

DOID 325: Thinking with Models

Spring 2016, TBA

Teaching Notes (and Class Record)

Steven O. Kimbrough

Draft: © November 5, 2016

Contents

Preface	ix
1 Class 1	1
2 Class 2	3
3 Class 3	5
4 Class 4	7
5 Class 8: W&R, chapter 3, Segregation	9
5.1 Teaching Notes (about 20 minutes)	9
5.2 Exercises (about 40 minutes)	10
5.3 Class Discussion (about 20 minutes)	10
6 Class 9: W&R, chapter 3, El Farol	11
6.1 Teaching Notes (about 20 minutes)	11
6.2 Exercises (about 40 minutes)	11
6.3 Class Discussion (about 20 minutes)	12

List of Figures

List of Tables

Preface

Chapter 1

Class 1

Chapter 2

Class 2

Chapter 3

Class 3

Chapter 4

Class 4

Chapter 5

Class 8: W&R, chapter 3, Segregation

5.1 Teaching Notes (about 20 minutes)

Focus now on some programming techniques.

1. Page 132: `n-of number patches then sprout`.
`n-of <agentset>` is a general approach to obtaining an agentset of size `number` from another agentset. The returned agentset has no duplicates.
`sprout` is a command to a patch, causing it to create turtles on top if. Works for breeds of turtles, too.
2. Page 133: `count (turtles-on neighbors)`. Go through this. We are in a turtle context... `(turtles-on neighbors)` returns an agentset of turtles with the property that they are on the `neighbors` of the current turtle. Nice, elegant construction.
3. Page 133: `with qualities` an agentset and can be arbitrarily complex. Note the syntax. Can have arbitrary boolean combinations inside `[...]`.
4. The book discusses `update-turtles` but not the enclosing procedure `update-variables`. Important to see how this works: turtles are moved individually; once all the turtles are moved, the updating happens. Discuss: Why is this necessary? This is why we need a `happy?` attribute.

5. Page 135: `set color (item (random number-of-ethnicities) colors)`. OK, explain how it works. Now, what if we want the ethnicities not to be equi-probable? What can we do?
6. Page 136: `random %-similar-wanted`. Discuss how this works. Note it's a bit odd. What about using `random-normal`? How would that work? Discuss other probability distributions supported by NetLogo and what they might be good for.

5.2 Exercises (about 40 minutes)

Form groups of 3 (2 or 4 as necessary). Discuss and write down and load to Canvas responses to the following.

1. Add agents that don't care, that is, are happy with no neighbors of their own color. Have them move occasionally. Develop a slider for the number of don't care agents of each color.

Explore the resulting model. How does it behave? Compare with the original model and the extensions from the book.
2. Percent similar is a key measure of performance (MoP) of the system as modeled. It only looks at immediate neighborhoods. Use `in-radius` to develop a more flexible MoP that looks at neighborhoods of arbitrary size (to be indicated by a slider). Be careful to count only `other turtles`.

5.3 Class Discussion (about 20 minutes)

Discuss the exercises with the entire class. What did people do?

To the extent there is time: Discuss what would be needed to get a reasonably accurate and well calibrated segregation model.

Chapter 6

Class 9: W&R, chapter 3, El Farol

6.1 Teaching Notes (about 20 minutes)

Focus now on some programming techniques and on the El Farol model itself.

1. Do a walkthrough of the El Farol code, explaining how it works.
2. Pages 143–4: Discuss `turtles-own` (again) and step through the handling of `reward`.
3. Page 145: Discuss `scale-color` and how it works.
4. Work through histograms carefully and explicitly.

6.2 Exercises (about 40 minutes)

Form groups of 3 (2 or 4 as necessary). Discuss and write down and load to Canvas responses to the following.

1. Write code to report the three best strategies and the three worst strategies, so that we can look at them. What do you find?
2. Do exercise 32 on page 155 of the book.
3. Implement a win-stay/lose-shift strategy and put it into the model.

6.3 Class Discussion (about 20 minutes)

Discuss the exercises with the entire class. What did people do?