Readme-file for the Replication of the Figures in
"Optimal Corporate Taxation with Financial Frictions"

Review of Economic Studies, 2022
Eduardo Dávila and Benjamin Hébert

The replication folder contains code to replicate all the figures in the paper.

Structure. There is a single main file (*run_all.m*) that generates all the figures and numbers in the body of the paper and the Online Appendix.

- The folder *figures* includes code functions that generate figures.
- The folder *functions* and *functions_aux* includes matlab functions.
- The folder *scripts* includes matlab scripts.
- The folder *output* contains the generated figures.
- The files *dynamic_run.mat*, *dynamic_run_0.mat*, and *dynamic_run_2.mat* contain the data generated by solving the model. Generating these files takes a while; if you only want to change plots or print different results and have already generated these files, comment out the part of *run_all.m* that calls *scripts/main_corporate_tax_dynamic.m*.

How to Run. The *run_all.m* must be run twice to generate everything in the paper. In that file, setting "run_sensitivity=0" will generate everything except two specific numbers mentioned in the paper, involving a re-calibration of the model with a different entry semi-elasticity. Setting "run_sensitivity=1" will generate the two numbers mentioned.

- Note that *run_all.m* will (via *scripts/print_results.m*) print named variables which are the numbers found in the paper (subsection "Quantitative Results") in the command window. In that subsection's order of appearance, the computed values in the paper are called:
    1. *tau_div*
    2. *rel_entry_pt*
    3. *rel_entry*
    4. *entry_mass_ratio*

5. *new_ratio* (expressed as ~1.102 instead of 10.2%)
6. *entry_only_effect*
7. *value_effect*
8. *current_ratio* (expressed as ~1.047 instead of 4.7%)
9. *total_ratio* (expressed as ~1.07 instead of 7%)

- *run_all.m* with "run_sensitivity=1" will print *tau_div_2* and *total_ratio_2* (with the semi-elasticity doubled) and then print *total_ratio_0* and *tau_div_0* (with the semi-elasticity set to zero). Note that the total_ratio variables are expressed as ≈ 1.169 and ≈ 0.986 instead of 16.9% and -1.4%.
- The parameters in Tables A1 and A2 are printed as *cal_params*, *cal_moments*, and *momentTargets*

- The quantitative figures in the paper are generated as follows (all in the *output* folder):
  - Figure 2: *prodfunc.eps*
  - Figure 3: *entry_big.eps*
  - Figure 4: *entry.eps*
  - Figure A1: *laffer.eps*
  - Figure A2: *dynamic.eps*

Run Times. The code was run on Matlab R2020b on a Intel(R) Xeon(R) CPU E5-2687W v4 @ 3.00GHz, with 64Gbs of RAM and Windows 10 Pro. It took about one hour to run main_corporate_tax_dynamic.m once.

The code was also run on Matlab 2022a on a 2021 Macbook Pro (M1 Pro chip) with 16Gbs of RAM. It took about 2.5 hours to run main_corporate_tax_dynamic.m once.

Running *run_all.m* with and without run_sensitivity=1 is therefore expected take between 3 and 7.5 hours on a desktop computer. The code uses parallel methods and may run faster in a many-core research computing environment.

Software. Required toolboxes: Statistics and Machine Learning Toolbox, Parallel Toolbox, Global Optimization Toolbox. Make sure the parallel toolbox is set to automatically create a parallel pool (or start one prior to running the scripts).

The Matlab file exchange utility "breakyaxis" and its license can be found in the MatlabUtilities sub-folder as part of the replication package.

Citation: MikeCF (2022). Break Y Axis (https://www.mathworks.com/matlabcentral/fileexchange/45760-break-y-axis), MATLAB Central File Exchange, accessed August 2022.

The files *tauchen.m* and *disclyap.m* are originally from Elmar Mertens' markovchains.zip file (https://www.elmarmertens.com/lecturenotes), accessed August 2022. They are licensed under the following terms (per that website): "All materials are provided on an "as is" basis. No warranty, but comments are of course welcome. Feel free to use them as long as you give proper reference to the source."