- ☐ 2-8 hours
- ☐ 8-24 hours
- ☐ 1-3 days
- ☐ 3-14 days
- ☑ > 14 days

Feasible in about 35 hours on a 32 core machine.

Approximate storage space needed:

- ☐ < 25 MBytes
- ☐ 25 MB - 250 MB
- ☑ 250 MB - 2 GB
- ☐ 2 GB - 3 GB
- ☐ 3 GB - 25 GB
- ☐ 25 GB - 250 GB
- ☐ > 250 GB

Approximate RAM memory needed:

- 24 GB (sequential estimation), and 2-3 GB per core when using parallel execution.

- ☐ Not feasible to run on a desktop machine, as described below.

Details

The main computational routines were last run on a **32-core Intel virtual client** with Intel(R) Xeon(R) Gold 6248R CPU (3.0 GHz), **Windows 11 Enterprise**, and **112 GB of RAM** using **MATLAB 2023b**. Computation took about **35 hours**.

Auxiliary code to collect MCMC estimates, and produce tables and figures was last run on a **laptop with Apple Silicon chip** (M1 pro), **macOS Sequoia 15.2**, and **MATLAB 2024b for macOS** with **16GB of RAM**.

Description of programs/code

- The default data files are fredsxMD20-2022-09.csv (all 20 variables covered) and fredsxMD20exYield-2022-09.csv (all data ex Yields). Both are based on the 2022-09 vintage of FRED-MD, available at https://research.stlouisfed.org/econ/mccracken/fred-databases/, and can also be reconstructed with the script generateFREDdata.m that is provided as part of our data folder.

To produce quasi-real estimates of each model defined in the paper, there are various `go*.m` files:

- goVAR.m for the standard linear VAR. (goVAR200811.m estimates the linear VAR using only data through 2008; these estimates are used as reference point in our analysis of parameter changes of various models described in the supplement.)

- goVARshadowrateGeneral.m for the general shadow rate VAR (applies only to the exYield dataset provided in fredsxMD20exYield-2022-09.csv).

- goVARshadowrateNonstructural.m for the unrestricted version of the non-structural shadow-rate VAR

- **goVARshadowrateBlockNonstructural.m** for the restricted version of the non-structural shadow-rate VAR

- In order to apply each dataset to each applicable model, use the MATLAB driver files `botgo*.m`. This set of driver files can also be re-created (and with different settings) by the Bash script **preparebatch.sh** as described further below.

Other estimates, tables and figures: To produce all tables and figures of the paper, as well as additional material as shown in the supplementary appendix, call **runtablesandfigures.m**, which calls the following scripts to create tables and figures:

- **oosEvaluationTables.m** produces forecast comparison tables (as in Tables 2 and 3 of the paper) based on output stored by the `goVAR*.m` scripts. Comparisons against a linear VAR that omits short-rate data, as in Table 4 of the paper, are compiled by **oosEvaluationTablesAgainstLongYieldsOnly.m**.
- **barplotBETAslopes.m** reports the ELB-specific coefficients estimated from the general shadow-rate VAR as reported in Figure 1 of the paper.
- **showShadowrates.m** produces figures of shadow-rate estimates as shown in Figure 2 in the paper, and stores the time series of shadow-rate estimates in **CCMMshadowrateEstimates.csv**. The shadow-rate estimates are obtained from output stored by the `goVAR*.m` scripts.
- **oosPlotForecastsSHADOWFFR.m** to plot predictive densities for shadow rate and federal funds rate at selected jump offs (as in Figure 3).
- **showPAIchanges.m** produces comparison figures of VAR transition coefficients as discussed in the supplement to our paper.

All core scripts are in the main directory. In addition, there are the following subdirectories:

- **data** contains raw input data, and the script **generateFREDdata.m** to produce input files, named `fredsxMD*.csv`, as needed by the estimation routines (copies of these input files are also provided in the main folder).
- **matlabtoolbox** for general utilities (also available at https://github.com/elmarmertens/em-matlabbox).
- By default, estimation results are stored as `*.mat` files in the main folder of the replication package.
- The subfolder `foo` will be created inside the main folder by our routines for creating tables and figures.

License for Code

The code is licensed under a BSD license. See **LICENSE.txt** for details. Toolbox code from Elmar Mertens is also licensed under a BSD license. See **matlabtoolbox/LICENSE.txt**.

Instructions to Replicators

The following instructions should be executed in a MATLAB session with the current working directory set to the main folder of the replication package:

Option 1

In a Linux or macOS shell:

- You can use the **makefile** provided with this replication package to perform the steps described above: `make all` performs estimation, creates all tables and figures and compiles the two LaTeX files mentioned above.
- By default, the **makefile** will execute the computational routines in parallel mode, which requires availability of MATLAB's Parallel Computing Toolbox. If parallel mode is not available, please replace the **makefile** calls to **goparbatch.sh** by calls to **goseqbatch.sh**.
- The **makefile** uses the scripts **goparbatch.sh** and **goseqbatch.sh** to launch MATLAB and execute our code. In Linux, these shell scripts assume that the command `matlab` is on the shell path. In macOS, each of these shell scripts defines an alias `matlab` that points to the installed MATLAB version. To adapt this setting to your environment, please edit line 12 in **goparbatch.sh** and **goseqbatch.sh** as needed.

Option 2

When launching our code in an interactive MATLAB session:

- If available, and prior to launching our routines, enable parallel processing of `parfor` loops to create a parallel pool in MATLAB, which requires the MATLAB Parallel Computing Toolbox; otherwise the loops will be executed sequentially. Choose a number of parallel workers suitable for your computing environment (in terms of available CPU and RAM memory). For

example, to use the default setting for your system, simply use `parpool`; in order to use the code with 32 workers use `parpool(32)`. When no `parpool` is created, `parfor` steps will be executed one-by-one. Depending on system defaults in your MATLAB installation, MATLAB might also launch `parpool` automatically when encountering the first `parfor` (or `spmd`) command. In principle, it should suffice to launch `parpool` only once per session. But, depending on system defaults, please note that the parallel pool may automatically terminate when idle (and thus needs to be launched again for further use).

- Call runbatch.m, which launches a sequence of out-of-sample runs for all models and datasets considered in the paper. Specifically, runbatch.m calls various `bot*.m` files. Each file creates a `*.mat` file with model-specific results data that is stored in the repository's main directory. (This step took us about 35 hours on a virtual client with 32 cores.)

- Call runtablesandfigures.m to create all tables and figures for the paper and its supplementary appendix (with the exception of Table 1). This script requires that runbatch.m has finished (and the resulting `*.mat` files are stored in the repository's main directory). Outputs are stored in `*.tex` format (for tables) and `*.eps` format (for figures) and stored in a subfolder `foo`. (This step takes a couple of minutes.)

- To re-create the list of variables provided in Table 1: In MATLAB, change the working directory to the subfolder data and execute generateFREDdata.m which creates datalist-fredsxMD20-2022-09.tex. The resulting TeX file corresponds to Table 1 in the paper (after some manual reformatting).

- Optional: When all tables and figures are stored in `foo`, you can compile the LaTeX files CCMMpaperTablesAndFigures.tex and CCMMsupplementTablesAndFigures.tex to collect the figures and tables from paper and supplement.

Details

- All scripts set the MATLAB path to point to toolboxes in matlabtoolbox. In addition, most scripts collect output in a temporary directory, which is by default created as subfolder `foo` within the main directory (if not present the folder will be created). Edit localtemp.m to change the location of this temp directory (or place an alternate copy of `localtemp.m` in the main directory of the repository).

- All necessary input files are already placed in the main folder of this repository. (See below for recreating or updating these inputs from source data files provided in the data folder of this repository.)

- Output is collected in a LaTeX file, which is also compiled at the end of each script (provided a LaTeX installation can be found on the path). To control the compilation of output, please edit finishwrap.m. To avoid collecting output files, comment out the call to initwrap in each script (and make sure to define instead a variable called `wrap` that is set to empty).

- To launch out-of-sample runs for a given model, use scripts called `go*.m` (or `bot*.m` as described in the next bullet). After computing the out-of-sample runs each of these `go*.m` scripts stores results for further post-processing in a `*.mat` file. While looping over out-of-sample runs in our `go*.m` files, individual iterations may encounter (rare) errors due numerical issues, which are reported on screen. These errors do not lead to abortion of the program; instead, they are caught by the program and the iteration in question is relaunched.

- Each `bot*.m` file is a copy of a corresponding `go*.m` file, but with presets that apply the model to a particular dataset. In Bash, you can create this set of `bot*.m` files, by calling the Bash script preparebatch.sh and pass the appropriate `go*.m` file names as argument. For example `sh preparebatch.sh goVAR*.m` prepares `bot*.m` files for all `go*.m` files in the repo and prepares estimation of each model with and without longer-run yields data (as needed in the paper). Also, if `goVAR.m` is included in the argument list when calling preparebatch.sh, an additional MATLAB driver file is created that estimates the linear VAR to data that omits shorter-run yields affected by the effective lower bound. The script preparebatch.sh can also be used to vary other parameters for the out-of-sample runs. For example, to create `botgo*.m` files that employ a different number of MCMC draws, change the value of `MCMCdraws` on line 19 of preparebatch.sh.

- The main directory contains the Bash scripts goparbatch.sh and goseqbatch.sh that can be used to launch a sequence of multiple MATLAB scripts from the shell. Both shell scripts expect that the names of the MATLAB scripts to be executed should be passed as argument list. Both shell scripts execute the arguments in sequence *and in separate MATLAB sessions*. In case of goparbatch.sh, the MATLAB session opens a parallel pool. For example, the shell command `sh gobatch.sh goVAR.m goVARshadowrateBlockNonstructural.m goVARhybrid.m` will launch a command line session of MATLAB, start a parallel pool, and then execute goVAR.m; once goVAR.m has been executed, the MATLAB session closes, a new one is reopened for execution of goVARshadowrateBlockNonstructural.m etc. (The shell script works with as many

command line arguments as supported by Bash and has been written for use on macOS and Linux.) Alternatively, MATLAB scripts can, of course, also be called interactively on the MATLAB GUI's command line. As noted above: In Linux, these shell scripts assume that the command `matlab` is on the shell path. In macOS, each of these shell scripts defines an alias `matlab` that points to the installed MATLAB version. To adapt this setting to your environment, please edit line 12 in goparbatch.sh and goseqbatch.sh as needed.

List of tables and programs

The provided code reproduces:

- ☑️ All numbers provided in text in the paper
- ☑️ All tables and figures in the paper
- ☐ Selected tables and figures in the paper, as explained and justified below.

Note: Table 1 of the paper consists only of text that lists data mnemonics and transformations and is hard-coded in the paper's TeX file.

Figure/Table #	Program	Line Number	Output file	Note
Table 1	data/generateFREDdata.m	395ff	data/datalist-fredsxMD20-2022-09.tex	manually reformatted for paper
Table 2	oosEvaluationTables.m	328	tripleComparisonQE-fredsxMD20exYield-2022-09-Standard-p12-vs-shadowrateGeneralVAR-p12-vs-nonstructuralVAR-p12-evalStart201001evalEnd201712.tex	
Table 3	oosEvaluationTables.m	328	tripleComparisonQE-fredsxMD20-2022-09-Standard-p12-vs-nonstructuralVAR-p12-vs-blocknonstructuralVAR-p12-evalStart201001evalEnd201712.tex	
Table 4	oosEvaluationTablesAgainstLongYieldsOnly.m	255	tripleComparisonQE-fredsxMD14longyields-2022-09-standardVAR-p12-vs-nonstructuralVAR-p12-vs-BlocknonstructuralVAR-p12-evalStart201001evalEnd201712.tex	
Figure 1	barplotBETAslopes.m	93ff	BETA-ELBshadowrateGeneralAR1SV-RATSbvarshrinkage-p12-202208.eps	
Figure 2, Panel a	showShadowrates.m	139ff	shadowrate1-p12-fredsxMD20exYield-2022-09-ELBnonstructuralAR1SV-vs-ELBshadowrateGeneralAR1SV-LSAP.eps	see line 184ff for tabulated values
Figure 2, Panel b	showShadowrates.m	164ff	shadowrate1-p12-fredsxMD20-2022-09-ELBblocknonstructuralAR1SV-wuxiakrippner-LSAP.eps	see line 184ff for tabulated values

Figure/Table #	Program	Line Number	Output file	Note
Figure 3, Panel a	oosPlotForecastsSHADOWFFR.m	147ff	FEDFUNDS-fredsxMD20-202209-blocknonstructuralshadowrateAR1SV-predictivedensity1-2020-03-WITHLEGEND.eps	see 197ff for tabulated values
Figure 3, Panel b	oosPlotForecastsSHADOWFFR.m	147ff	FEDFUNDS-fredsxMD20-202209-blocknonstructuralshadowrateAR1SV-predictivedensity1-2020-09.eps	see 197ff for tabulated values
Table B.1	oosEvaluationTables.m	328	tripleComparisonQE-fredsxMD20exYield-2022-09-Standard-p12-vs-shadowrateGeneralVAR-p12-vs-nonstructuralVAR-p12-evalStart201001evalEnd202208.tex	
Table B.2	oosEvaluationTables.m	328	tripleComparisonQE-fredsxMD20-2022-09-Standard-p12-vs-nonstructuralVAR-p12-vs-blocknonstructuralVAR-p12-evalStart201001evalEnd202208.tex	supplement
Table B.3	oosEvaluationTablesAgainstLongYieldsOnly.m	255	tripleComparisonQE-fredsxMD14longyields-2022-09-standardVAR-p12-vs-nonstructuralVAR-p12-vs-BlocknonstructuralVAR-p12-evalStart201001evalEnd202208.tex	supplement
Figure C.1 , Panel a	showPAIchanges.m	174ff	PAI-intercept-standardVARAR1SV-fredsxMD20-2022-09	supplement
Figure C.1 , Panel b	showPAIchanges.m	174ff	PAI-intercept-ELBblocknonstructuralAR1SV-fredsxMD20-2022-09	supplement
Figure C.1 , Panel c	showPAIchanges.m	212ff	PAI-lag1-standardVARAR1SV-fredsxMD20-2022-09	supplement
Figure C.1 , Panel d	showPAIchanges.m	212ff	PAI-lag1-ELBblocknonstructuralAR1SV-fredsxMD20-2022-09	supplement
Figure C.1 , Panel e	showPAIchanges.m	251ff	PAI-lagOTHER-standardVARAR1SV-fredsxMD20-2022-09	supplement
Figure C.1 , Panel f	showPAIchanges.m	251ff	PAI-lagOTHER-ELBblocknonstructuralAR1SV-fredsxMD20-2022-09	supplement
Figure C.2 , Panel a	showPAIchanges.m	174ff	PAI-intercept-standardVARAR1SV-fredsxMD20exYield-2022-09	supplement
Figure C.2 , Panel b	showPAIchanges.m	174ff	PAI-intercept-ELBblocknonstructuralAR1SV-fredsxMD20exYield-2022-09	supplement

Figure/Table #	Program	Line Number	Output file	Note
Figure C.2 , Panel c	showPAIchanges.m	212ff	PAI-lag1-standardVARAR1SV-fredsxMD20exYield-2022-09	supplement
Figure C.2 , Panel d	showPAIchanges.m	212ff	PAI-lag1-ELBblocknonstructuralAR1SV-fredsxMD20exYield-2022-09	supplement
Figure C.2 , Panel e	showPAIchanges.m	251ff	PAI-lagOTHER-standardVARAR1SV-fredsxMD20exYield-2022-09	supplement
Figure C.2 , Panel f	showPAIchanges.m	251ff	PAI-lagOTHER-ELBblocknonstructuralAR1SV-fredsxMD20exYield-2022-09	supplement

References

Krippner, Leo (2015), Zero Lower Bound Term Structure Modeling: A Practitioner's Guide. Palgrave Macmillan.

McCracken, Michael W. and Serena Ng (2016), "FRED-MD: A monthly database for macroeconomic research." Journal ofBusiness & Economic Statistics, 34 (4), 574–589.

Wu, Jing Cynthia and Fan Dora Xia (2016), "Measuring the macroeconomic impact of monetary policy at the zero lower bound." Journal of Money, Credit and Banking, 48 (2-3), 253–291.