

segnet

This repository contains the replication files for: Mele, Angelo (2019), “Does school desegregation promote diverse interactions? An equilibrium model of segregation within schools”, American Economic Journal: Economic Policy.

The paper provides an empirical application of the model developed in Mele, Angelo (2017), “A structural model of dense network formation”, Econometrica, 85(3), pp 825-850

Data are from the National Longitudinal Study of Adolescents Health (Add Health). Because of the sensitive nature of the data they are available under contract from UNC Chapel Hill. Researchers interested in the data can apply at <https://www.cpc.unc.edu/projects/addhealth/documentation/restricteduse>

I cannot post the result of the estimations or simulations, because they contain the network data and the information about the students. If you have any problem replicating these results, please contact me at angelo.mele@jhu.edu

REMARK: many of these codes run for a very long time (some for days).

REMARK: all the results in the paper were obtained using a Dell Precision T7610 PC with Intel Xeon CPU E5-2697 v2 @ 2.70GHz with 48 threads and 64GB RAM.

1. extract data

Before running the R scripts, you need to use Stat Transfer to transform the .stc files into .xpt files.

The list of files is

- ICPSR_27022/DS0001/DA27022P1.stc
- ICPSR_27021/DS0001/DA27021P1.stc

Then run the following file to extract the data and save in .csv.

`extract_addhealth.R`

2. create variables and datasets

Need to run the following scripts in sequence.

1. `sample_addhealth.R` : this will collect the sample used in estimation.
2. `datasets_addhealth.R`: this will generate the datasets

This will create the files used for descriptive stats.

The final datasets for the estimation using Rmpi and ergm are obtained by running the following script.

3. `datasets_for_estimation.R`: this creates the files for estimation using parallelization through the package Rmpi.

This will create the files used in estimation. It creates school fractions of each race and edge covariates to include in the formula for estimation. It will generate a set of .Rdata files from the saturated sample of Add Health, which is the one used in the estimation exercise

3. estimation

Rmpi package in r

To run the estimation we need to use MPI in R. Notice that your installation will depend on the OS you are using. My codes were run on a Windows machine, with Windows 7.

Rmpi package installation

Custom network statistics in ergm

The payoff functions for the indirect links are not included in the standard `ergm` package. To add these statistics you need to download the `ergm.userterms` package, designed to include new statistics.

The instruction on how to use this package are here: [Link to vignette in JOSS](#).

Hunter DR, Goodreau SM, Handcock MS (2013). `ergm.userterms`: A Template Package for Extending `statnet`, *Journal of Statistical Software* 52(2), 1-25, URL <http://www.jstatsoft.org/v52/i02/>.

Our function is here: `InitErgmTerm.users.R`

And the corresponding `.c` code is here: `changestats.users.c`

Estimation codes

The results of the estimation are reported in Table 2. This table contains the output of several estimation runs, which are contained in the appendix. The following codes reproduce the results of Table 2, but the complete posterior estimates are reported in Tables 3 to 8 in the appendix.

The codes for estimation are:

Table 4

This corresponds to model (2) in Table 2.

Run 1: `Rmpi_estimation_run1.R`

Run 2: `Rmpi_estimation_run2.R`

Run 3: `Rmpi_estimation_run3.R`

I run the code 3 times, to find a good starting value and to improve the proposal distribution. In the first run I use a random walk proposal with diagonal variance, that is each parameter is independent. This is clearly suboptimal. I use 10000 parameter simulations and 1000 network simulations per parameter. This finds a suitable starting value in few hours. We then use the estimated series of parameters to compute the covariance matrix to input in the random walk for the second run. We repeat for another run, and in the third run we use 100000 simulations and 5000 network simulations.

Table 3

This corresponds to model (1) in table 2.

Run 1: `Rmpi_estimation_dirutil_run1.R`

Run 2: `Rmpi_estimation_dirutil_run2.R`

For this simpler model that does not contain payoffs from link externalities, we only need two runs of the simulations, as it converges quite fast.

Table 5

This corresponds to model (4) in Table 2 in the paper.

Run 1: Rmpi_estimation_newspeg_run1.R
Run 2: Rmpi_estimation_newspeg_run2.R
Run 3: Rmpi_estimation_newspeg_run3.R

Table 6

This corresponds to model (3) in Table 2 in the paper.

Run 1: Rmpi_estimation_newspeg_dirutil_run1.R
Run 2: Rmpi_estimation_newspeg_dirutil_run2.R

For this simpler model that does not contain payoffs from link externalities, we only need two runs of the simulations, as it converges quite fast.

Table 7

This corresponds to model (6) in Table 2 in the paper.

Run 1: Rmpi_estimation_newspeg_allsat_run1.R
Run 2: Rmpi_estimation_newspeg_allsat_run2.R
Run 3: Rmpi_estimation_newspeg_allsat_run3.R

Table 8

This corresponds to model (5) in Table 2 in the paper.

Run 1: Rmpi_estimation_newspeg_allsat_dirutil_run1.R
Run 2: Rmpi_estimation_newspeg_allsat_dirutil_run2.R

4. policy counterfactuals

First I generate the datasets for each policy swap using the following code. For each school, I swap the students of different racial group (or income group) and generate a new school dataset containing the new mix of students.

datasets_for_policy.R

FOR each of the datasets generated by the previous code, I run the **ergm** simulation to generate equilibrium networks implied by the new school composition.

To run the policy counterfactuals I use the following code.

Policy simulations: policy_simulation.R

I then summarize the results of these simulations using standard network and segregation measures, using the following code.

Policy summaries: policy_summary.R

Policy Figures: figures_policy.R