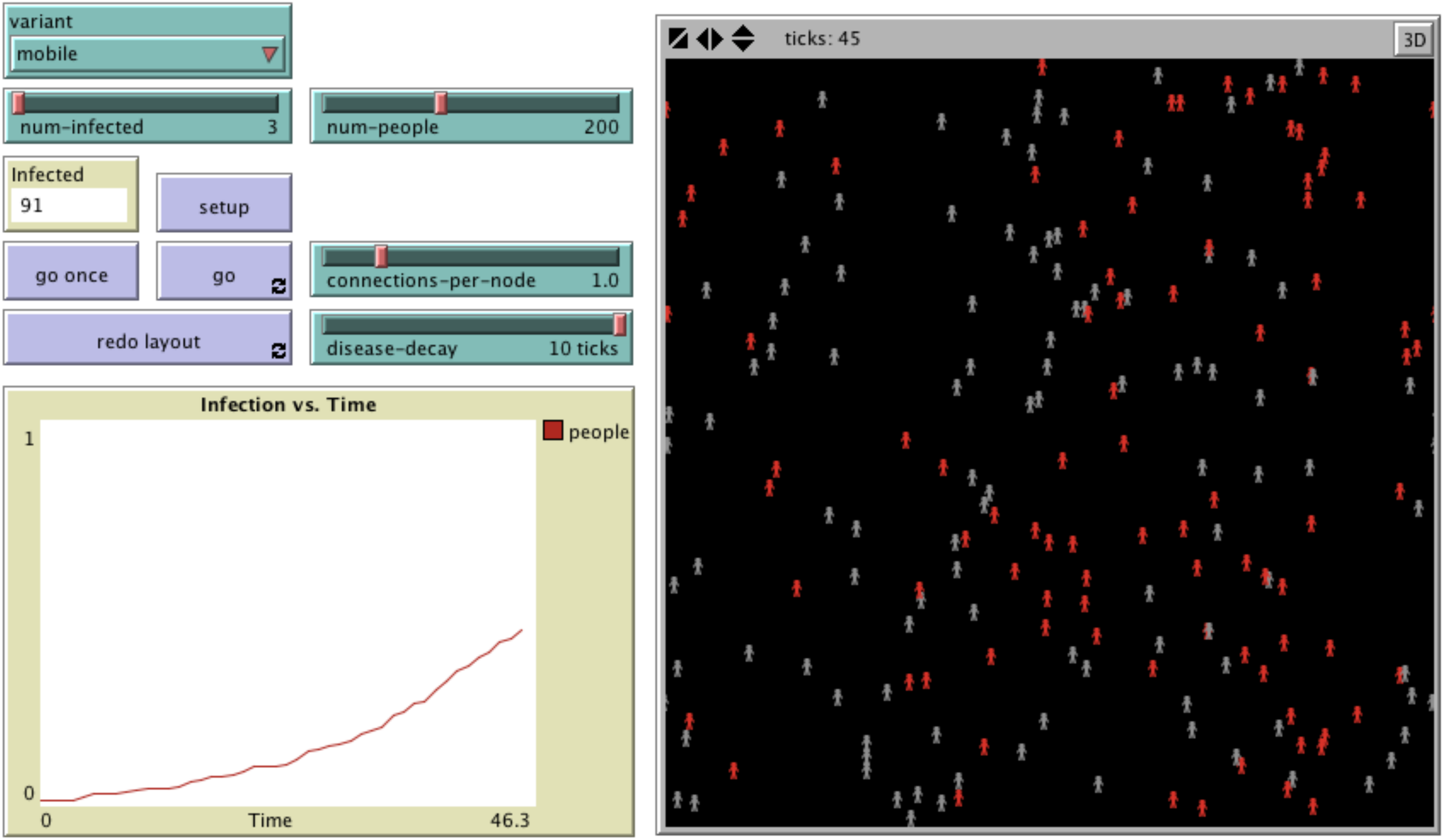
Analyzing ABMs

Bill Rand
Associate Professor of Business Management
Poole College of Management
North Carolina State University

Now that I have an Agent-Based Model how can I use it?

The Spread of Disease

- If a disease infects 5 people before the original person recovers (r0 = 5), then in three steps of the infection, the disease will infect 156 people (fast growth).
- On the other hand, the number of possible infections will quickly tail off because the disease may run out of new hosts (resource limits).
- Other Complications
  - No truly uniform mixing
  - Different susceptibilities to disease
  - Environmental interactions



variant: mobile

num-infected 3

num-people 200

Infected 91

connections-per-node 1.0

disease-decay 10 ticks

ticks: 45

Spread of Disease

An Initial Investigation

- What if we decide to investigate how population density affects the spread of disease?
- A couple of researchers decide to run experiments to investigate this
- We examine how population density affects the time to 100% infection

Different Results

First Run

Population	50	100	150	200
Time to 100% Infection	419	188	169	127

Second Run

Population	50	100	150	200
Time to 100% Infection	305	263	118	126

Maybe we need to do more runs?

Raw Data

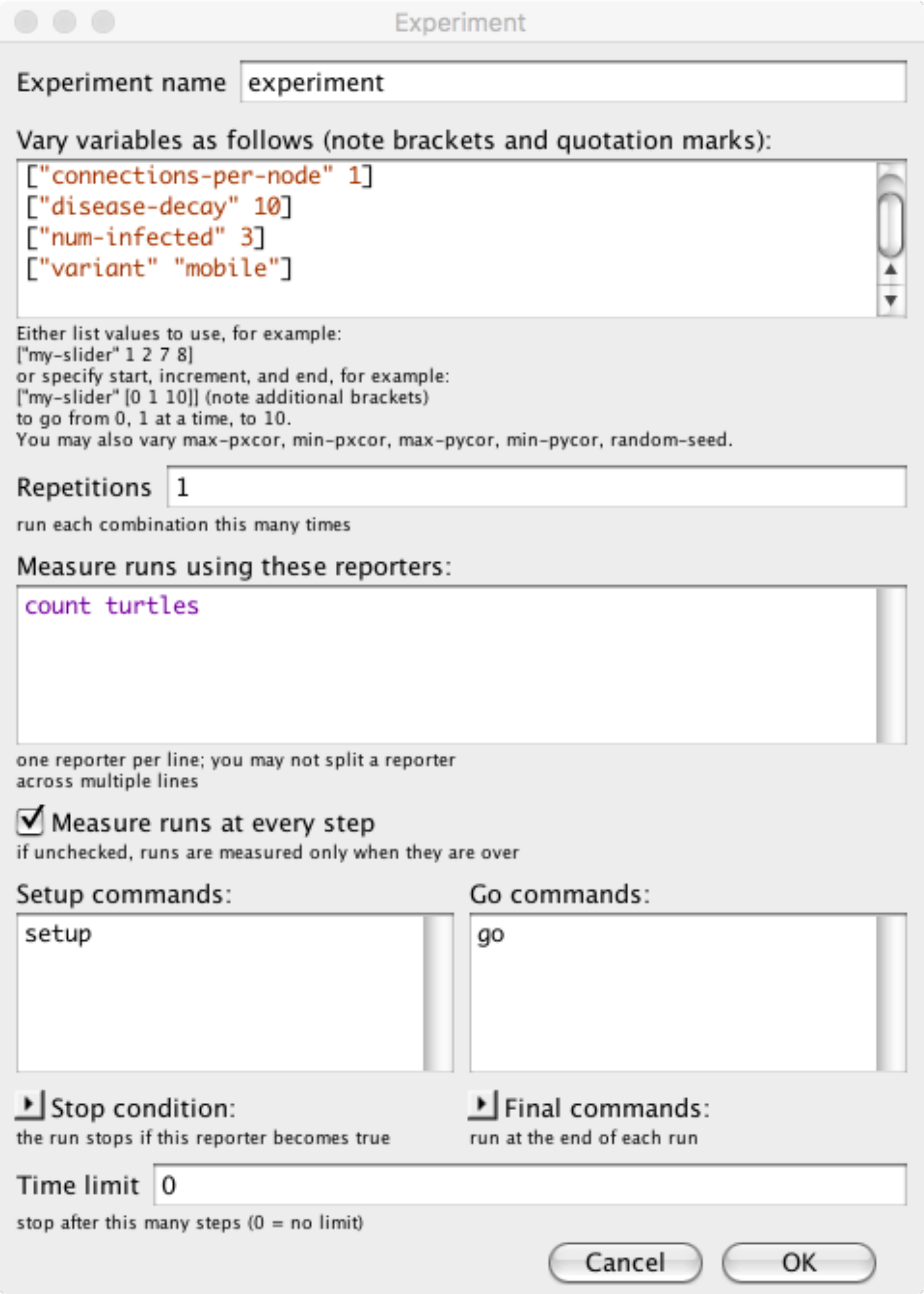
Population	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10
50	419	365	305	318	323	337	432	380	430	359
100	188	263	256	205	206	205	201	181	202	231
150	169	118	163	146	143	167	137	121	140	140
200	127	126	113	111	133	129	109	101	105	133

The Necessity of Multiple Runs

- Since many of these models are stochastic, we need to run the model several times to understand the behavior
- There are several different ways to execute multiple runs in NetLogo or in any ABM platform

Within the Model

```
repeat 10 [
  set num-people 50
  setup
  while [ count turtles with [ not infected? ] > 0 ]
    [ go ]
  print ticks
]
```



Experiment name: experiment

Vary variables as follows (note brackets and quotation marks):

```
["connections-per-node" 1]
["disease-decay" 10]
["num-infected" 3]
["variant" "mobile"]
```

Either list values to use, for example:
["my-slider" 1 2 7 8]
or specify start, increment, and end, for example:
["my-slider" [0 1 10]] (note additional brackets)
to go from 0, 1 at a time, to 10.
You may also vary max-pxcor, min-pxcor, max-pycor, min-pycor, random-seed.

Repetitions: 1

run each combination this many times

Measure runs using these reporters:

```
count turtles
```

one reporter per line; you may not split a reporter
across multiple lines

☑ Measure runs at every step

if unchecked, runs are measured only when they are over

Setup commands:

```
setup
```

Go commands:

```
go
```

Stop condition:

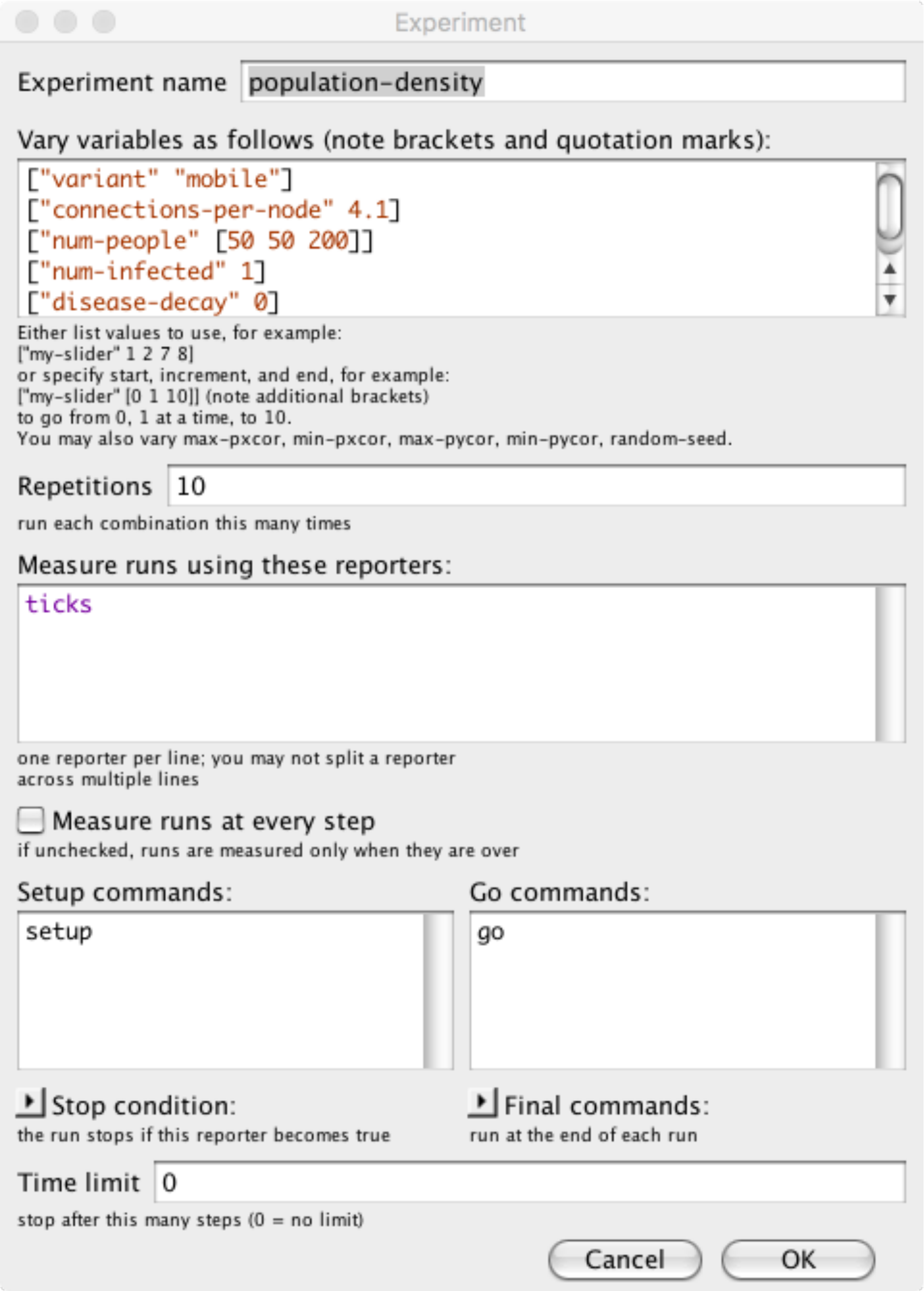
the run stops if this reporter becomes true

Final commands:

run at the end of each run

Time limit: 0

stop after this many steps (0 = no limit)



Experiment name: population-density

Vary variables as follows (note brackets and quotation marks):

```
["variant" "mobile"]
["connections-per-node" 4.1]
["num-people" [50 50 200]]
["num-infected" 1]
["disease-decay" 0]
```

Either list values to use, for example:
["my-slider" 1 2 7 8]
or specify start, increment, and end, for example:
["my-slider" [0 1 10]] (note additional brackets)
to go from 0, 1 at a time, to 10.
You may also vary max-pxcor, min-pxcor, max-pycor, min-pycor, random-seed.

Repetitions: 10

run each combination this many times

Measure runs using these reporters:

```
ticks
```

one reporter per line; you may not split a reporter
across multiple lines

□ Measure runs at every step

if unchecked, runs are measured only when they are over

Setup commands:

```
setup
```

Go commands:

```
go
```

Stop condition:

the run stops if this reporter becomes true

Final commands:

run at the end of each run

Time limit: 0

stop after this many steps (0 = no limit)

Raw Data

BehaviorSpace results (NetLogo 5.3.1)						
Spread of Disease.nlogo						
population-density						
09/26/2016 15:07:32:848 -0400						
min-pxcor	max-pxcor	min-pycor	max-pycor			
-20	20	-20	20			
[run number]	variant	connections-per-node	num-people	num-infected	disease-decay	[step]
2	"mobile"	4.1	50	1	0	
7	"mobile"	4.1	50	1	0	
5	"mobile"	4.1	50	1	0	
8	"mobile"	4.1	50	1	0	
6	"mobile"	4.1	50	1	0	
1	"mobile"	4.1	50	1	0	
4	"mobile"	4.1	50	1	0	
11	"mobile"	4.1	100	1	0	
10	"mobile"	4.1	50	1	0	
3	"mobile"	4.1	50	1	0	

Summary Statistics

Summary statistics provide us with a way of characterizing a large data set.

Easy to do in R.

Population	Mean	Std. Dev.
50	366.8	47.39385802
100	213.8	27.40154091
150	144.4	17.65219533
200	118.7	12.12939497

Mean

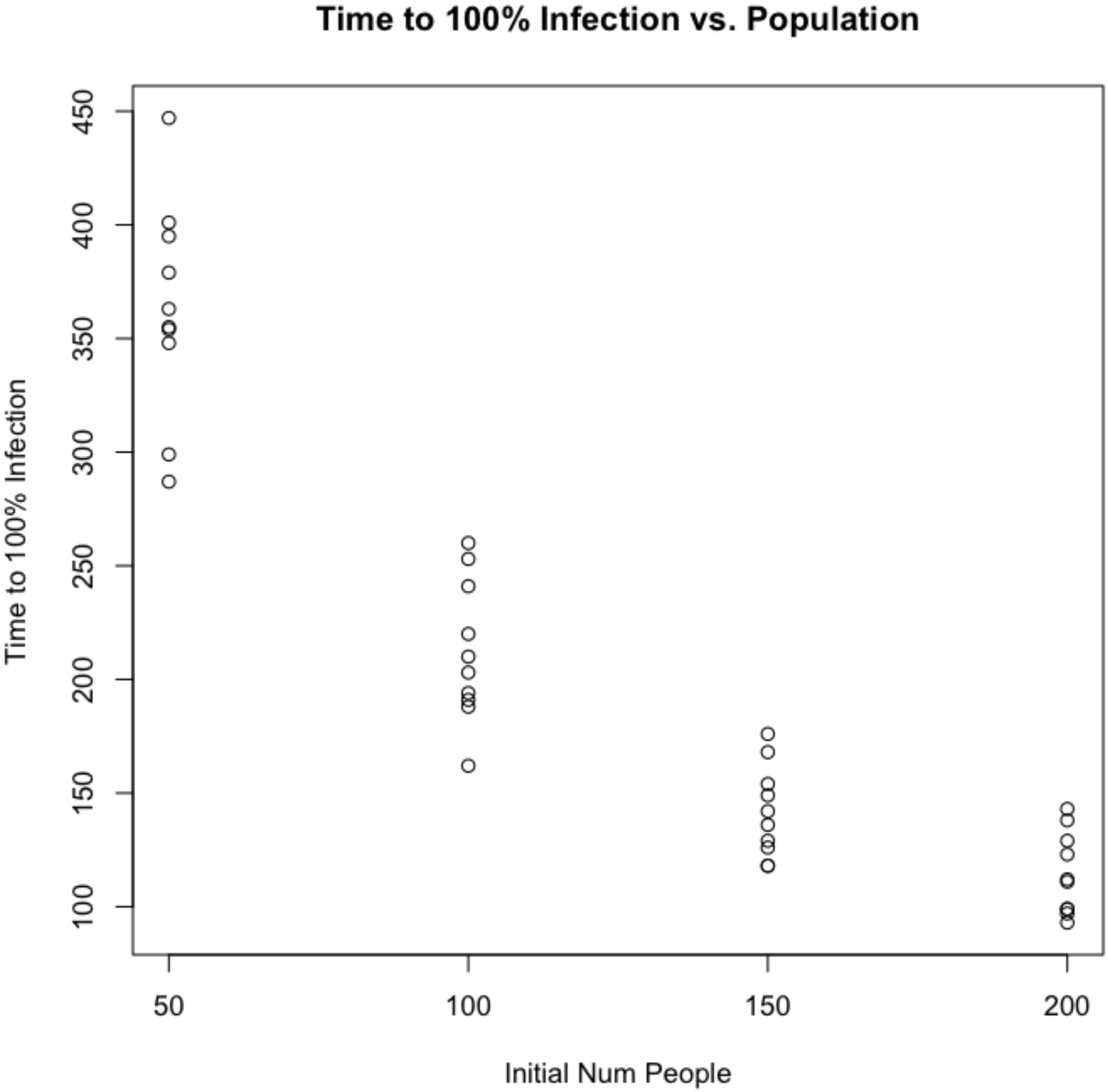
$$\mu = \frac{x_1 + x_1 + \cdots + x_n}{N}$$

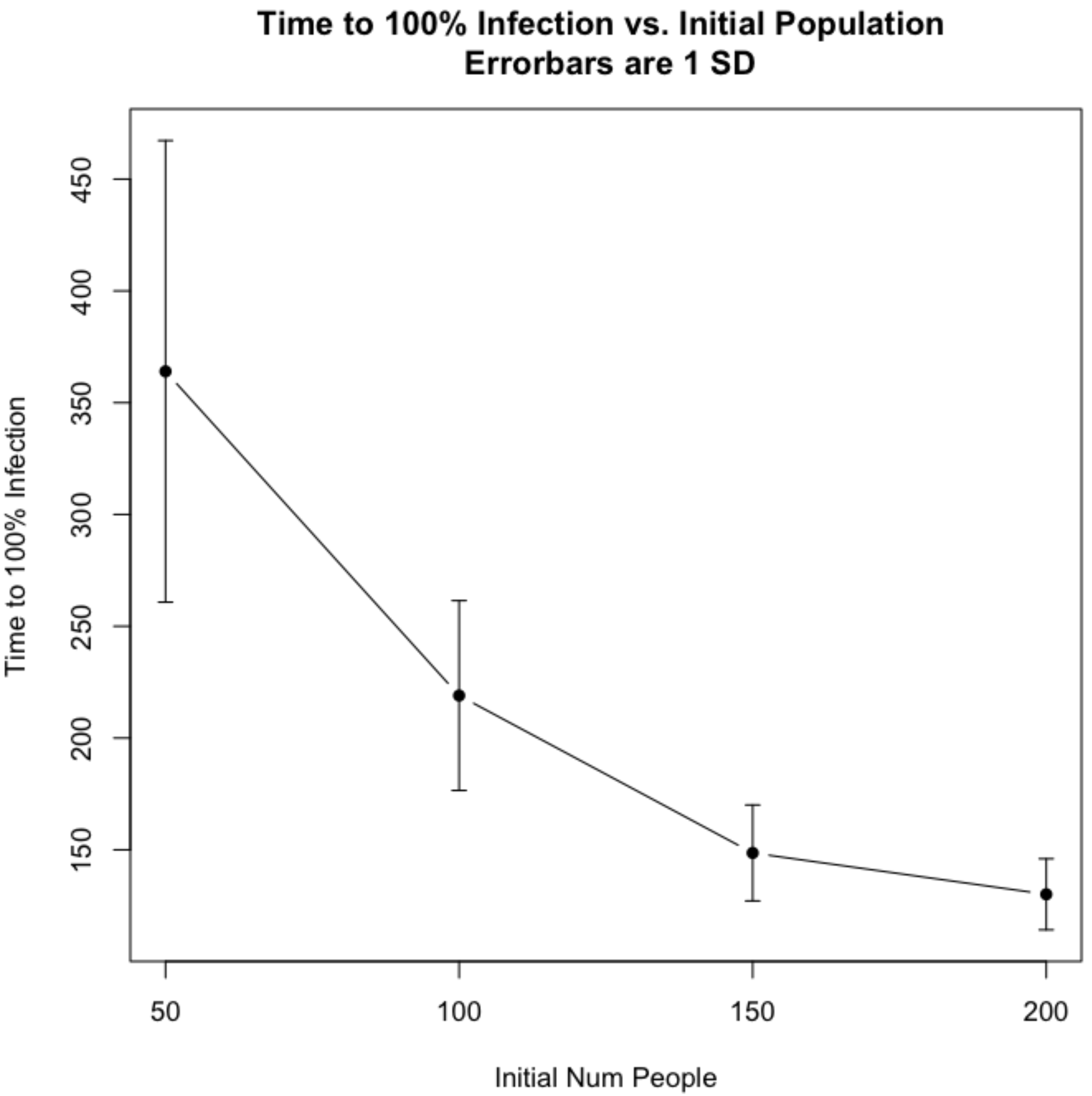
Standard Deviation

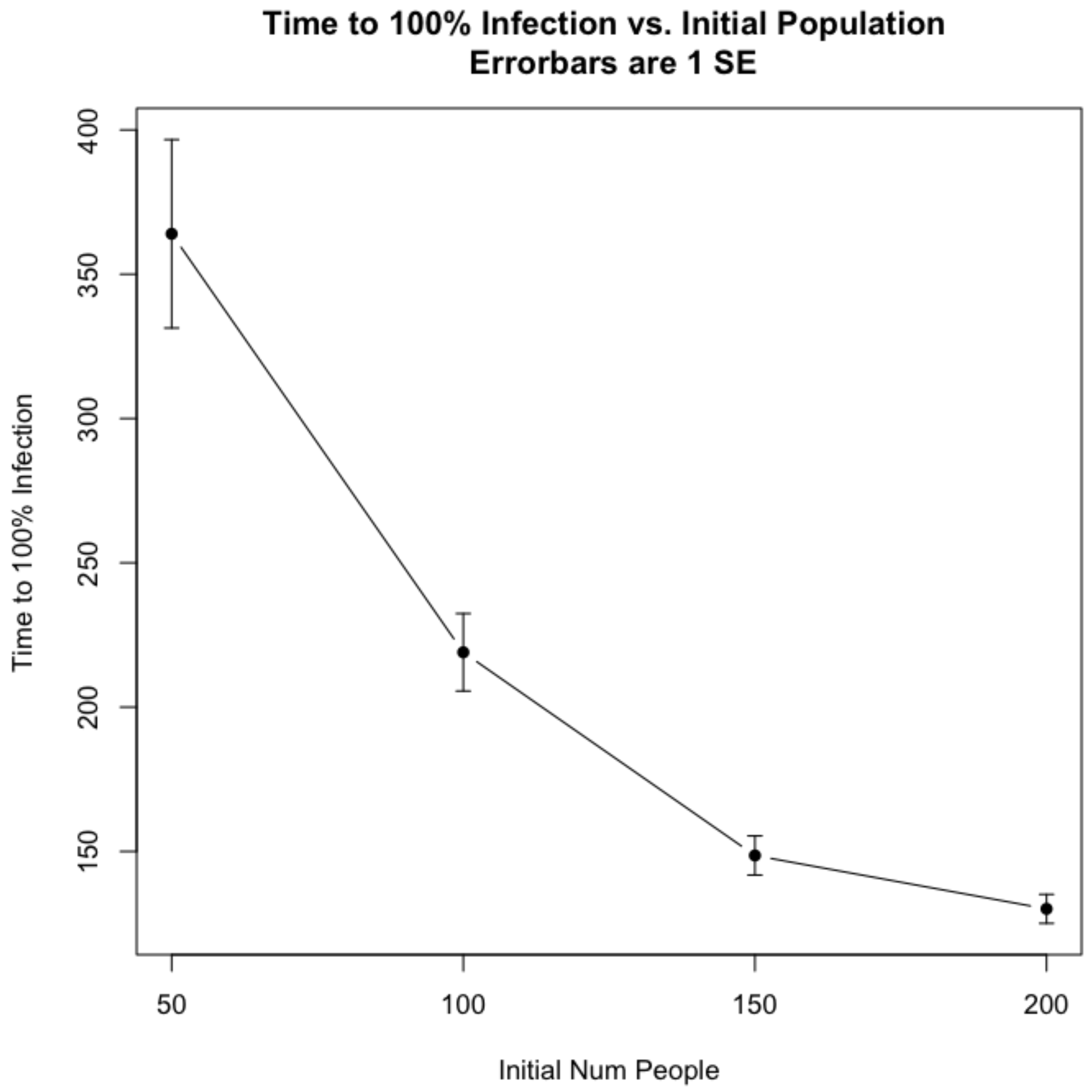
$$\sigma = \sqrt{\frac{1}{N}[(x_1 - \mu)^2 + (x_2 - \mu)^2 + \cdots + (x_n - \mu)^2]}$$

Creating Graphs

- Graphs provide a visual way to summarize the data
- They are not better or worse than summary statistics just different
- Many different graphs have been created
- Always useful to start by looking at all data
- Again, easy to do in R

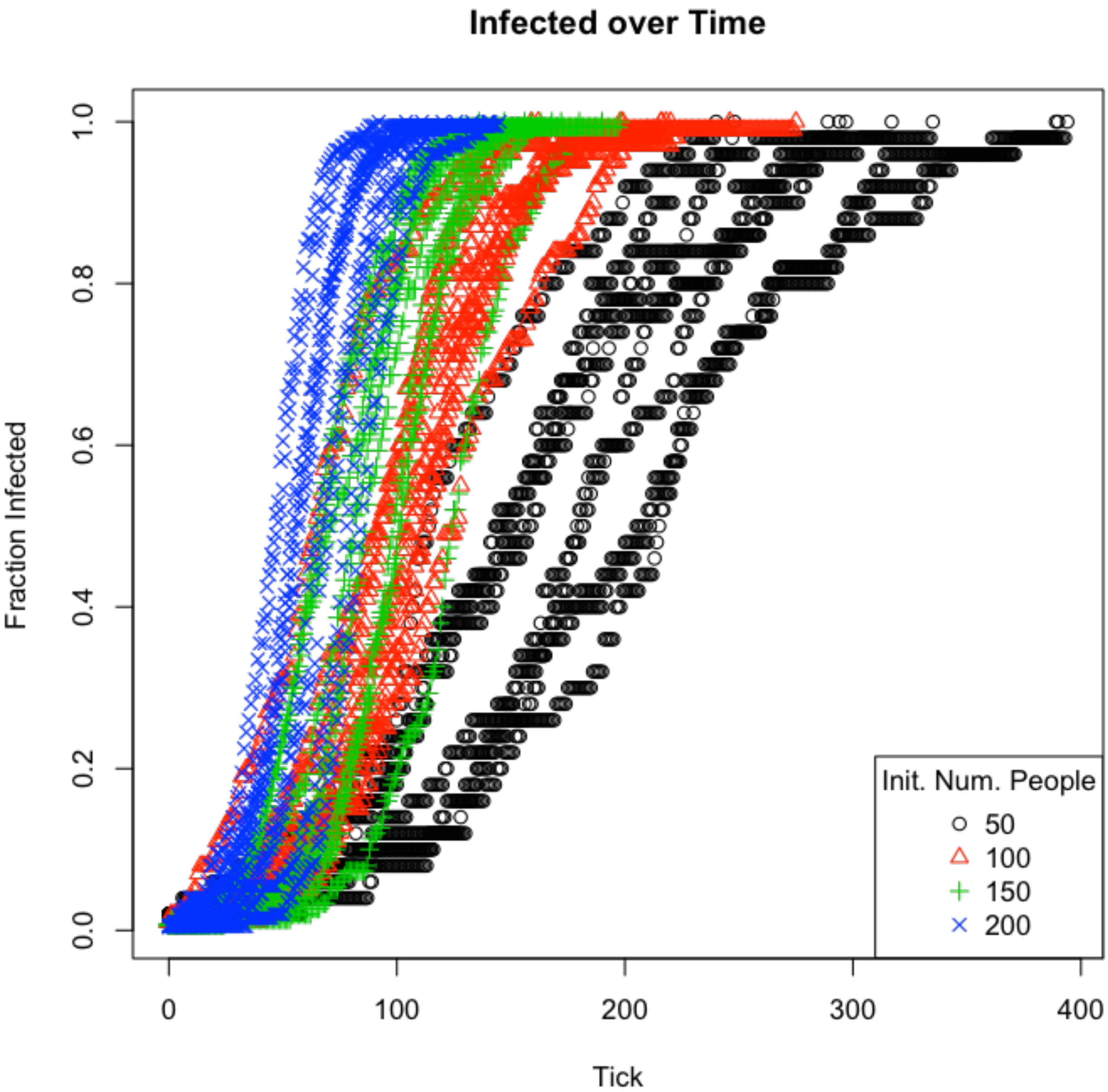

Time to 100% Infection vs. Population
Time to 100% Infection
450
400
350
300
250
200
150
100
50
100
150
200
Initial Num People

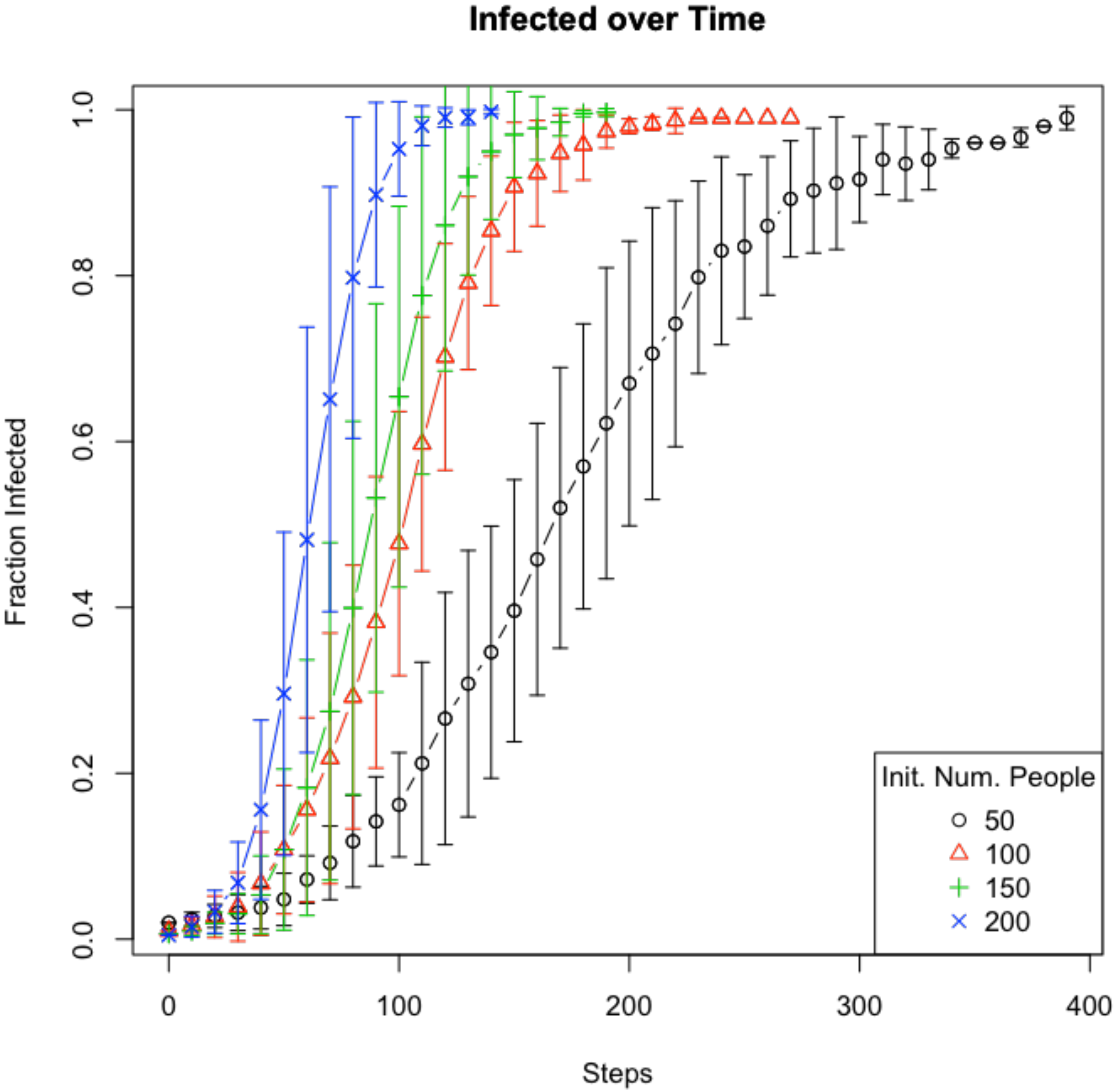

Time to 100% Infection vs. Initial Population
Errorbars are 1 SD
Time to 100% Infection
150
200
250
300
350
400
450
50
100
150
200
Initial Num People


Time to 100% Infection vs. Initial Population
Errorbars are 1 SE
Time to 100% Infection
400
350
300
250
200
150
50
100
150
200
Initial Num People

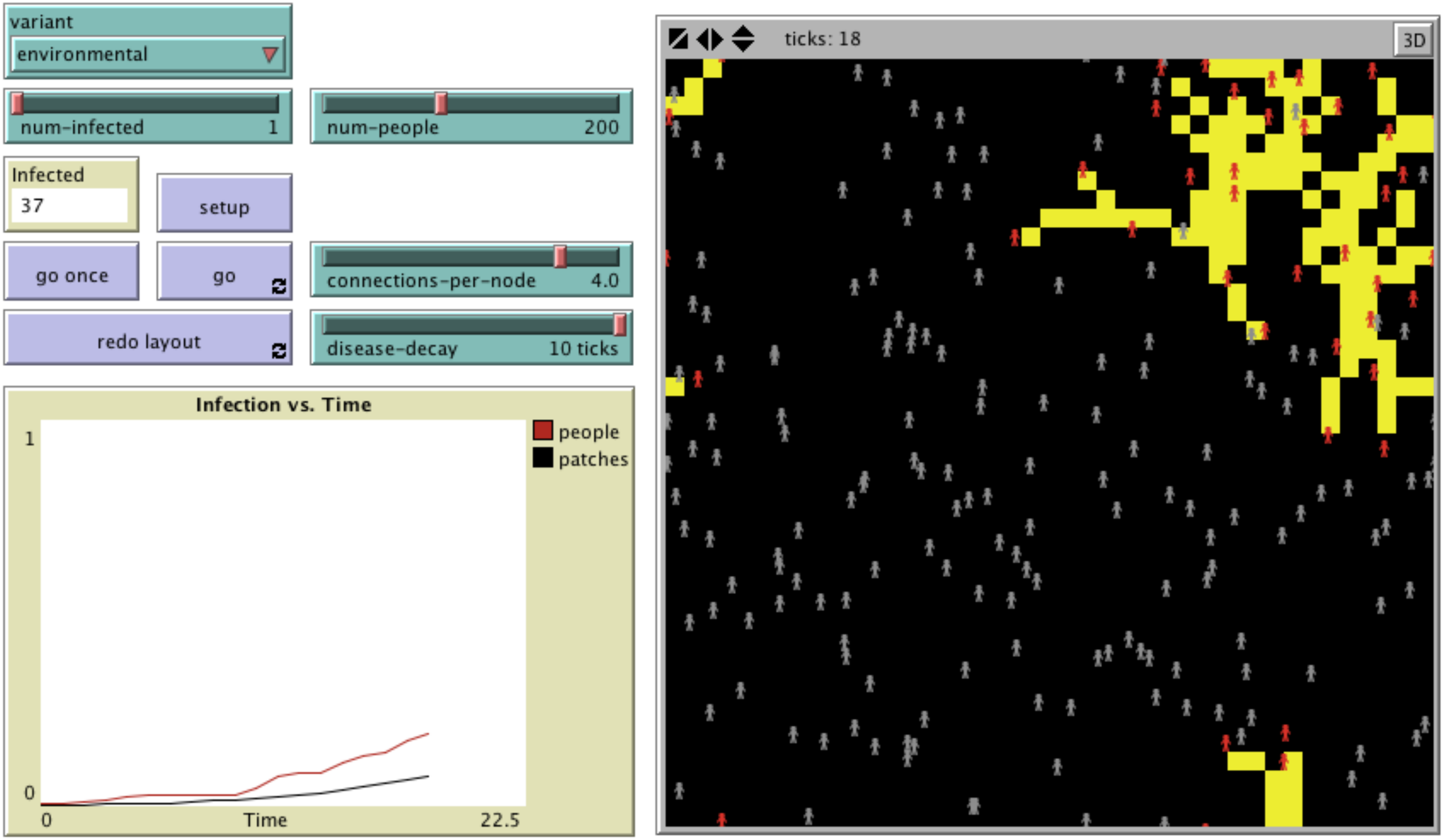
Time Series Graphs

- The graphs so far are all end of run results
- One important aspect of ABM is that you also have transitionary data recorded
- Runtime graphs are useful to look at, but they require lots of data


Infected over Time
Fraction Infected
0.0
0.2
0.4
0.6
0.8
1.0
Tick
0
100
200
300
400
Init. Num. People
○ 50
△ 100
+ 150
× 200


Infected over Time
Fraction Infected
0.0
0.2
0.4
0.6
0.8
1.0
Steps
0
100
200
300
400
Init. Num. People
○ 50
△ 100
+ 150
× 200

An Environmental Variant



variant: environmental

num-infected 1

num-people 200

Infected: 37

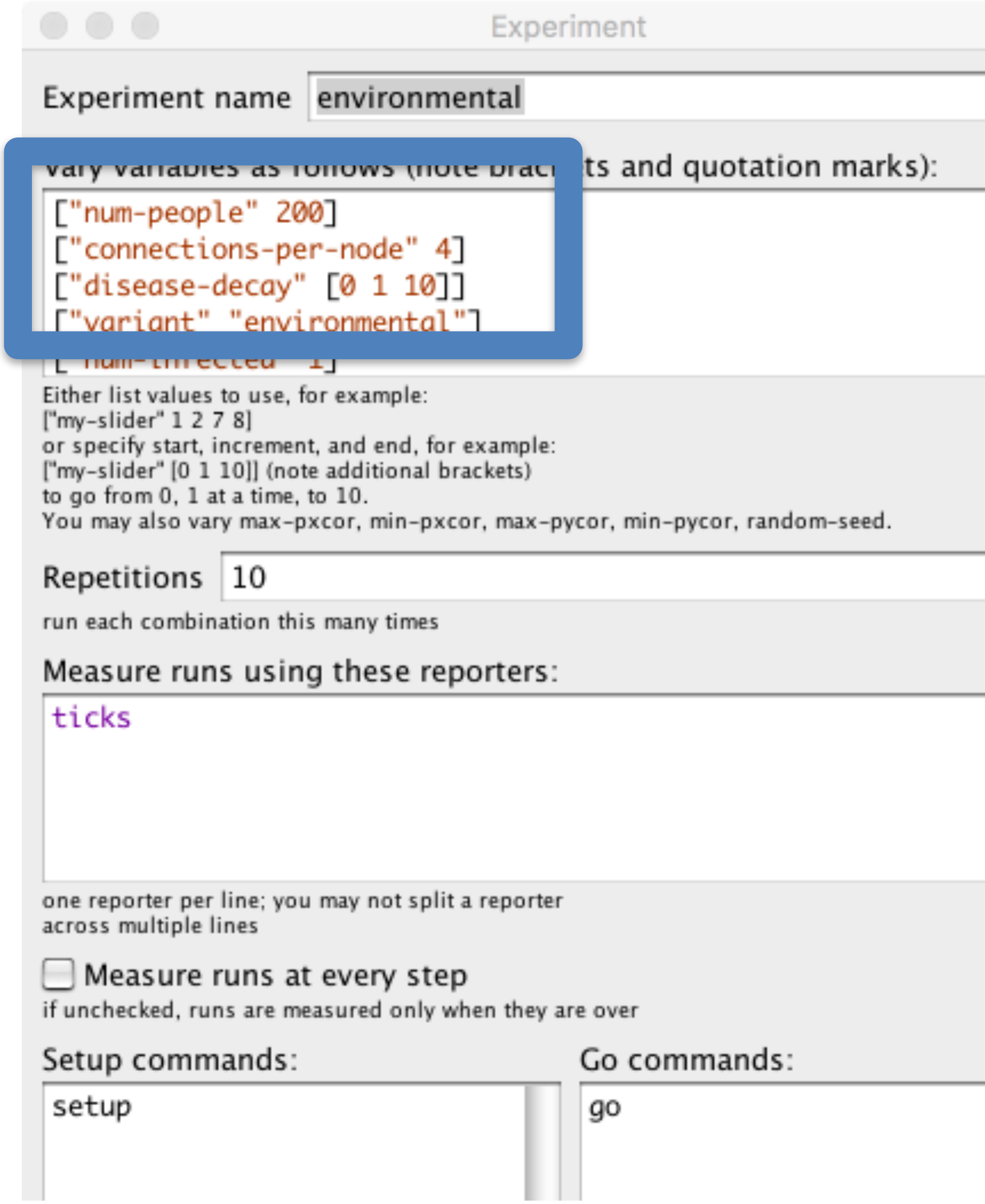
connections-per-node 4.0

disease-decay 10 ticks

ticks: 18

Environmental Changes

- Set number of people to a constant
- Add new variable called disease decay
- Anyone entering an area where the disease still is will be infected



Experiment name environmental

Vary variables as follows (note brackets and quotation marks):

```
["num-people" 200]
["connections-per-node" 4]
["disease-decay" [0 1 10]]
["variant" "environmental"]
["num-infected" 1]
```

Either list values to use, for example:
["my-slider" 1 2 7 8]
or specify start, increment, and end, for example:
["my-slider" [0 1 10]] (note additional brackets)
to go from 0, 1 at a time, to 10.
You may also vary max-pxcor, min-pxcor, max-pycor, min-pycor, random-seed.

Repetitions 10

run each combination this many times

Measure runs using these reporters:

```
ticks
```

one reporter per line; you may not split a reporter across multiple lines

☐ Measure runs at every step

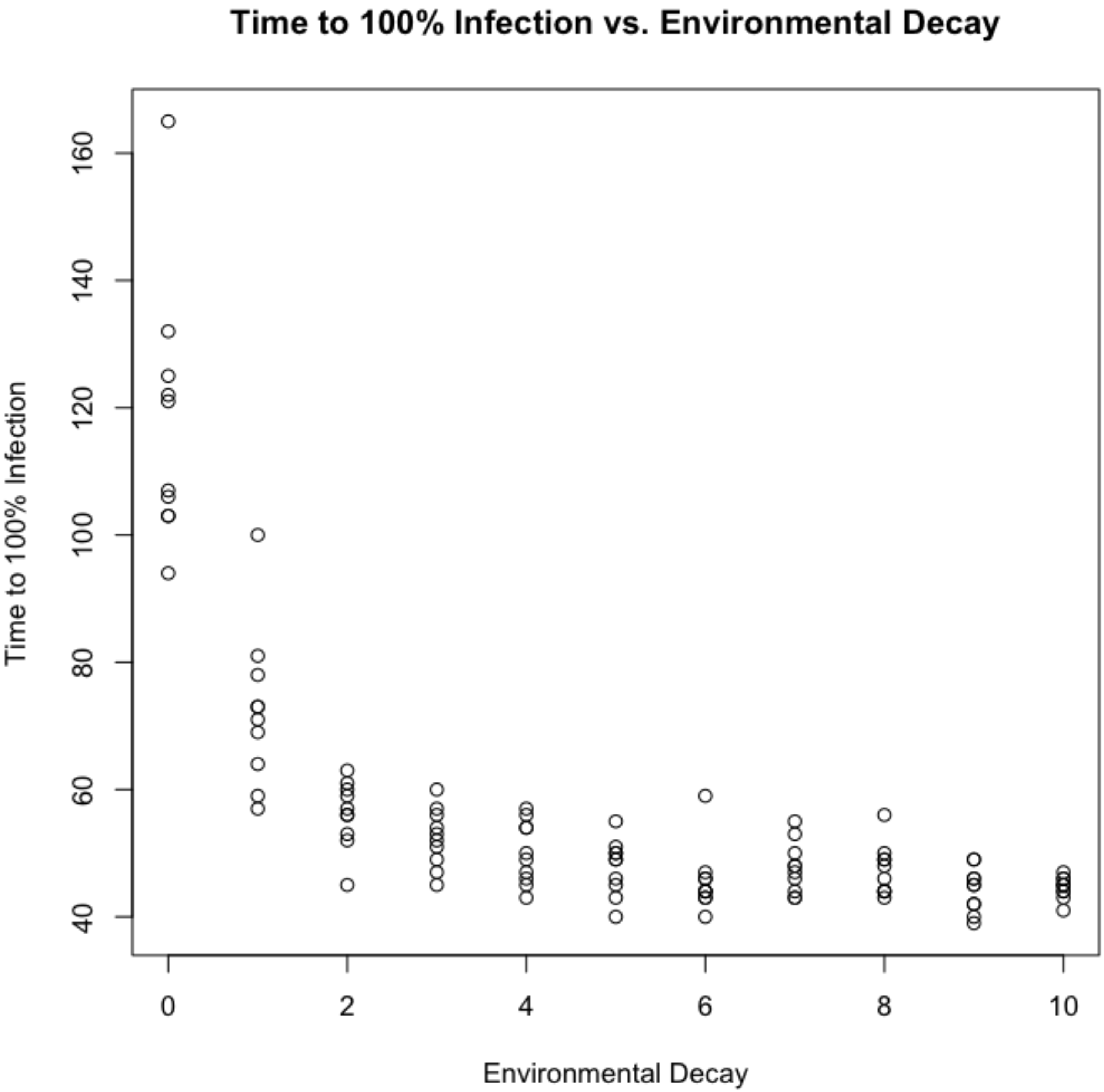
if unchecked, runs are measured only when they are over

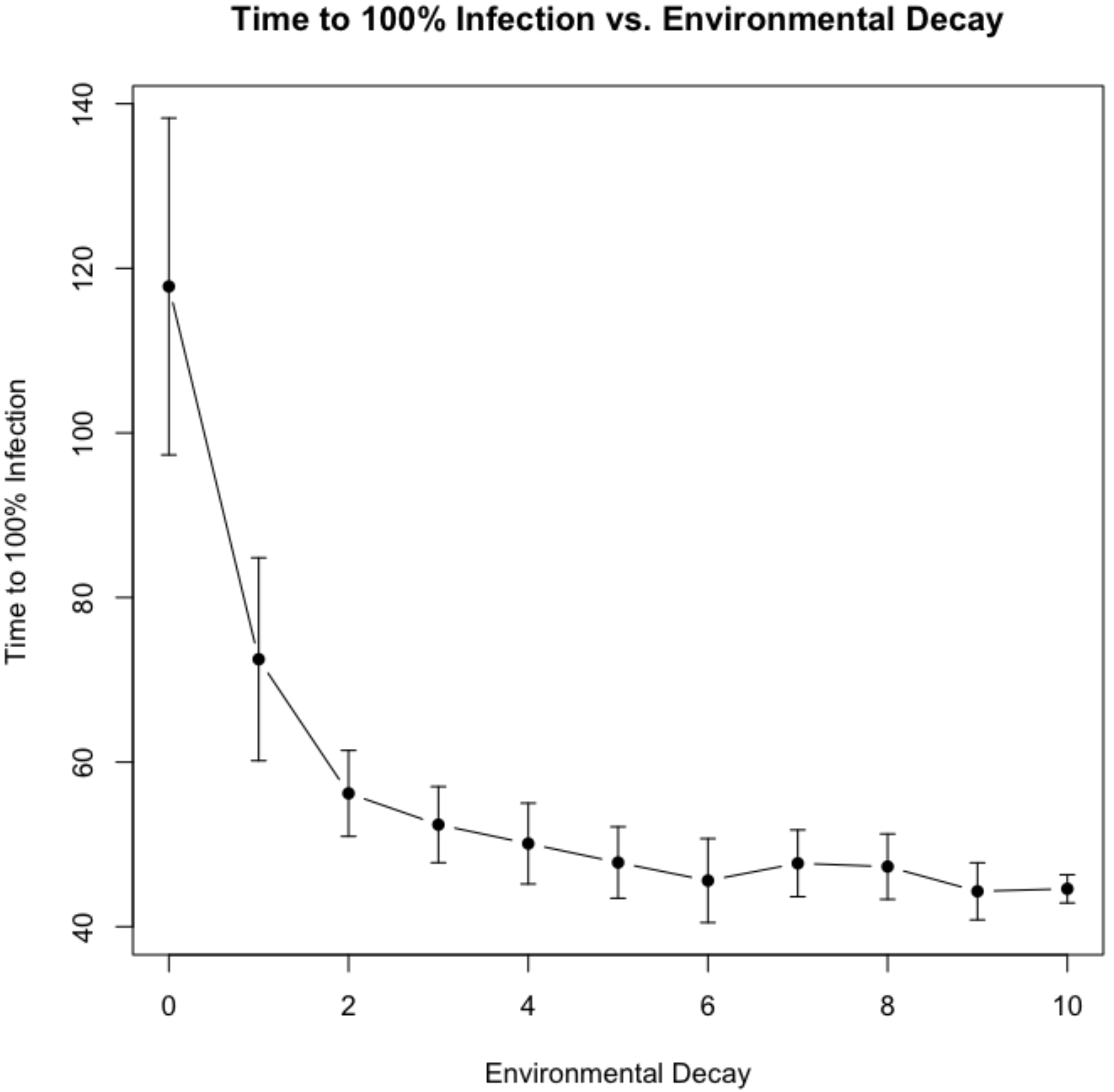
Setup commands:

```
setup
```

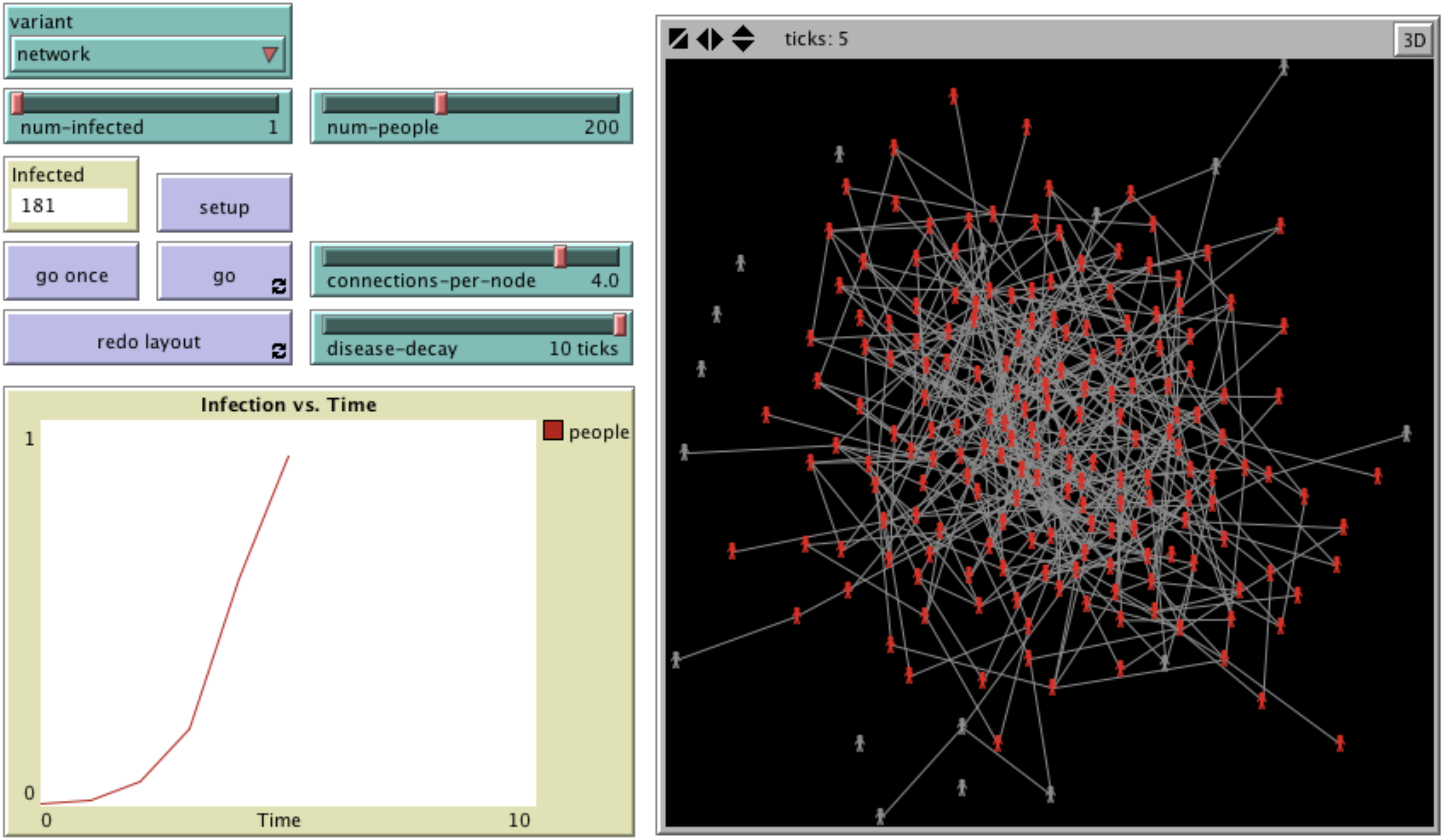
Go commands:

```
go
```


Time to 100% Infection vs. Environmental Decay
Time to 100% Infection
40
60
80
100
120
140
160
Environmental Decay
0
2
4
6
8
10


Time to 100% Infection vs. Environmental Decay
Time to 100% Infection
140
120
100
80
60
40
0
2
4
6
8
10
Environmental Decay

A Network Variant



variant
network

num-infected 1

num-people 200

Infected
181

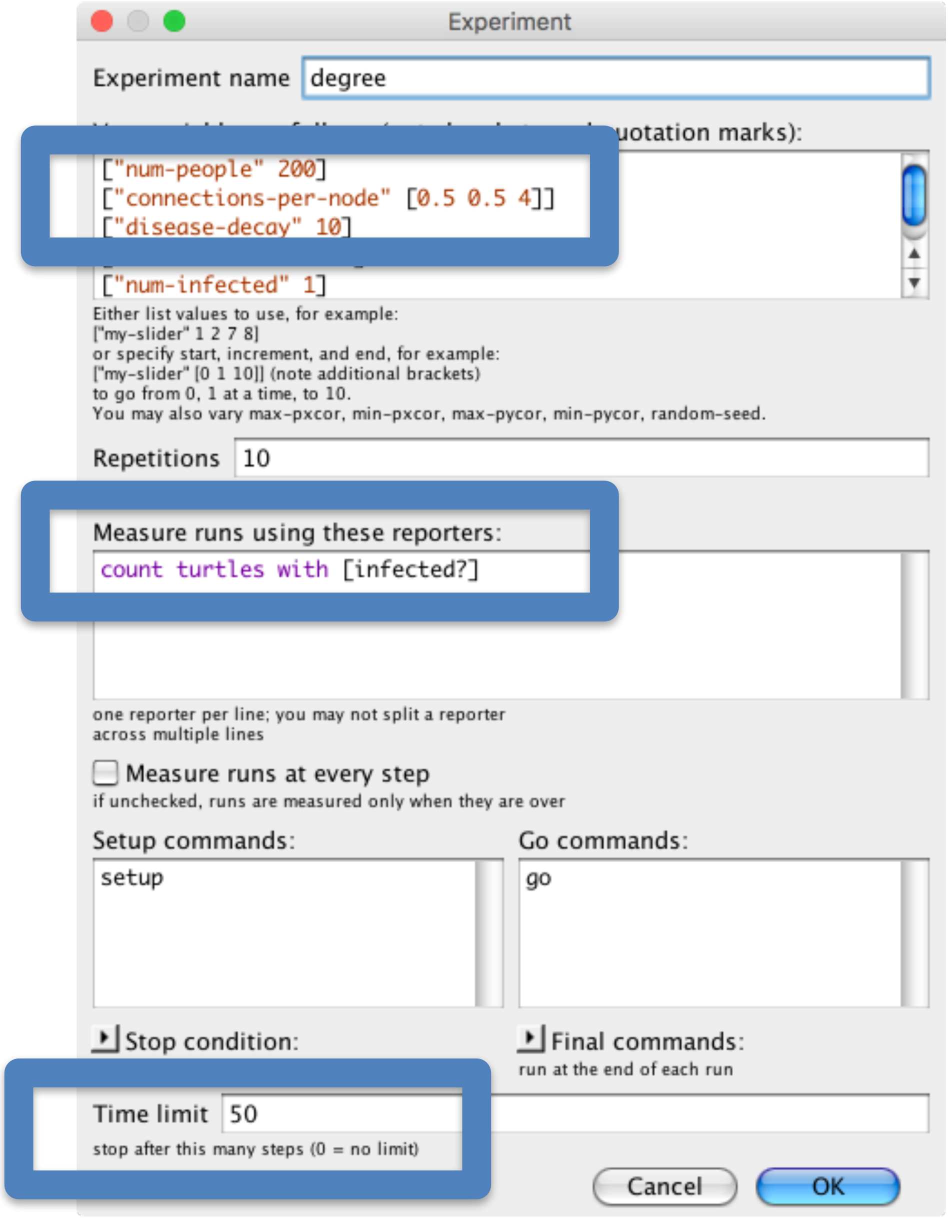
connections-per-node 4.0

disease-decay 10 ticks

ticks: 5

Network Changes

- 100% Infection is no longer guaranteed
- Measure spread until a certain time
- Record number infected at that time



Experiment name: degree

uotation marks):

```
["num-people" 200]
["connections-per-node" [0.5 0.5 4]]
["disease-decay" 10]
["num-infected" 1]
```

Either list values to use, for example:
["my-slider" 1 2 7 8]
or specify start, increment, and end, for example:
["my-slider" [0 1 10]] (note additional brackets)
to go from 0, 1 at a time, to 10.
You may also vary max-pxcor, min-pxcor, max-pycor, min-pycor, random-seed.

Repetitions: 10

Measure runs using these reporters:

```
count turtles with [infected?]
```

one reporter per line; you may not split a reporter across multiple lines

□ Measure runs at every step
if unchecked, runs are measured only when they are over

Setup commands:

```
setup
```

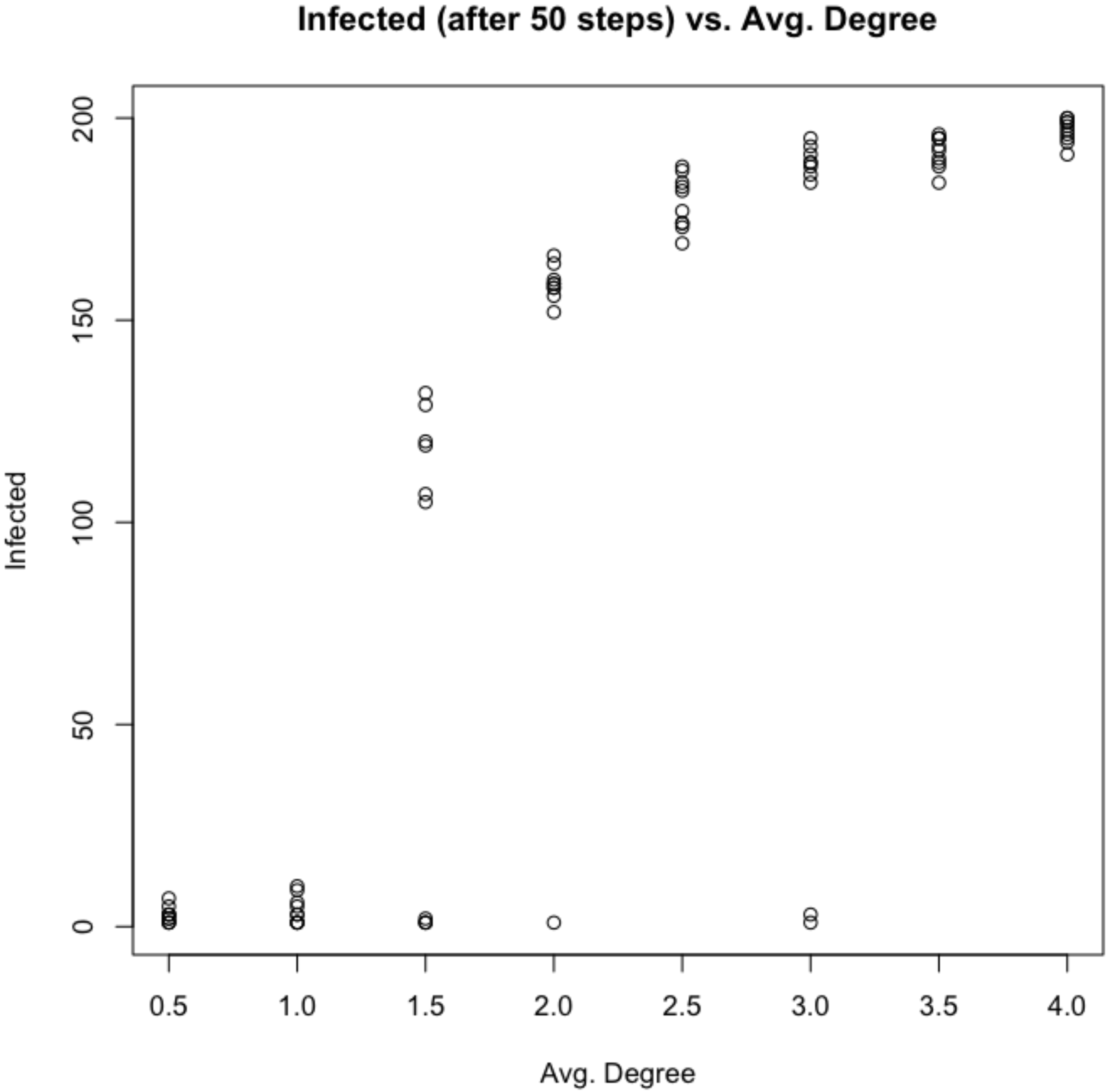
Go commands:

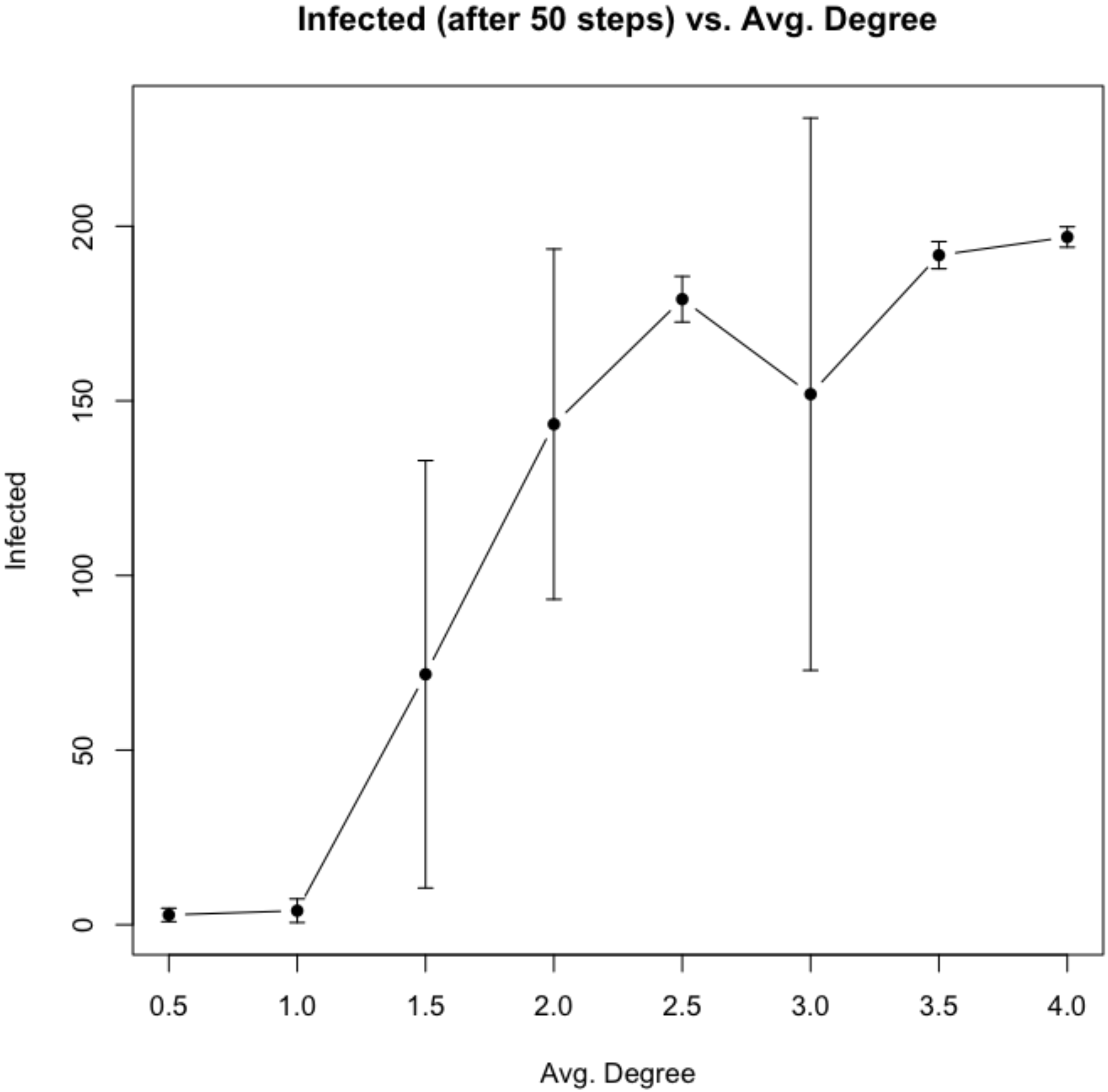
```
go
```

Stop condition:

Final commands:
run at the end of each run

Time limit: 50
stop after this many steps (0 = no limit)


Infected (after 50 steps) vs. Avg. Degree
Infected
0
50
100
150
200
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Avg. Degree

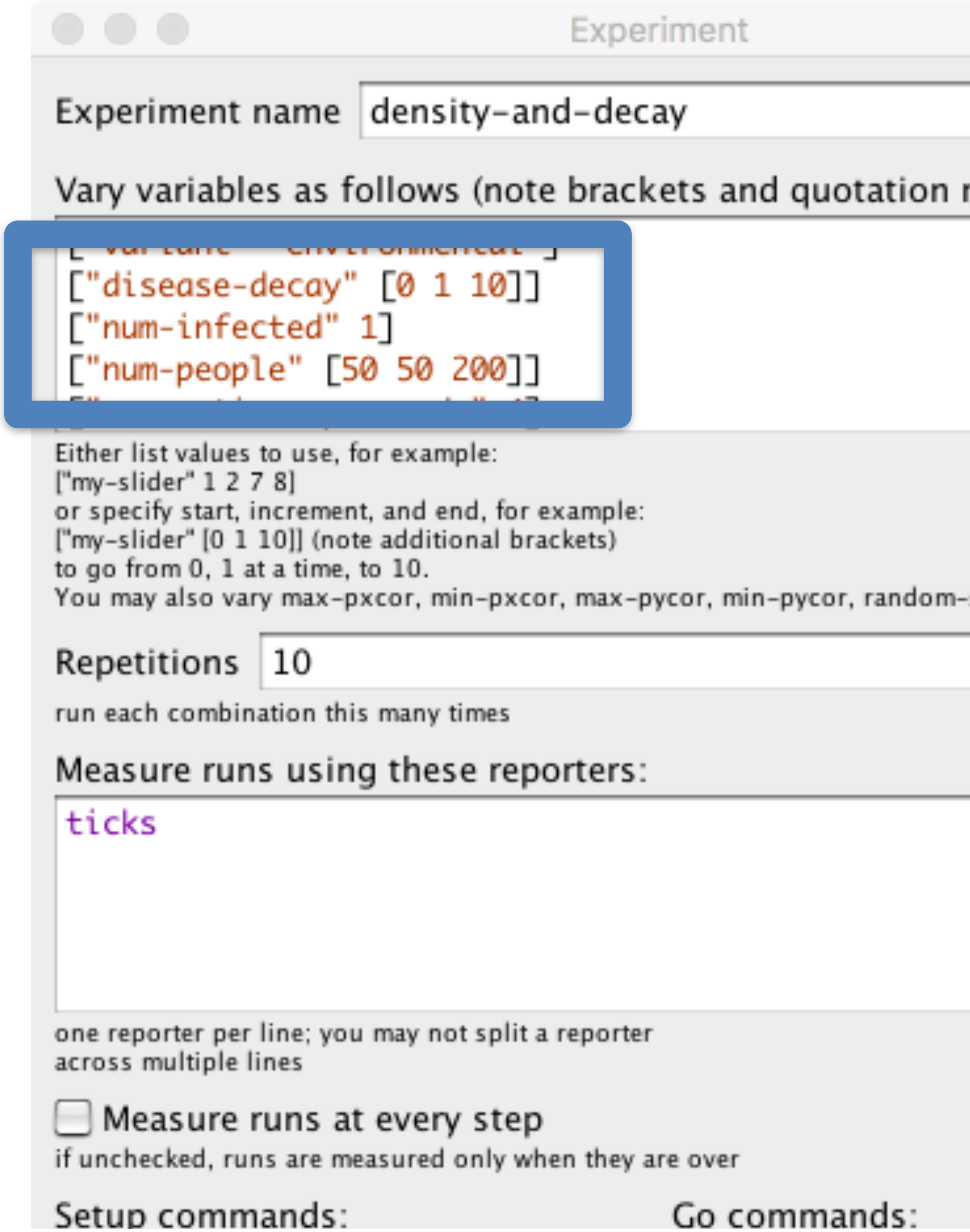

Infected (after 50 steps) vs. Avg. Degree
Infected
0
50
100
150
200
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Avg. Degree

Multivariate Analysis

- Our analysis so far has been about one input and one output
- What if you wanted to look at multiple inputs simultaneously?
  1. Establish base settings and vary from that
  2. Aggregating over all other variables
  3. 3D Charts of combinations
  4. Regression

Density and Decay

- We will vary population density and disease decay simultaneously
- Observe the results
- 440 Runs



Experiment name density-and-decay

Vary variables as follows (note brackets and quotation

```
["variant" "environmental"]
["disease-decay" [0 1 10]]
["num-infected" 1]
["num-people" [50 50 200]]
```

Either list values to use, for example:
["my-slider" 1 2 7 8]
or specify start, increment, and end, for example:
["my-slider" [0 1 10]] (note additional brackets)
to go from 0, 1 at a time, to 10.
You may also vary max-pxcor, min-pxcor, max-pycor, min-pycor, random-

Repetitions 10

run each combination this many times

Measure runs using these reporters:

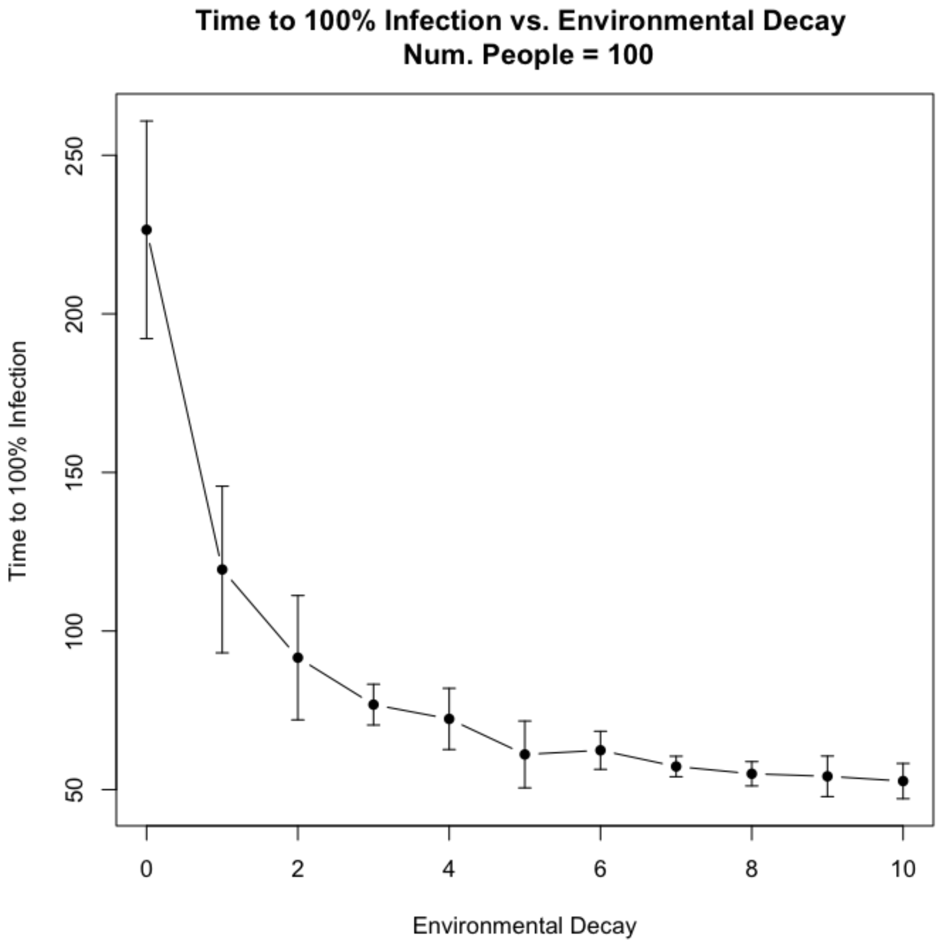
```
ticks
```

one reporter per line; you may not split a reporter across multiple lines

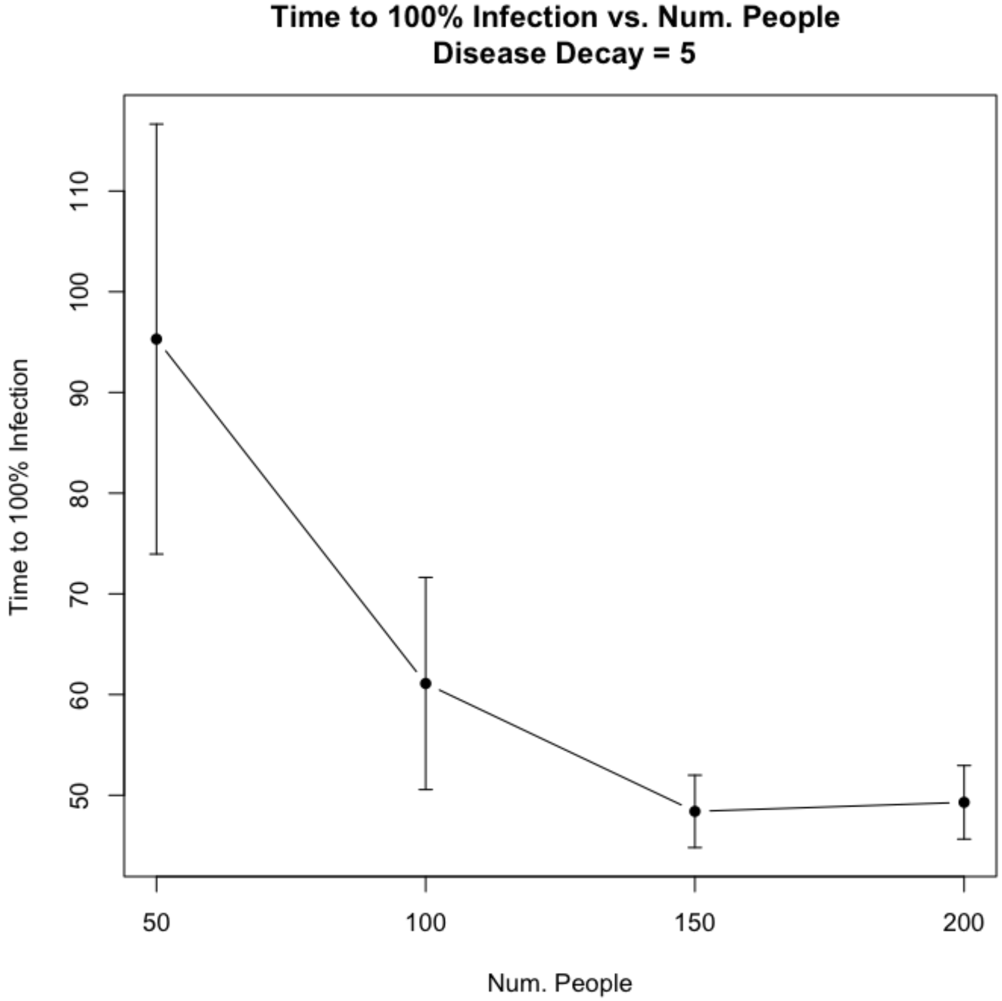
□ Measure runs at every step

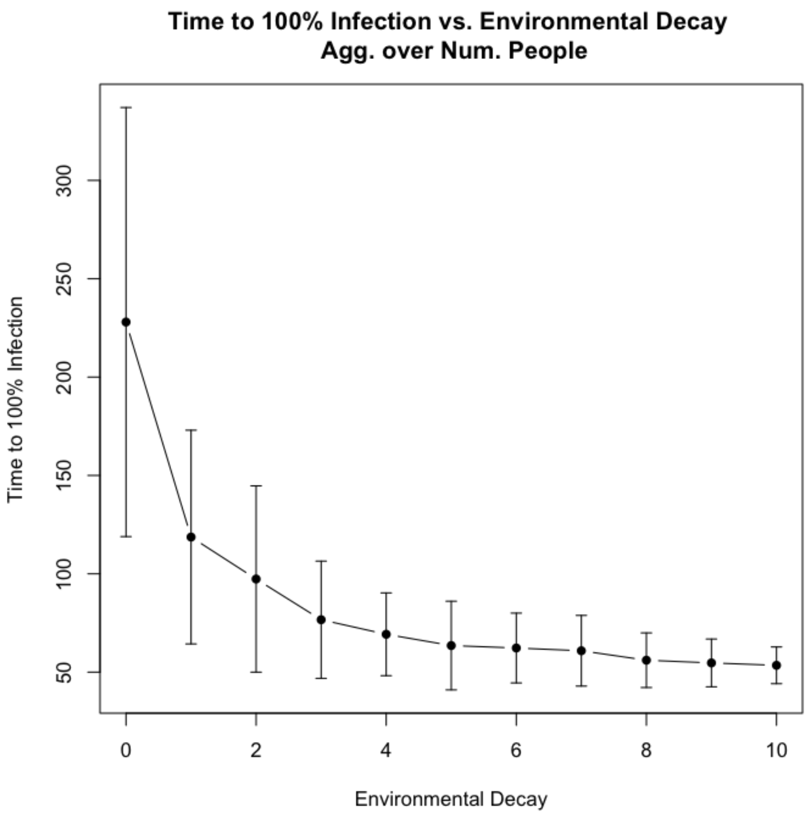
if unchecked, runs are measured only when they are over

Setup commands: Go commands:

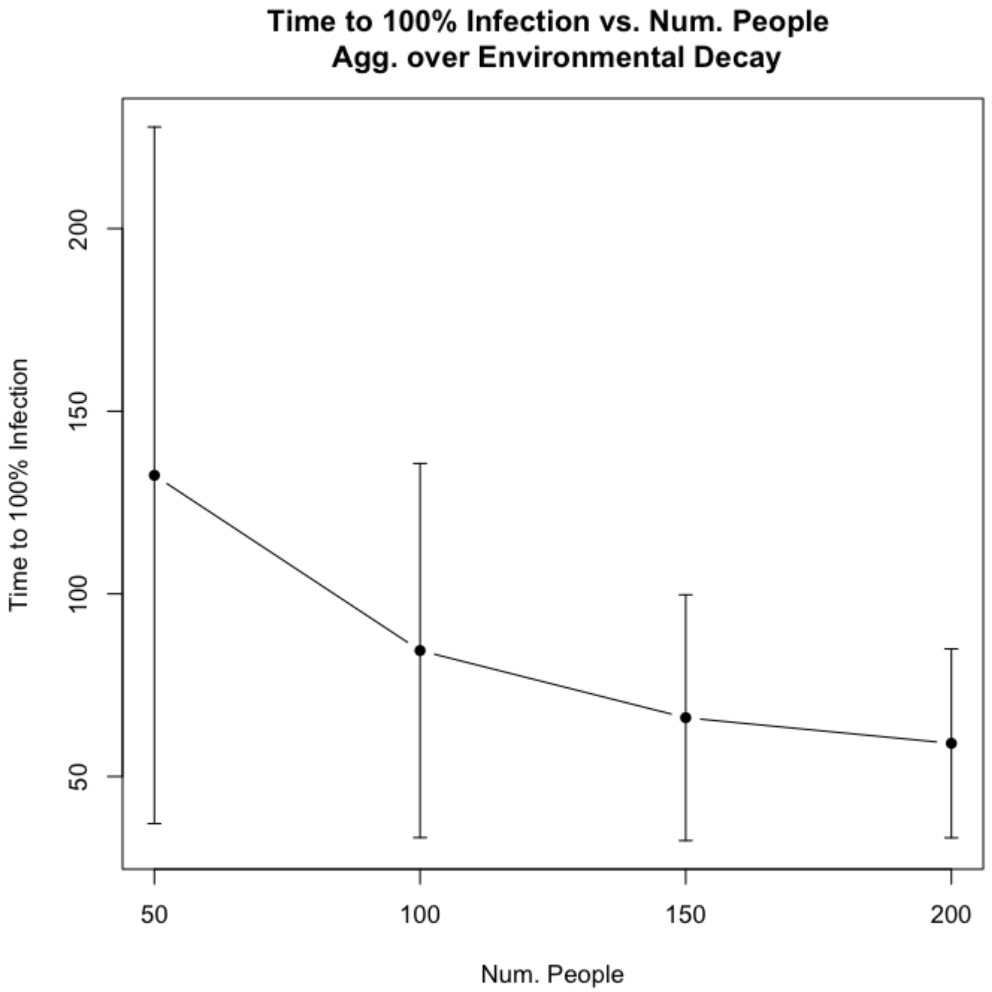


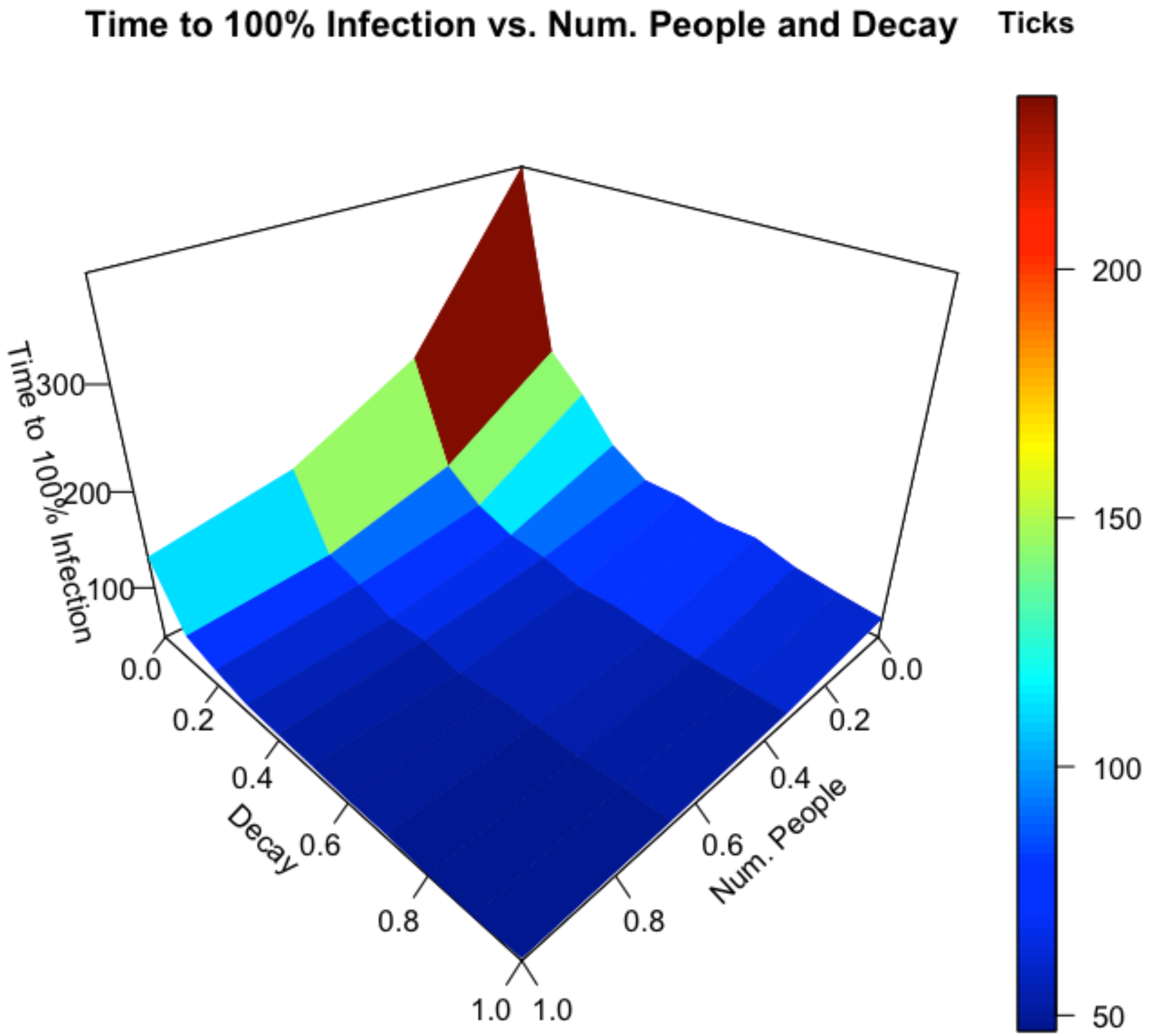
1. Base Case and Variants





2. Aggregating over Variation

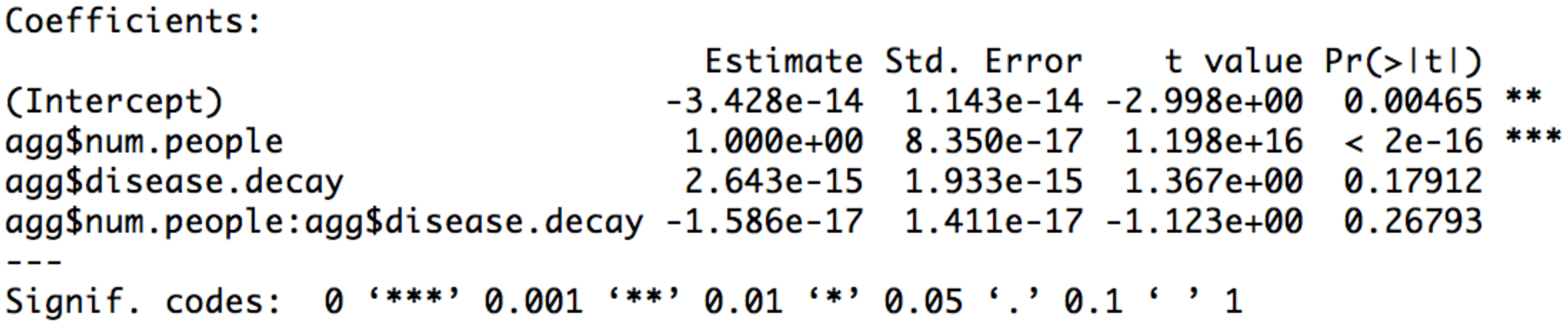



Time to 100% Infection vs. Num. People and Decay
Ticks
200
150
100
50
Time to 100% Infection
300
200
100
Decay
0.0
0.2
0.4
0.6
0.8
1.0
Num. People
0.0
0.2
0.4
0.6
0.8
1.0

3. 3D Plots

4. Regression

- If the model is really complex we can regress our output variable on all of our inputs
- This gives us a sense of how critical each input is to driving the output
- We can also look at the interactions



```
Coefficients:
                                 Estimate Std. Error    t value Pr(>|t|)
(Intercept)                    -3.428e-14  1.143e-14 -2.998e+00  0.00465 **
agg$num.people                  1.000e+00  8.350e-17  1.198e+16  < 2e-16 ***
agg$disease.decay               2.643e-15  1.933e-15  1.367e+00  0.17912
agg$num.people:agg$disease.decay -1.586e-17  1.411e-17 -1.123e+00  0.26793
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```