Complex Network

Quiz 13

Wang Biyuan

18M38156

Department of Computer Science

Lecturer: Tsuyoshi Murata

January 28, 2019

Code

```
!pip install -q ndlib
!pip install -q bokeh

import networkx as nx
import matplotlib.pyplot as plt
import numpy as np
import ndlib.models.epidemics.SIRModel as sir

# Network Definition
g = nx.erdos_renyi_graph(1000, 0.1)
print(nx.info(g))


# Model Selection
model = sir.SIRModel(g)


import ndlib.models.ModelConfig as mc

# Model Configuration
config = mc.Configuration()
config.add_model_parameter('beta', 0.001)
config.add_model_parameter('gamma', 0.01)
config.add_model_parameter("percentage_infected", 0.05)

model.set_initial_status(config)

# Simulation
iterations = model.iteration_bunch(200)
trends = model.build_trends(iterations)


# Visualization
from bokeh.io import output_notebook, show
output_notebook() # there will be no output without this

from ndlib.viz.bokeh.DiffusionTrend import DiffusionTrend

# Diffusion trend
viz = DiffusionTrend(model, trends)
p = viz.plot(width=400, height=400)


# show(p)

# Prevalence plot
from ndlib.viz.bokeh.DiffusionPrevalence import DiffusionPrevalence
viz_2 = DiffusionPrevalence(model, trends)
p_2 = viz_2.plot(width=400, height=400)
```

```python
# show(p2)

# Multiple plot
from ndlib.viz.bokeh.MultiPlot import MultiPlot
vm = MultiPlot()
vm.add_plot(p)
vm.add_plot(p_2)
m = vm.plot()
show(m)
```

```python
!pip install -q ndlib
!pip install -q bokeh

import networkx as nx
import matplotlib.pyplot as plt
import numpy as np
import ndlib.models.epidemics.SIRModel as sir

# Network Definition
g = nx.complete_graph(1000)
print(nx.info(g))

# Model Selection
model = sir.SIRModel(g)

import ndlib.models.ModelConfig as mc

# Model Configuration
config = mc.Configuration()
config.add_model_parameter('beta', 0.001)
config.add_model_parameter('gamma', 0.01)
config.add_model_parameter("percentage_infected", 0.05)
model.set_initial_status(config)

# Simulation
iterations = model.iteration_bunch(200)
trends = model.build_trends(iterations)

# Visualization
from bokeh.io import output_notebook, show
output_notebook() # there will be no output without this
from ndlib.viz.bokeh.DiffusionTrend import DiffusionTrend

# Diffusion trend
viz = DiffusionTrend(model, trends)
p = viz.plot(width=400, height=400)
# show(p)

# Prevalence plot
from ndlib.viz.bokeh.DiffusionPrevalence import DiffusionPrevalence
```

```python
viz2 = DiffusionPrevalence(model, trends)
p2 = viz2.plot(width=400, height=400)
# show(p2)

# Multiple plot
from ndlib.viz.bokeh.MultiPlot import MultiPlot
vm = MultiPlot()
vm.add_plot(p)
vm.add_plot(p2)
m = vm.plot()
show(m)
```

```python
!pip install -q ndlib
!pip install -q bokeh

import networkx as nx
import matplotlib.pyplot as plt
import numpy as np
import ndlib.models.epidemics.SIRModel as sir

# Network Definition
g = nx.star_graph(1000)
print(nx.info(g))

# Model Selection
model = sir.SIRModel(g)

import ndlib.models.ModelConfig as mc

# Model Configuration
config = mc.Configuration()
config.add_model_parameter('beta', 0.001)
config.add_model_parameter('gamma', 0.01)
config.add_model_parameter("percentage_infected", 0.05)
model.set_initial_status(config)

# Simulation
iterations = model.iteration_bunch(200)
trends = model.build_trends(iterations)

# Visualization
from bokeh.io import output_notebook, show
output_notebook() # there will be no output without this
from ndlib.viz.bokeh.DiffusionTrend import DiffusionTrend

# Diffusion trend
viz = DiffusionTrend(model, trends)
p = viz.plot(width=400, height=400)
# show(p)

# Prevalence plot
from ndlib.viz.bokeh.DiffusionPrevalence import DiffusionPrevalence
```

```
viz2 = DiffusionPrevalence(model, trends)
p2 = viz2.plot(width=400, height=400)
# show(p2)

# Multiple plot
from ndlib.viz.bokeh.MultiPlot import MultiPlot
vm = MultiPlot()
vm.add_plot(p)
vm.add_plot(p2)
m = vm.plot()
show(m)
```

```
!pip install -q ndlib
!pip install -q bokeh

import networkx as nx
import matplotlib.pyplot as plt
import numpy as np
import ndlib.models.epidemics.SIRModel as sir

# Network Definition
g = nx.cycle_graph(1000)
print(nx.info(g))

# Model Selection
model = sir.SIRModel(g)

import ndlib.models.ModelConfig as mc

# Model Configuration
config = mc.Configuration()
config.add_model_parameter('beta', 0.001)
config.add_model_parameter('gamma', 0.01)
config.add_model_parameter("percentage_infected", 0.05)
model.set_initial_status(config)

# Simulation
iterations = model.iteration_bunch(200)
trends = model.build_trends(iterations)

# Visualization
from bokeh.io import output_notebook, show
output_notebook() # there will be no output without this
from ndlib.viz.bokeh.DiffusionTrend import DiffusionTrend

# Diffusion trend
viz = DiffusionTrend(model, trends)
p = viz.plot(width=400, height=400)
# show(p)

# Prevalence plot
from ndlib.viz.bokeh.DiffusionPrevalence import DiffusionPrevalence
```

```python
viz2 = DiffusionPrevalence(model, trends)
p2 = viz2.plot(width=400, height=400)
# show(p2)

# Multiple plot
from ndlib.viz.bokeh.MultiPlot import MultiPlot
vm = MultiPlot()
vm.add_plot(p)
vm.add_plot(p2)
m = vm.plot()
show(m)
```

```python
!pip install -q ndlib
!pip install -q bokeh

import networkx as nx
import matplotlib.pyplot as plt
import numpy as np
import ndlib.models.epidemics.SIRModel as sir

# Network Definition
g = nx.barbell_graph(500,0)
print(nx.info(g))

# Model Selection
model = sir.SIRModel(g)

import ndlib.models.ModelConfig as mc

# Model Configuration
config = mc.Configuration()
config.add_model_parameter('beta', 0.001)
config.add_model_parameter('gamma', 0.01)
config.add_model_parameter("percentage_infected", 0.05)
model.set_initial_status(config)

# Simulation
iterations = model.iteration_bunch(200)
trends = model.build_trends(iterations)

# Visualization
from bokeh.io import output_notebook, show
output_notebook() # there will be no output without this
from ndlib.viz.bokeh.DiffusionTrend import DiffusionTrend

# Diffusion trend
viz = DiffusionTrend(model, trends)
p = viz.plot(width=400, height=400)
# show(p)

# Prevalence plot
from ndlib.viz.bokeh.DiffusionPrevalence import DiffusionPrevalence
```

```python
viz2 = DiffusionPrevalence(model, trends)
p2 = viz2.plot(width=400, height=400)
# show(p2)

# Multiple plot
from ndlib.viz.bokeh.MultiPlot import MultiPlot
vm = MultiPlot()
vm.add_plot(p)
vm.add_plot(p2)
m = vm.plot()
show(m)
```

```python
!pip install -q ndlib
!pip install -q bokeh

import networkx as nx
import matplotlib.pyplot as plt
import numpy as np
import ndlib.models.epidemics.SIRModel as sir

# Network Definition
g = nx.lollipop_graph(720,280)
print(nx.info(g))

# Model Selection
model = sir.SIRModel(g)

import ndlib.models.ModelConfig as mc

# Model Configuration
config = mc.Configuration()
config.add_model_parameter('beta', 0.001)
config.add_model_parameter('gamma', 0.01)
config.add_model_parameter("percentage_infected", 0.05)
model.set_initial_status(config)

# Simulation
iterations = model.iteration_bunch(200)
trends = model.build_trends(iterations)

# Visualization
from bokeh.io import output_notebook, show
output_notebook() # there will be no output without this
from ndlib.viz.bokeh.DiffusionTrend import DiffusionTrend

# Diffusion trend
viz = DiffusionTrend(model, trends)
p = viz.plot(width=400, height=400)
# show(p)

# Prevalence plot
from ndlib.viz.bokeh.DiffusionPrevalence import DiffusionPrevalence
```

```python
viz2 = DiffusionPrevalence(model, trends)
p2 = viz2.plot(width=400, height=400)
# show(p2)

# Multiple plot
from ndlib.viz.bokeh.MultiPlot import MultiPlot
vm = MultiPlot()
vm.add_plot(p)
vm.add_plot(p2)
m = vm.plot()
show(m)
```

```python
viz2 = DiffusionPrevalence(model, trends)
p2 = viz2.plot(width=400, height=400)
# show(p2)

# Multiple plot
from ndlib.viz.bokeh.MultiPlot import MultiPlot
vm = MultiPlot()
```

Results

1. Do simulations with same parameters (# of nodes, beta, gamma, percentage infected) but with different network structures such as:
1) complete graph (clique)
2) star graph
3) cycle graph
4) barbell graph
5) lollipop graph

2. Show an example of Diffusion Trend which is quite different from that of Erdos-Renyi network (together with used network structure).
Cycle graph has a very different diffusion trend comparing with Erdos-Renyi. Because a cycle graph has all nodes connected one by one to each other. If there is one node got infected, its neighbours will be infected and affect their neighbours. Therefore, the susceptibility is very high (close to 1).
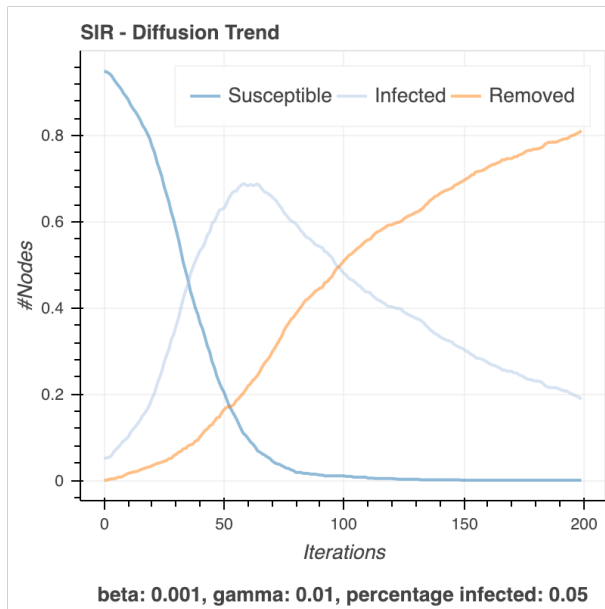
Figure 1: Erdos-Renyi
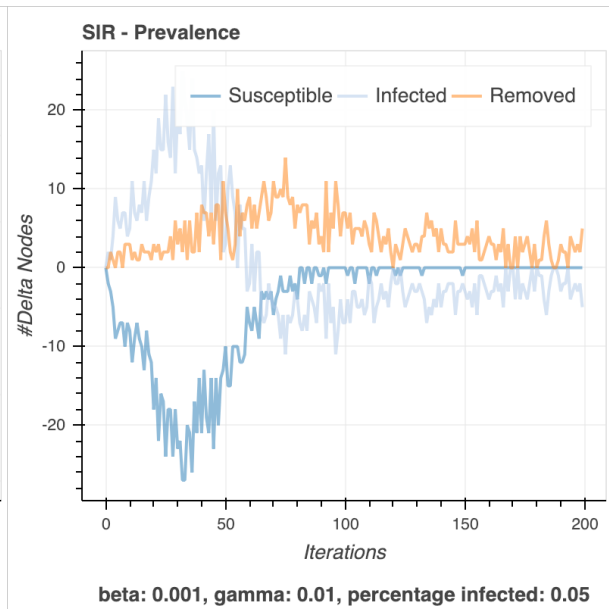
Figure 2: Erdos-Renyi
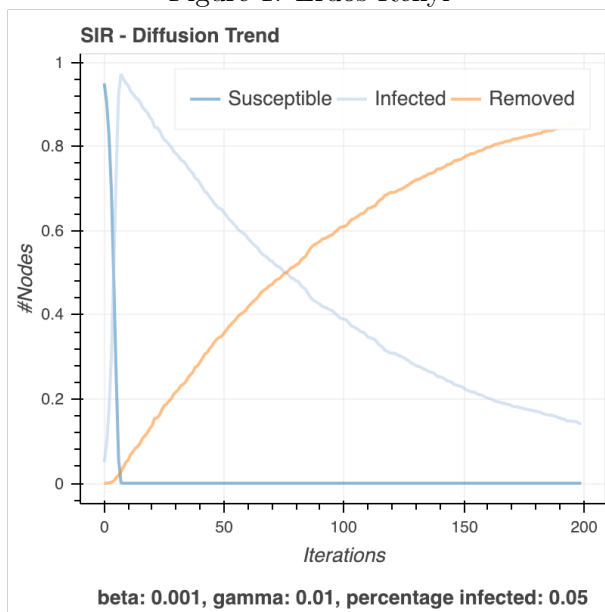
Figure 3: complete
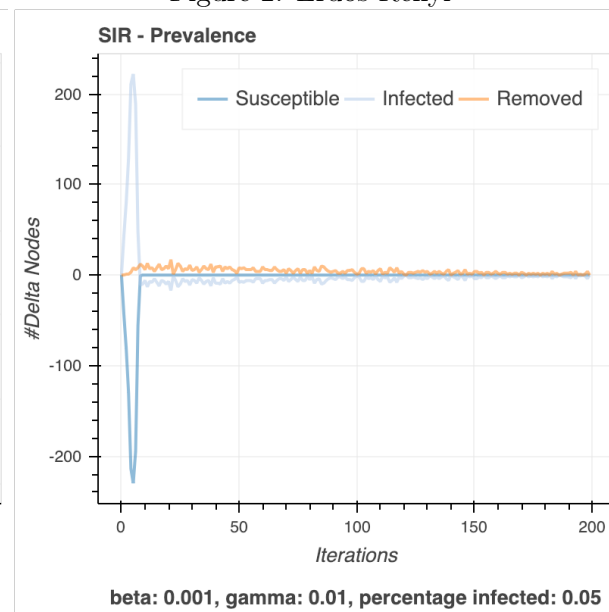
Figure 4: complete
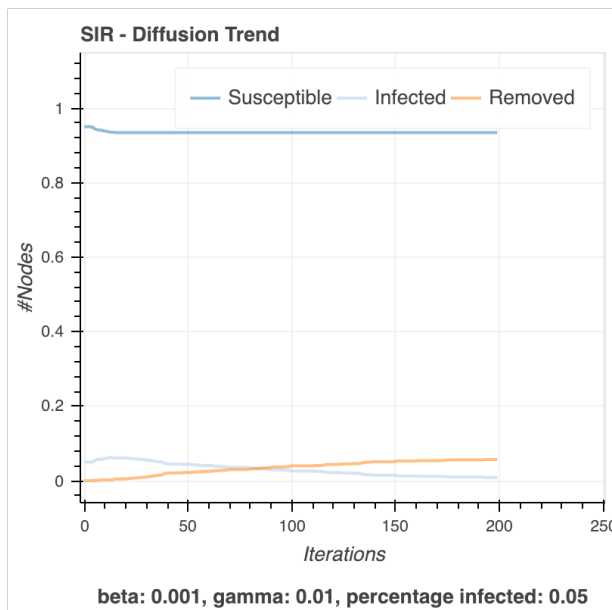
Figure 5: star
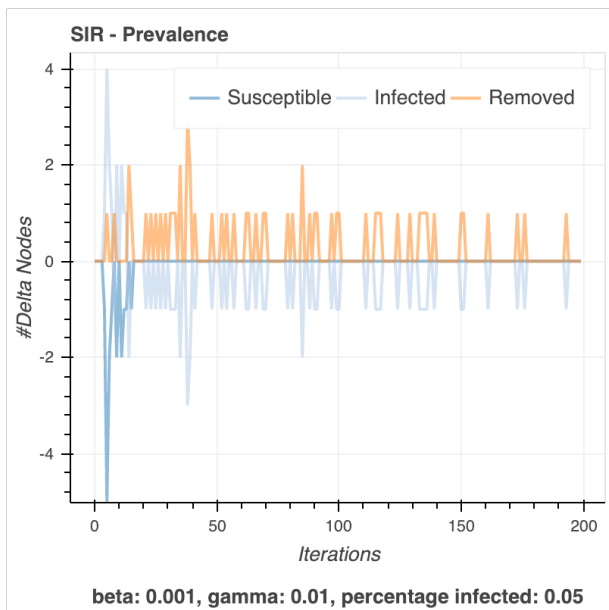
Figure 6: star
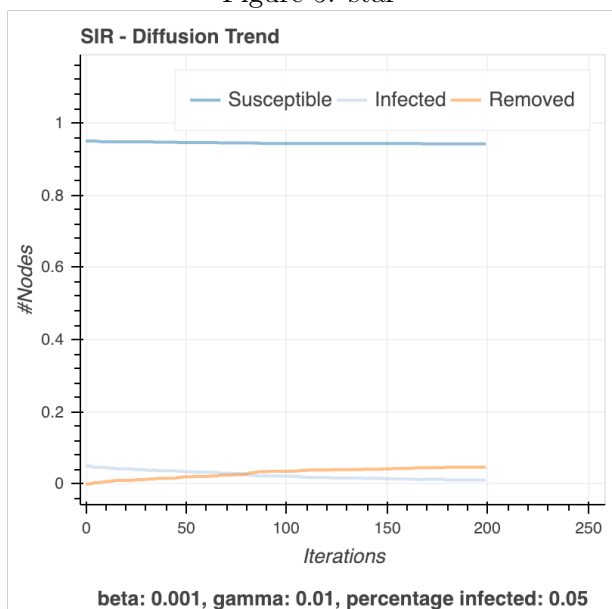
Figure 7: cycle

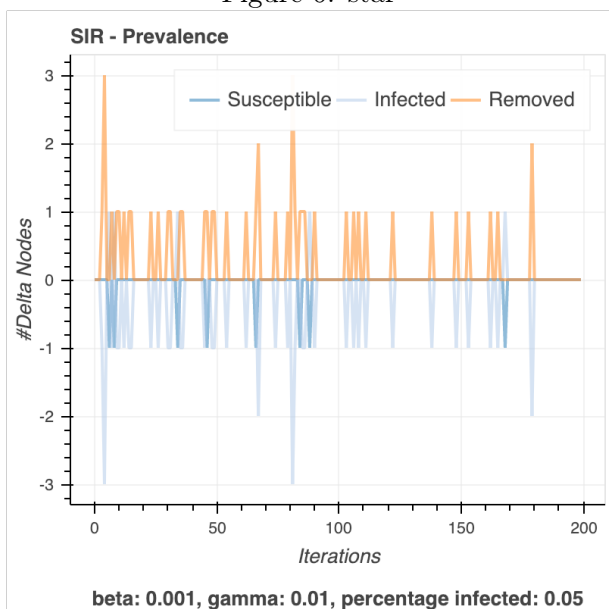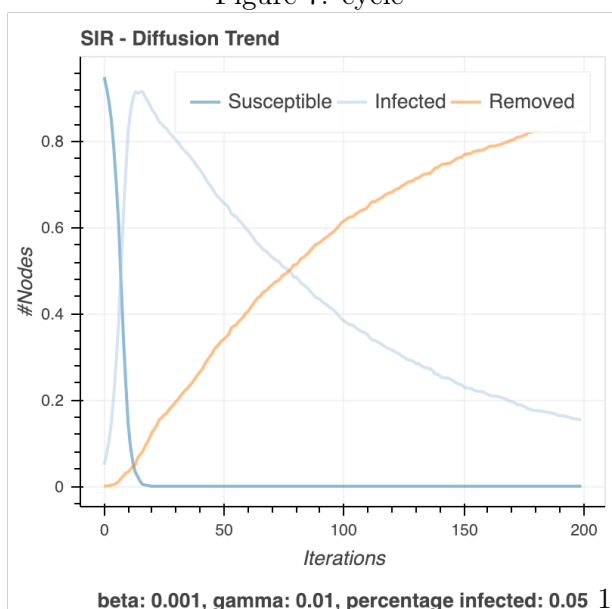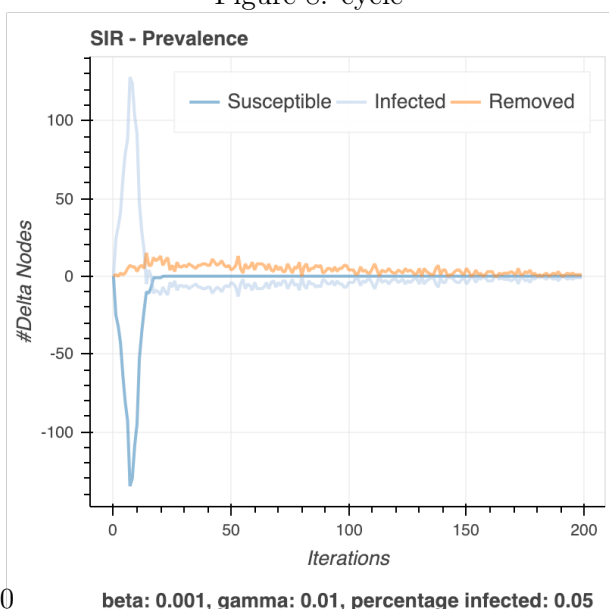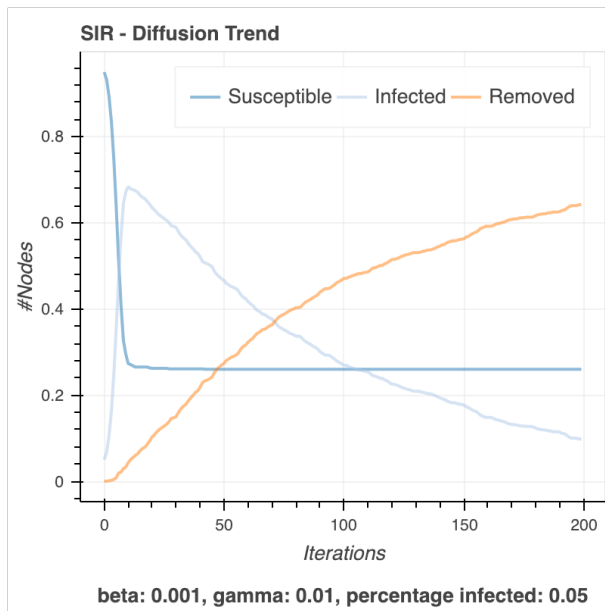Figure 8: cycle
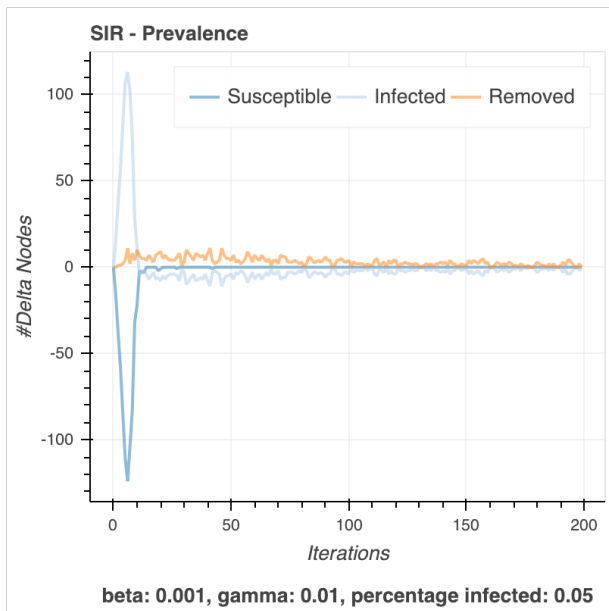

Figure 9: barbell

Figure 10: barbell

Figure 11: lollipop

Figure 12: lollipop