The Heterogeneous Effects of Government Spending: It's All About Taxes
README FILE

Axelle Ferriere* and Gaston Navarro†

January 2024

Below we explain how to replicate all results in the paper. We start describing replication of model results and then move to replication of empirical results. Please let us know if you have comments or questions.

1 Model Results

The `MODEL` folder contains all files needed to replicate the quantitative findings in the paper. The `CODES` folder contains all FORTRAN codes, as well as output of the codes in `.txt` files, and MATLAB files to compute various moments of the calibrations. The `MATLAB_FIGURES` folder reports all MATLAB files to produce the plots, which are stored in the folder titled `FIGURES`.

The `CODES` folder is organized as follows.

- The `BENCHMARK` folder contains the benchmark calibration and results. It contains five subfolders.
  - The `CALIB` computes the calibration of the benchmark economy. In this folder, the `MATLAB` subfolder also contains MATLAB files to compute the distribution of *lpe*, *mpc* and wealth (Tables 1, 2, 3, and 4, and numbers reported in Section 3.3, 3.4), and to compute numbers in Section 4.2.
  - The `CG` folder computes the transition for the "Constant Progressivity" case, while the `MG` folder computes the transition for the "Higher Progressivity" case.

*Paris School of Economics: axelle.ferriere@psemail.eu
†Federal Reserve Board: gaston.m.navarro@frb.gov

- The CG_DECOMP and MG_DECOMP folders produce the results of the direct and indirect effects of the spending shock in Section 5.3. There are three cases: choosing index_decomp on line 60 in the Main.f90 file computes the paths with all equilibrium taxes and prices (case 1), only indirect effects (case 2) and only direct effects (case 3).

The BENCHMARK folder is used to produces Figures 1 to 4.

- The ROBUST folder contains additional quantitative results in the main text.
  - The FLATLPE and FLATMPC folders are used to produce Figure 5 of the paper. In each folder, there are three subfolders, CALIB, CG and MG, to compute the calibration and the transition under the two tax schemes; in the CALIB folder, a MATLAB subfolder also computes distributions of *lpe* and *mpc* (Tables 6, 7, 8 and 9).
  - The NOMCONS and HIGHDEBT folders produce results for the "Accommodative Monetary Policy" and "High Public Debt" cases presented in Figure 6. In each folder, the folders CG and MG compute transitions under the two tax schemes. There is no CALIB file as the steady sate is as in the benchmark.
  - The FLEXWAGES and NEOCLASS folders produce results for the "Flexible Wages" and "Frictionless Economy" cases presented in Figure 7. In each folder, the folders CG and MG compute transitions under the two tax schemes. There is no CALIB file as the steady sate is as in the benchmark.
- The ROBUST_APPEND folder contains additional quantitative results presented in Appendix.
  - The HETB folder contains the (β, B) model presented in Appendix A.6.1. It contains three subfolders, CALIB, CG and MG, to compute the calibration and the transition under the two tax schemes used to produce Figure 16; in the CALIB folder, a MATLAB subfolder also computes distributions of *lpe* and *mpc* (Figure 15).
  - The PROFITMORE and PROFITLESS present models with alternative rules for the distribution of profits. Each folder contains two subfolders, CG and MG, used to produce Figure 17.
  - The remaining five folders are used to produce Figure 18. In particular, STICKWAGES computes in CG and MG transitions with more sticky wages in the benchmark calibration. FLATLPE_FLEXWAGES and FLATMPC_FLEXWAGES computes in CG and MG transitions with flexible prices, while FLATLPE_STICKWAGES and FLATMPC_STICKWAGES computes in CG and MG transitions with more sticky prices; the calibrations for

these two economies are in FLATLPE and FLATMPC in the ROBUST folder described above.

All tolerances for steady states and transitions are at 1e-5, except for the (β, B) model and the "Higher Progressivity" shock with more sticky wages, which were harder to converge, and for which errors are below 3e-5 and 2e-5, respectively. However, multipliers were stable across iterations.

Note that, in each folder, the Command.txt file specifies the command to compile and run the codes, with the options for starting guesses.

Computational requirements and running time. The FORTRAN codes were run on the BigTex High Performance Computing Group at the Federal Reserve Bank of Dallas. To compute Jacobians, we used 12 nodes with 16 cores per nodes. After the Jacobian computation, we compute transitions using 1 node with 16 cores. Codes were compiled with Intel compiler (update 19.0.3.199) using a MKL library. For each exercise, the Jacobian computations takes about 20 minutes; and the transition computation can take between 30 minutes to 6 hours depending on the exercise.

We use MATLAB for plots and simple computations. We use MATLAB R2022b on a computer using macOS 14.2.1. Some of the MATLAB codes use the CompEcon library.[1] Running time for plots is below 20 seconds, and the more demanding computations we do in MATLAB take no more than 5 mins.

2 Empirical Results

The EMPIRICAL folder contains all files needed to replicate empirical findings in the paper. This includes the databases, the STATA files used for regressions, and the MATLAB files used for plots. To simplify replication of our results, we provide separate codes for each figure/table/result whenever possible. We provide a more detailed description of all codes below.

Databases We use five databases in the paper.

- DATA_MACRO_FN.csv contains macro and tax variables used to run all empirical experiments using macro variables at the US level.

  In particular, military news, GDP, GDP deflator, government spending, unemployment, population, 3-month T-bill, and fiscal deficits, are from Ramey and Zubairy (2018); investment and hours are from Ramey (2011); for wages, we use non-farm real hourly compensation from FRED (Federal Reserve Bank of St. Louis).

[1] The Compecon library can be downloaded at: https://github.com/PaulFackler/CompEcon.

Marginal tax rates are borrowed from Barro and Redlick (2011) and Mertens and Olea (2018). To compute average tax rates, we build measures of fiscal revenues using data from the SOI (Internal Revenue Service) and from NIPA (U.S. Bureau of Economic Analysis). Income measures are borrowed from Piketty and Saez (2003). Data on average tax distribution is from Piketty, Saez and Zucman (2018).

See Appendix B.1 and Appendix B.3 for a more detailed description of all variables.

- The state-level tax responses on Appendix C.5 uses the data `DATA_STATE_FN.csv`, also under the folder `EMPIRICAL`. All data comes from NIPA (U.S. Bureau of Economic Analysis). See Appendix C.5 for a description of the variables.

- The multipliers comparison with Ramey and Zubairy (2018) uses the data under folder `RZ_codes`. We obtained these data from Ramey's website (link).

- The wealth distribution computation uses the 1983 wave of the Survey of Consumer Finances (SCF, Board of Governors of the Federal Reserve Board). We added the 1983 SCF wave under the folder `EMPIRICAL`. See Appendix B.2.1 for more details on the variables description.

- Some computations related to the *mpc* use the Survey of Households Income and Wealth (SHIW, Banca d'Italia) database. The original SHIW data is made available by Banca D'Italia. We use the 2010 and 2016 waves, which we added under the folder `EMPIRICAL`. We also use the SHIW data provided by Jappelli and Pistaferri (2020), which we added under the folder `EMPIRICAL`

LPM Results Below we explain how to replicate the local projections in our paper

- The MATLAB code `FIGURE_8_12_21.m` generates Figures 8, 12, and 21 in the paper.

- To replicate Figure 9, first run the STATA code `LPM_FIGURE_9.do` and then run the MATLAB code `FIGURE_9.m` which generates the plot.

- To replicate Figure 10, first run the STATA code `LPM_FIGURE_10.do` and then run the MATLAB code `FIGURE_10.m` which generates the plot.

  - To generate the three plots in Figure 10, `LPM_FIGURE_10.do` has to be run three times changing the corresponding variable in line 40. See STATA code for more instructions.

- To replicate Figure 11, first run the STATA codes `LPM_FIGURE_11_lngov.do` and `LPM_FIGURE_11_deficit.do`, and the run the MATLAB code `FIGURE_11.m` which generates the figure.

- To replicate Figure 13, run the MATLAB code `FIGURE_13.m`. The STATA code `LPM_FIGURE_14.do` needs to be run before, because it generates and stores the shocks that are plotted in Figure 13.

- To replicate Figure 14, first run the STATA code `LPM_FIGURE_14.do` setting `pvar` to *pfed* on line 50. Then run the MATLAB code `FIGURE_14.m` which generates the plot.

- To replicate Figure 19, run the MATLAB code `FIGURE_19.m`. The STATA code `LPM_FIGURE_14.do` needs to be run before, because it generates and stores the shocks that are plotted in Figure 19.

- Most robustness cases in Table 11 and Table 12 are generated by the STATA file `LPM_TABLE_11_12.do`. Select the relevant robustness case on line 42 of the STATA code. Each case outputs an excel file with the numbers in the table. See STATA code for more instructions.

  - The **Expansion & slack** robustness check of Table 11 is generated by the STATA code `LPM_TABLE_11_unemp.do`. It generates an excel file with the numbers in the table.

  - The **Sign** robustness check of Table 11 is generated by the STATA code titled `LPM_TABLE_11_sign.do`. It generates an excel file with the numbers in the table.

- To replicate Figure 20, first run the STATA code `LPM_FIGURE_20.do` and then run the MATLAB code `FIGURE_20.m` which generates the plot.

- To replicate Figure 22, first run the STATA code `LPM_FIGURE_14.do` setting `pvar` to *psoc* on line 50. Then run the MATLAB code `FIGURE_22.m` which generates the plot.

- To replicate Figure 23, first run the STATA code `LPM_FIGURE_23.do` and then run the MATLAB code `FIGURE_23.m` which generates the plot.

  - To generate the three plots in Figure 23, `LPM_FIGURE_23.do` has to be run three times changing the corresponding dates. In particular, set the case on line 34 of the STATA code to set each case. Each case will also set the right time trend. See STATA code for more instructions.

- To replicate Figure 24, first run the STATA code `LPM_FIGURE_24.do` and then run the MATLAB code `FIGURE_24.m` which generates the plot.

  - To generate the two plots in Figure 24, `LPM_FIGURE_24.do` has to be run two times changing the corresponding dates. In particular, set the case on line 38 of the STATA code to set each case. See STATA code for more instructions.

- To replicate Figure 25, first run the STATA code `LPM_FIGURE_25.do` and then run the MATLAB code `FIGURE_25.m` which generates the plot. Run the codes for Figure 10 before, since those results are used to produce Figure 25.

- Figure 26 is produced using several codes, some of which we borrow from Ramey and Zubairy (2018).

  - First, run the STATA code `LPM_FIGURE_26.do`. Run this code twice, for the cases: (1) benchmark, and (2) no MTR. Each case can be selected on line 43. See STATA code for more instructions.

  - Second, run the STATA code `jordagk_twoinstruments.do` under the `RZ_code` folder. This is the code provided by Ramey and Zubairy (2018), to which we only changed the samples dates use for the regressions. Run this code twice: (1) for the 1889-2012 sample (as in Ramey and Zubairy, 2018 sample), and (2) for for 1913-2006 case (as in our sample). Each case can be selected on line 19. See STATA code for more instructions.

  - Finally, run the MATLAB code `FIGURE_26.m` which produces the plot.

- To replicate Figure 27, first run the STATA code `LPM_FIGURE_27.do` and then run the MATLAB code `FIGURE_27.m` which generates the plot.

  - Run the STATA code `LPM_FIGURE_27.do` four times to generate the fours irfs on Figure 27. In particular, set the left-hand-side variable `yvar` to "gdp" or "gov" on line 39, and the shock to `shock` "bp" or "news" on line 42. These four possible combinations give the four irfs on Figure 27.

- To replicate Figure 28 and Figure 29, run the STATA code `LPM_FIGURE_28_29.do` and then run the MATLAB code `FIGURE_28_29.m` which generates the plot.

  - Run the STATA code `LPM_FIGURE_28_29.do` six times to generate the six irfs on Figure 28 and Figure 29. In particular, set the case on line 38 of the STATA code. See STATA code for more instructions.

Micro databases Below we explain how to replicate the results using the SCF and the SHIW databases

- The STATA code `SCF_MAIN.do` computes the wealth distribution reported in Table 2. See Appendix B.2.1 for more details.

- The STATA code `SHIW_MAIN.do` computes the correlations between *mpc* and labor income, as well as *mpc* and total income, which we report in Section 3.4. See Appendix B.2.2 for more details.

Miscellaneous We use two measures of progressivity in the paper: *P[fed]*, based on federal income taxes only, and *P[ss]*, which also includes social security taxes. Appendix B.3.1 explains how to construct the progressivity measures using average marginal tax rates (AMRT) and average tax rates [ATR], while Table 10 reports the numbers. The database `DATA_MACRO_FN.csv` contains the variables needed to compute *P[fed]* (labeled as `MITR` and `AITR`) as well as to compute *P[ss]* (labeled as `MTR` and `ATR`).

Computational requirements We use STATA 15 version and MATLAB R2022b on a computer using macOS 14.2.1. For the instrumental variable estimations on STATA, we use `ivreg2` package we can be installed by typing "`ssc install ivreg2`" on the command window. STATA run time is less than 30 seconds per script.

Additionally, for running the replication codes from Ramey and Zubairy (2018), one further needs to install the packages: `ranktest`, `weakivtest`, `weakiv`, and `avar`. These packages can be installed as: "`ssc install ranktest`", "`ssc install weakivtest`", "`ssc install weakiv`", and "`ssc install avar`".

References

Banca d'Italia (2018). Survey on Household Income and Wealth. https://www.bancaditalia.it/statistiche/tematiche/indagini-famiglie-imprese/bilanci-famiglie/index.html. Accessed on June 2022.

Barro, R. and Redlick, C. (2011). Macroeconomic effects from government purchases and taxes. *The Quarterly Journal of Economics*, **126** (1), 51–102.

Board of Governors of the Federal Reserve Board (2006). Survey of Consumer Finances. https://www.federalreserve.gov/econres/scfindex.htm. Accessed on June 2022.

Federal Reserve Bank of St. Louis (2023). FRED. https://fred.stlouisfed.org/series/COMPRNFB. Accessed on May 2023.

Internal Revenue Service (2022). SOI Tax Stats. https://www.irs.gov/statistics/soi-tax-stats-individual-time-series-statistical-tables. Accessed on March 2022.

Jappelli, T. and Pistaferri, L. (2020). Reported mpc and unobserved heterogeneity. *American Economic Journal: Economic Policy*, **12** (4), 275–97.

Mertens, K. and Olea, J. L. M. (2018). Marginal tax rates and income: New time series evidence*. *The Quarterly Journal of Economics*, p. qjy008.

Piketty, T. and Saez, E. (2003). Income inequality in the united states, 1913-1998. *Quarterly Journal of Economics*, **1** (118), 1–39.

—, — and Zucman, G. (2018). Distributional national accounts: Methods and estimates for the united states. *The Quarterly Journal of Economics*, **133** (2), 553–609.

Ramey, V. A. (2011). Identifying government spending shocks: It's all in the timing. *Quarterly Journal of Economics*, **1** (126), 51–102.

— and Zubairy, S. (2018). Government spending multipliers in good times and in bad: Evidence from u.s. historical data. *Journal of Political Economy*, **126** (2), 850–901.

U.S. Bureau of Economic Analysis (2022a). National Income and Product Accounts, National Data. https://apps.bea.gov/iTable/?ReqID=19. Accessed on March 2022.

U.S. Bureau of Economic Analysis (2022b). National Income and Product Accounts, Regional Data. https://apps.bea.gov/iTable/?ReqID=70. Accessed on April 2022.