Magolego SNA - Epidemic Models

Contents

```
# To actually install EpiModel you need a DLPK (http://www.gnu.org/software/glpk/) library, which can b

library("EpiModel")
```

SIR Epidemic Model

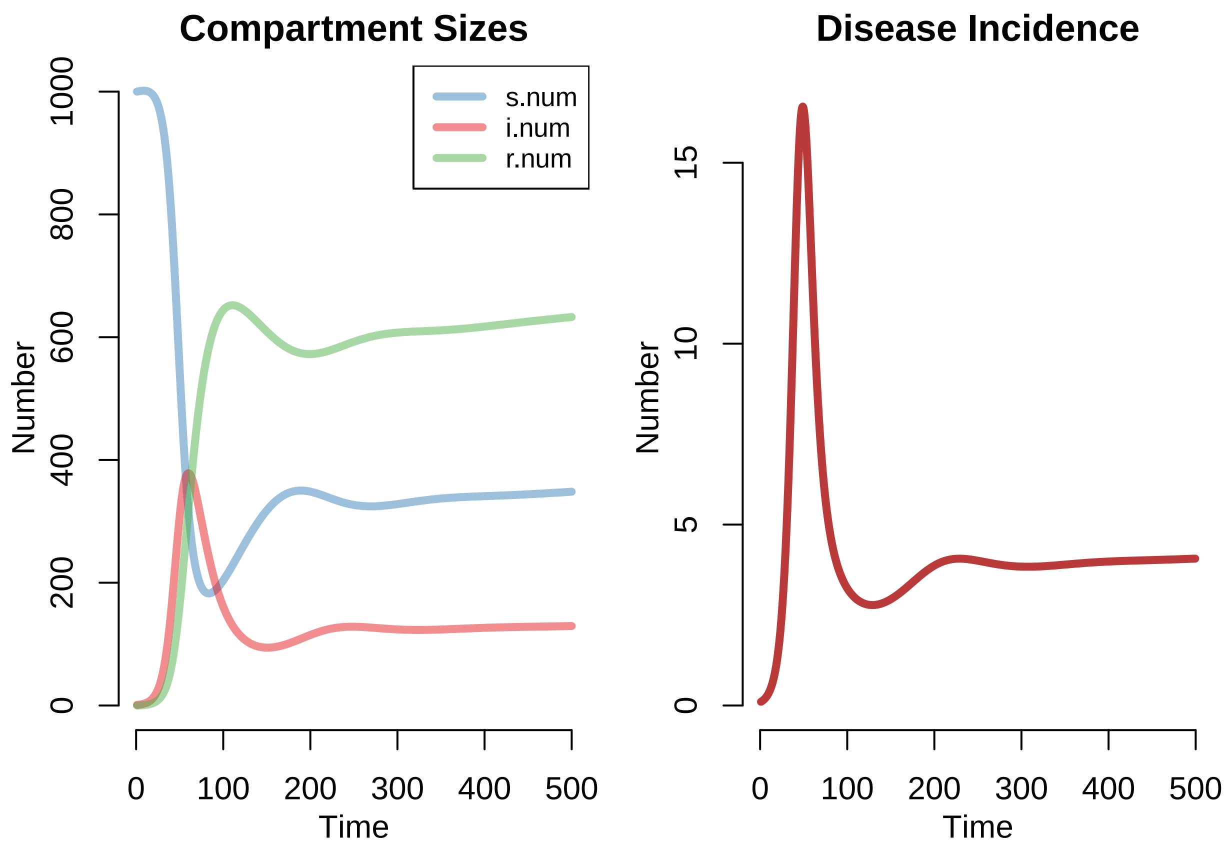
SIR in EpiModel Library

In a Susceptible-Infectious-Recovered (SIR) model, infected individuals recover from disease into a life-long recovered state; they are never again susceptible to disease. In this section, we model an SIR disease by adding to our basic SI model a recovery process. We also introduce demographic processes so that persons may enter and exit the population through births and deaths.

In `EpiModel`, introducing these new transition processes into the model is straightforward. In param.dcm, parameters for the *recovery rate*, *birth rate*, and *state-specific death rate* must be entered. These model parameters imply that the birth rate is slightly higher than the underlying death rate among susceptibles, and that there is disease-induced mortality because the di.rate is larger than the other two death rates. In init.dcm, node that it is necessary to specify the number of initially recovered, even if that is 0. Finally, in control.dcm, the *dt* argument may be used to obtain model results in fractional time units

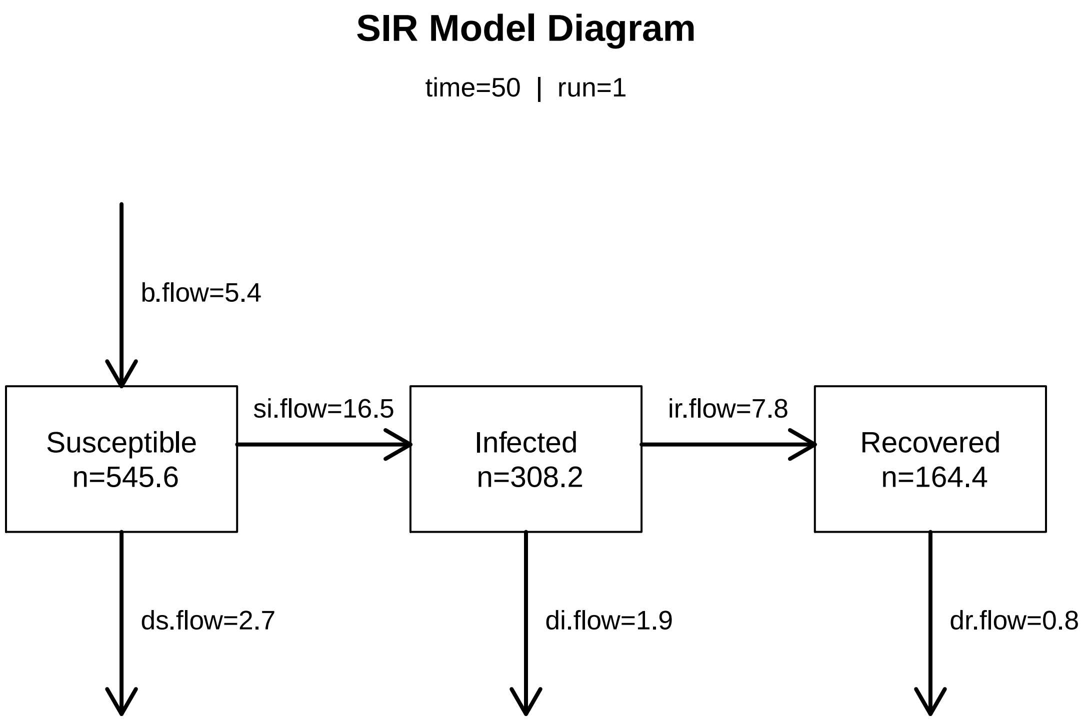
```
param <- param.dcm(inf.prob = 0.2, act.rate = 1, rec.rate = 1/20,
                   b.rate = 1/95, ds.rate = 1/100, di.rate = 1/80,
                   dr.rate = 1/100)
init <- init.dcm(s.num = 1000, i.num = 1, r.num = 0)
control <- control.dcm(type = "SIR", nsteps = 500, dt = 0.5)
mod <- dcm(param, init, control)
```

Next we plot the results of the model with several plot arguments set to non-default values



This plot provides a standard state-flow diagram that is often presented in the epidemiological literature

```
par(mfrow = c(1, 1))
comp_plot(mod, at = 50, digits = 1)
```



The plot shows the three state sizes and flows at t=50. This plot is also built into the summary function through the comp_plot argument to that function.

Simulating the Epidemic Model

In contrast to DCMs and ICMs, which solve or simulate the epidemic system with one function, network models require multiple steps:

- The network is initialized
- The network model is parameterized
- The network model is fit with netest
- The network model is diagnosed with netdx
- A dynamic network is simulated given the model fit
- Epidemic processes are simulated on top of the dynamic simulated network

```
size <- 500
nw <- network.initialize(size, directed = FALSE)
dissolution <- ~ offset(edges)
duration <- 100
coef.diss <- dissolution_coefs(dissolution, duration)
formation <- ~ edges + concurrent
target.stats <- c(50, 25)
est <- netest(nw,
               formation,
               target.stats,
```

```
                coef.diss,
                verbose = FALSE)
```

```
## Starting maximum likelihood estimation via MCMLE:
## Iteration 1 of at most 20:
## The log-likelihood improved by 0.09784
## Step length converged once. Increasing MCMC sample size.
## Iteration 2 of at most 20:
## The log-likelihood improved by 0.003704
## Step length converged twice. Stopping.
##
## This model was fit using MCMC.  To examine model diagnostics and check for degeneracy, use the mcmc.
```

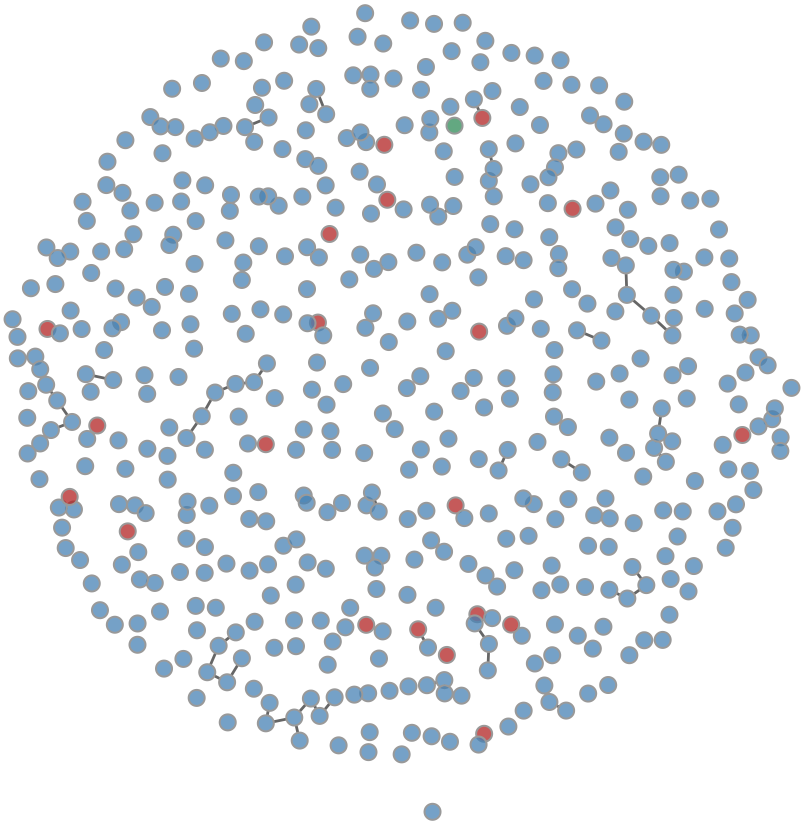
```
dx <- netdx(est, nsims = 10, nsteps = 500)
```

```
##
## Network Diagnostics
## -----------------------
## - Simulating 10 networks
##   |**********|
## - Calculating formation statistics
## - Calculating duration statistics
##   |**********|
## - Calculating dissolution statistics
##   |**********|
##
```

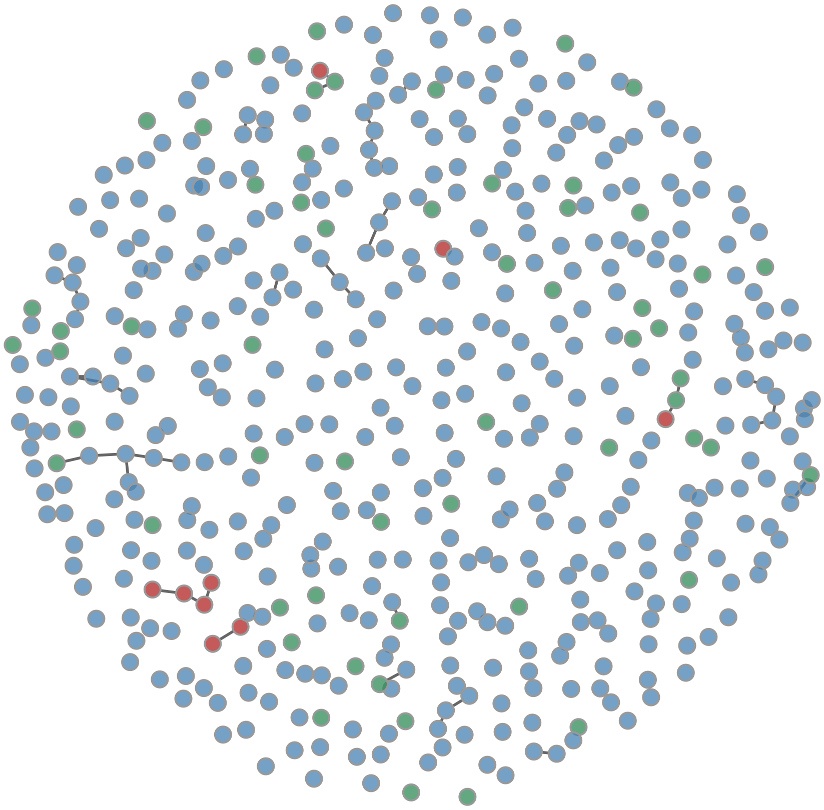
```
param <- param.net(inf.prob = 0.8, act.rate = 50, rec.rate = 0.01)
init <- init.net(i.num = 20, r.num = 1)
control <- control.net(type = "SIR", nsims = 5, nsteps = 500,
                       verbose.int = 0)
sim <- netsim(est, param, init, control)
```

```
##
## * Starting Network Simulation
## Sim = 1/5
## Sim = 2/5
## Sim = 3/5
## Sim = 4/5
## Sim = 5/5
```

```
par(mar = c(0,0,0,0))
plot(sim, type = "network",
     col.status = TRUE, at = 1, sims = 1)
```



```
par(mar = c(0,0,0,0))
plot(sim, type = "network",
     col.status = TRUE, at = 500, sims = 1)
```



```r
summary(sim, at = 100)
```

```
##
## EpiModel Summary
## =======================
## Model class: netsim
##
## Simulation Details
## -----------------------
## Model type: SIR
## No. simulations: 5
## No. time steps: 500
## No. NW modes: 1
##
## Model Statistics
## ------------------------------
## Time: 100
## ------------------------------
##             mean     sd   pct
## Suscept.   455.6  5.941 0.911
## Infect.     18.2  2.588 0.036
## Recov.      26.2  4.438 0.052
## Total      500.0  0.000 1.000
## S -> I       0.0  0.000    NA
## I -> R       0.2  0.447    NA
## ------------------------------
```

```r
summary(sim, at = 250)
```

```
##
## EpiModel Summary
## =======================
## Model class: netsim
##
## Simulation Details
## -----------------------
## Model type: SIR
## No. simulations: 5
## No. time steps: 500
## No. NW modes: 1
##
## Model Statistics
## ------------------------------
## Time: 250
## ------------------------------
##            mean      sd    pct
## Suscept.    445  11.358  0.890
## Infect.      11   6.364  0.022
## Recov.       44   6.892  0.088
## Total       500   0.000  1.000
## S -> I        0   0.000     NA
## I -> R        0   0.000     NA
## ------------------------------
```

```r
summary(sim, at = 500)
```

```
##
## EpiModel Summary
## =======================
## Model class: netsim
##
## Simulation Details
## -----------------------
## Model type: SIR
## No. simulations: 5
## No. time steps: 500
## No. NW modes: 1
##
## Model Statistics
## ------------------------------
## Time: 500
## ------------------------------
##             mean      sd    pct
## Suscept.   430.8  23.732  0.862
## Infect.      5.8   4.207  0.012
## Recov.      63.4  20.182  0.127
## Total      500.0   0.000  1.000
## S -> I       0.0   0.000     NA
## I -> R       0.0   0.000     NA
## ------------------------------
```