Epidemic models 1

Ben Bolker

2021-02-22 ©BMB (except textbook material/other images)

motivation

- P & I data from Philadelphia 1918 flu:

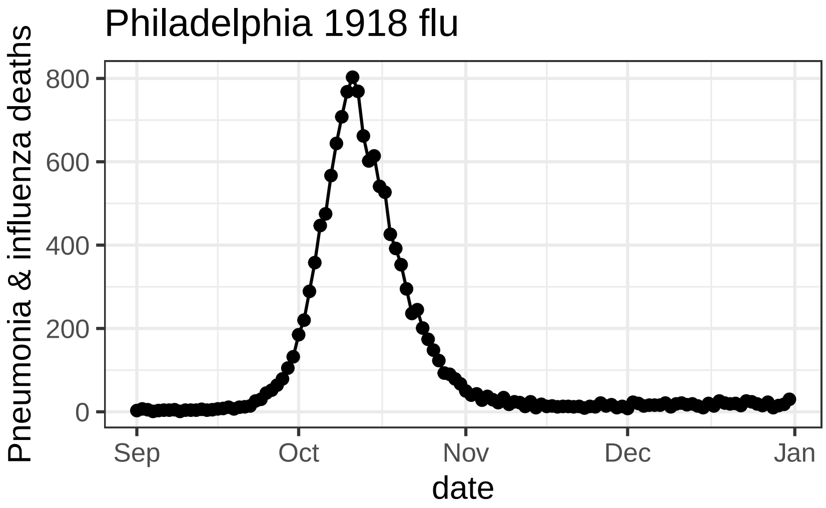


Figure 1: Phila. 1918 flu data

what do we want to figure out?

what shall we assume?

- classify individuals as S, I (**compartmental** model; **microparasite** or **intensity-independent**)
- disease is transmitted from S to I
- $S \to I$ instantaneously (zero latent period, no E)
- population is **homogeneous** (no heterogeneity in susceptibility, infectiousness, contact)
- fixed population size (birth = migration = 'natural' death = 0)
- transmission rate is time-invariant

- assumption 2 is OK (Pasteur, Koch's postulates . . .)
- all the rest are approximations

  start simple!

- parsimony

- robustness?
- applicability/estimation?

Levins (1966) (also Orzack and Sober (1993), Levins (1993), Weisberg (2007))

exponential growth

- one variable (=1D model)
- how does disease spread? → equation

---

what variables should we use?

- time (t)
- state variable: incidence, prevalence, death rate, death toll (= cumulative death?)
- deaths loosely connected to transmission

  but deaths are observed!

---

when are deaths a good **proxy** for incidence?

- infection -> death time is fixed
- homogeneity? (might not matters?)
- mortality curve is shifted epidemic

(COVID context … we observe case reports, number of tests, hospitalizations, and deaths)

- **incidence**: number of infections per unit time (rate or flow)
- **prevalence**: number of currently infected people (quantity or stock)

  prevalence is closer to the **mechanism**

---

model components:

- $I(t)$ (state variable: prevalence)
- $I(0)$ (initial conditions)
- β (parameter) = avg contacts **per susceptible per infective per unit time**

$$I(t + \Delta t) \approx I(t) + \beta I(t)\Delta t$$

Take $\lim \Delta t \to 0$ (and solve):

$$\frac{dI}{dt} = \beta I \to I(t) = I(0)exp(\beta t)$$

model criticism

- Ignored discrete nature of individuals
- Ignored time-varying β (e.g. **diurnal** fluctuations)
- Ignored finite infectious periods (recovery/death)

Next: What if we make infectious periods finite? (i.e., including recovery (**clearance**) or death

$$dI/dt = \beta I - \gamma I$$

mean infectious period

$$\begin{aligned} I(t) &= I(0)\exp(-\gamma t) \\ \text{proportion uninfected} &= \exp(-\gamma t) \\ \text{proportion infected} &= 1 - \exp(-\gamma t)(= \text{CDF} := C(t)) \\ \text{PDF} : &= C'(t) = \gamma \exp(-\gamma t) \\ \text{substitute } x &= \gamma t \quad \to \quad dx = \gamma\, dt \\ \text{mean} &= E[t] = \int t \exp(-\gamma t)\, dt = \int x \exp(-x)\, dx/\gamma = 1/\gamma \end{aligned}$$

dimensional analysis

rates and characteristic times/scales

- is I a proportion or a density or a number ... ?
- what are the units of β, γ ?

nondimensionalization

- standardize any values that can be eliminated **without loss of (mathematical) generality**
- what can we do here?
- $\gamma = 1$
- I ? (depends on how we have defined it initially) $\to I/N$

compare with data???

Original scale:

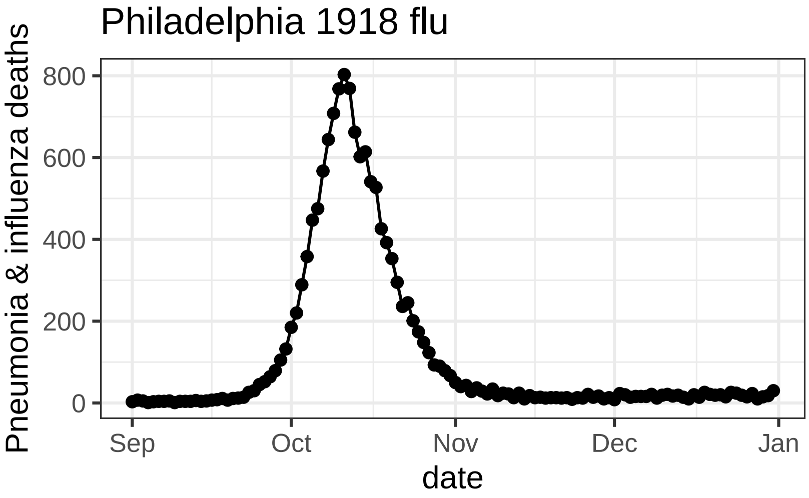


Figure 2: Philadelphia P&I

Log scale:

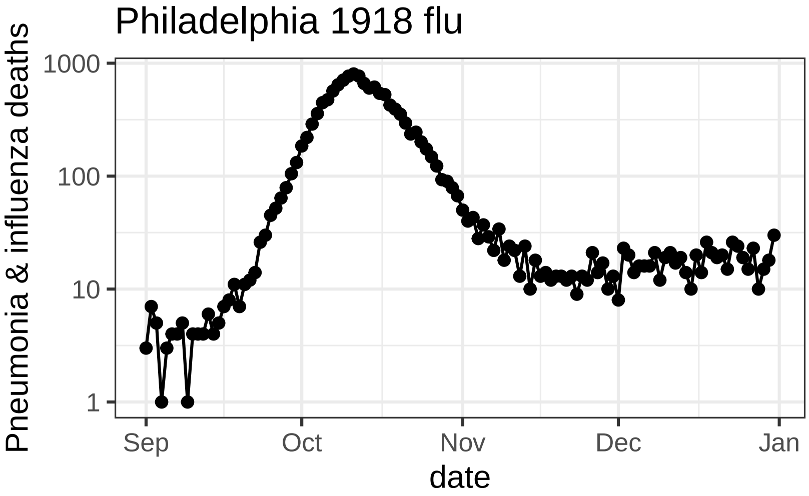


Figure 3: Philadelphia P&I, log scale

- Fit a straight line through the straight part of the curve
- slope is βN
- “intercept” is $\log(I(0))$ (zero is defined in a tricky way)

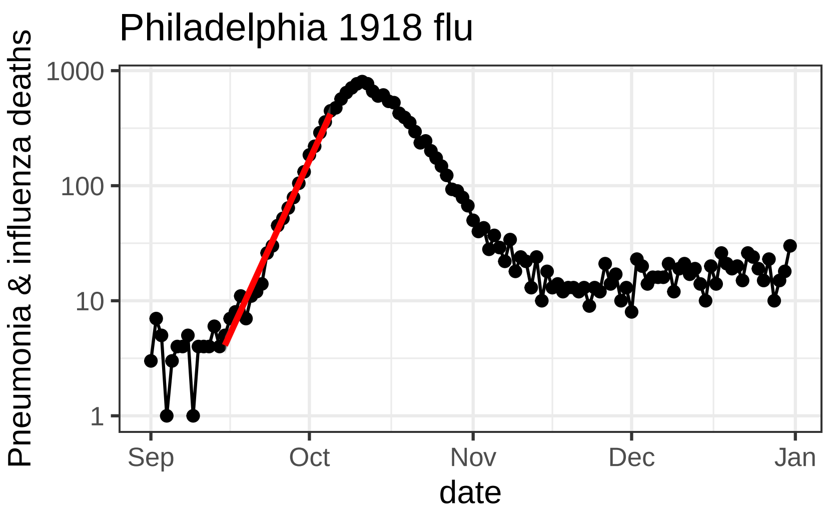
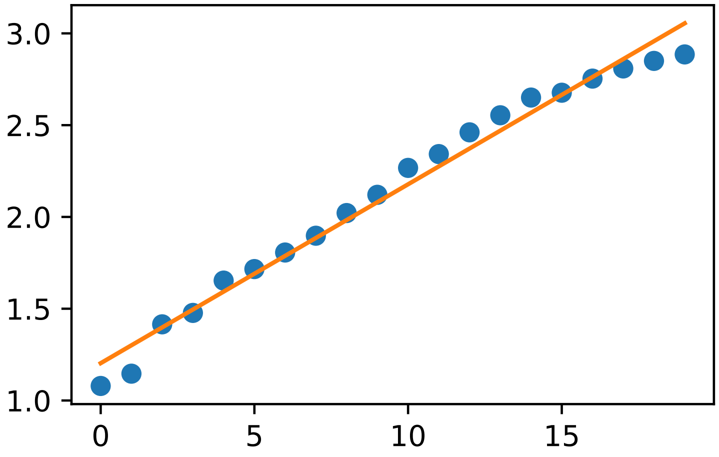


Figure 4: log-scale flu with regression

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
dd = pd.read_csv("data/pim_us_phila_city_1918_dy.csv")
## plt.plot(dd.pim)
## plt.plot(np.log10(dd.pim))
t = np.arange(20)
lw = np.log10(dd.pim)[20:40]
plt.plot(t,lw,'o')
## https://docs.scipy.org/doc/scipy/reference/generated/scipy.linalg.lstsq.html
## https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html
ta = t.reshape(-1,1)  ## make this into a column vector
reg = LinearRegression().fit(ta,lw)
plt.plot(t,reg.intercept_+reg.coef_[0]*t)
```



model assessment

- math is super-easy!
- clear, testable predictions
- parameter estimation is easy
- only consistent over a short time window
  - small t: arbitrarily close to zero
  - large t: ridiculous

Simple (SI) epidemic

- what are we missing?
- **depletion of susceptibles**
- let's take a step back and ignore death & recovery for now

$$dS/dt = -\beta SI$$
$$dI/dt = \beta SI$$

This looks 2D **but** what if we assume $S + I = N$ is constant? Then $S = N - I$

$$dI/dt = \beta(N - I)I$$

How do we solve this? **Partial fractions**

$$
\begin{aligned}
\frac{dI}{\beta(N-I)I} &= dt \\
dI\left(\frac{A}{N-I}+\frac{B}{I}\right) &= dI\cdot\frac{A+B(N-I)}{I(N-I)} \\
A = B; \quad B &= 1/N \\
\frac{1}{\beta N}(-\log(N-I)+\log(I))\Bigg|_{I(0)}^{I} &= t-t_0 \\
(-\log(N-I)+\log(I))\Bigg|_{I(0)}^{I} &= (\beta N)(t-t_0) \quad (\text{set } t_0 = 0) \\
\log\left(\frac{I}{N-I}\right)-\log\left(\frac{I(0)}{N-I(0)}\right) &= \beta N t \\
\log\left(\frac{I}{N-I}\right) &= \beta N t + -\log\left(\frac{I(0)}{N-I(0)}\right) \\
\frac{I}{N-I} &= \exp(\beta N t)\frac{I(0)}{N-I(0)} \equiv Q \\
I &= Q(N-I) \\
I(t)(1+Q) &= QN \\
I(t) &= \frac{QN}{1+Q} = \frac{N}{1+\frac{1}{Q}} \\
&= \frac{N}{1+\left(\frac{N-I(0)}{I(0)}\right)\exp(-\beta N t)}
\end{aligned}
$$

?? $\equiv I(0)\exp(\beta N t)/(1+(I0/N)(\exp(\beta N t)-1))$??

Qualitative analysis

- $I \ll N$? exponential growth
- **per capita growth rate** ($(dI/dt)/I = d(\log(I))/dt$) decreases monotonically with increasing I
- asymptotic behaviour? equilibria? periodic orbits?
- periodic orbits impossible in 1D (uniqueness of flows)

equilibrium analysis

- $I = 0$, **disease free equilibrium** (DFE)
- $I = N$, **endemic equilibrium** (EE)

  Stability? (Assume $\beta > 0$)

- **local asymptotic stability**
- **global asymptotic stability** (Lyapunov functions)

model criticism/conclusions

(Comparison to metapop, logistic growth model)

SIR model

Basic SIR model

- put the pieces together

$$
\begin{aligned}
\frac{dS}{dt} &= -\beta S I \\
\frac{dI}{dt} &= \beta S I - \gamma I \\
\frac{dR}{dt} &= \gamma I
\end{aligned}
$$

- really 2D (because $S + I + R = N$)
- rescale to $N = 1$ (S, I, R as proportions)

Numerical solution (R version):

```r
## define gradient function
SIRgrad <- function(t, y, parms) {
    g <- with(as.list(c(y,parms)), {
        c(-beta*S*I, beta*S*I-gamma*I, gamma*I)
    })
    return(list(g))
}
library(deSolve)
## initial conditions and parameters
y0 <- c(S=0.99, I=0.01, R=0)
p0 <- c(beta=4, gamma=1)
tvec <- seq(0,8,length=101)
## solve (LSODA by default)
sir_R <- ode(y=y0, times=tvec, parms=p0, func=SIRgrad)

## plot
par(las=1,bty="l") ## cosmetic
matplot(tvec, sir_R[,-1],
        type="l", lwd=2, ## solid lines, thicker
        xlab="time",ylab="proportion")
legend("right",names(y0), col=1:3, lty=1:3, lwd=2)
```

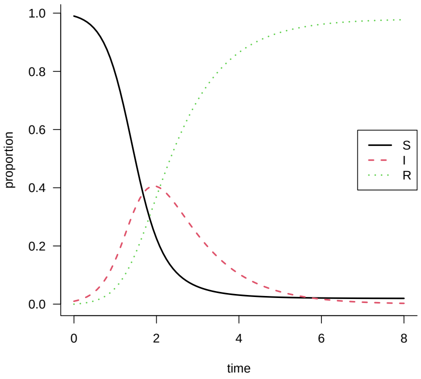


Figure 5: SIR model (R)

Phase plane plot

```
par(las=1,bty="l") ## cosmetic
plot(I~S,type="l",data=as.data.frame(sir_R))
with(as.data.frame(sir_R), points(S,I, cex=0.75,pch=16))
```

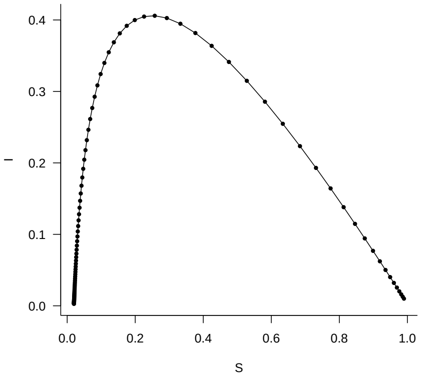
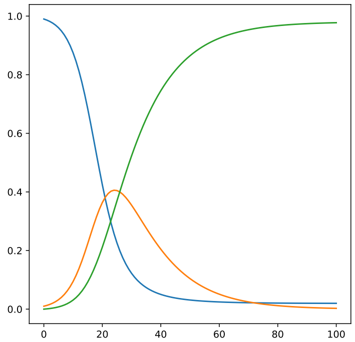


Figure 6: SIR phase plane (R)

Solve using Python

```
import numpy as np
import scipy.integrate
def SIR_grad(x,t,params):
   """basic gradient definitions for SIR model"""
   beta,gamma = params   ## unpack parameters
   S,I,R = x             ## unpack state variables
   return(np.array([-beta*S*I, beta*S*I-gamma*I, gamma*I]))

t_vec = np.linspace(0,8,101)
params = (4,1) ## extra parameters (beta, gamma)
y0 = (0.99, 0.01, 0)
SIR_sol1 = scipy.integrate.odeint(SIR_grad,
                                  y0=y0,
                                        t=t_vec,
                                        args=(params,))

## https://community.rstudio.com/t/how-to-display-the-plot-in-the-python-chunk/22039/3
import matplotlib.pyplot as plt
fig, ax = plt.subplots()
ax.plot(SIR_sol1);
plt.show()
```



dimensional analysis

- initial growth rate (time^{-1}) $\beta - \gamma$
- mean infectious period $1/\gamma$ (time)
- basic reproduction number $\mathcal{R}_0 = \beta/\gamma$

initial growth rate

$$\begin{aligned}\frac{dI}{dt} &= \beta S - \gamma I \\ &= (\beta S - \gamma) I \\ &\approx (\beta - \gamma) I \qquad \text{near } DFE\end{aligned}$$

or calculate **Jacobian** $(\partial X_i / \partial X_j)$:

$$\begin{pmatrix} -\beta I & -\beta S & 0 \\ \beta I & \beta S - \gamma & 0 \\ 0 & \gamma & 0 \end{pmatrix}$$

Evaluate at DFE ($\{1, 0, 0\}$):

$$\begin{pmatrix} 0 & -\beta & 0 \\ 0 & \beta - \gamma & 0 \\ 0 & \gamma & 0 \end{pmatrix}$$

Eigenvalues of this are pretty boring! But useful approach.

Per capita rates

In general we can express *per capita* gradients in X as gradients of $\log(X)$:

$$\begin{aligned}
\frac{dX}{dt} &= Xf(X, Y, Z, \ldots) \\
\frac{\frac{dX}{dt}}{X} &= f(X, Y, Z, \ldots) \\
\frac{d\log(X)}{dt} &= f(X, Y, Z, \ldots)
\end{aligned}$$

Another way to see that $\beta - \gamma$ is the slope on the log scale.

Stability of DFE

- $\beta > \gamma$ ($r > 0$)
- $\beta/\gamma > 1$ ($\mathcal{R}_0 > 1$)

Local asymptotic stability **or**

- $\frac{dI}{dt} = \beta SI - \gamma I$
- non-dimensionalize: $\gamma = 1$, $\beta = \mathcal{R}_0$
- $\frac{dI}{dt} = (\mathcal{R}_0 S - 1)I$
- $\frac{d\log I}{dt} = \mathcal{R}_0 S - 1$

Since $S \leq 1$, $\mathcal{R}_0 < 1 \rightarrow$ deriv of $\log I$ is always negative (don’t really need the last step)

Automated analysis

```
library(phaseR)

## -------------------------------------------------------------------------------
## phaseR: Phase plane analysis of one- and two-dimensional autonomous ODE systems
## -------------------------------------------------------------------------------
##
## v.2.1: For an overview of the package’s functionality enter: ?phaseR
##
## For news on the latest updates enter: news(package = "phaseR")

par(las=1,bty="l",xaxs="i",yaxs="i") ## cosmetic
SIRgrad_2d <- function(t, y, parms) {
    g <- with(as.list(c(y,parms)), {
        c(-beta*S*I, beta*S*I-gamma*I)
    })
    return(list(g))
}
## plot(0:1,0:1,type="n",xlab="S",ylab="I")
f1 <- flowField(SIRgrad_2d,
          xlim=c(0,1),
          ylim=c(0,1),
          parameters=p0,
          state.names=c("S","I"),
          add=FALSE)
n1 <- nullclines(SIRgrad,
                 xlim=c(0,1),
                 ylim=c(0,1),
                 parameters=p0,
                 state.names=c("S","I"))
t1 <- trajectory(SIRgrad_2d,parameters=p0,
           state.names=c("S","I"),
           ## n=10,
           y0=y0[1:2],
           tlim=c(0,5))
```

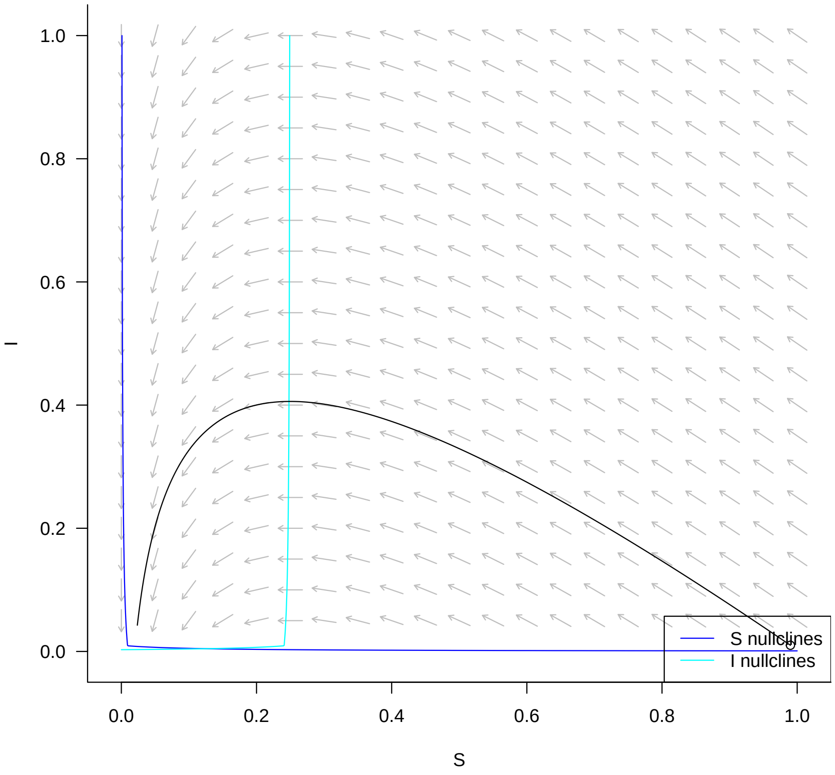


Figure 7: phase plane analysis in R

```
phasePlaneAnalysis(SIRgrad_2d,xlim=c(0,1),
                   parameters=p0,
                   state.names=c("S","I"),
                   ylim=c(0,1))
```

Solution

- can't get analytical solution for $S(t)$, $I(t)$
- **but**: we can solve for $I(S)$:

$$
\begin{aligned}
\frac{dI}{dS} &= \frac{dI/dt}{dS/dt} = -1 + \frac{1}{\mathcal{R}_0 S} \\
\int_{I(0)}^{I} (t)\, dI &= \int_{S(0)}^{S(t)} \left(-1 + \frac{1}{\mathcal{R}_0 S}\right) dS \\
I - I(0) &= -(S - S(0)) + \frac{1}{\mathcal{R}_0} \log(S/S(0)) \\
I + S - (I(0) + S(0)) &= \frac{1}{\mathcal{R}_0} \log(S/S(0))
\end{aligned}
$$

Final size calculations

- $t \to \infty$:

$$
(I_\infty + S_\infty) - (I(0) + S(0)) = \frac{1}{\mathcal{R}_0} \log S_\infty / S(0)
$$

- newly invading pathogen: $S \approx 1$, $I(0) \ll 1$ (≈ 0), $I_\infty \to 0$
- in the limit $I(0) \to 0$:

$$
S_\infty) - 1 = \frac{1}{\mathcal{R}_0} \log S_\infty
$$

- "final size" $Z = 1 - S_\infty$
- $-Z = \frac{1}{\mathcal{R}_0} \log(1 - Z)$

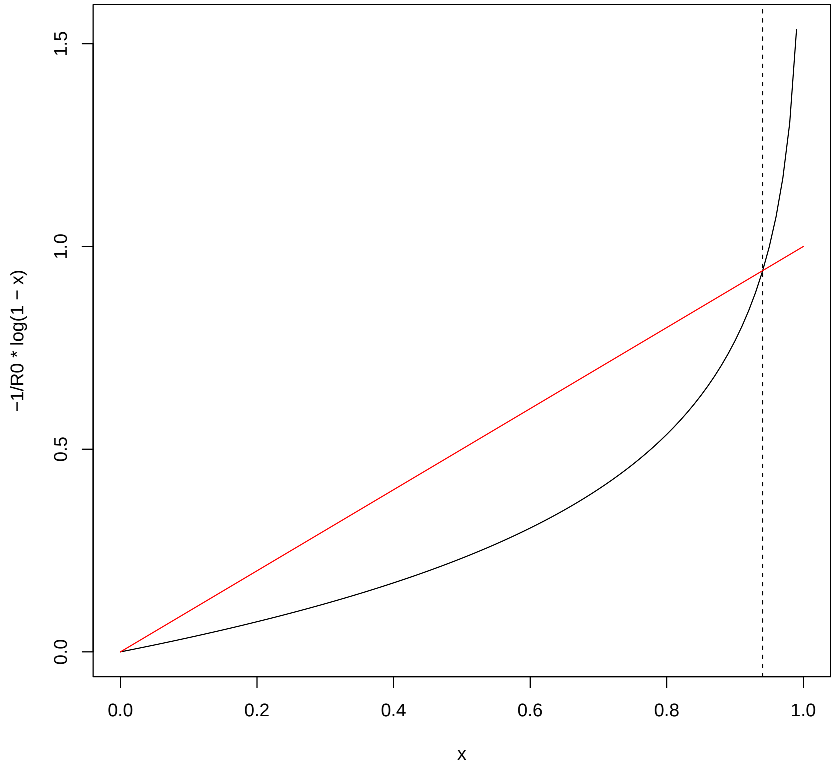
Lambert W functions

- How do we solve this?
- Newton's method (or whatever)
- *Lambert W* (Corless et al. 1996): solves $W \exp(W) = Z$

```
import sympy as sym
z, R = sym.symbols('z R')
sym.solve(sym.Eq(z,-1/R*sym.log(1-z)),z)

## [(R + LambertW(-R*exp(-R)))/R]
```

```
finalsize <- function(R0) {
    1+1/R0*lambertW(-R0*exp(-R0))
}
```

```
R0 <- 3
curve(-1/R0*log(1-x), from=0,to=1)
curve(1*x, add=TRUE,col="red")
library(emdbook)
abline(v=finalsize(R0),lty=2)
```



Epidemic threshold

Assuming vaccination (or other perfect *prophylaxis* [protection]) at rate p

$$R_0 = 1 - 1/p$$

speed-based intervention:

$$\begin{aligned} \beta S I - (\gamma + \phi) I &< 0 \\ I(\beta - \gamma - \phi) &< 0 \\ \phi &> (\beta - \gamma) = r \end{aligned}$$

Comparing Epidemic threshold vs. final size

```
library(emdbook)
finalsize <- function(R0) {
    1+1/R0*lambertW(-R0*exp(-R0))
}
par(las=1,bty="l")
curve(finalsize(x),from=1,to=10,xlab=expression(R[0]),
      ylab="proportion")
curve(1-1/x, add=TRUE, col=2)
legend("bottomright",
       c("final size","herd immunity threshold"),
       col=1:2, lty=1)
```

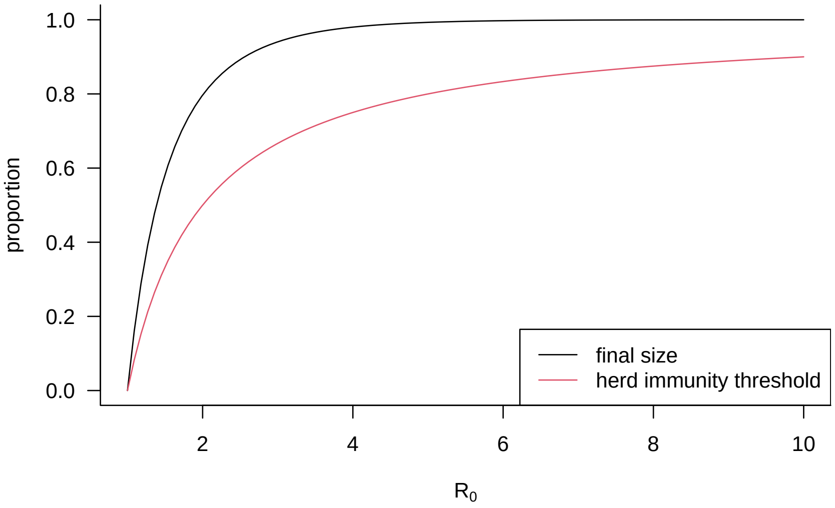


Figure 8: final size vs herd immunity

Estimating R from data

- Euler-Lotka equation

$$\begin{aligned}
I(t) &= \int_0^t I(t-\tau)K(\tau)\,d\tau \\
I(0)\exp(rt) &= \int_0^t I(0)\exp(r(t-\tau))K(\tau)\,d\tau \\
1 &= \int_0^t \exp(-r\tau)K(\tau)\,d\tau \\
1 &= \int_0^t \exp(-r\tau)\mathcal{R}_0 g(\tau)\,d\tau \\
\frac{1}{\mathcal{R}_0} &= \int_0^t \exp(-r\tau)g(\tau)\,d\tau
\end{aligned}$$

SIRS/SIR with vital dynamics

- models of **endemic** disease

- e.g. "childhood diseases" (measles, mumps, rubella, pertussis, polio, chickenpox, . . .)
  - directly transmitted, acute, immunizing
- SIRS model: influenza (evolution), coronaviruses/SARS-CoV-2 (maybe???), cholera (King et al. 2008)
- SIR model *with vital dynamics*

$$
\begin{aligned}
\frac{dS}{dt} &= \mu N - \beta/NSI - \mu S \\
\frac{dI}{dt} &= \beta/NSI - \gamma I - \mu I \\
\frac{dR}{dt} &= \gamma I - \mu R
\end{aligned}
$$

- Balanced population (birth rate = μN). Can consider more complex demography but often don't need to
  - disease-induced death rates low relative to natural mortality
  - demographic time scales much longer than epidemic time scales (exceptions: **chronic** diseases like tuberculosis, HIV/AIDS, diseases of non-human animals an dplants . . .)
- Scaling β/N is much easier for dealing with applications/real data, scaling $N = 1$ is easier for doing math
- $\mathcal{R}_0$ is $\beta/(\mu+\gamma)$ ($\approx \beta/\gamma$ for most human diseases)

Most of the following is taken from Brauer, Castillo-Chavez, and Feng (2019)

$$
\begin{aligned}
S^* &= \frac{\mu+\gamma}{\beta} = 1/\mathcal{R}_0 \quad \text{(this is \textbf{very general})} \\
I^* &= \frac{\mu}{\mu+\gamma} - \frac{\mu}{\beta} = \frac{\mu}{\beta}(\mathcal{R}_0 - 1)
\end{aligned}
$$

- at equilibrium the **force of infection** is βI^*, so the **average age at infection** is $A = 1/(\beta I^*)$
- average lifespan is $L = 1/\mu$
- $L/A = \beta I^*/\mu = \mathcal{R}_0 - 1$
  - another way to estimate $\mathcal{R}_0$! (also, $S^* = 1/\mathcal{R}_0$)
  - tells us something about risk by age, effects of vaccination

Jacobian at EE:

$$
\begin{pmatrix}
-\mu\mathcal{R}_0 & -(\mu+\gamma) \\
\mu(\mathcal{R}_0 - 1) & 0
\end{pmatrix}
$$

Trace = $-\mu\mathcal{R}_0$, Det = $\mu(\mu+\gamma)(\mathcal{R}_0 - 1)$

$$
\begin{aligned}
\lambda &= (1/2)\left(-\mu\mathcal{R}_0 \pm \sqrt{\mu^2\mathcal{R}_0^2 - 4\mu(\mathcal{R}_0-1)(\mu+\gamma)}\right) \\
&\approx (1/2)\left(-\mu\mathcal{R}_0 \pm \sqrt{-4\mu(\mathcal{R}_0-1)\gamma}\right) \\
&= -\frac{\mu\mathcal{R}_0}{2} \pm i\sqrt{(1/L)(L/A)\gamma} \\
&= -\frac{\mu\mathcal{R}_0}{2} \pm i\cdot 1/\sqrt{A\tau}
\end{aligned}
$$

where $\tau = 1/\gamma$ is the infectious period. Both parts of the eigenvalue have units of time^{-1}.

- the period is 2π *divided by* the imaginary part = $2\pi\sqrt{A\tau}$
- e.g. for measles $A \approx 5$yr, $\tau \approx 1/26$, epidemic interval is `2*pi*sqrt(5/26)` $\approx$ 2.76 years.
- **Damping factor**: amplitude decreases by a factor $\exp(-\mu\mathcal{R}_0 T/2)$ per cycle (factor of 1/2 because we are measuring the size of the excursions, not the size of the envelope)

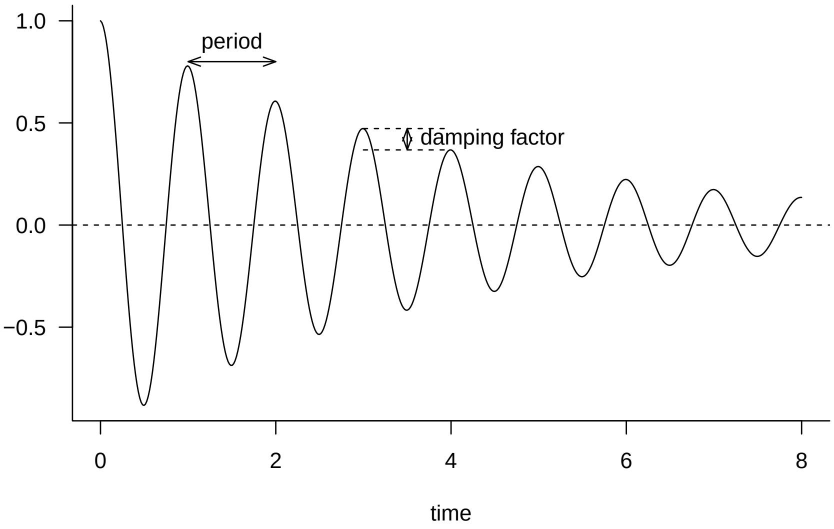


Figure 9: damping

Stochasticity

Allen (2017)

Reed-Frost model

- notes on history: 1927-1928, 1951-1952 (Abbey 1952; Reed 1951), 1976 (Frost 1976)

- household infection model: fixed (small) population, discrete infection generations
- start with **index cases**
- then allow infection: probability of infection = $1-(1-p_i)^I$
- expected number of infections = $S\left(1-(1-p_i)^I\right)$
- $\mathcal{R}_0 = p_i N$
- **hazard** interpretation
  - probability of infection per I per small Δt = β
  - prob of **non-infection** by 1 inf at time τ = $\exp(-\beta\tau) = 1-p_i$
  - prob of non-inf by I inf = $\exp(-\beta\tau)^I = \exp(-(\beta I)\tau)$
  - hazard $\equiv$ FOI
- can do standard analysis of equilibria, stability, etc. ($\mathbf{X}_{t+1} = \mathbf{X}_t$; stability based on $|\lambda| \lessgtr 1$)
- what is $\mathcal{R}_0$?
- r vs R relationship: $\mathcal{R}_0 = (1+\kappa r\bar{G})^{1/\kappa}$
  - κ is the the reciprocal of the *shape parameter* ($\kappa = \mathrm{CV}^2$)
  - SIR: $\kappa = 1$, $\mathcal{R}_0 = 1 + r\bar{G}$. R-F: $\kappa \to 0$ so $\mathcal{R}_0 = \exp(r\bar{G})$

discrete-time stochastic R-F

- stochastic version: $I_{t+1} \sim \mathrm{Binom}\left(S_t, 1-(1-p_i)^{I_t}\right)$

```
set.seed(101)
rf <- function(nt,y0,p_i) {
    res <- numeric(nt)
    S <- y0[["S"]]
    I <- res[1] <- y0[["I"]]
    for (t in 2:nt) {
        I <- rbinom(1, prob=1-(1-p_i)^I, size=S)
        S <- S-I
        if (I==0) break
        res[t] <- I
    }
    return(res)
}
r0 <- rf(20, y0=c(S=99,I=1), p_i=0.02)
r1 <- replicate(300, rf(20, y0=c(S=99,I=1), p_i=0.02))
par(las=1,bty="l")
matplot(r1,type="l",col=adjustcolor("black",alpha.f=0.2),lty=1)
lines(rowMeans(r1),col=2,lwd=3)
```

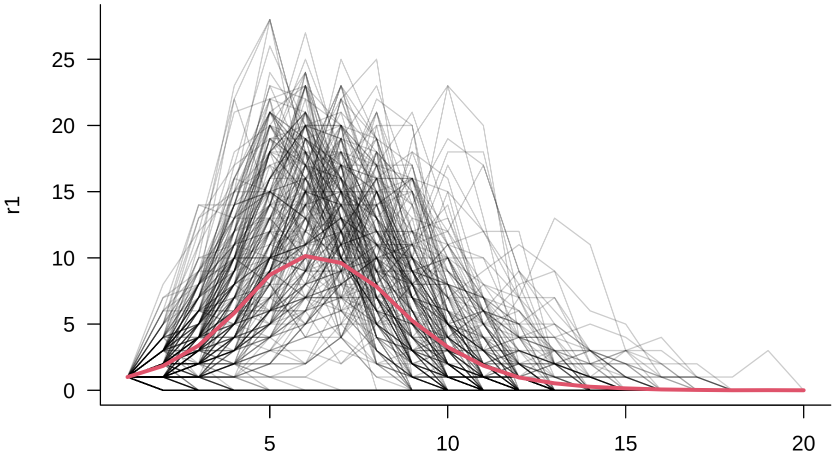


Figure 10: Reed-Frost ensemble

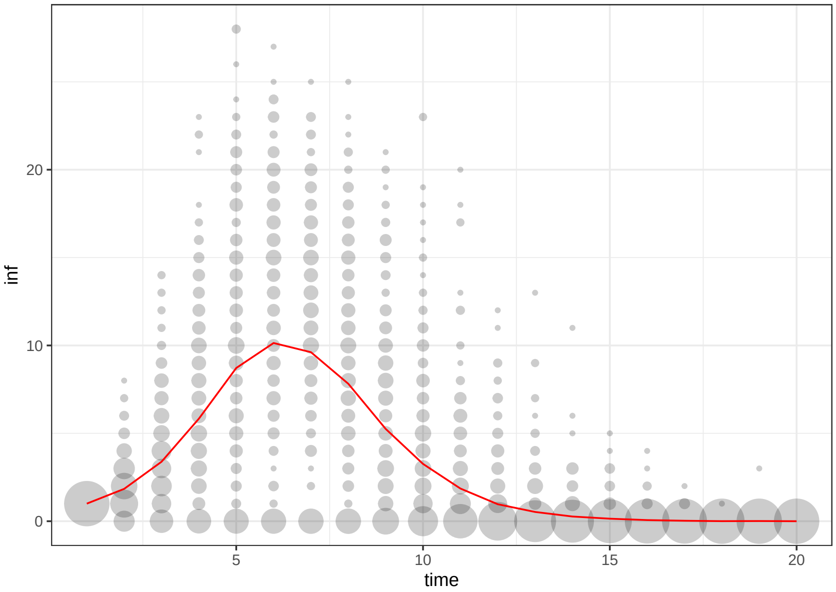


Figure 11: Reed-Frost ensemble 2

Gillespie algorithm

- define all processes in terms of *rates* (SIR: βSI, γI), rather than gradients
- *Poisson process*: exponentially distributed

```
library(GillespieSSA2)
parms <- c(beta = 5 , gamma = 1, N=100)
final_time <- 10
initial_state <- c(S = 99, I=1, R=0)
reactions <- list(
  reaction("beta*S*I/N",  c(S = -1, I=+1), name="transmission"),
  reaction("gamma*I",    c(I = -1, R = +1), name="recovery")
)
set.seed(1)
g1 <- ssa(initial_state,reactions,final_time,parms,
          method=ssa_exact(),
          sim_name="SIR")

plot_ssa(g1) + geom_step()
```

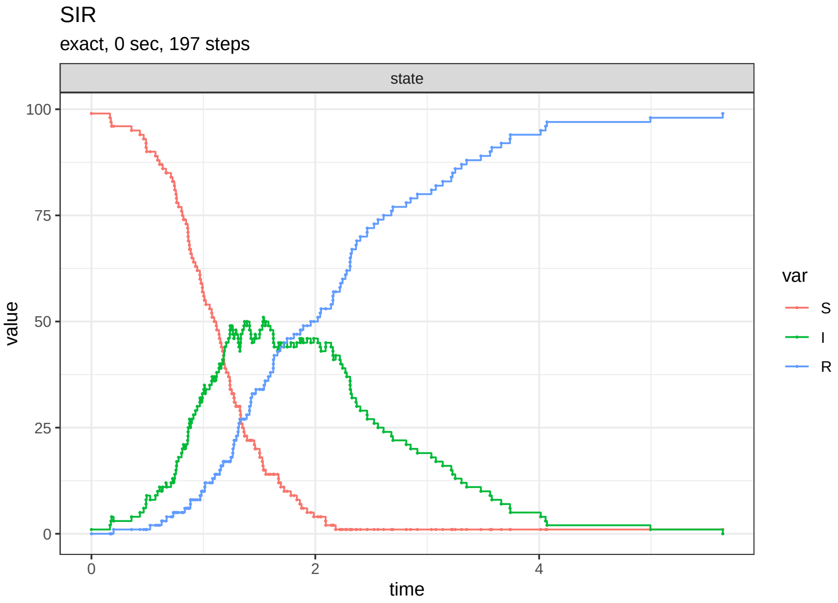
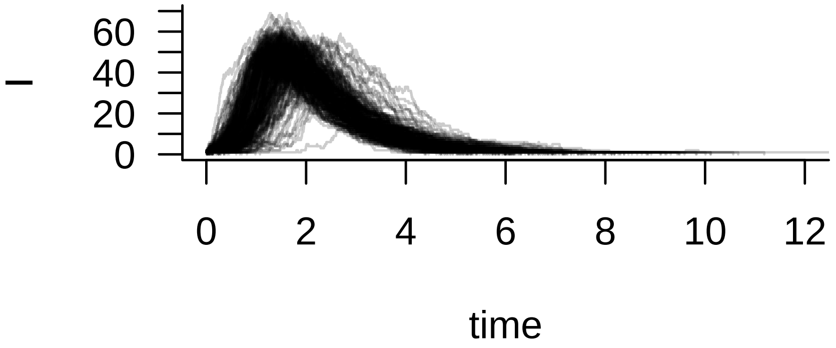


Figure 12: Gillespie realization



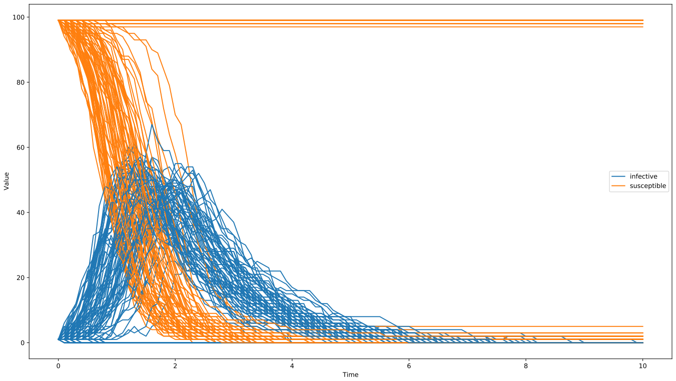
(See here)

```
import gillespy2
class SIRv(gillespy2.Model):
    def __init__(self, parameter_values=None):
        # First call the gillespy2.Model initializer.
        gillespy2.Model.__init__(self, name='SIRv')
        # Define parameters for the rates of creation and dissociation.
        ## can't use expressions in reaction rates ... ?? scale by N
        beta = gillespy2.Parameter(name='beta', expression=0.05)
        gamma= gillespy2.Parameter(name='gamma', expression=1)
        N= gillespy2.Parameter(name='N', expression=100)
        self.add_parameter([beta, gamma, N])
        # Define variables for the molecular species representing M and D.
        S = gillespy2.Species(name='susceptible', initial_value=99)
        I = gillespy2.Species(name='infective',   initial_value=1)
        self.add_species([S, I])
        r_inf = gillespy2.Reaction(name="r_infection", rate=beta, reactants={S:1,I:1}, products={I:2})
        r_rec = gillespy2.Reaction(name="r_recovery", rate=gamma, reactants={I:1}, products={})
        self.add_reaction([r_inf, r_rec])
        # Set the timespan for the simulation.
        self.timespan(np.linspace(0, 10, 101))

model = SIRv()
results = model.run(number_of_trajectories=100)

results.plot();
plt.show()
```

stochastic ODEs

- take limits of very large population size, short time, so that stochastic changes become a *Wiener process* (continuous, non-differentiable, changes are Gaussian)

Figure 13: Gillespie realization (Python)



$$dX(t) = \underbrace{f(X(t))\,dt}_{\text{deterministic}} + \underbrace{G(X(t))\,dW(t)}_{\text{stochastic}}$$

- $f(X(t))$ is the original gradient vector
- $G(X(t))$:
  - need $(G\,dW)(G\,dW)^T$ to equal the **covariance** of ΔX.

$$C = E\begin{pmatrix} (\Delta S)^2 & \Delta S \Delta I \\ \Delta S \Delta I & (\Delta I)^2 \end{pmatrix} = \begin{pmatrix} \beta SI/N & -\beta SI/N \\ -\beta SI/N & \beta SI/N + \gamma I \end{pmatrix} \Delta t$$

Need the **matrix square root** of C, i.e. G such that $GG^T = C$. (For large/complex methods we probably want to use a numerical method such as **Cholesky decomposition**. In this case:

$$G = \begin{pmatrix} -\sqrt{\beta SI/N} & 0 \\ \sqrt{\beta SI/N} & -\sqrt{\gamma I} \end{pmatrix}$$

- **Euler-Maruyama** method: Euler integration + noise scaled by $\sqrt{\Delta t}$

```
def SIR_2d_grad_stoch(x,t,params):
   """basic gradient definitions for SIR model"""
   beta,gamma,N = params   ## unpack parameters
   S,I = x                ## unpack state variables
   incid = beta*S*I/N
   grad = np.array([-incid, incid-gamma*I])
   G = np.matrix([[-np.sqrt(incid), 0], [np.sqrt(incid), -np.sqrt(gamma*I)]])
   return((grad,G))
```

```
def em_step(y0, t, params, func, dt):
    """take a single Euler-Maruyama step"""
    grad, G  = func(y0, t, params)
    nx = len(y0)
    stoch = np.matmul(G, np.random.normal(size=(nx,1)))*np.sqrt(dt)
    return y0 + grad*dt + stoch.reshape(nx)

dt = 0.001
t_vec = np.arange(0,8,0.001)
params = (4,1,100) ## extra parameters (beta, gamma,N)
y0 = (99, 1)
em_step(y0, 0, params, SIR_2d_grad_stoch, 0.001)
```

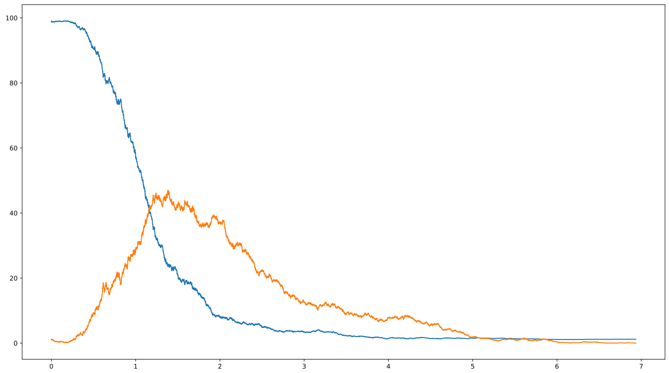
```
## matrix([[98.9106254 ,  1.08772802]])
```

```
np.random.seed(101)
res = np.zeros(shape=(len(t_vec),2))
res[0,:] = y0
for i in range(1,len(t_vec)):
    res[i,:] = em_step(res[i-1,:], t, params, SIR_2d_grad_stoch, dt)
```

```
## /home/bolker/.local/share/r-miniconda/envs/r-reticulate/bin/python:7: RuntimeWarning: invalid value enc
```

```
plt.plot(t_vec,res);
plt.show()
```

Figure 14: Euler-Maruyama realization



Sustained oscillations in epidemic systems

Problem

Using measles data from Ontario:

```
on_meas <- read.csv("data/meas_ca_on__1939-89_wk.csv",skip=3)
plot(cases~numdate,data=on_meas,type="l")
```

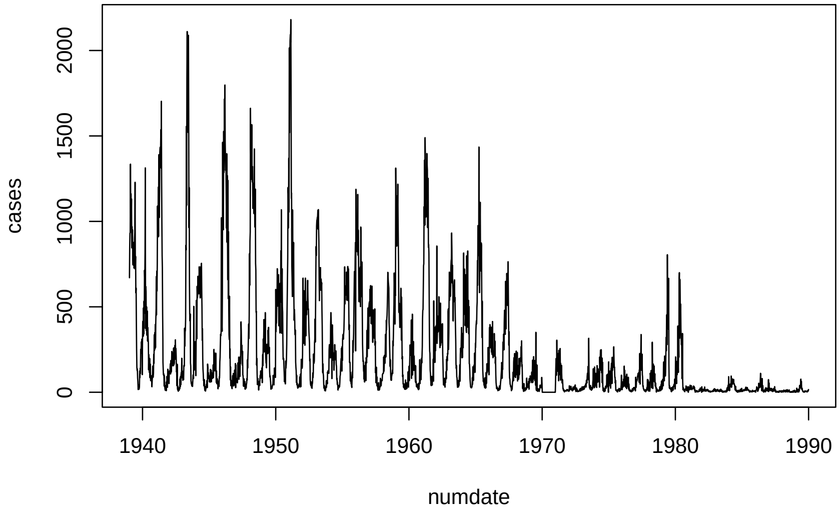


Figure 15: Ontario measles data

```
library(dplR)
m <- on_meas[on_meas$numdate<1970,]
## https://rstudio-pubs-static.s3.amazonaws.com/9428_1197bd003ebd43c49b429f22ea4f36e5.html
wave.out <- morlet(m$cases,m$numdate)
wave.out$period <- wave.out$period/52
wavelet.plot(wave.out,key.col=heat.colors(10),useRaster=TRUE)
```

Bartlett cycles

```
N <- 100000; R0 <- 6; infper <- 1/26
parms <- c(beta = R0/infper , gamma = 1/infper, N=N, mu=1/50)
eq <- with(as.list(parms), c(S=N/R0, I=N*(mu/beta)*(R0-1),
                             R=N*(1-1/R0-(mu/beta)*(R0-1))))
final_time <- 200
reactions <- list(
  reaction("mu*N",  c(S = +1), name="birth"),
  reaction("mu*S",  c(S = -1), name="S_death"),
  reaction("mu*I",  c(I = -1), name="I_death"),
  reaction("mu*R",  c(R = -1), name="R_death"),
  reaction("beta*S*I/N",  c(S = -1, I=+1), name="transmission"),
  reaction("gamma*I",     c(I = -1, R=+1), name="recovery"),
  reaction("immig", c(I=+1, R=-1), name="cheat")
)
set.seed(3)
```

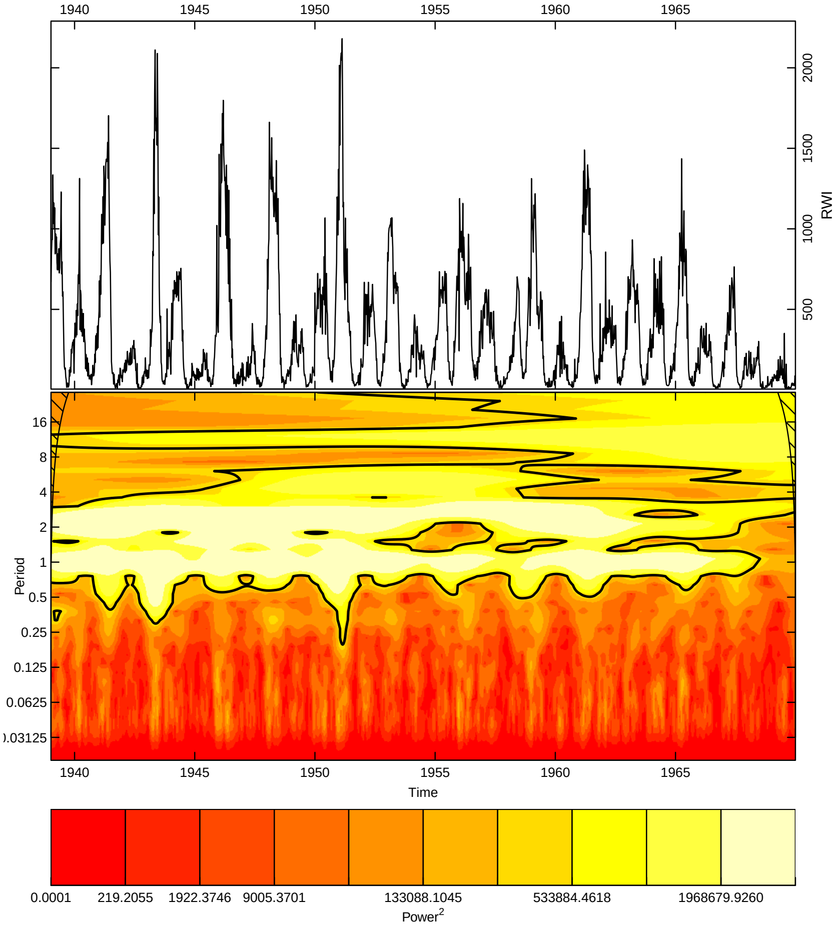


Figure 16: wavelet analysis of measles data

```
g2 <- ssa(round(eq),
          reactions,
          final_time,
          c(parms,immig=30),
          census_interval=1/52,
          method=ssa_exact(), ## ode_em(noise_strength=100),
          sim_name="SIR_vital")
```

```
with(g2,plot(state[(52*190):nrow(state),"I"],type="l"))
```

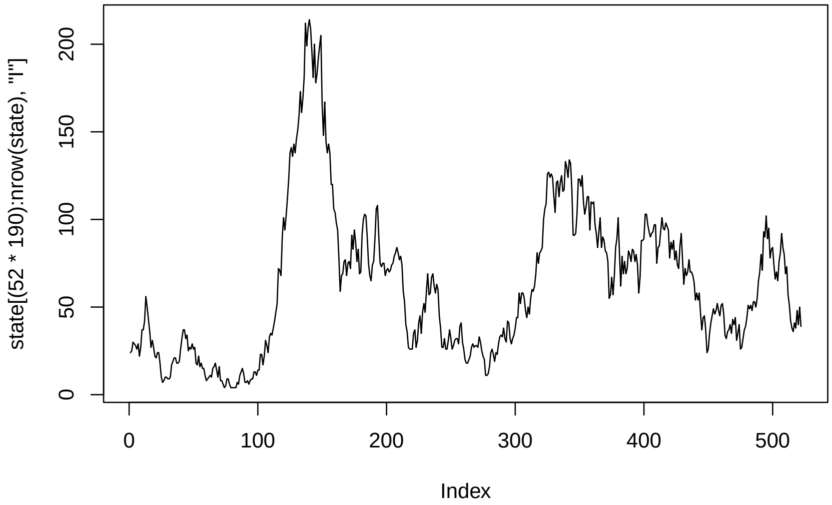


Figure 17: Bartlett cycles

seasonal forcing

```
N <- 1; R0 <- 15; infper <- 1/26
parms <- c(beta = R0/infper , gamma = 1/infper, N=N, mu=1/50, delta=0.2)
eq <- with(as.list(parms), c(S=N/R0, I=N*(mu/beta)*(R0-1),
                             R=N*(1-1/R0-(mu/beta)*(R0-1))))
SIRgradv <- function(t, y, parms) {
    g <- with(as.list(c(y,parms)), {
        beta1 <- beta*(1+delta*sin(2*pi*t))
        c(-beta1*S*I/N+mu*(N-S), beta1*S*I/N-(mu+gamma)*I, gamma*I-mu*R)
    })
    return(list(g))
}
mm <- ode(y=eq, times=seq(0,50,by=1/52), func=SIRgradv, parms=parms)
```

```
plot(I~time, as.data.frame(mm),type="l")
```

See Earn (2009) for more (PDF here) . . .

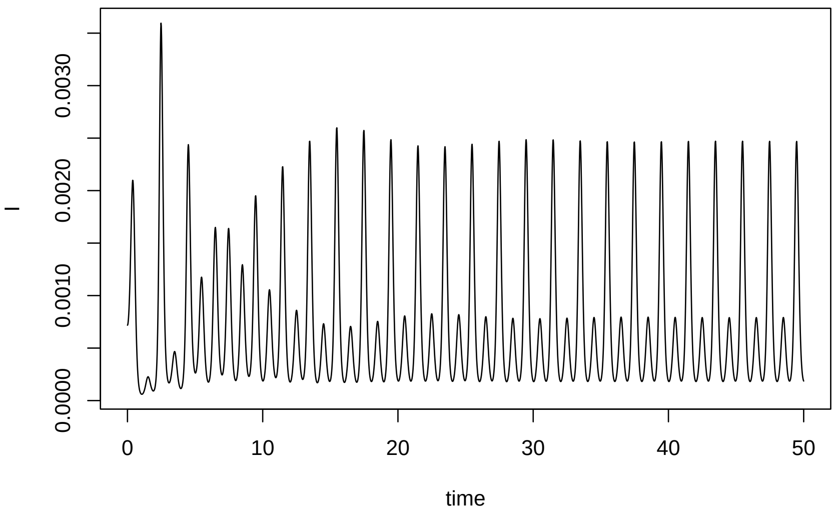


Figure 18: seasonally forced SIR

References

Abbey, Helen. 1952. "An Examination of the Reed-Frost Theory of Epidemics." *Human Biology* 24 (3): 201–33. https://search.proquest.com/docview/1301828990/citation/D63A6C9BEF294594PQ/1.

Allen, Linda J. S. 2017. "A Primer on Stochastic Epidemic Models: Formulation, Numerical Simulation, and Analysis." *Infectious Disease Modelling* 2 (2): 128–42. https://doi.org/10.1016/j.idm.2017.03.001.

Brauer, Fred, Carlos Castillo-Chavez, and Zhilan Feng. 2019. "Endemic Disease Models." In *Mathematical Models in Epidemiology*, edited by Fred Brauer, Carlos Castillo-Chavez, and Zhilan Feng, 63–116. Texts in Applied Mathematics. New York, NY: Springer. https://doi.org/10.1007/978-1-4939-9828-9_3.

Corless, Robert M., G. H. Gonnet, D. E. G. Hare, D. J. Jeffrey, and D. E. Knuth. 1996. "On the Lambert W Function." *Advances in Computational Mathematics* 5 (4): 329–59. https://doi.org/10.1007/BF02124750.

Earn, David J. D. 2009. "Mathematical Epidemiology of Infectious Diseases." In *Mathematical Biology*, edited by Mark Lewis, Mark Chaplain, James Keener, and Philip Maini. Vol. 14. IAS/Park City Mathematics Series. American Mathematical Society. https://doi.org/https://doi.org/10.1090/pcms/014.

Frost, Wade Hampton. 1976. "Some Conceptions of Epidemics in General." *American Journal of Epidemiology* 103 (2): 141–51. https://doi.org/10.1093/oxfordjournals.aje.a112212.

King, Aaron A., Edward L. Ionides, Mercedes Pascual, and Menno J.

Bouma. 2008. "Inapparent Infections and Cholera Dynamics." *Nature* 454 (7206): 877–80. https://doi.org/10.1038/nature07084.

Levins, R. 1966. "The Strategy of Model Building in Population Biology." *American Scientist* 54: 421–31. https://www.jstor.org/stable/27836590.

Levins, Richard. 1993. "A Response to Orzack and Sober: Formal Analysis and the Fluidity of Science." *Quarterly Review of Biology* 68 (4): 547–55.

Orzack, Steven Hecht, and Elliott Sober. 1993. "A Critical Assessment of Levins's the Strategy of Model Building in Population Biology (1966)." *Quarterly Review of Biology* 68 (4): 533–46.

Reed, Lowell. 1951. "Epidemic Theory: What Is It? (Part 1 of 4)." https://www.youtube.com/watch?v=xYaFebO3kAU.

Weisberg, Michael. 2007. "Forty Years of 'the Strategy': Levins on Model Building and Idealization." *Biology & Philosophy* 21 (5): 623–45. https://doi.org/10.1007/s10539-006-9051-9.