

Introduction to Fitting an SIR model practical

Dr Juliette Unwin

Modern Statistics and Machine Learning
for Population Health in Africa

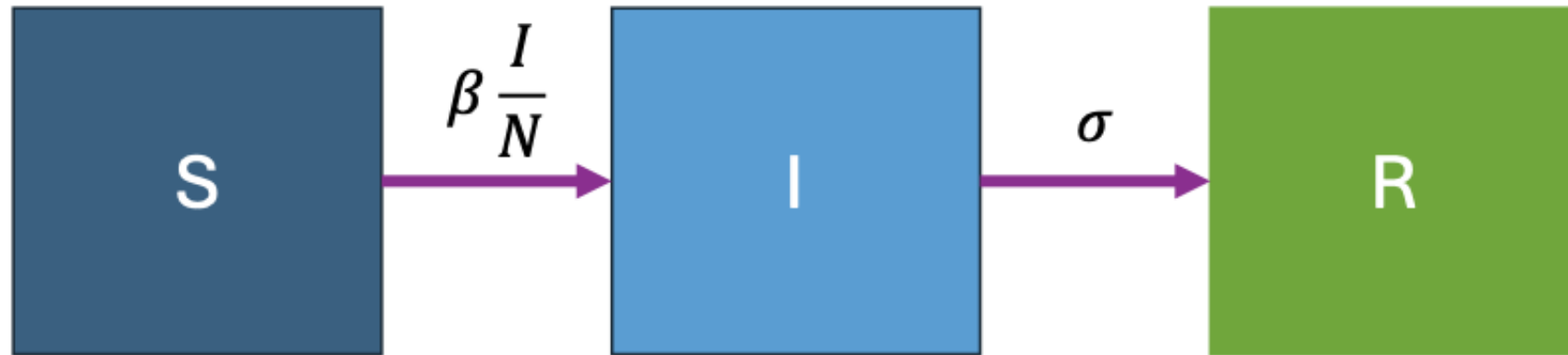
23rd February 2024

Fitting two different models: SIR model



MRC Centre for
Global Infectious
Disease Analysis

Imperial College
London

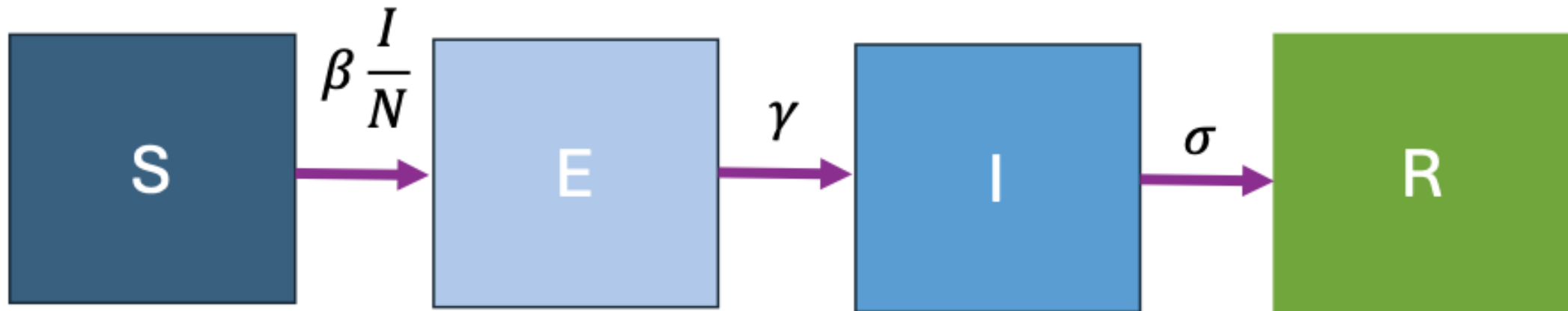


Fitting two different models: SEIR model



MRC Centre for
Global Infectious
Disease Analysis

Imperial College
London



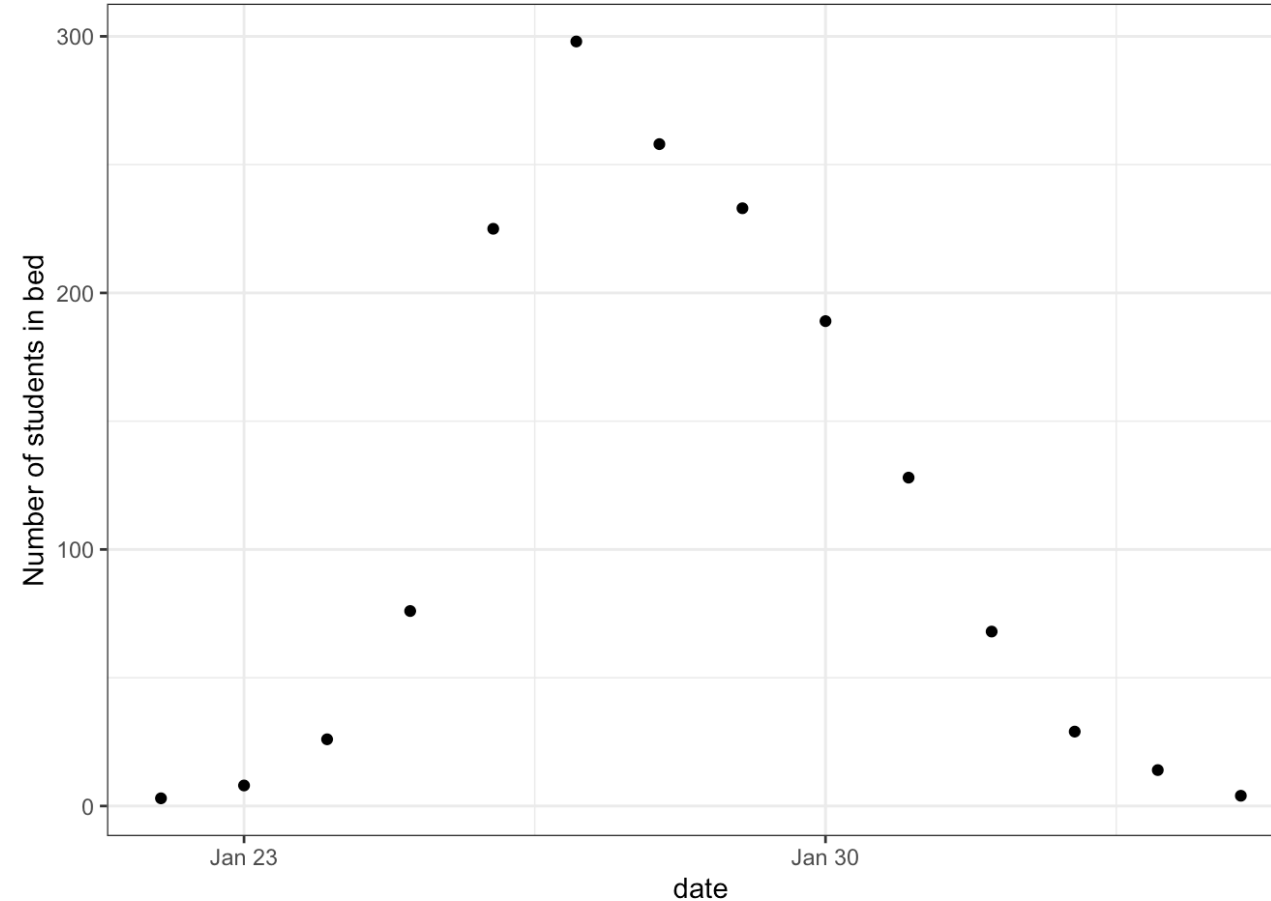
To two types of data: Prevalence



MRC Centre for
Global Infectious
Disease Analysis

Imperial College
London

We examine an outbreak of influenza A (H1N1) in 1978 at a British boarding school. The data consists of the daily number of students in bed (prevalence data), spanning over a time interval of 14 days.



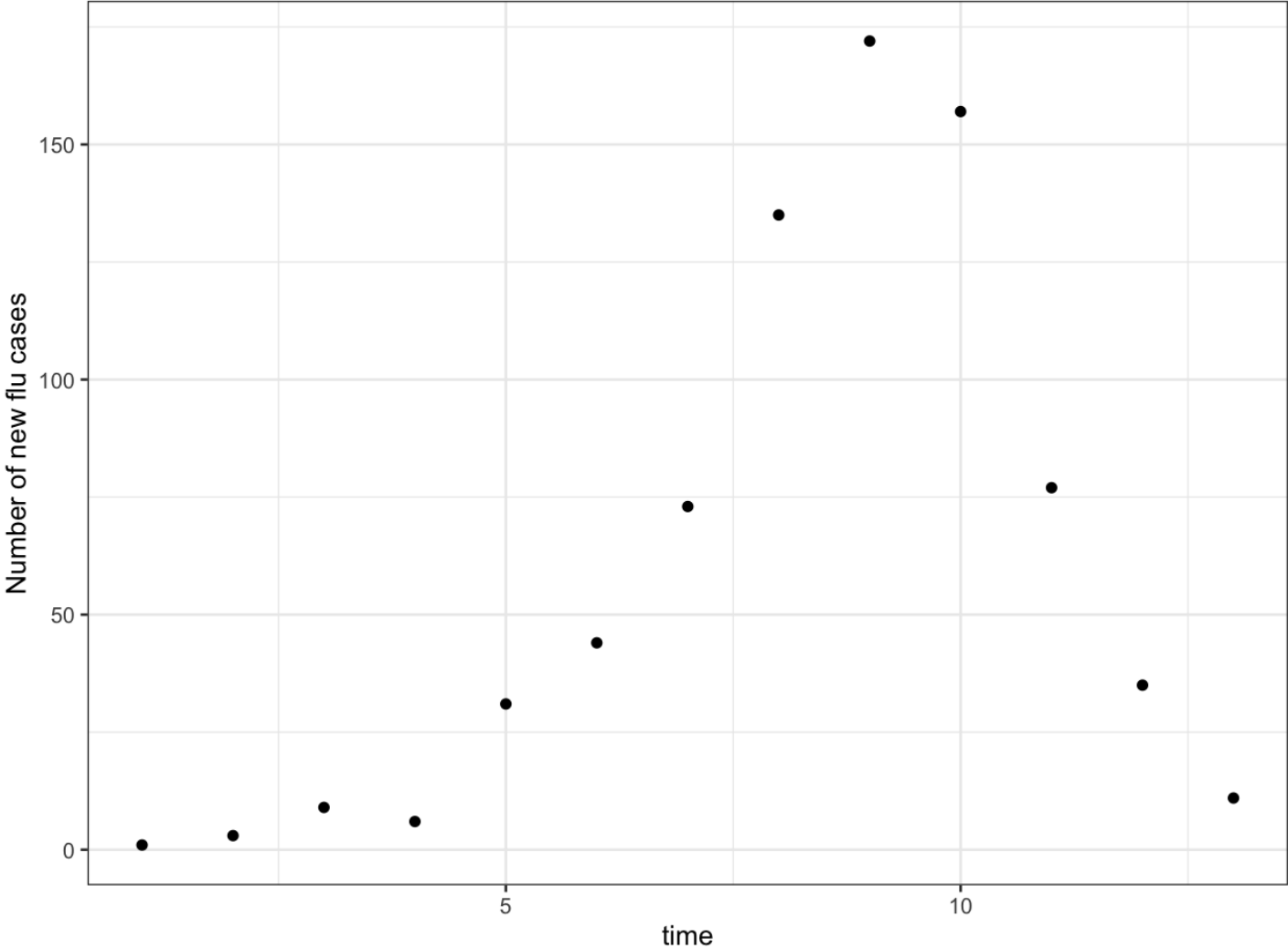
To two types of data: Incidence



MRC Centre for
Global Infectious
Disease Analysis

Imperial College
London

New cases of flu each day



Stan model file



MRC Centre for
Global Infectious
Disease Analysis

Imperial College
London

```
functions {  
  vector sir(real t,  
             vector y,  
             real beta,  
             real sigma,  
             real N) {  
  
    vector[3] dydt;  
  
    real S = y[1];  
    real I = y[2];  
    real R = y[3];  
  
    dydt[1] = -beta * I * S / N;  
    dydt[2] =  beta * I * S / N - sigma * I;  
    dydt[3] =  sigma * I;  
  
    return dydt;  
  }  
}  
data {  
}  
parameters {  
}  
transformed parameters{  
}  
model {  
}  
generated quantities {  
}
```

Fitting the model in python



MRC Centre for
Global Infectious
Disease Analysis

Imperial College
London

```
1 import numpy as np
2
3 # Set the population size
4 N = 763
5
6 # Define initial conditions
7 i0 = 1
8 s0 = N - i0
9 r0 = 0
10 y0 = [s0, i0, r0]
11
12 # Make the data struture
13 stan_data = {
14     "n_days": len(df.in_bed),
15     "y0": y0,
16     "t0": 0,
17     "t": np.arange(1, len(df.in_bed)+1),
18     "N": N,
19     "cases": df.in_bed
20 }
```

Then we compile our stan model.

```
1 sir_model = CmdStanModel(stan_file = 'sir_model.stan')
2
```

Finally we run MCMC. For this problem, it suffices to use Stan's defaults. Note that, as is standard practice, we run 4 Markov chains.

```
1 fit_sir_model = sir_model.sample(data = stan_data,
2     iter_sampling = 2000,
3     chains = 4,
4     seed = 0)
```