Econ 21410 - Problem Set II

Schelling's segregation and related models*

April 10, 2014

This homework should be done in LaTeX The homework will be graded on correctness, but will also heavily weight clarity and professionalism. Being able to produce clean, clear, well documented write-ups and code is an important skill which will be rewarded. Its better to not do some of the harder parts than to turn in an incomprehensible document. Your R script as well as a log-file should be submitted. Alternatively, use knitr to print your code inline in your latex document.

Make sure to write code which is clear and flexible. Read the whole problem before you begin coding. Some parameters will change and the code should be written in a way to make this easy to implement. We will re-use code in this course. Flexibility and documentation now will save you headaches later in the quarter. Remember to properly indent your code!

SUBMISSION: The homework must be emailed to Oliver and myself by 2p.m. Monday, April the 14th. The email must include a pdf with the filename lastname_pset2.pdf and R code called lastname_pset2_code.R where "lastname" should be replaced with your last name. The subject of your email should be [ECON 21410: pset2 submission]

Remember that asking and answering questions on our github page, coming to office hours to ask questions, and contributing to the class wiki are all worth participation credit, which is 10% of your grade in this class.

1 Control-Flow in R

Complete the following exercise on if/for/while loops in R (use correct indentation!):

1. initialize a variable z to 0
2. write a for-loop that runs from 1 to 1000, call the current iteration i.
3. have the code print out what iteration it is on every 100 iterations
4. in each iteration draw a random value from c(1,2,3)
5. if the drawn random value is equal to 3, set it equal to 1
6. if the drawn random value is equal to 2, set it equal to 1
7. if the drawn random value is equal to 1, add it to z
8. After the loop ends, print the value of z.

*Please email johneric@uchicago.edu and obrowne@uchicago.edu if you have questions.

2 Schelling's Segregation Model

Consider a version of Schelling's Segregation Model implemented in the following way:

1. $\{n_1, ..., n_N\}$ are generated where n^r are red, n^g are green, and n^b are blue.
2. Each individual is initially placed (uniformly) at random on $[0, 1] \times [0, 1]$
3. Start with individual n_1 (who has type $t \in (r, b, g)$) then proceed with the algorithm.
    a) Take individual $n_{i,t}$. If *at least* j_t of their m_t closest neighbors are the same color as them, move on to n_{i+1},
    b) if fewer than j_t of their m_t closest neighbors are the same color as them, randomly draw a new living location and move on to n_{i+1}.
    c) Continue this process until no individuals remain who wish to move.

Machinery for the Schelling Model:

1. write a function that calculates the distances between a coordinate point (x_i, y_i) and a vector of coordinate points $[(x, y)]$. This should return a vector of distances
2. Write a function which simulates Schelling's Segregation model. The function should take the size of each population, j_t and m_t for each type. The function should allow for up to three types.
    - The function should output a plot (or the data necessary to output a plot) of the initial distribution of agents and the final distribution of agents (even better if it outputs some intermediate plots)
    - The function should return the data for the final allocation of individuals.
    - The function should return the number "cycles" the algorithm takes
3. We would like to study the "amount" of segregation in an our "city". To do this, we will write a function which will compute three different segregation metrics. The function should take the simulated final data from your Schelling function and return the three following metrics (focus on the first metric, questions involving the second two indexes will in total be not be worth more than 5% of the grade and are more challenging):
    - Similar neighbors index: For each individual, calculate the proportion of their m_t nearest neighbors that are the same type as them. Take the average of this number across individuals.
    - Dissimilarity index: grid up the $[0, 1] \times [0, 1]$ city into "blocks" of size 0.2×0.2. Using these blocks have the function return the dissimilarity index.[1]
    - Gini index: Using the same grid as above, have the function return the gini index.

[1] Calculating the dissimilarity index and the gini index are a fair amount of additional work. These will be worth substantially fewer points than the similar neighbor index. Make sure you finish the rest of the problem set first.

Output for Schelling

1. Run a "baseline" model with two populations of 250. Let each group care about their 8 nearest neighbors and lets assume members from both groups are "happy" if half of their nearest 8 neighbors are the same color as them.

    - Make a plot of the initial distribution of individuals and the final distribution of individuals.

    - Run your model from a few different starting seeds and see how stable your results are above and discuss (2-5 sentences)

2. Make a plot showing how the number of iterations changes as you increase the populations of the two groups (symmetrically).

3. Make a table showing how run-time increases as you increase the populations of the two groups (symmetrically) (hint, see the command "system.time()").

4. Calculate the similar-neighbor index, discuss (1-4 sentences).

5. Make a plot of how the similar-neighbor index changes as you increase the ratio of nearest neighbors that need to be of the same type for the individual to be happy from .1 to .9.

6. Make a plot showing how the similar-neighbor index changes as you increase the number of individuals in each population from 50 to 500.

7. Make a plot showing how the similar-neighbor index changes based on the number of nearest neighbors considered (from 5 to 30).

Differences between populations: Now lets consider how the model changes when the two populations do not have the same characteristics.

1. Keeping the populations at 250, let the first population be happy if $\frac{6}{8}$ of it's nearest neighbors are of the similar type while the second population is happy if $\frac{3}{8}$ of its nearest neighbors are of similar type.

2. Produce a plot showing the new allocation of individuals.

3. What is the value of the similar-neighbor index? (run the function a few times with different random seeds to see how much this varies). Briefly discuss.

4. Now let there be 500 of the first population, but only 100 of the second population. Again make a plot showing the final spatial allocation and calculate the similar-neighborhood index. How did this change? Discuss in 2-8 sentences.

Extending the model to three populations Now lets evaluate if the model changes when we introduce a third population.

1. Let each population have 150 individuals. Assume they are happy if $\frac{3}{8}$ of their neighbors are the same as them.

2. Produce a plot showing the new final allocation of individuals.

3. What is the similar-neighborhood index? (run the function a few times with different random seeds to see how much this varies). Briefly discuss.

4. Now let there be 500 of the first population, but only 100 in the second and third population. Let the first population be happy if $\frac{9}{12}$ of it's nearest neighbors are of the similar type while the second and third population are happy is happy if $\frac{2}{12}$ of its nearest neighbors are the same. What sort of model or scenario would this correspond to? Run this model several times and look at the final distribution of results. What is the "take away" of this model which we could use to test or inform our work with real data? Discuss (no more than a page).

Alternative Segregation Indexes Take your functions which calculate the dissimilarity-index and gini-index and compare how these vary compared to the similar-neighborhood index.[2] Start by using the baseline model, but also compare the three indexes for some of the alternative two-population models.[3]

3 Code Review

A surprisingly large part of coding is learning from and incorporating code others have already written. In this problem you will download code for a function which "simulates a peer-effects model". Its your job to back out how the model works and answer some general questions about the code.

1. Study the code and give an over-view of how this peer-effects model works. Don't discuss how the code works, but rather, sketch the model I used when implementing this code.

2. Are some people more affected by peers than others in this model?

3. What do the for-statement on line 31 and the while-statement on line 46 do (1-2 sentences each)?

4. Why do I output three different plots in the code? What do they each show?

5. What do k, n, and s do in the code?

6. How does changing s and k affect the code (1-3 sentences)?

7. Reuse you similar-neighbor index function from above to calculate the degree of segregation in the baseline model I run in my .R file. How does this model's level of segregation compare to our baseline Schelling model?

[2] As mentioned above, this portion of the homework will be worth notably fewer points than the other sections, so do this last, and do not worry too much if you are not able to get your alternative functions to work.

[3] This is purposefully unstructured (like most real work). Figure out what is interesting about the differences and discuss.

Research

- Suggest one research idea based around the models we considered in this homework (no more than 3 sentences).
- List three topics you may be interested in doing research on. These can be broad, such as "The returns to community college", or very narrow, such as "Advertising for for the Xbox One and the PlayStation 4". As undergraduates, its helpful to stick to things that really interest you or things you really care about (anything going on in your home state?).[4]
- If you are a 4th year who wrote a BA which you want to extend for this course, write a 3 sentence summary of your BA, then write 1-5 sentences on how you may extend it for this course.[5]

Side Projects

1. Download census tract data on race for the city of Chicago. Calculate the gini and dissimilarity indexes using your function. If possible make a map in R of these results. Write up no more than 1 page (not including figures) discussing your results (3 points)
2. Rewrite some or all of the code for this problem in Julia or C++ with Rcpp[6] (up to 3 points)
3. write a wiki entry with the examples for any of the following R commands: "order()", "which()", "apply()", "sapply()/lapply()" (u p to 1 point, no more than 1 per person).
4. write a wiki entry about how to time your code (up to 1 point).
5. write a wiki entry on ggplot2 (and the wrapper qplot). This is an extremely powerful plotting tool. Please provide examples and their corresponding plots. (up to 1 point, but multiple people can contribute and extend with different examples, different plot types, etc)
6. Build a shiny app (http://www.rstudio.com/shiny/) which lets users configure and run the Schelling model (or the grid Schelling Model) using a graphical user interface (up to 3 points).
7. Use knitr to add animated plots to your pdfs (animations only work in newer versions of Adobe products) (1.5 points)
8. Read the Thomas Schelling's 1971 and/or 1969 paper and write a short (2-5 page) summary and review of his paper(s). If you read both papers, discuss the difference between the two (up to 3 points). (http://www.tandfonline.com/doi/pdf/10.1080/0022250X.1971.9989794) (http://www.jstor.org/stable/pdfplus/1823701.pdf)

[4] For example, as an undergraduate I could have maybe have tried to write a hedonic pricing model for resell of high-end acoustic guitars (maybe scraping the data off of sales sites), or I could have written a paper on how Alaska's economy is counter-cyclical with rest of the Nation and the resulting impact of United States' monetary policy on Alaska's economy.

[5] Broad ideas are fine. If you don't have a good answer, come talk to us at office hours sometime in the next week or two

[6] If you tackle Rcpp send me an email and I can try to help you get started. You will most likely want to use RcppArmadillo, which links to the fantastic Armadillo C++ library which has syntax somewhat similar to matlab

Useful Resources

www.census.gov/hhes/www/housing/housing_patterns/pdf/app_b.pdf