Replication of Geography of Development

Klaus Desmet, Dávid Krisztián Nagy, Esteban Rossi-Hansberg

September 23, 2016

Contents

1 Running the code for the first time

In order to obtain figures and tables in *Geography of Development*, one needs to obtain the relevant variables by running the code as follows (for a thorough description of what each file does and how to run them, see the *Readme* file).

1. Run `instantaneous_trade_costs.m`
2. Run `fast_marching.m`
3. Run `reduce_trmult.m`
4. Run `ownmult_calc.m`
5. Run `gd_a_tau_vect.m`
6. Run `gd_m2_vect.m`
7. Run `main.m`

   NB: The `main.m` code is loading all the resulting files generated by each of the previous steps, which are needed to create the data.
8. Run `distelasticity.m`

These steps are the general procedure. When obtaining counterfactual results, some modifications to this procedure need to be done. See below.

2 Maps

- Figure 1
  These maps are obtained by running `gd_a_tau_vect.m` which plots the series for $\tau_0(r)$, $\bar{a}(r)$ and $\bar{a}(r)/u_0(r)$, whilst `gd_m2.m` generates and plots the vector of migration costs $m_2(r)$.
  Note that they are not saved by the code, one needs to save them manually.

- Figure 2 and 3
  These figures are automatically generated and saved in the folder Maps when running `main.m`. In addition to the figures in the paper, the code will also create these maps for $T = 200$.
  Note that the folder Maps is already included in the Replication file and is empty. It needs to exist and be in the same folder as `main.m` and `maps.m` before running the code.

- Figure 6 and 7
  In order to eliminate migration costs in the model (setting $\vartheta = 0$), one has to modify `model.m` by uncommenting line 40. In `maps.m`, modify the title of the saved files in line 64 (we suggest to simply modify `_1000`).
  Next run `main.m`.
  This procedure is valid for any change of the migration costs over all the cells, which corresponds to simply re-parametrizing ϑ.

3 Tables

- Table 2
  The correlations of log population and population growth at the country level are computed in `main.m` for the years 1995 and 1990 only (line 44 onward). The results are displayed in the command window.
  In order to generate the results for previous decades apply the following procedure:

  1. Load the historical data files for population levels (at the country-level) `popminusX`, where `X` indicates the number of years back from 2000 (20, 30, 40, 50, 87, 130).
     Note that beyond 10 years, the data are only available *at the country level* and are not included in the Replication file.
  2. Modify `main.m` as follows: In lines 46 to 49 and in line 57 replace `popminus5` by `popminusX` (as well as the variable names in lines 57 up to 63 and the text displayed lines 45 and 56).
  3. In lines 46 onward, replace `l_b(:,5)` with `l_b(:,X)`

  NB: Be careful to save the output of the code in a different file each time (modify the title of the file in `main.m` lines 77 to 91).

- Table 3
  This table requires to run the model 8 times. Each time the same modification is required in `model.m`: in line 40 replace the exponent of `m2.^x` where x takes on the values for ϑ given in the table. Run `main.m` after each modification.
  NB: Be careful to save the output of the code in a different file each time (modify the title of the file in `main.m` lines 77 to 91).
  The results are then obtained by computing the percentage changes between the results of each counterfactual with respect to $\vartheta = 1$ (benchmark model).
  Column 1 corresponds to variable `PDV_realgdp_w`, column 2 to `PDV_u_w`, and column 3 to `PDV_u2_w`.

In order to obtain the fourth column of the table, one needs to compute the variable `migr_ctry` (generated in `results.m` and saved in `main.m`) at $t = 1$.

- Table 7
  This table requires to run the model 16 times. For each line of the table, the model needs to be simulated (by running `main.m`) twice : the first time without modifying the `m2.mat` matrix, the second time by uncommenting line 40 in `model.m` and thereby setting $\vartheta = 0$.

  We describe each line in detail:

  - Line 1: benchmark model - the only modification required is the one mentioned above.
  - Line 2: In `initialize.m` (line 71), change `Omega`. Then run the model twice, as detailed above.
  - Line 3: In `initialize.m` (line 69), change `alpha`. Then run the model twice, as detailed above.
  - Line 4: In `model.m`, modify `lambda` in line 19. In order to get the appropriate results thereafter, one needs to run the code twice, as described above.
  - Line 5: In `model.m` modify `gamma2` in line 21, and modify `eta` in line 22 to 1.011. Then run the code twice as described above.
  - Line 6: Spatial diffusion is obtained by modifying equation (8) in the following way : In `model.m`, modify `eta` in line 22 to 0.945; comment line 55 and uncomment lines 56 and 57; and comment line 154 and uncomment line 155. Then run the code as described above.
  - Line 7: In order to modify the trade costs, one needs to recreate the actual `trmult_reduced.mat` matrix by directly modifying `reduce_trmult.m`. This is done by adding `*1.2` at the end of line 12. It then needs to be run again.
    Once modified, run `main.m` as described above.
  - Line 7: In order to modify the trade costs, one needs to recreate the actual `trmult_reduced.mat` matrix by directly modifying `reduce_trmult.m`. This is done by adding `*1.2` at the end of line 12. It then needs to be run again.
    Once modified, run `main.m` as described above.

  NB: Be careful to save the output of the code in a different file each time (modify the title of the file in `main.m` lines 77 to 91).

In a final step one needs to use variables `PDV_realgdp_w`, `PDV_u_w` and `PDV_u2_w` and compute percentage changes between the benchmark result (Line 1 of the table) for $\vartheta = 1$ and the results obtained (with $\vartheta = 1$ and $\vartheta = 0$) to reproduce columns 1, 2 and 3 respectively.

4 Graphs

- Figures 4, 5, 8
  These figures are generated in the code in `plots.m`, but need to be saved manually. In order to be able to compare the three scenarios in each graph, one needs to modify `model.m` in line 40 (uncomment it and adapt the exponent of `m2 = m2^x`), before running `main.m`.
  Finally one can use Matlab's graphic editor to merge the graphs in order to obtain the exact figures from the paper.

- Figures 11, 12, 13
  These figures are generated in the code in `plots.m`, but need to be saved manually. In order to be able to compare the four scenarios in each graph, one will need to run each of the steps 4 to 7 described in Section 1 after the following modifications:

  - In order to change ϑ, uncomment line 40 of `model.m` in each of the subsequent cases.
  - In order to get the counterfactuals of Figure 11, change the value of `Omega` in `initialize.m` (line 71).
  - In order to get the counterfactuals of Figure 12, change the value of `gamma2` in line 21 of `model.m`, and modify `eta` in line 22 to 1.011. Then run the model.
  - In order to get the counterfactuals of Figure 13, in `model.m` modify `eta` in line 22 to 0.945; comment line 55 and uncomment lines 56 and 57; and comment line 154 and uncomment line 155. Then run the code as described above.

  After running `main.m` and saving the individual graphs manually each time, one can use Matlab's graphic editor to merge the graphs in order to obtain the exact figures from the paper.