Read Me

Ryan Chahrour[*] Kyle Jurado[†]

Boston College Duke University

February 3, 2021

Abstract

Data and replication instructions for *Recoverability and Expectations-Driven Fluctuations*, published in the Review of Economic Studies.

[*]Department of Economics, Boston College, Chestnut Hill, MA 02467, U.S.A. Email: ryan.chahrour@bc.edu

[†]Department of Economics, Duke University, Durham, NC 27708, U.S.A. Email: kyle.jurado@duke.edu

Data availability and Provenance:

1. Cumulative nominal stock returns were downloaded from the CRSP database via WRDS (download from https://wrds-www.wharton.upenn.edu) and deflated by the GDP deflator. Access to this dataset requires a subscription. The series is the quarterly nominal NYSE stock return including distributions, selected after clicking on Stock File Indexes. A copy of this data is included in this archive.

2. Aggregate utilization adjusted TFP is taken from the San Francisco Federal Reserve (downloaded from https://www.frbsf.org/economic-research/indicators-data/total-factor-productivity-tfp/). A copy of this data is included in this archive.

3. The remaining series were dowloaded from the FRED database maintained by the Federal Reserve Bank of St. Louis (https://fred.stlouisfed.org). The variables used are services consumption (FRED code PCESV), nondurable consumption (PCND), durable consumption (PCDG), gross private domestic investment (GDPI), hours worked in the non-farm business sector (HOANBS), the GDP deflator (GDPDEF), 3-month Treasury yields (TB3MS), and population (CNP16OV). The data were download in Matlab using the DataStream toolbox, using the `get_FredData.m` command included in the replication files. A copy of this data is included in this archive.

4. The original replication files for Forni et al. (2017) were downloaded from https://doi.org/10.3886/E114132V1. A copy of this data (including original license) is included in this archive.

Dataset List:

Data file	Source	Notes	Provided?
data/fred_data.mat	St Louis FRED	Downloaded w. Datafeed toolbox	Yes
data/wrds_crisp.xlsx	CRSP	Stock Returns	Yes
data/quarterly_tfp.xlsx	San Francisco Federal Reserve	Utilization-adjusted TFP	Yes
fgls_orignal/	AEJ: Macro/Open:ICSPSR	Replication code for Forni et al. (2017)	Yes

Software:

This code was last run/tested in Mac OSX 10.14 using Matlab R2017a. The code requires Matlab's Symbolic Toolbox and Parallel Computing Toolbox. Original download of FRED data also used Matlab's Datafeed toolbox. No additional toolboxes are needed to replicate the results here.

The files have also all been run on Red Hat Linux Cluster, `Linux 3.10.0-1062.el7.x86_64`.

Hardware:

These programs have no special hardware requirements. However, some steps (noted below) are time-consuming, taking up to 18 hours on a 4-core 2018 iMac. The full set of code was most recently run in 4.5 hours on a Red Hat Linux cluster using 24 cores and 32 GB of RAM .

Instructions for Replication:

The replication code is organized according to each task. Helper functions that are reused several times across directories are stored in directories labeled `/*_box`. Each directory (other than `/*_box`) contains a file label `contents.txt` that summarizes the contents of that directory.

All computational steps can be performed at once by calling:

```
>> run_all
```

from the root directory. Re-computing all results from scratch takes around 4.5 hours on a modern cluster. To reproduce only figures, without recomputing all results from scratch call:

```
>> run_fast
```

from the root directory. This takes less than minute on a laptop.

Step-by-step instructions for replicating each figure in the paper, with or without using saved results, are below. All `>>cd` commands assume you begin in the root `/public_code` directory.

Setting up & data transformation

- Before running any code, add all auxiliary files to the path by calling:

  ```
  >> do_setup
  ```

  from the root directory.

- To load and transform the historical US data mat-file from the original series download from FRED and CRSP:

  ```
  >> cd data
  >> make_data
  ```

  These steps save the required variables to `/empirical/data_final.mat`, a copy of which is already included in the `/empirical` directory.

- Figure 1 is a conceptual diagram. No code is included.

Main empirical results

- To reproduce Figures 2, 3 and 4:

  ```
  >> cd empirical
  >> genplots_paper
  ```

  The results used for these figures are saved in `noise.mat` and `tfp_freq.mat`, which are included in the directory.

- To reproduce all calculations involved in making Figures 2, 3 and 4:

  ```
  >> cd empirical
  >> runall
  ```

  This steps takes about 10 hours on a quad-core iMac and about 2 hours on a modern cluster with 24 cores. To speed it up (at the cost of reducing the number of bootstrap samples from 1000 as in paper) uncomment line 22 in `genvars.m`.

Monte Carlo exercises

- To reproduce both panels of Figure 5 in the Appendix:

  ```
  >> cd monte_carlo_ylong
  >> genplots_paper
  ```

  These steps use results saved in `noise.mat`, `tfp_freq.mat`, and `truth.mat`, which are included in the directory.

- To reproduce both panels of Figure 6 in the Appendix:

```
>> cd monte_carlo_yshort
>> genplots_paper
```

These steps use results saved in `noise.mat`, `tfp_freq.mat`, and `truth.mat`, which are included in the directory.

- To re-generate results used in the Monte-Carlo exercises, simulate data by calling:

```
>> cd monte_carlo_sim
>> generate_file
>> monte_simulate
```

These steps simulate N=1000 artificial data samples and saves them to `/monte_carlo_yshort/monte_carlo.mat`, `/monte_carlo_ylong/monte_data.mat`, and `/fgls_compare/monte_data.mat`. Because the saved data is large, these files *are not included in the directory* and must be regenerated before proceeding to the next steps.

- To reproduce all calculations involved in making Figure 5 in the Appendix:

```
>> cd monte_carlo_ylong
>> runall
```

This step take about 2 minutes on a quad-core iMac.

- To reproduce all calculations involved in making Figure 6 in the Appendix:

```
>> cd monte_carlo_yshort
>> runall
```

This steps takes about 7 hours on a quad-core iMac, or 2 hours on a 24 core server. To speed it up (at the cost of reducing sample size from 1000 as in paper) uncomment line 12 in `genvars.m`.

Comparison to Forni et al. (2017)

Footnote 14 in the paper refers to a Monte Carlo exercise comparing our approach to identification to the one proposed by Forni et al. (2017). In the exercise, we simulate data from the Blanchard et al. (2013) model, using the parameterization estimated by Chahrour and Jurado (2018). This is the same data generating process we use for the Monte Carlo exercise in Appendix B of our paper.

The Forni et al. (2017) (FGLS) procedure requires specifying a variable that "reveals" the latent signal observed by agents in the model. For this, we assume that the econometrician can directly observe the signal itself. We adopt this strong assumption in order to give their procedure the best possible chance at succeeding in our comparison.

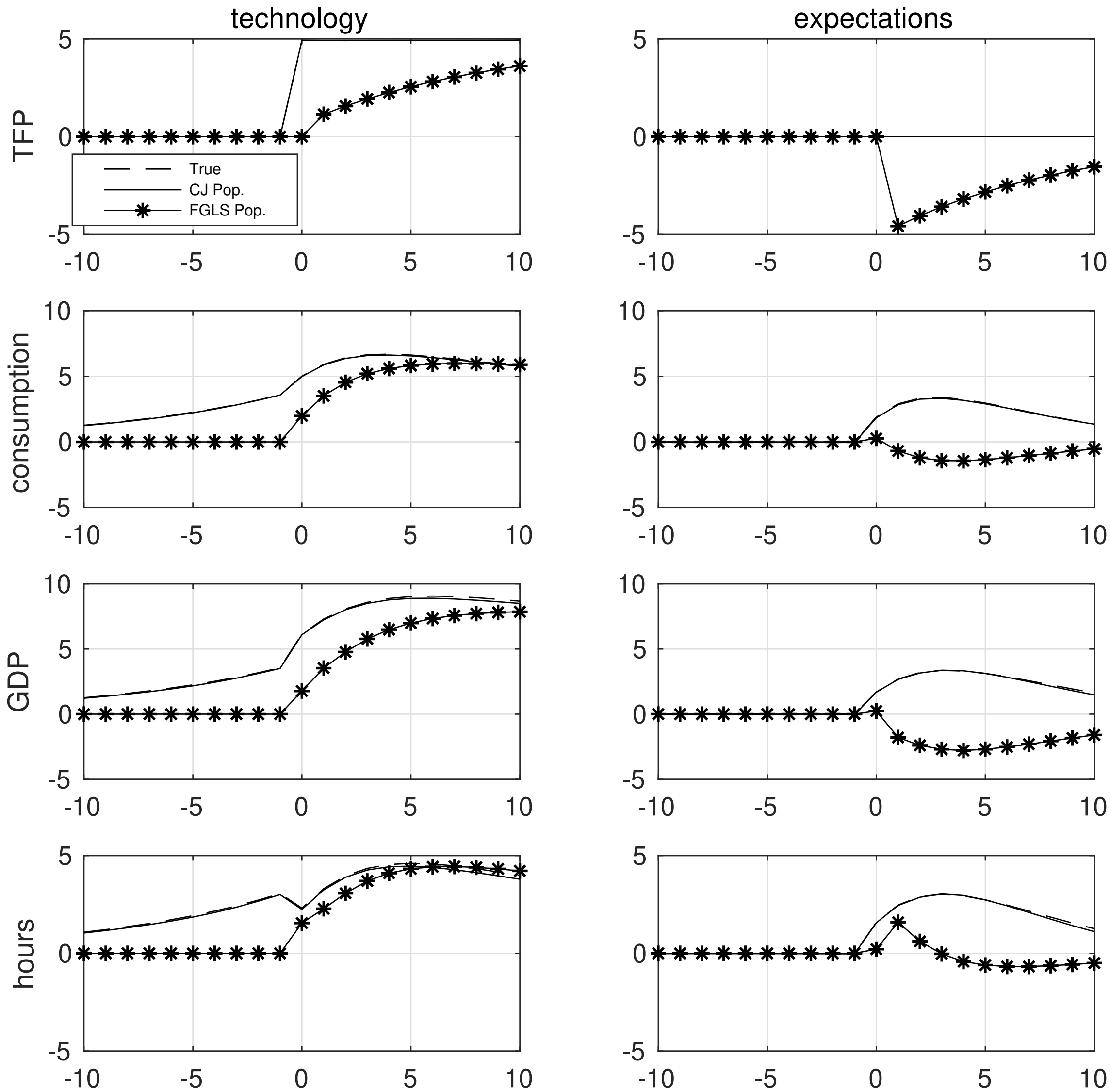

Figure I: Comparing CJ and FGLS procedures in population.

We performed two exercises that correspond directly to the Monte Carlo exercises we present in our Appendix. First, as in Figure (5a) from Appendix B, we compute the impulse responses to the technological and expectational disturbances identified by the two procedures *in population*. The results are presented here in Figure (I).

As described in Appendix B, our procedure (CJ) nearly perfectly identifies the structural responses (the solid and dashed lines lie on top of one another). By contrast, the FGLS procedure delivers responses that are substantially different from the true ones. The figure highlights two especially important differences between our procedures.

First, our procedure explicitly imposes the restriction that the expectational disturbance cannot affect productivity *at any horizon*, while the FGLS procedure does not. With this empirically realistic data generating process, the FGLS procedure estimates a large response of technology to the expectational disturbance; in fact, it is of the same order of magnitude as the response of TFP to a technological disturbance (two panels in the first row). This raises serious questions about the interpretation of this disturbance as a purely expectational disturbance, or "noise."

Second, our approach is better able to capture the long period of gradual anticipation in endogenous variables in response to the technological disturbance. For example, our procedure correctly captures the fact that consumption begins to respond gradually to the technological disturbance at least ten periods before TFP, but the FGLS procedure estimates only a single period of anticipation, in which consumption responds but TFP does not (first and second panels in the left column).

We next perform a similar experiment with 1000 samples of length $T = 284$ periods, as in Figure (6a) of Appendix B. The results are presented here in Figure (II). This figure shows that, according to the FGLS procedure, the true responses to technological disturbances are frequently outside the 90% intervals generated by the Forni et al. (2017b) procedure. Moreover, the corresponding intervals to the expectations shock are quite wide and always contain zero. By contrast, the 90% intervals generated by our procedure always contain the truth.

This exercise demonstrates that, with an empirically-realistic information structure and data generating process, our procedure delivers very different results than the procedure of FGLS.

Performing the FGLS Monte Carlo comparison

- To reproduce Figure I above:

```
>> cd monte_carlo_ylong
>> genplots_fgls
```

  This step uses results saved in `noise.mat`, `tfp_freq.mat`, and `FGLS_compare.mat`, versions of which are already included in the respective directories.

- To reproduce Figure II above:

```
>> cd monte_carlo_yshort
>> genplots_fgls
```

  This step uses results saved in `noise.mat`, `tfp_freq.mat`, and `FGLS_compare.mat`, versions of which are already included in the respective directories.

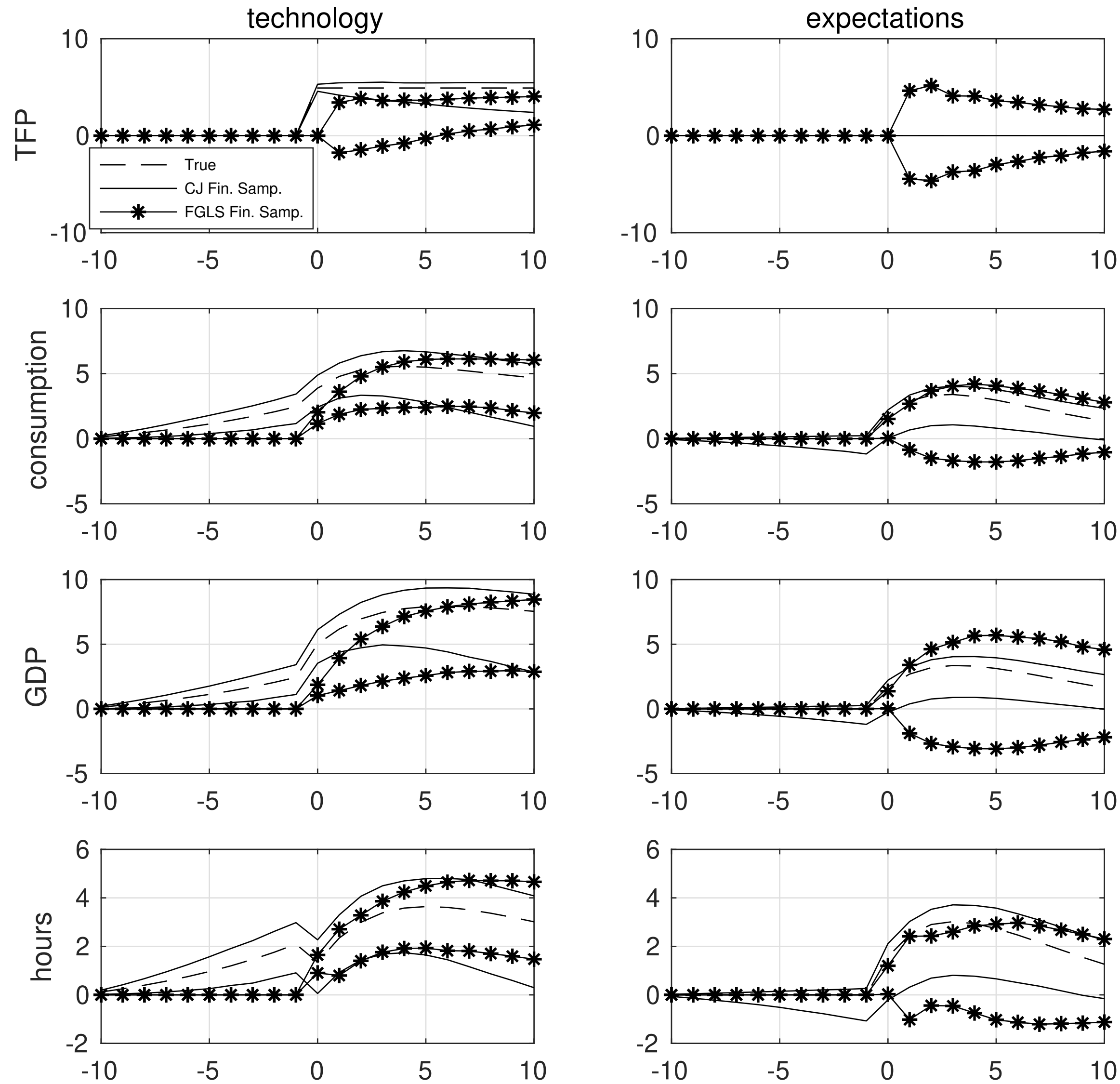

Figure II: Comparing CJ and FGLS procedures in finite sample.

- To re-generate these results from scratch, first complete the steps above to create and save the simulation samples `/fgls_compare/monte_dat.mat`.

  Now, run:

  ```
  >> cd fgls_compare
  >> setup_fgls
  ```

  This last step copies the entire contents of `/fgls_original` (which contains the FGLS replication files downloaded from `https://doi.org/10.3886/E114132V1`) into `/fgls_compare`.

  Next, call:

```
>> gen_monte
```

This step generates the FGLS responses for the same 1000 Monte Carlo samples used in our own Monte Carlo exercise, and saves the results to `/monte_carlo_ylong/FGLS_compare.mat` and `/monte_carlo_yshort/FGLS_compare.mat` for use in Figures I and II.

To restore the `/fgls_compare` directory to its initial state:

```
>> clean_up
```

Now, to reproduce Figure I above:

```
>> cd ../monte_carlo_ylong
>> genplots_fgls
```

Or, to reproduce Figure II above:

```
>> cd ../monte_carlo_yshort
>> genplots_fgls
```

References

Blanchard, O. J., J. L'Huillier, and G. Lorenzoni (2013). News, Noise, and Fluctuations: An Empirical Exploration. *American Economic Review 103*(7), 3045–3070.

Center for Research in Security Prices. CRSP Stock Market Indexes. Wharton Research Data Services, Accessed from https://wrds-www.wharton.upenn.edu/pages/get-data/center-research-security-prices-crsp/ on June 7, 2019.

Chahrour, R. and K. Jurado (2018). News or Noise? The Missing Link. *American Economic Review 108*(7), 1702–36.

Federal Reserve Bank of San Francisco. Total Factor Productivity. Accessed from https://www.frbsf.org/economic-research/indicators-data/total-factor-productivity-tfp/ on June 1, 2019.

Federal Reserve Bank of St. Louis. FRED database. Accessed from https://fred.stlouisfed.org/ on June 4, 2019.

Forni, M., L. Gambetti, M. Lippi, and L. Sala (2017). Noisy News in Business Cycles. *American Economic Journal: Macroeconomics 9*(4), 122–52.