korenmiklos / **Diversification-Through-Trade-Replication**

Caselli, Francesco, Miklós Koren, Milan Lisicky, Silvana Tenreyro. 2019. "Replication Package for Diversification Through Trade." The Quarterly Journal of Economics.

Edit

Manage topics

5 commits | **1** branch | **0** releases | **1** contributor | CC-BY-4.0

Branch: **master** ▾ | New pull request | Create new file | Upload files | Find File | Clone or download ▾

korenmiklos Add data/ | Latest commit `a6400d1` 5 minutes ago

data	Add data/	5 minutes ago
experiments	Initial commit	3 hours ago
notebooks	Move output to output folder	21 minutes ago
output	Move output to output folder	21 minutes ago
.gitignore	Initial commit	3 hours ago
LICENSE.md	Add license	3 hours ago
Makefile	Move output to output folder	21 minutes ago
README.md	Add data/	5 minutes ago
calibrate_params.jl	Initial commit	3 hours ago
calibration_utils.jl	Initial commit	3 hours ago
equilibrium.jl	Initial commit	3 hours ago
figure.jl	Initial commit	3 hours ago
install.jl	Initial commit	3 hours ago
output.jl	Initial commit	3 hours ago
read_data.jl	Initial commit	3 hours ago
save.jl	Initial commit	3 hours ago
scenario_template.jl	Initial commit	3 hours ago
system_test.jl	Initial commit	3 hours ago
table.jl	Initial commit	3 hours ago
unit_test.jl	Initial commit	3 hours ago
utils.jl	Initial commit	3 hours ago

README.md

Replication Data for: 'Diversification Through Trade'

Citation and license

Please cite as

```
Caselli, Francesco; Koren, Miklós; Lisicky, Milan; Tenreyro, Silvana, 2019, "Replication Data for:
'Diversification Through Trade'", The Quarterly Journal of Economics.
```

The software provided here (`.jl` and `.ipynb` files) is licensed under licensed under a Creative Commons Attribution 4.0 International License. To use the data provided in the `data` folder, please check the license terms of the original data vendors.

Data

Data are from the following sources.

- EU Klems Database (March 2008)
- UN National Accounts (2012)
- UNIDO INDTSTAT2
- Penn World Table 7.1
- World Development Indicators (October 2015)
- UN Comtrade (2015)

The core sample of countries include the United States, Mexico, Canada, Australia, China, Japan, South Korea, India, Colombia, the United Kingdom, a composite of France and its overseas departments, Germany, Italy, Spain, Portugal, a composite of Belgium and Luxembourg, the Netherlands, Finland, Sweden, Norway, Denmark, Greece, Austria and Ireland. Other countries are merged as "Rest of the World."

The data are disaggregated into 24 sectors: agriculture (including mining and quarrying), 22 manufacturing sectors, and services, all available in US dollars for the core countries and the Rest of the world (ROW). The 22 manufacturing sectors correspond to the industries numbered 15 to 37 in the ISIC Rev. 3 classification (36 and 37 are bundled together).

The final dataset is obtained by combining different sources and some estimation. The details are described in the Appendix of Caselli, Koren, Lisicky and Tenreyro (2019). The input data is stored, in comma delimited format, in the `data` folder.

Requirements

The code runs Julia 0.6 on Mac OS X and Linux. (We have not tested it on Windows.) The necessary Julia packages are installed by `install.jl` or `make install`.

Workflow

The `Makefile` runs all the necessary computations in the correct order. If you want to reproduce the tables in our paper, run `make` or `make tables`.

Both Julia and Make can run in parallel. Computing the equilibrium for a given set of parameters takes about 1 minute on a single 3.8GHz CPU core. Each period's equilibrium can be computed in parallel (in Julia) and each scenario can be computed in parallel (in Make). With 44 scenarios (see below) and 36 time periods, total run time should be about 1600 minutes. The Makefile is set to run 10 Julia threads in parallel (`PROCS = -p10`). If you have fewer cores, set `PROCS` accordingly. You can then run Make jobs in parallel with `make -j4 tables`.

To run the code in an AWS EC2 instance, follow the steps in `notebooks/aws-recipe.md`. You need to launch an instance from an EC2 image with Julia 0.6 preinstalled.

The core logic is in two files.

- `equilibrium.jl` calculates the equilibrium prices and quantities given a set of parameters.
- `calibrate_params.jl` calibrates parameters to match a set of data moments.

These modules are called by each scenario to be computed. An **experiment** is a set of common parameters applied to our economy *before calibration*. For example, `theta=4` is an experiment and so is `theta=8`. A **scenario** is a particular (often counterfactual) parametrization of an experiment. For example, given calibrated trade costs `kappa`, a scenario might reset those to their 1972 value.

As an example, consider the following two scenarios: `experiments/CES2/actual/scenario.jl` and `experiments/CES2/kappa1972/scenario.jl`. They belong to the same experiment, `experiments/CES2`, which calibrates its parameters *after* setting `sigma=2` in `experiments/CES2/init_parameters.jl`.

The `actual` scenario does not change any of the calibrated parameters. The `kappa1972` scenario replaces all `kappa` with the 1972 values in the same country and sector (see `experiments/CES2/kappa1972/change_parameters.jl`).

The results of each run are saved in `results.jld2` (a JLD2 Julia file format) in the *scenario* folder, such as `experiments/CES2/kappa1972/results.jld2`. Most of our tables require comparisons of four scenarios (with and without trade cost changes, with and without sectoral shocks). These tables are saved in the *experiment* folder, such as `experiments/CES2/output_table.csv`.

Technical details

The calibration algorithm is described in Section III in the paper. The equilibrium solution algorithm is described in Section II.B. The basic outline of the equilibrium solution can be described by four nested loops.

Algorithm

- The **outer loop** solves for equilibrium labor shares, given expected wages. Expectations are taken over `S` possible realizations of future productivity shocks.
- The **adjustment loop** solves for equilibrium deviations from pre-decided labor shares, in response to shocks to productivity. This only runs when labor adjustment costs are finite, otherwise labor shares remain at their preassigned value.
- The **middle loop** solves for the equilibrium sectoral expenditure shares using resource constraints and market clearing conditions.
- The **inner loop** solves for the equilibrium intermediate goods prices across countries, given a *fixed* set of sectoral expenditure shares. The rest of the prices can be computed algebraically.

Data structures

All parameters are stored in the dictionary `parameters`. Parameters can vary by m (destination country), n (source country), j (sector) and t (time). They are stored in a 4-dimensional array with indexes (m,n,j,t). If a certain dimension is not relevant for a parameter, that dimension is retained as a singleton dimension. This is to ensure that arrays are conforming to one another in size.

Similarly, all random variables computed in the equilibrium are stored as 4-dimensional arrays with m, n, j and s (state of the world) as indexes. This ensures conformability of variables and easy formulas such as

```
R_nt = sum(w_njt .* L_njt ./ beta_j, 3)
```

where the summation is across the third, j, dimension.

Output

Tables and figured are saved in the `output` folder.

File	Exhibit	Script
output/Figure2.pdf	Figure 2	notebooks/Create Figure 2.ipynb
output/table1.csv	Table 1	experiments/baseline
output/table2.csv	Table 2	notebooks/Volatility by decade.ipynb
output/table3.csv	Table 3	experiments/trade_imbalance
output/table4left.csv	Table 4, left panel	experiments/theta2
output/table4right.csv	Table 4, right panel	experiments/theta8
output/table5left.csv	Table 5, left panel	experiments/rho0005
output/table5center.csv	Table 5, center panel	experiments/labor_adjustment
output/table5right.csv	Table 5, right panel	experiments/rho002
output/table6left.csv	Table 6, left panel	experiments/CES0.5
output/table6right.csv	Table 6, right panel	experiments/CES1.5
output/table7.csv	Table 7	experiments/no_io_linkages
output/table8left.csv	Table 8, left panel	experiments/no_china

File	Exhibit	Script
output/table8right.csv	Table 8, right panel	experiments/china_1972
output/trade_share.pdf	Supplementary figure	notebooks/Compare scenarios.ipynb

References

- EU KLEMS Database, March 2008. Marcel Timmer, Mary O'Mahony & Bart van Ark, The EU KLEMS Growth and Productivity Accounts: An Overview, University of Groningen & University of Birmingham. http://www.euklems.net/euk08i.shtml
- Julia: A Fresh Approach to Numerical Computing. Jeff Bezanson, Alan Edelman, Stefan Karpinski and Viral B. Shah (2017) SIAM Review, 59: 65–98. doi: 10.1137/141000671. url: https://julialang.org/research/julia-fresh-approach-BEKS.pdf.
- Penn World Table 7.1, 2012. Alan Heston, Robert Summers and Bettina Aten, Penn World Table Version 7.1 Center for International Comparisons of Production, Income and Prices at the University of Pennsylvania, November 2012. https://www.rug.nl/ggdc/productivity/pwt/pwt-releases/pwt-7.1
- UN Comtrade, 2015. "United Nations Commodity Trade Statistics Database." United Nations Staistics Division. https://comtrade.un.org/
- UN National Accounts, 2012. "National Accounts Official Country Data. Table 2.1 Value added by industries at current prices (ISIC Rev. 3)." United Nations Statistics Division. http://data.un.org/Data.aspx?d=SNA&f=group_code%3a201
- UNIDO INDTSTAT 2, 2019. "UNIDO Industrial Statistics Database at the 2-digit level of ISIC (Revision 3)." United Nations Industrial Development Organization. https://www.unido.org/researchers/statistical-databases
- World Development Indicators, October 2015. "World Development Indicators." The World Bank. http://databank.worldbank.org/data/download/archive/WDI_excel_2015_10.zip