Discrete-time deterministic models

Modern Techniques in Modelling

LONDON SCHOOL of HYGIENE & TROPICAL MEDICINE

Introduction

LONDON SCHOOL of HYGIENE &TROPICAL MEDICINE

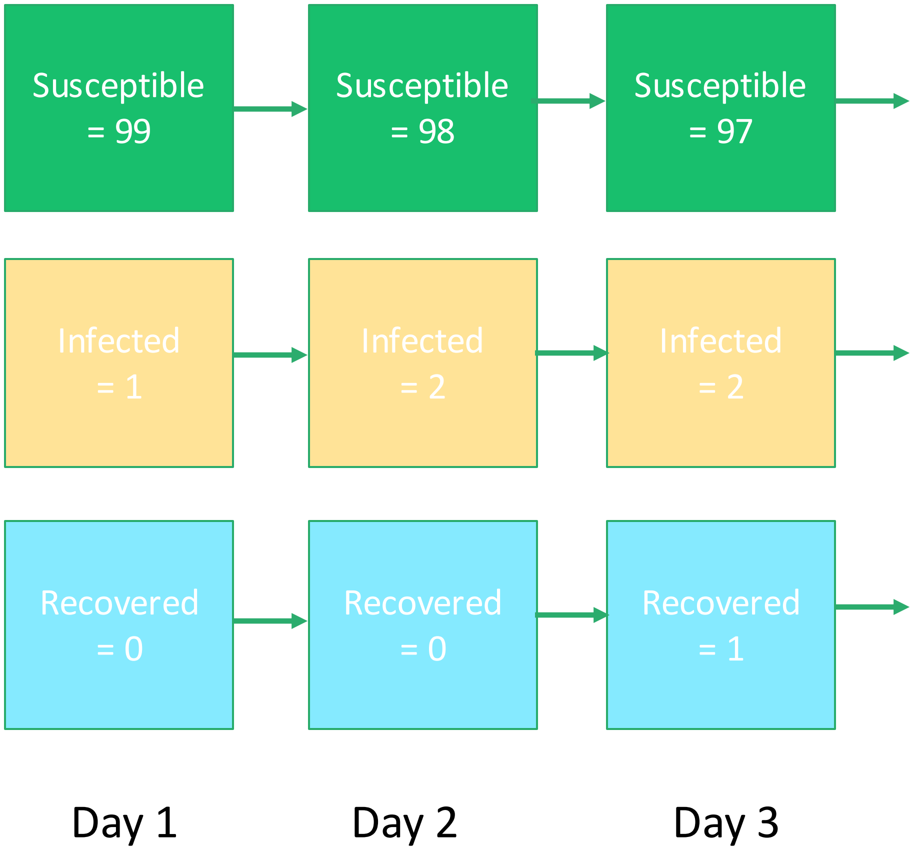
Introduction

- Welcome to the lecture on Discrete-time deterministic models

- This session's objectives:
    - Understand the concept of discrete-time models
    - Learn about difference equations
    - Explore the SIR (Susceptible-Infectious-Recovered) model in discrete time
    - Implement and analyse a discrete-time SIR model in R

Discrete-time deterministic models

Often we may want to model an epidemic in terms of a discrete time step, e.g. from one day to the next:



Discrete-time deterministic models

This can be modelled by considering the current state of the epidemic, *y*, at time *t:*

$$\mathbf{y}(t) = \big(S(t),\ I(t),\ R(t)\big)$$

that updates with each time step by some function of the current state*:*

$$\mathbf{y}(t+1) = \mathbf{y}(t) + g(\mathbf{y}(t))$$

This is known generally as a **difference equation**, the solution of which can be solved exactly by iteratively applying the update function for each discrete time step.

Example: If you have £100 in a bank account and earn 5% interest yearly, next year you'd have: £100 * (1 + 0.05) = £105

Discrete-time deterministic models

Consider an SIR model for a closed population that follows the following rules:

- Susceptible individuals become infected at a rate proportional to the size of the product of *susceptible* and *infectious* populations:

$$S(t+1) = S(t) - \beta * S(t) * I(t)$$

where β is the per capita infection rate.

Discrete-time deterministic models

- Recovery from infection grants life-long immunity

$$R(t+1) = R(t) + \gamma * I(t)$$

Where γ is the recovery rate.

(The mean time spent infectious is then $1/\gamma$ (1/rate = duration)).

Discrete-time deterministic models

The only transitions between states in our model are infection and recovery

$$I(t+1) = I(t) + \beta * S(t) * I(t) - \gamma * I(t)$$

Which adds those being infected and subtracts those who recover

SIR example

We start with:

- 1% of the population are infected, $I(0) = 0.01$
- no recovered, $R(0) = 0$, and
- the remainder of the population is susceptible, $S(0) = 0.99$,
- for $N(t) = 1$, the total population.

Our system is therefore:

$$
\begin{aligned}
S(t+1) &= S(t) - \beta * S(t) * I(t), & S(0) &= 0.99 \\
I(t+1) &= I(t) + \beta * S(t) * I(t) - \gamma * I(t), & I(0) &= 0.01 \\
R(t+1) &= R(t) + \gamma * I(t), & R(0) &= 0
\end{aligned}
$$

SIR example

This system can also be written in vector form as:

$$\begin{bmatrix} S \\ I \\ R \end{bmatrix}_{t+1} = \begin{bmatrix} S \\ I \\ R \end{bmatrix}_t + \begin{bmatrix} -\beta SI \\ \beta SI - \gamma I \\ \gamma I \end{bmatrix}_t$$

Here the subscript indicates the time value and the term on the right is our *update vector.*

SIR example

In R, we can write a function to return our update vector with three elements:

```
update_sir <- function(t, y, parms){
  S <- y[1]
  I <- y[2]
  R <- y[3]

  beta  <- parms['beta']
  gamma <- parms['gamma']

  out <- c(- beta*S*I,
           + beta*S*I - gamma*I,
           + gamma*I)

  return(out)
}
```

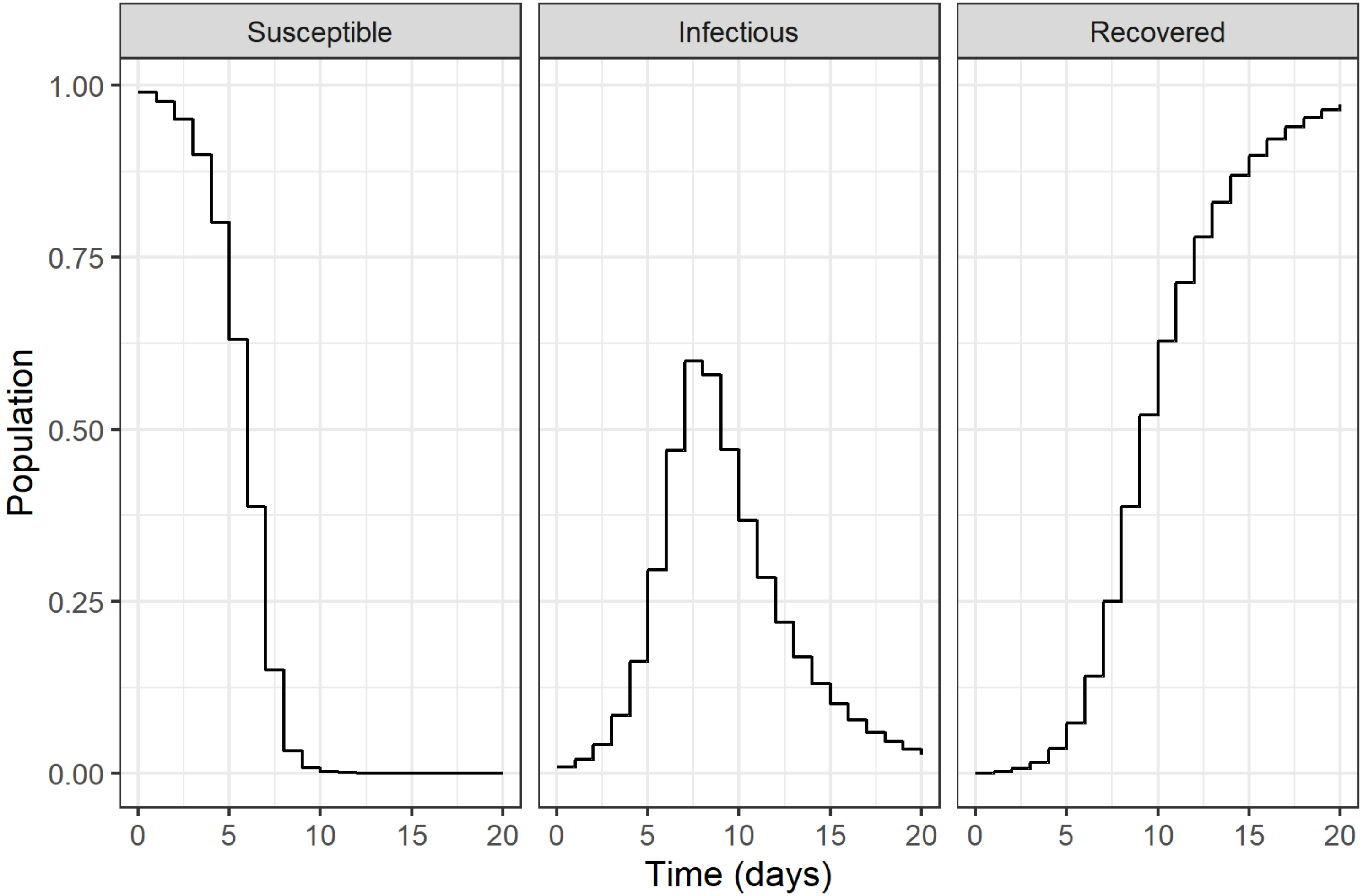
Discrete-time deterministic SIR

```
parms_sir <- c(beta  = 1.3,
               gamma = 0.23)

time_sir <- seq(0, 20, by = 1)

y_sir    <- matrix(data = NA,
                   nrow = length(time_sir),
                   ncol = 3)

# initial values at t=0
y_sir[1, ] <- c(0.99, 0.01, 0)

for (i in 2:(nrow(y_sir))){

  y_sir[i,] <- y_sir[i - 1,] +
    update_sir(t     = time_sir[i],
               y     = y_sir[i - 1, ],
               parms = parms_sir)
}
```

Discrete-time deterministic SIR

```
y_sir_df <- as.data.frame(y_sir)
names(y_sir_df) <- c('Susceptible',
                     'Infectious',
                     'Recovered')
y_sir_df <- cbind(time = time_sir,
                  y_sir_df)

head(y_sir_df)
##   time Susceptible Infectious  Recovered
## 1    0   0.9900000 0.01000000 0.00000000
## 2    1   0.9771300 0.02057000 0.00230000
## 3    2   0.9510006 0.04196833 0.00703110
## 4    3   0.8991151 0.08420110 0.01668382
## 5    4   0.8006967 0.16325327 0.03605007
## 6    5   0.6307654 0.29563626 0.07359832
```

Discrete-time deterministic SIR



Discrete-time deterministic SIR

Our more general form of the update is

$$\mathbf{y}(t+1) = f\left(t, \mathbf{y}(t), \boldsymbol{\theta}(t)\right)$$

…but parameters may or may not change with time *t.*

Summary

LONDON SCHOOL of HYGIENE & TROPICAL MEDICINE

Summary

- Difference equations are defined for discrete time steps, and can be solved by iterating over those time steps.
- Discrete time models can be specified either as:

$\mathbf{y}(t + \Delta t) = f(\mathbf{y}(t), t, \boldsymbol{\theta}, \Delta t)$ to transform current state

$\mathbf{y}(t + \Delta t) = y(t) + \Delta t\, g(\mathbf{y}(t), t, \boldsymbol{\theta})$ to update current state

- $f(\cdot), g(\cdot)$ can be any function that captures the dynamics of the physical system we're interested in.

Looking forward

- We’ll now look at implementing discrete-time models in R in the practical.
- Later today you’ll start to learn about extending to continuous time models with **differential equations.**
- Later in the week you’ll learn about update functions that model infection and recovery as probabilistic events.

Practical

LONDON SCHOOL of HYGIENE & TROPICAL MEDICINE