# The Schelling Segregation Model

Simulating human behavior can be complicated. People have individual needs, wants and preferences that form the goals they pursue throughout their lives. People do not pursue their goals in isolation; they compete and cooperate with other people in their families, work, school and community.

Attempts to simplify the problem of representing human behavior by using aggregate mathematical equations or statistical populations of individuals have been shown to miss important properties of real world behavior. 1,2,3 The problem of building a model of human behavior has to encompass both the individual decisions and the environment which is created by and reacts to those decisions.

One of the earliest researchers to tackle the problem of systematically modeling the interaction of individual choices with a larger environment is the Nobel Prize winning economist Thomas C. Shelling. Schelling has a number of interesting accomplishments to his name that are worth viewing on Wikipedia, but what interests us here is his segregation model.

Schelling built his original segregation model to understand the problem of racial segregation in the United States 4, but he later expanded the model to cover other types of segregation including gender, language, age and income5.

Schelling’s segregation model beautifully illustrates how to simplify individual choice and environmental interaction while preserving the fundamental properties of each. The Schelling segregation model is simple, but the results can be surprising. Schelling showed that a small preference for similar neighbors can lead to a highly segregated neighborhood.

A Schelling model consists of people or households in a neighborhood. Each person has a type (white, black, brown, etc.) and a preference for what kind of neighbors would make them happy. A neighborhood is an arrangement of lots that can either be empty or contain people. Schelling experimented with both one and two dimensional neighborhoods. A person will be happy if their immediate neighbors meet their preference or they will be unhappy and want to move. Preferences only come into play when a person has at least one neighbor. A person surrounded by empty lots is assumed to be happy.

|  |  |  |
| --- | --- | --- |
| O | X | O |
| O | X | O |
| X | O | O |

Figure 1 Sample Neighborhood

For the example neighborhood in (**Figure 1 Sample Neighborhood**) composed of X’s and O’s the center X has eight neighbors with two X’s and six O’s. If the center X has a preference for having neighbors like itself of .25 or less it will be happy otherwise it will want to move.

The cycle of the Schelling model is as follows:

1. People are laid out on a neighborhood grid.
2. The unhappy people are identified.
3. Unhappy people try to move to a different place in the neighborhood.
4. Repeat the cycle as desired.

An implementation of a Schelling segregation model is provided in the file SchellingSegregationModel.py. The SchellingSegregationModel code can handle modeling segregation on discrete attributes like race, gender, and religion or on continuous attributes like age and income. The provided model assumes that unhappy people can move to any empty lot or swap places with any other unhappy person.

A critical part of the model is the Neighborhood class which contains the grid of lots in the neighborhood. The neighborhood grid forms a torus, or doughnut shape, meaning that the grid wraps around its edges. The lots may be filled with EmptyLot classes or they may be filled with implementations of the SchellingAgent class representing people. The move() function on the Neighborhood class contains the code that combines the empty lots and unhappy people and initiates a random swap of unhappy pairs.

The people in the model all descend from the SchellingAgent class. The base SchellingAgent is designed to represent discrimination on discrete variables like race or gender. The LikesSameAgent class represents a person who wants a minimal level of similar neighbors with a discrete attribute like race. The LikesOthersAgent likes a neighborhood with a minimal level of neighbors with a discrete attribute that is not the same as itself. The ContinuousSchellingAgent class extends the SchellingAgent class to add support for calculating similarity for continuous attributes like age and income. The assumption for continuous variables is that similarity does not need to be exact, close will do. The ContinuousLikesSameAgent and ContinuousLikesOtherAgent are the continuous attribute equivalents of the LikesSameAgent and LikesOtherAgent classes.

A quick introduction to the model code is provided in the function demo(). By default demo builds a segregation model between X’s and O’s with a preference for similar neighbors in a grid of 400 squares. The demo people would like to have .3 of their neighbor the same as themselves.

|  |
| --- |
| **>>> import SchellingSegregationModel**  **>>> demo()**  **[(0, (0.192, 0.4881)), (1, (0.0774, 0.6012)), (2, (0.0458, 0.6482)), (3, (0.0287, 0.676)), (4, (0.0143, 0.696)), (5, (0.0115, 0.705)), (6, (0.0115, 0.7123)), (7, (0.0, 0.719))]** |

Example 1 SchellingSegregationModel demo

The demo() function code returns statistics for each turn on neighborhood unhappiness and neighborhood similarity. So on turn 0 of the code run in **Example 1 SchellingSegregationModel demo** the neighborhood was 19% unhappy and the aggregate similarity was 49%. By the end of the run (in only seven turns!) the neighborhood unhappiness was at 0% and the local similarity was at 72%.

Notes:

1.

2.

3.

4.

5.

## References

1. Micromotives and Macrobehavior, Thomas C. Schelling, W.W. Norton and Co. 2006
2. Simulation for the Social Scientist, Nigel Gilbert, Klaus G. Troitzsch, Open University Press, 2009
3. “Models of Segregation”, Thomas C. Schelling, The American Economic Review, Vol. 59 No. 2, May 1969[[1]](#footnote-1)

1. [↑](#footnote-ref-1)