The most popular model to model epidemics is the so-called [SIR](http://www.public.asu.edu/~hnesse/classes/sir.html) model – or [Kermack-McKendrick](https://en.wikipedia.org/wiki/Kermack-McKendrick_theory). Consider a population of size N, and assume that S is the number of susceptible, I the number of infectious, and R for the number recovered (or immune) individuals, \displaystyle {\begin{aligned}&{\frac {dS}{dt}}=-{\frac {\beta IS}{N}},\\[6pt]&{\frac {dI}{dt}}={\frac {\beta IS}{N}}-\gamma I,\\[6pt]&{\frac {dR}{dt}}=\gamma I,\end{aligned}}so that \displaystyle{{\frac{dS}{dt}}+{\frac {dI}{dt}}+{\frac {dR}{dt}}=0}which implies that S+I+R=N. In order to be more realistic, consider some (constant) birth rate \mu, so that the model becomes\displaystyle {\begin{aligned}&{\frac {dS}{dt}}=\mu(N-S)-{\frac {\beta IS}{N}},\\[6pt]&{\frac {dI}{dt}}={\frac {\beta IS}{N}}-(\gamma+\mu) I,\\[6pt]&{\frac {dR}{dt}}=\gamma I-\mu R,\end{aligned}}Note, in this model, that people get sick (infected) but they do not die, they recover. So here, we can model [chickenpox](https://en.wikipedia.org/wiki/Chickenpox), for instance, not SARS.

The dynamics of the infectious class depends on the following ratio:\displaystyle{R\_{0}={\frac {\beta }{\gamma +\mu}}} which is the so-called [basic reproduction number](https://en.wikipedia.org/wiki/Basic_reproduction_number) (or reproductive ratio). The effective reproductive ratio is R\_0S/N, and the turnover of the epidemic happens exactly when R\_0S/N=1, or when the fraction of remaining susceptibles is R\_0^{-1}. As shown in [Directly transmitted infectious diseases:Control by vaccination](https://science.sciencemag.org/content/215/4536/1053), if S/N<R\_0^{-1}[/latex] the disease (the number of people infected) will start to decrease.</p> <p>Want to see it  ? Start with</p> <p>2f8f7d80e757ec80aa013429cb8a03e4011</p> <p>for the parameters. Here,  [latex]R\_0=4. We also need starting values

|  |  |
| --- | --- |
| 1  2  3  4  5 | epsilon = .001  N = 1  S = 1-epsilon  **I** = epsilon  R = 0 |

Then use the [ordinary differential equation solver](https://www.rdocumentation.org/packages/deSolve/versions/1.27.1/topics/ode?tap_a=5644-dce66f&tap_s=10907-287229), in R. The idea is to say that \boldsymbol{Z}=(S,I,R) and we have the gradient \frac{\partial \boldsymbol{Z}}{\partial t} = SIR(\boldsymbol{Z})where SIR is function of the various parameters. Hence, set

|  |  |
| --- | --- |
| 1  2 | p = **c**(mu = 0, N = 1, **beta** = 2, **gamma** = 1/2)  start\_SIR = **c**(S = 1-epsilon, **I** = epsilon, R = 0) |

The we must define the time, and the function that returns the gradient,

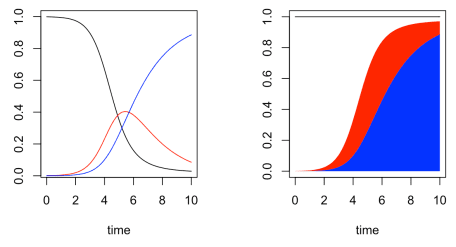
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | times = **seq**(0, 10, **by** = .1)  SIR = **function**(**t**,Z,p){  S=Z[1]; **I**=Z[2]; R=Z[3]; N=S+**I**+R  mu=p["mu"]; **beta**=p["beta"]; **gamma**=p["gamma"]  dS=mu\*(N-S)-**beta**\*S\***I**/N  dI=**beta**\*S\***I**/N-(mu+**gamma**)\***I**  dR=**gamma**\*I-mu\*R  dZ=**c**(dS,dI,dR)  **return**(dZ)} |

To solve this problem use

|  |  |
| --- | --- |
| 1  2 | **library**(deSolve)  resol = ode(y=start\_SIR, times=times, func=SIR, parms=p) |

We can visualize the dynamics below

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | **par**(mfrow=**c**(1,2))  **t**=resol[,"time"]  **plot**(**t**,resol[,"S"],type="l",xlab="time",ylab="")  **lines**(**t**,resol[,"I"],**col**="red")  **lines**(**t**,resol[,"R"],**col**="blue")  **plot**(**t**,**t**\*0+1,type="l",xlab="time",ylab="",ylim=0:1)  **polygon**(**c**(**t**,**rev**(**t**)),**c**(resol[,"R"],**rep**(0,**nrow**(resol))),**col**="blue")  **polygon**(**c**(**t**,**rev**(**t**)),**c**(resol[,"R"]+resol[,"I"],**rev**(resol[,"R"])),**col**="red") |

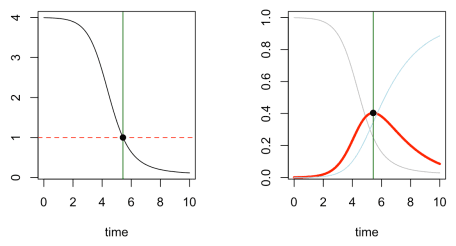


We can actually also visualize the effective reproductive number is R\_0S/N, where

|  |  |
| --- | --- |
| 1 | R0=p["beta"]/(p["gamma"]+p["mu"]) |

The effective reproductive number is on the left, and as we mentioned above, when we reach 1, we actually reach the maximum of the infected,

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | **plot**(**t**,resol[,"S"]\*R0,type="l",xlab="time",ylab="")  **abline**(h=1,lty=2,**col**="red")  **abline**(v=**max**(**t**[resol[,"S"]\*R0&gt;=1]),**col**="darkgreen")  **points**(**max**(**t**[resol[,"S"]\*R0&gt;=1]),1,pch=19)  **plot**(**t**,resol[,"S"],type="l",xlab="time",ylab="",**col**="grey")  **lines**(**t**,resol[,"I"],**col**="red",lwd=3)  **lines**(**t**,resol[,"R"],**col**="light blue")  **abline**(v=**max**(**t**[resol[,"S"]\*R0&gt;=1]),**col**="darkgreen")  **points**(**max**(**t**[resol[,"S"]\*R0&gt;=1]),**max**(resol[,"I"]),pch=19) |



And when adding a \mu parameter, we can obtain some interesting dynamics on the number of infected,

|  |  |
| --- | --- |
| 1  2  3  4  5 | times = **seq**(0, 100, **by**=.1)  p = **c**(mu = 1/100, N = 1, **beta** = 50, **gamma** = 10)  start\_SIR = **c**(S=0.19, **I**=0.01, R = 0.8)  resol = ode(y=start\_SIR, **t**=times, func=SIR, p=p)  **plot**(resol[,"time"],resol[,"I"],type="l",xlab="time",ylab="") |

