Lesson 4.
Case study:
Measles in large and small towns

Aaron A. King, Edward Ionides, and Kidus Asfaw

June 30, 2020

Outline

Objectives

- To display a published case study using plug-and-play methods with non-trivial model complexities.
- To show how extra-demographic stochasticity can be modeled.
- To demonstrate the use of covariates in **pomp**.
- To demonstrate the use of profile likelihood in scientific inference.
- To discuss the interpretation of parameter estimates.
- To emphasize the potential need for extra sources of stochasticity in modeling.

Challenges in inference from disease dynamics I

- Understanding, forecasting, managing epidemiological systems increasingly depends on models.
- Dynamic models can be used to test causal hypotheses.
- Real epidemiological systems:
  - are nonlinear
  - are stochastic
  - are nonstationary
  - evolve in continuous time
  - have hidden variables
  - can be measured only with (large) error
- Dynamics of infectious disease outbreaks illustrate this well.
- Measles is the paradigm for a nonlinear ecological system that can be well described by low-dimensional nonlinear dynamics.

Challenges in inference from disease dynamics II

- A tradition of careful modeling studies have proposed and found evidence for a number of specific mechanisms, including
  - a high value of R_0 (c. 15–20)
  - under-reporting
  - seasonality in transmission rates associated with school terms
  - response to changing birth rates
  - a birth-cohort effect
  - metapopulation dynamics
  - fadeouts and reintroductions that scale with city size
  - spatial traveling waves
- Much of this evidence has been amassed from fitting models to data, using a variety of methods.

Challenges in inference from disease dynamics III

- See Rohani and King (2010) for a review of some of the high points.

Measles in England and Wales I

- We revisit a classic measles data set, weekly case reports in 954 urban centers in England and Wales during the pre-vaccine era (1950–1963).
- We examine questions regarding:
- measles extinction and recolonization
- transmission rates
- seasonality
- resupply of susceptibles
- We use a model that
  1. expresses our current understanding of measles dynamics
  2. includes a long list of mechanisms that have been proposed and demonstrated in the literature
  3. cannot be fit by existing likelihood-based methods
- We examine data from large and small towns using the same model, something no existing methods have been able to do.

Measles in England and Wales II

- We ask: does our perspective on this disease change when we expect the models to explain the data in detail?
- What bigger lessons can we learn regarding inference for dynamical systems?

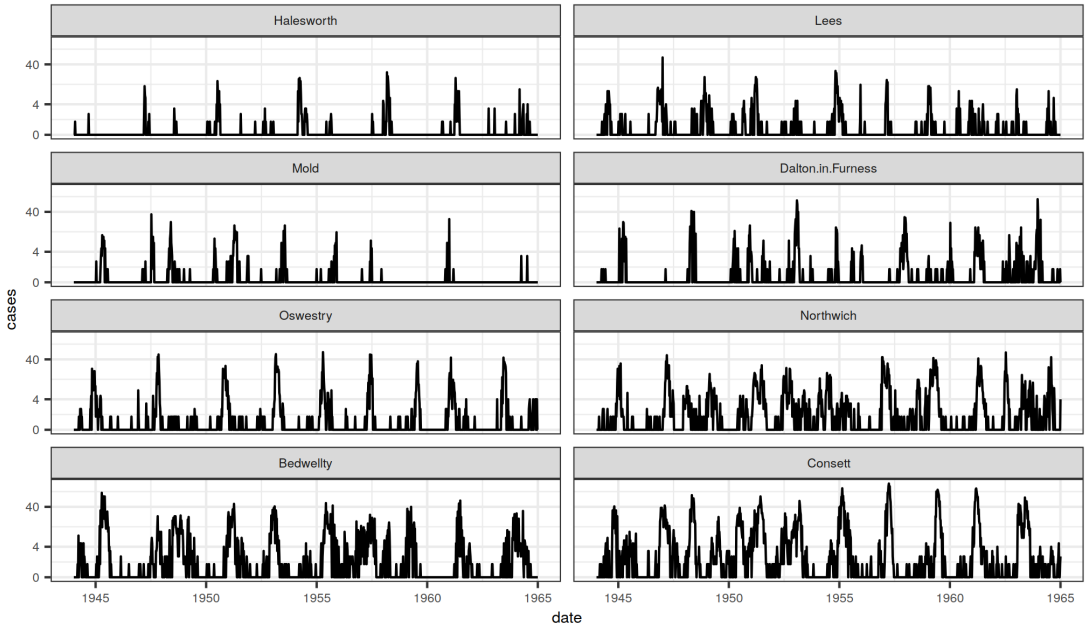
He, Ionides, & King, *J. R. Soc. Interface* (2010) I

Data sets:

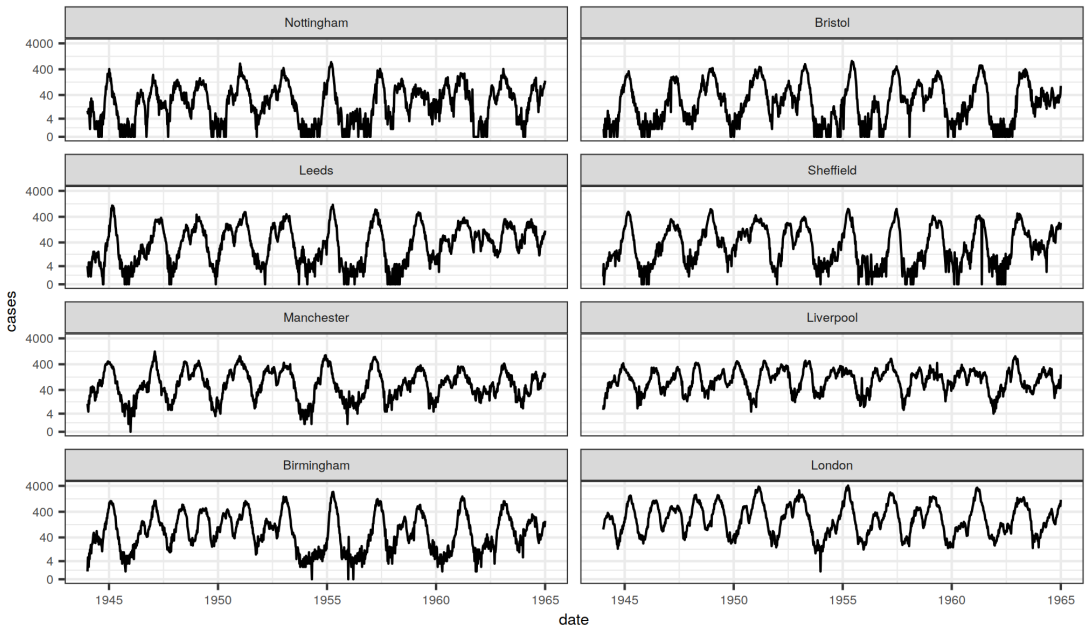
- Twenty towns, including
  - 10 largest
  - 10 smaller, chosen at random
- Population sizes: 2k–3.4M
- Weekly case reports, 1950–1963
- Annual birth records and population sizes, 1944–1963

Map of cities in the analysis

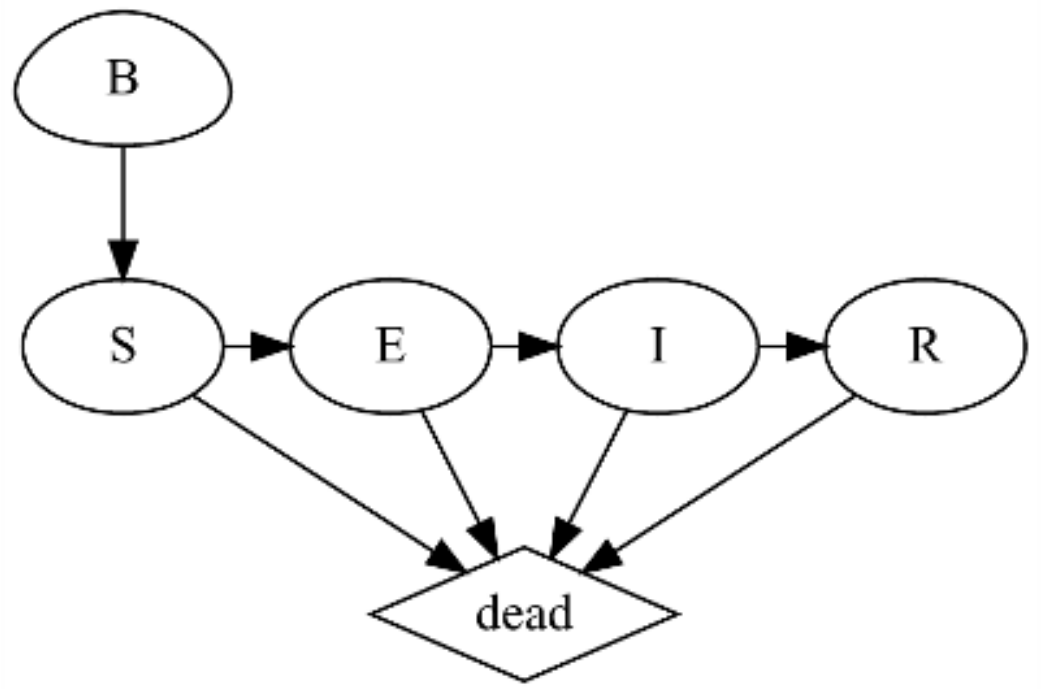
City case counts I: smallest 8 cities



City case counts II: largest 8 cities



Continuous-time Markov process model



Continuous-time Markov process model I

- Covariates:
  - $B(t)$ = birth rate, from data
  - $N(t)$ = population size, from data
- Entry into susceptible class:

$$\mu_{BS}(t) = (1-c)\,B(t-\tau) + c\,\delta(t-\lfloor t \rfloor) \int_{t-1}^{t} B(t-\tau-s)\,ds$$

  - c = cohort effect
  - τ = school-entry delay
  - $\lfloor t \rfloor$ = most recent 1 September before t
- Force of infection:

$$\mu_{SE}(t) = \frac{\beta(t)}{N(t)}\,(I+\iota)\,\zeta(t)$$

  - ι = imported infections

Continuous-time Markov process model II

- $\zeta(t)$ = Gamma white noise with intensity σ_{SE} [He *et al.* (2010); Bhadra *et al.* (2011)]
- school-term transmission:

$$\beta(t) = \begin{cases} \beta_0 \left(1 + a(1-p)/p\right) & \text{during term} \\ \beta_0 \left(1 - a\right) & \text{during vacation} \end{cases}$$

  - a = amplitude of seasonality
  - $p = 0.7589$ is the fraction of the year children are in school.
  - The factor $(1-p)/p$ ensures that the average transmission rate is β_0.

- Overdispersed binomial measurement model:
$\text{cases}_t \,|\, \Delta N_{IR} = z_t \sim \text{Normal}\left(\rho\, z_t, \rho\,(1-\rho)\, z_t + (\psi\, \rho\, z_t)^2\right)$

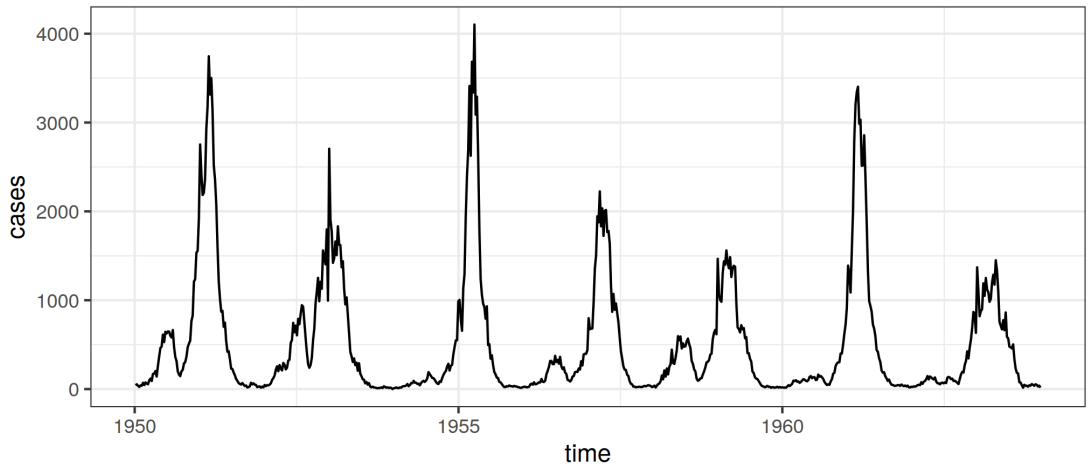
Implementation in **pomp**

- We'll load the packages we'll need, and set the random seed, to allow reproducibility.
- Note that we'll be making heavy use of the **tidyverse** methods.
- Also, we'll be using **ggplot2** for plotting: see this brief tutorial.
- Finally, we'll use the convenient **magrittr** syntax, which is explained here.

Data and covariates

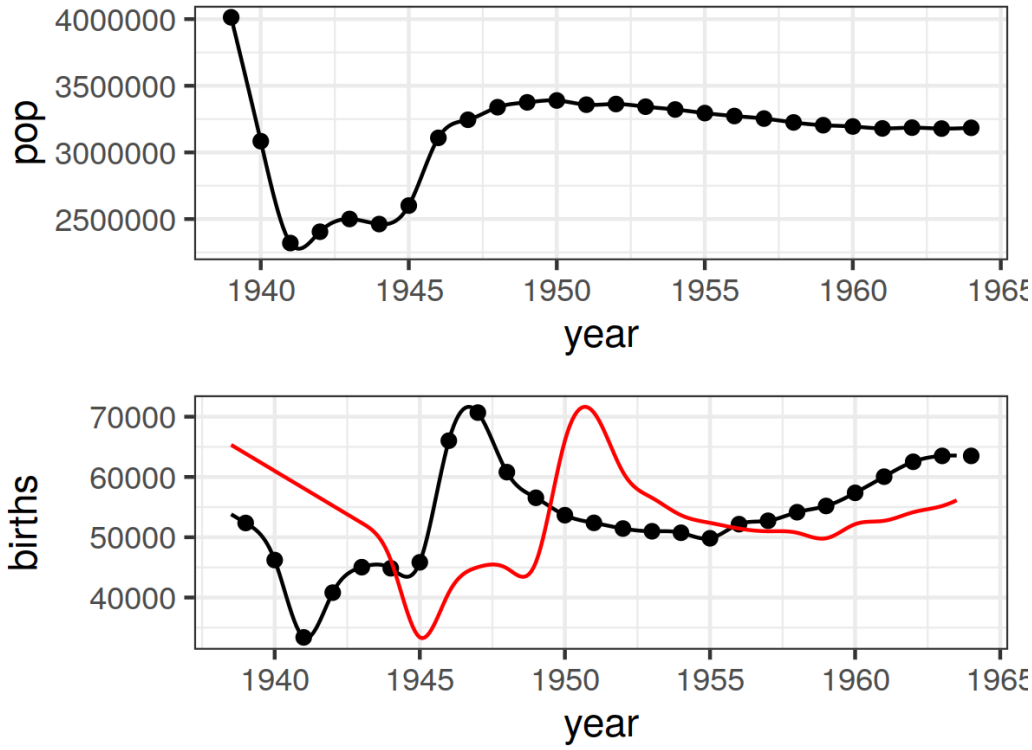
- We load the data and covariates. The data are measles reports from 20 cities in England and Wales.
- We also have information on the population sizes and birth-rates in these cities; we'll treat these variables as covariates.
- We will illustrate the pre-processing of the measles and demography data using London as an example.

Data and covariate plots I



Now, we smooth the covariates. Note that we delay the entry of newborns into the susceptible pool.

Data and covariate plots II



The partially observed Markov process model

We require a simulator for our model. Notable complexities include:

1. Incorporation of the known birthrate.
2. The birth-cohort effect: a specified fraction (`cohort`) of the cohort enter the susceptible pool all at once.
3. Seasonality in the transmission rate: during school terms, the transmission rate is higher than it is during holidays.
4. Extra-demographic stochasticity in the form of a Gamma white-noise term acting multiplicatively on the force of infection.
5. Demographic stochasticity implmented using Euler-multinomial distributions.

Implementation of the process model I

```
double beta, br, seas, foi, dw, births;
double rate[6], trans[6];

// cohort effect
if (fabs(t-floor(t)-251.0/365.0) < 0.5*dt)
  br = cohort*birthrate/dt + (1-cohort)*birthrate;
else
      br = (1.0-cohort)*birthrate;

// term-time seasonality
t = (t-floor(t))*365.25;
if ((t>=7&&t<=100) ||
    (t>=115&&t<=199) ||
    (t>=252&&t<=300) || (t>=308&&t<=356))
    seas = 1.0+amplitude*0.2411/0.7589;
else
    seas = 1.0-amplitude;
```

Implementation of the process model II

```
// transmission rate
beta = R0*(gamma+mu)*seas;
// expected force of infection
foi = beta*pow(I+iota,alpha)/pop;
// white noise (extrademographic stochasticity)
dw = rgammawn(sigmaSE,dt);
rate[0] = foi*dw/dt;  // stochastic force of infection
rate[1] = mu;                       // natural S death
rate[2] = sigma;                 // rate of ending of latent stage
rate[3] = mu;                       // natural E death
rate[4] = gamma;                 // recovery
rate[5] = mu;                       // natural I death

// Poisson births
births = rpois(br*dt);
```

Implementation of the process model III

```
// transitions between classes
reulermultinom(2,S,&rate[0],dt,&trans[0]);
reulermultinom(2,E,&rate[2],dt,&trans[2]);
reulermultinom(2,I,&rate[4],dt,&trans[4]);

S += births   - trans[0] - trans[1];
E += trans[0] - trans[2] - trans[3];
I += trans[2] - trans[4] - trans[5];
R = pop - S - E - I;
W += (dw - dt)/sigmaSE;  // standardized i.i.d. white noise
C += trans[4];           // true incidence
```

- In the above, C represents the true incidence, i.e., the number of new infections occurring over an interval.

Implementation of the process model IV

- Since recognized measles infections are quarantined, we argue that most infection occurs before case recognition so that true incidence is a measure of the number of individuals progressing from the I to the R compartment in a given interval.

State initializations

We complete the process model definition by specifying the distribution of initial unobserved states. The following codes assume that the fraction of the population in each of the four compartments is known.

```
rinit <- Csnippet("
double m = pop/(S_0+E_0+I_0+R_0);
S = nearbyint(m*S_0);
E = nearbyint(m*E_0);
I = nearbyint(m*I_0);
R = nearbyint(m*R_0);
W = 0;
C = 0;
")
```

The measurement model I

- We'll model both under-reporting and measurement error.
- We want $\mathbb{E}[\text{cases}|C] = \rho\, C$, where C is the true incidence and $0 < \rho < 1$ is the reporting efficiency.
- We'll also assume that $\mathrm{Var}[\text{cases}|C] = \rho\,(1-\rho)\,C + (\psi\,\rho\,C)^2$, where ψ quantifies overdispersion.
- Note that when $\psi = 0$, the variance-mean relation is that of the binomial distribution. To be specific, we'll choose cases—C $\sim f(\cdot|\rho,\psi,C)$, where

$$
\begin{aligned}
&f(c|\rho,\psi,C)\\
&\qquad = \Phi(c+\tfrac{1}{2}, \rho\,C, \rho\,(1-\rho)\,C + (\psi\,\rho\,C)^2) -\\
&\qquad\quad \Phi(c-\tfrac{1}{2}, \rho\,C, \rho\,(1-\rho)\,C + (\psi\,\rho\,C)^2)
\end{aligned}
$$

where $\Phi(x,\mu,\sigma^2)$ is the c.d.f. of the normal distribution with mean μ and variance σ^2.

The measurement model II

The following computes $\mathbb{P}[\text{cases}|C]$.

```
dmeas <- Csnippet("
double m = rho*C;
double v = m*(1.0-rho+psi*psi*m);
double tol = 1.0e-18;
if (cases > 0.0) {
  lik = pnorm(cases+0.5,m,sqrt(v)+tol,1,0)-pnorm(cases-0.5,m,sqrt(v)+tol,
} else {
  lik = pnorm(cases+0.5,m,sqrt(v)+tol,1,0)+tol;
}
")
```

Case simulations

The following codes simulate $\text{cases}|C$.

```
rmeas <- Csnippet("
double m = rho*C;
double v = m*(1.0-rho+psi*psi*m);
double tol = 1.0e-18;
cases = rnorm(m,sqrt(v)+tol);
if (cases > 0.0) {
  cases = nearbyint(cases);
} else {
  cases = 0.0;
}
")
```

Constructing the 'pomp' object

```
dat %>%
  pomp(t0=with(dat,2*time[1]-time[2]),
       time="time",
       rprocess=euler(Csnippet(rproc),delta.t=1/365.25),
       rinit=rinit,
       dmeasure=dmeas,
       rmeasure=rmeas,
       covar=covariate_table(covar,times="time"),
       accumvars=c("C","W"),
       statenames=c("S","E","I","R","C","W"),
       paramnames=c("R0","mu","sigma","gamma","alpha","iota",
                    "rho","sigmaSE","psi","cohort","amplitude",
                    "S_0","E_0","I_0","R_0")
       ) -> m1
```

Estimates from He *et al.* (2010)

He *et al.* (2010) estimated the parameters of this model. The full set is included in the R code accompanying this document, where they are read into a data frame called `mles`.
We verify that we get the same likelihood as He *et al.* (2010).
Change eval to T

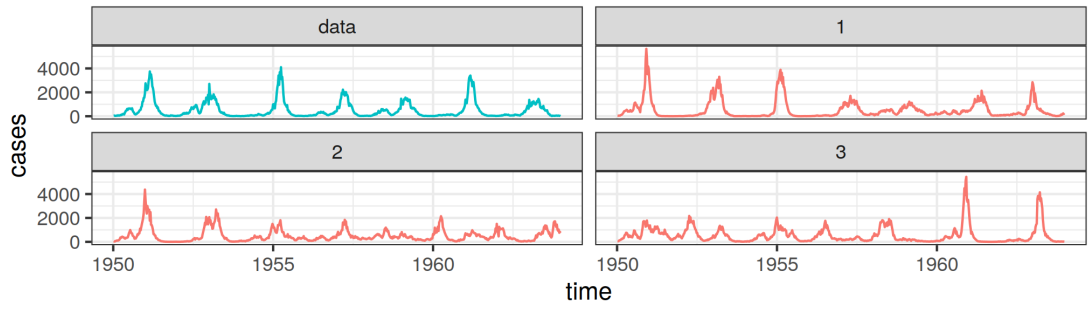
```
library(foreach)
library(doParallel)
library(doRNG)

registerDoParallel()
registerDoRNG(998468235L)

foreach(i=1:4) %dopar% {
  library(pomp)
  pfilter(m1,Np=10000,params=theta)
} -> pfs
logmeanexp(sapply(pfs,logLik),se=TRUE)
```

Simulations at the MLE

```
m1 %>%
  simulate(params=theta,nsim=3,format="d",include.data=TRUE) %>%
  ggplot(aes(x=time,y=cases,group=.id,color=(.id=="data")))+
  guides(color=FALSE)+
  geom_line()+facet_wrap(~.id,ncol=2)
```



Parameter transformations

- The parameters are constrained to be positive, and some of them are constrained to lie between 0 and 1.
- We can turn the likelihood maximization problem into an unconstrained maximization problem by transforming the parameters.
- Specifically, to enforce positivity, we log transform, to constrain parameters to $(0, 1)$, we logit transform, and to confine parameters to the unit simplex, we use the log-barycentric transformation.

```
pt <- parameter_trans(
  log=c("sigma","gamma","sigmaSE","psi","R0"),
  logit=c("cohort","amplitude"),
  barycentric=c("S_0","E_0","I_0","R_0")
)
m1 %>%
  pomp(partrans=pt,
       statenames=c("S","E","I","R","C","W"),
       paramnames=c("R0","mu","sigma","gamma","alpha","iota",
                    "rho","sigmaSE","psi","cohort","amplitude",
```

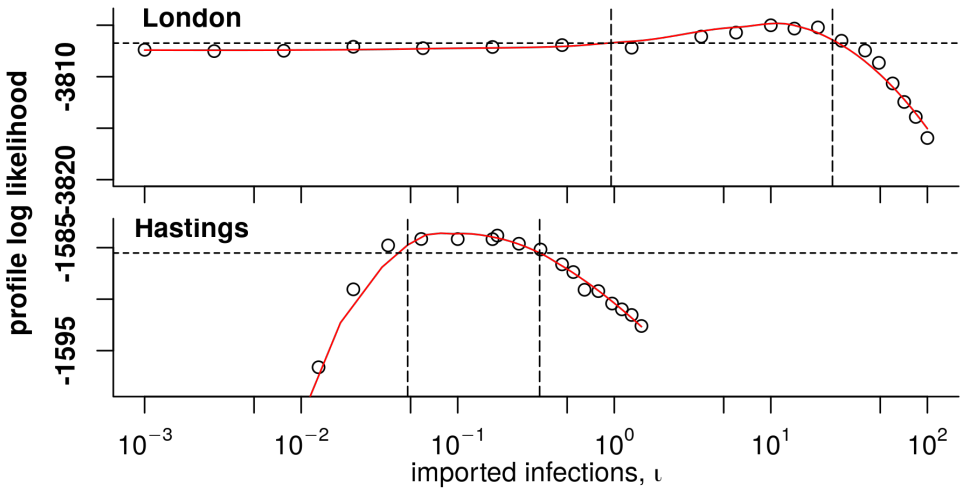
Results from He *et al.* (2010)

The linked document shows how a likelihood profile can be constructed using IF2 The fitting procedure used is as follows:

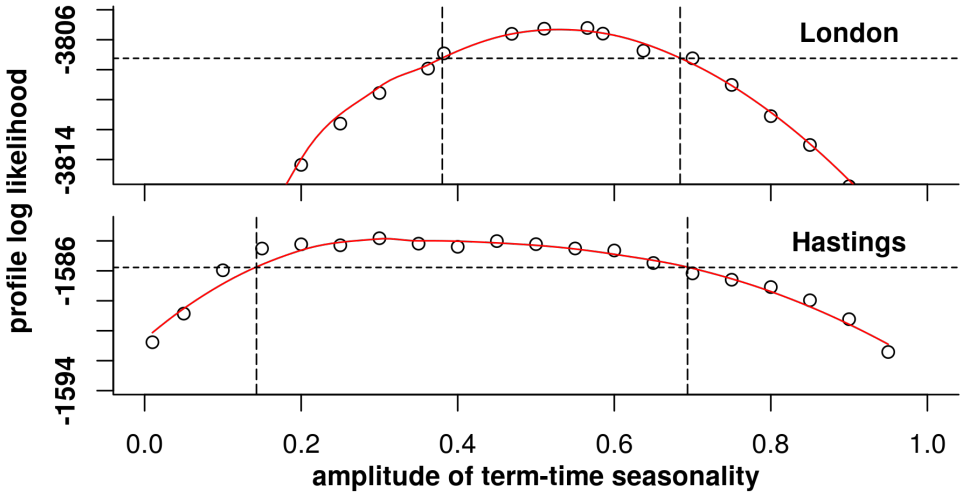
- A large number of searches were started at points across the parameter space.
- Iterated filtering was used to maximize the likelihood.
- We obtained point estimates of all parameters for 20 cities.
- We constructed profile likelihoods to quantify uncertainty in London and Hastings.

Imported infections

$$\text{force of infection} = \mu_{SE} = \frac{\beta(t)}{N(t)}\,(I+\iota)\,\zeta(t)$$

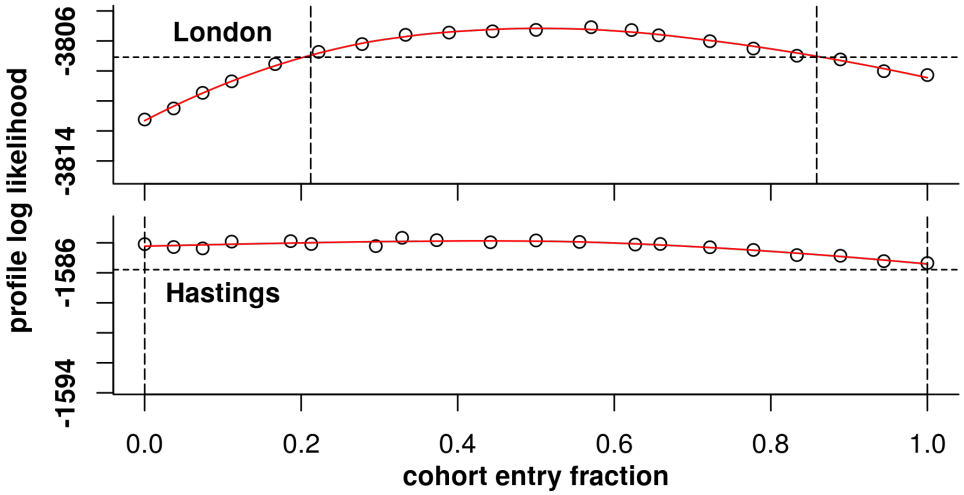


Seasonality



Notable findings

Cohort effect



Notable findings

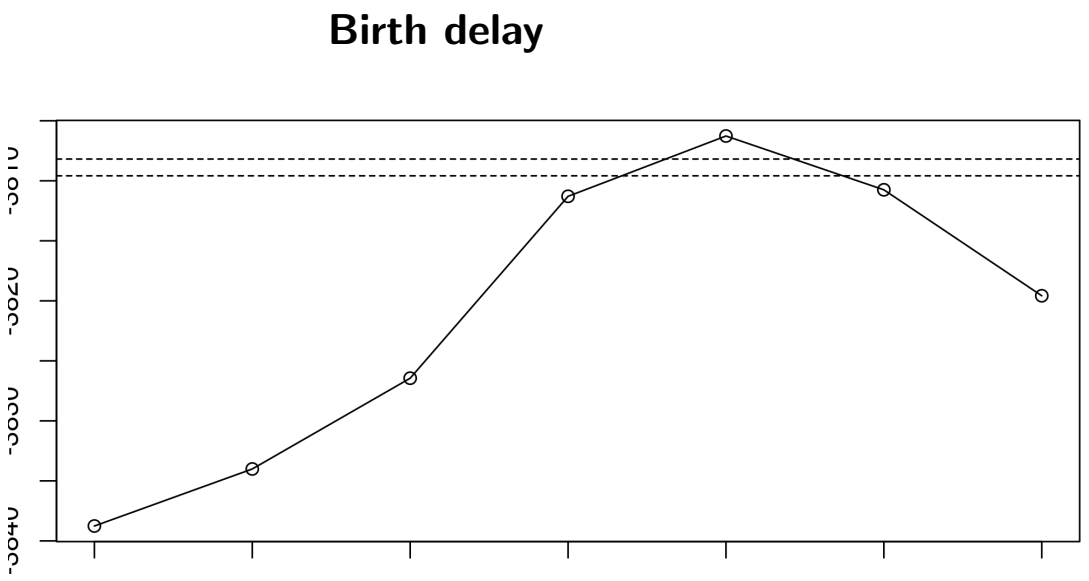
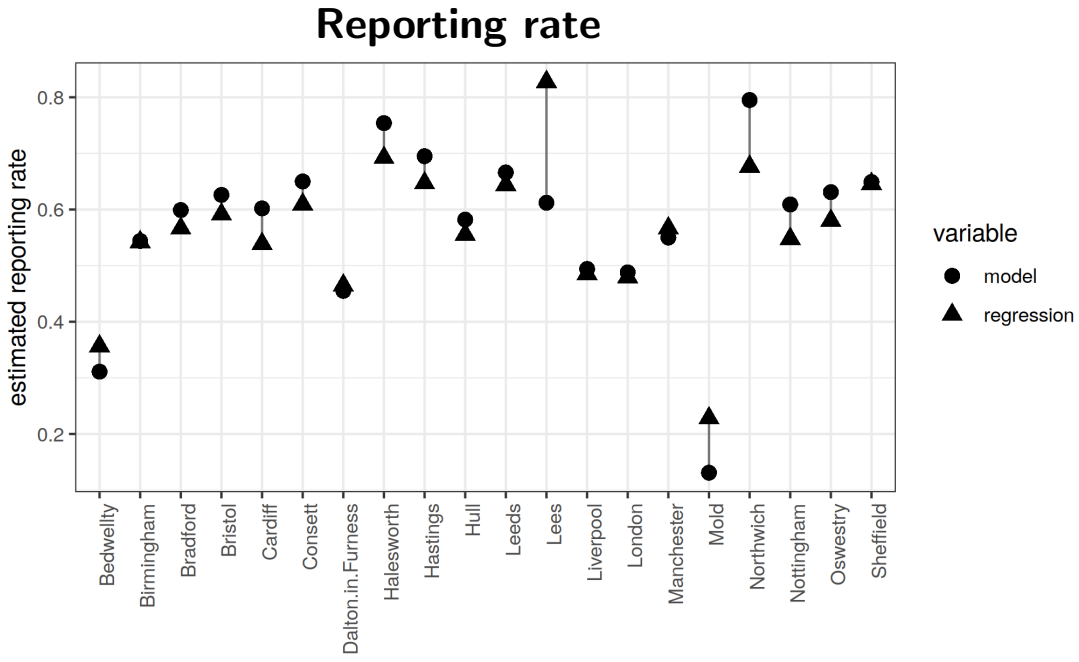


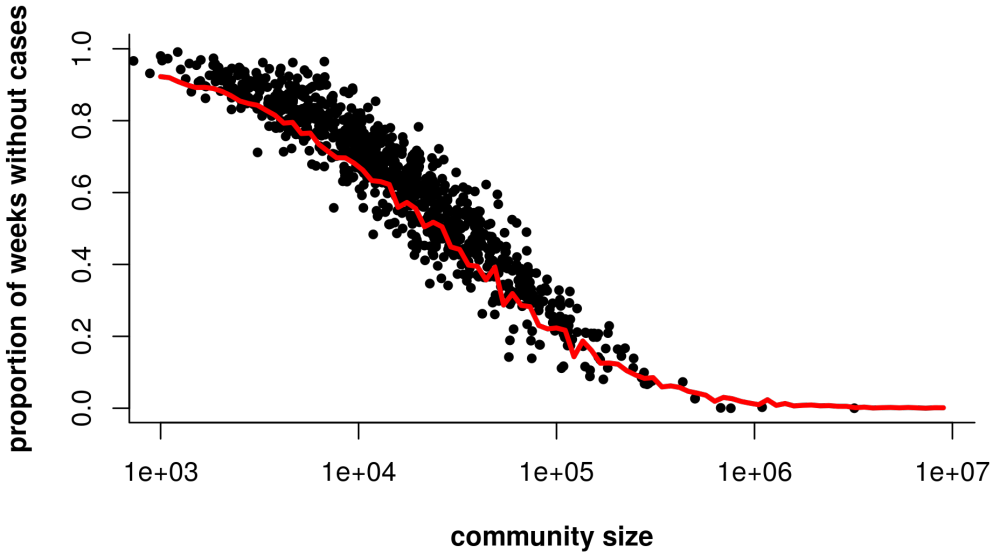
Figure: Profile likelihood for birth-cohort delay, showing 95% and 99% critical values of the log likelihood.

Notable findings



Notable findings

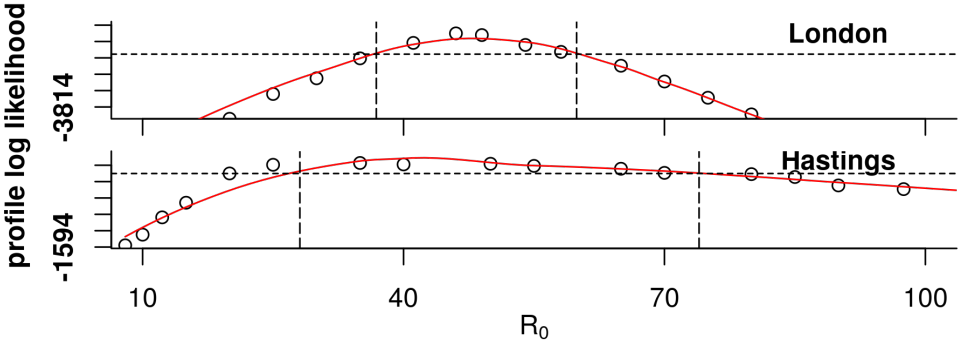
Predicted vs observed critical community size



Problematic results

R_0 estimates inconsistent with literature

- Recall that R_0 : a measure of how communicable an infection is.
- Existing estimates of R_0 (c. 15–20) come from two sources: serology surveys, and models fit to data using feature-based methods.



Problematic results

Parameter estimates

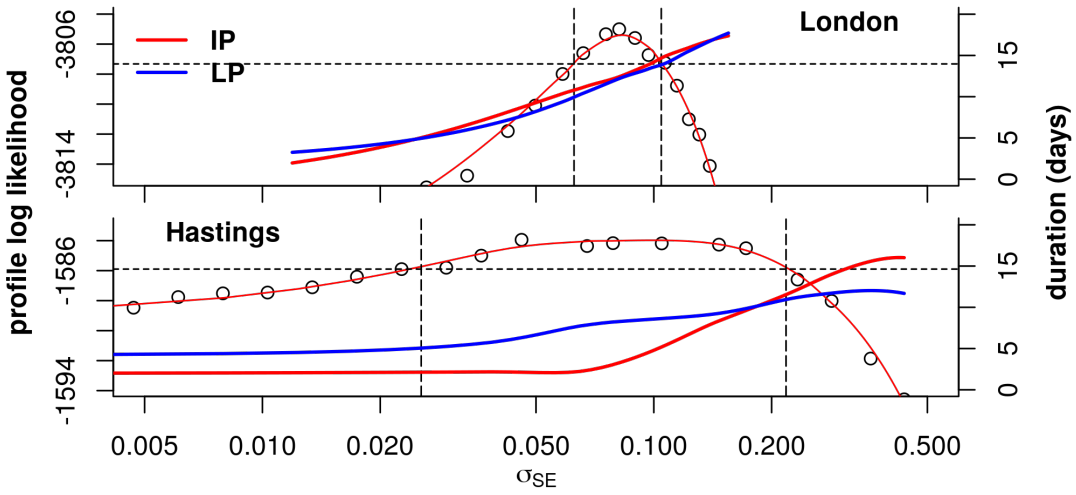
	pop	R0	amplitude	LP	IP
Halesworth	2170	33.100	0.381	7.870	2.290
Lees	4250	29.700	0.153	8.510	2.050
Mold	6410	21.400	0.271	5.930	1.780
Dalton.in.Furness	10600	28.300	0.203	5.480	1.980
Oswestry	11000	52.900	0.339	10.300	2.710
Northwich	18300	30.100	0.423	8.510	3.020
Bedwellty	28900	24.700	0.160	6.820	3.030
Consett	39100	35.900	0.200	9.080	2.660
Hastings	65700	34.200	0.299	7.000	5.440
Cardiff	245000	34.400	0.223	9.870	3.090
Bradford	294000	32.100	0.236	8.510	3.360
Hull	302000	38.900	0.221	9.180	5.460
Nottingham	307000	22.600	0.157	5.720	3.700
Bristol	443000	26.800	0.203	6.190	4.940
Leeds	510000	47.800	0.267	9.480	10.900
Sheffield	515000	33.100	0.313	7.230	6.380
Manchester	704000	32.900	0.290	11.100	6.940
Liverpool	802000	48.100	0.305	7.900	9.800
Birmingham	1120000	43.400	0.428	8.510	11.600
London	3390000	56.800	0.554	13.100	12.500
r	NA	0.455	0.301	0.322	0.946

	alpha	iota	rho	psi	sigmaSE
Halesworth	0.948	0.00912	0.754	0.641	0.0748
Lees	0.968	0.03110	0.612	0.681	0.0802
Mold	1.040	0.01450	0.131	2.870	0.0544

Problematic results

Extrademographic stochasticity

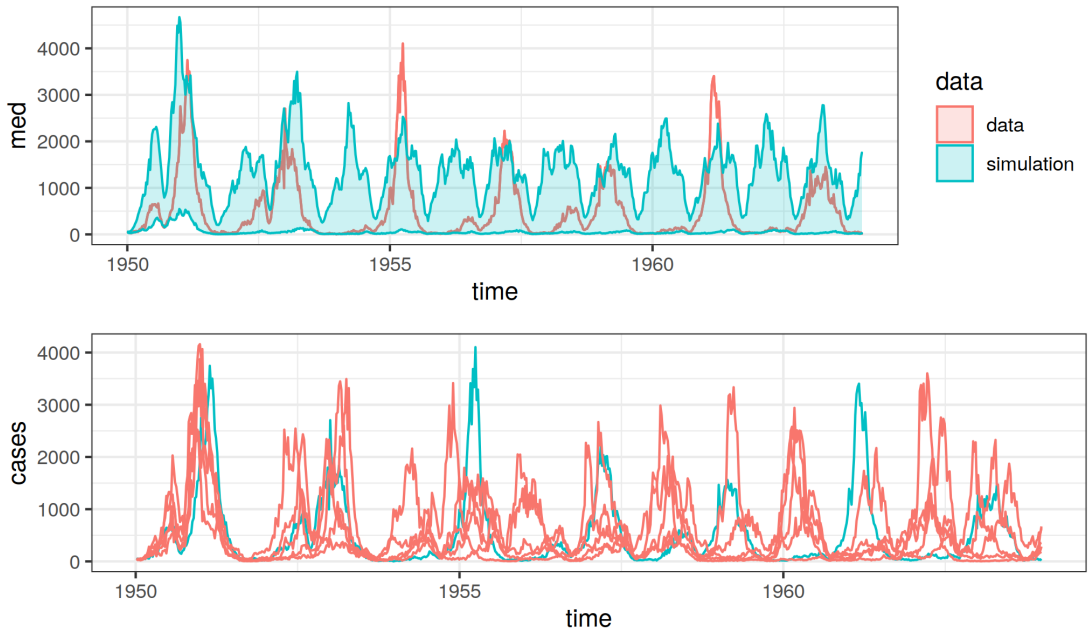
$$\mu_{SE} = \frac{\beta(t)}{N(t)} (I + \iota) \zeta(t)$$



Questions

- What does it mean that parameter estimates from the fitting disagree with estimates from other data?
- How can one interpret the correlation between infectious period and city size in the parameter estimates?
- How do we interpret the need for extrademographic stochasticity in this model?

Simulations at the MLE



Exercise 4.1. Reformulate the model

- Modify the He *et al.* (2010) model to remove the cohort effect. Run simulations and compute likelihoods to convince yourself that the resulting codes agree with the original ones for 'cohort $= 0$'.
- Now modify the transmission seasonality to use a sinusoidal form. How many parameters must you use? Fixing the other parameters at their MLE values, compute and visualize a profile likelihood over these parameters.

Exercise 4.2. Extrademographic stochasticity

Set the extrademographic stochasticity parameter $\sigma_{SE} = 0$, set $\alpha = 1$, and fix ρ and ι at their MLE values, then maximize the likelihood over the remaining parameters.

- How do your results compare with those at the MLE? Compare likelihoods but also use simulations to diagnose differences between the models.

References I

Bhadra A, Ionides EL, Laneri K, Pascual M, Bouma M, Dhiman R (2011). "Malaria in Northwest India: Data analysis via partially observed stochastic differential equation models driven by Lévy noise." *Journal of the American Statistical Association*, **106**, 440–451. doi: 10.1198/jasa.2011.ap10323.

He D, Ionides EL, King AA (2010). "Plug-and-play inference for disease dynamics: measles in large and small populations as a case study." *Journal of the Royal Society, Interface*, **7**, 271–283. doi: 10.1098/rsif.2009.0151.

Rohani P, King AA (2010). "Never mind the length, feel the quality: the impact of long-term epidemiological data sets on theory, application and policy." *Trends in Ecology & Evolution*, **25**(10), 611–618. doi: 10.1016/j.tree.2010.07.010.

License, acknowledgments, and links

- This lesson is prepared for the Simulation-based Inference for Epidemiological Dynamics module at the 2020 Summer Institute in Statistics and Modeling in Infectious Diseases, SISMID 2020.
- The materials build on previous versions of this course and related courses.
- Licensed under the Creative Commons Attribution-NonCommercial license. Please share and remix non-commercially, mentioning its origin. 
- Produced with R version 4.0.2 and **pomp** version 3.0.2.1.

Back to course homepage
R code for this lesson