Replicating the output for the benchmark specification (uniform truncation)

François LeGrand, Alaïs Martin-Baillon, Xavier Ragot

December, 17 2022

This set of files replicates the graphs and data of *Should monetary policy care about redistribution? Optimal monetary and fiscal policy with heterogeneous agents* in the case of **the main specification** in the case of the uniform truncation (Section 5 of the paper).

Be careful of not mixing up these files with other replicating files and they should be located in an independent folder.

How to run?

In this order:

1. Open `main.ipynb` and execute all cells. This is a `Julia` notebook. Tested on release v1.8.0.
2. Run `main.m` in `Matlab` or `Octave`. Requires Dynare. Tested on Octave v6.2.0 and Dynare 5.1.

Remark. At the very top of `main.m`, the variable `bprint` can be parametrized to display second-order moments. By default, the lattter are not displayed.

The details

The `Julia` files takes care of computing the steady state, while the `main.m` simulates the model in the presence of aggregate shocks.

The output of the `Julia` file is a file `steady_state_dynare.mat` that will used by `Octave` / `Matlab`.

The outputs of the `Octave` / `Matlab` are the `IRFs.png` file plotting IRFs and possibly second-order moments (according to parametrization choice).

The steady state computation

The steady state is computed thanks to six `Julia` notebooks.

- `Main.ipynb`: Solves the steady-state model and returns the truncated model (as `steady_state_dynare.mat` for `Dynare`, saved in the current folder);
- `Structures.ipynb`: Structures and parameter calibration from targets;
- `Utils.ipynb`: Contains some useful functions;
- `SolveAiyagari.ipynb`: Solves the Aiyagari model;
- `Projection.ipynb`: Computes the steady-state truncated model;
- `Weights.ipynb`: Computes the steady-state Lagrange multipliers.

Each of above files are commented and self-explained.

Simulating the model with aggregate shocks

The file `main.m` simulates the model for the three economies of the paper:

- Economy 1: no time-varying tax but optimal inflation;
- Economy 2: no time-varying tax and constant inflation;
- Economy 3: time-varying tax and optimal inflation.

The outcomes of the program can be parametrized as follows:

- The variable `bprint` can be set to `0` for no model details (including no second-order moments) and to `1` for model details. Default value is `0`.

The `Octave` / `Matlab` actually writes three `Dynare` codes `code_dynare_1.mod`, `code_dynare_2.mod`, `code_dynare_3.mod` which correspond to the three economies. Each of this code is then solved in `Dynare`. These files, as interim Dynare files are created in the current folder.

To delete all interim files, you can run the following `bash` commands (be careful with spaces!):

```
$> rm -R +code_dynare_*
$> rm -R code_dynare_*
```