

Replication code for

Halpern, László, Miklós Koren and Ádám Szeidl. 2015. "Imported Inputs and Productivity." *American Economic Review*.

Please cite the above paper when using any of these programs.

Instructions

Our data preparation and estimation code use Stata 13 and Python 2.7 and run under Unix-like systems (Max OS X and Linux distributions).

We have included a `Makefile` that runs all the code used in calculating the descriptive statistics, the point estimation of parameter values, the bootstrap, and the figures of simulation results. Unzip all the files, and once you have the dataset in `data/impmanuf.dta`, just run

```
cd code
make all
```

You may need to edit the `Makefile` to call the appropriate Stata executable (`stata-se` in our case).

Workflow

All the data processing and estimation scripts are under the folder `code/`. The other folders are placeholders for the necessary inputs and outputs of these scripts. For example, data is read from `data/` (please see the note on data access below), intermediate simulation results are saved under `doc/simulations`, whereas graphs are saved under `text/graphs`.

The estimation workflow consists of the following broad steps. The `Makefile` contains all the dependencies fully describing the workflow.

1. Descriptive statistics. `descriptives.do`, `import_margins.do`, `numbers-in-text.do` calculate descriptive statistics reported in Tables 1 and 2 and throughout the text.
2. Estimate a given regression specification. Specifications are given in separate do-files. For example, `code/map/table3_baseline.do` contains the necessary settings for the baseline specification in Table 3. These cannot be called by themselves, but are started in Stata by `do estimate_specification table3_baseline 0`. `estimate_specification.do` is the master script, calling several preprocessing (`prepare_for_estimation.do`) and estimation scripts (`estimate_Gn.do`, `run_the_estimation.do`, `removevar.do`). The estimation results are saved in a `.csv` file, for example, `doc/mapreduce/table3_baseline/0.csv`.
3. Bootstrap. Bootstrap repetitions are estimated the same way as the point estimates, but each on a different bootstrap sample. The samples are drawn conditional on a random seed running from 1 through 500. For example, to draw a new random sample with random seed 27, run `do estimate_specification table3_baseline 27`, which will save the estimated parameter values into `doc/mapreduce/table3_baseline/27.csv`.
4. Collect estimation results into tables. `reducer.do` and `sterrors.do` calculate the

- standard errors and other statistics based on the bootstrap, and `write_results_to_excel.py` collects the several specifications into one table.
5. Estimate fixed costs. Given the estimated parameter values, `fixedcost.do` estimates fixed costs and reports various statistics about them.
 6. Calculate the simulated equilibrium in a counterfactual scenario. This is done by `counterfactual.do` and `simulate_entry.do`.
 7. Draw figures based on several counterfactual scenarios. This is done by `collapse-and-draw-graph.do`.
 8. Create production-ready figures. This is done by `all_graphs.do`.

Data access

Our estimates are based on a panel dataset of import transactions, balance sheets and earnings statements of Hungarian firms for the period 1992 to 2003. The “IEHAS-CeFiG Hungary” dataset is described in detail in

Békés, Gábor, Balázs Muraközy, and Péter Harasztosi. 2011. “Firms and Products in International Trade: Evidence from Hungary.” *Economic Systems Research* 35 (1): 4–24.

Because the data is confidential, we cannot make it available in this replication package.

Researchers interested in replicating our results with this same data, or conducting other academic research on this data, can access the dataset and the necessary data processing scripts at the premises of the Institute of Economics of the Hungarian Academy of Sciences. Please refer to <http://www.mtaki.hu/english/> or contact the corresponding author, Miklos Koren at korenm@ceu.edu.