# Bayesian Inference of Reproduction Number from Epidemic and Genetic Data

Alicia Gill
joint work with Xavier Didelot, Richard Everitt, Jere Koskela and Tim Vaughan

Algorithms seminar
10th May 2024

# What's the problem?

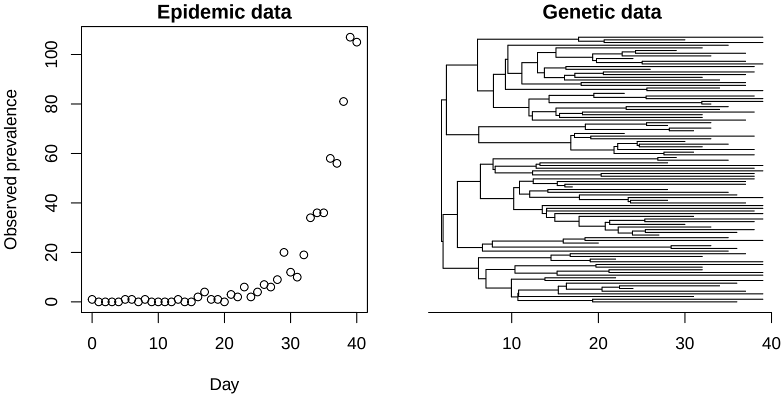
The reproduction number $R(t)$ represents the average number of secondary infections caused by each infected individual.

Problems:

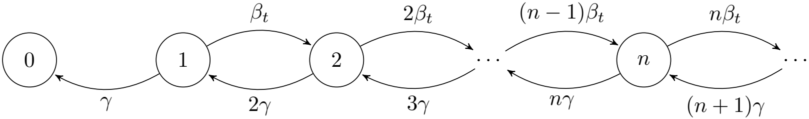
- Epidemic data may be noisy/incomplete
- Trees (used to represent the genetic data) are not directly informative about epidemiological processes like $R(t)$.

Aim:
The aim is to use epidemic data and genetic data in a joint model to estimate $R(t)$.

# Data


Epidemic data
Observed prevalence
0
20
40
60
80
100
0
10
20
30
40
Day
Genetic data
10
20
30
40

# Modelling the epidemic (1/2)

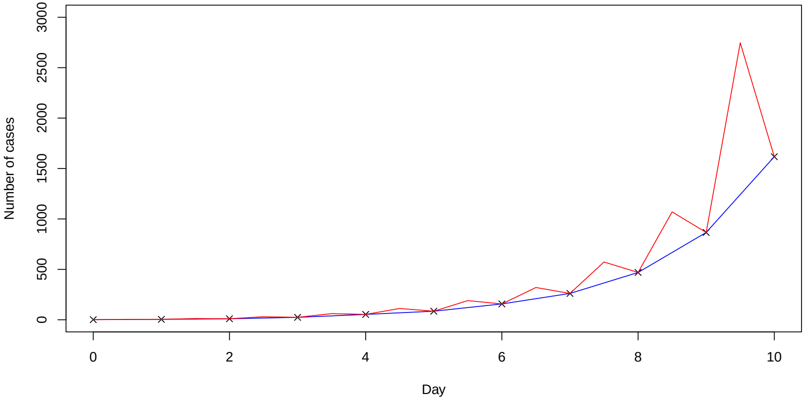


In a birth-death model of disease outbreak, the reproduction number is

$$R(t) = \frac{\beta_t}{\gamma}.$$

# Modelling the epidemic (2/2)

Let $X_n$ denote the number of cases on day $n$.



$$B_n \mid X_{n-1} = x_{n-1}, \beta_n \sim \text{Poisson}(\beta_n x_{n-1})$$
$$D_n \mid X_{n-1} = x_{n-1}, \gamma \sim \text{Poisson}(\gamma x_{n-1})$$

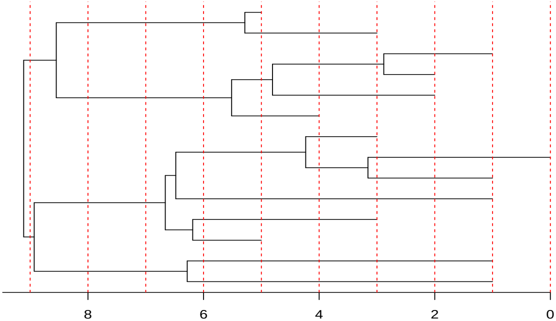
# Modelling the observed epidemic

Let $Y_n$ denote the observed prevalence on day $n$.

$$Y_n \mid X_n = x_n \sim \text{Binomial}(x_n, \rho)$$

where $\rho$ is the reporting probability.

# Modelling the phylogeny



| Days from present, $n$ | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| # lineages, $A_n$ | 2 | 4 | 4 | 8 | 10 | 10 | 10 | 8 | 6 | 1 |
| # coalescences, $C_n$ | 1 | 2 | 0 | 4 | 2 | 2 | 1 | 1 | 0 | 0 |

We want to model the number of coalescences on day $n$ as a binomial distribution with $\binom{A_n}{2}$ trials and success probability $p_n$.

# Backward-in-time

In a Kingman's coalescent model[1], two lineages coalesce exponentially with rate $1/N_e(t)$ where $N_e(t)$ denotes the effective population size at time $t$. Overall coalescence rate is

$$\lambda(t) = \binom{A_t}{2} \frac{1}{N_e(t)}.$$

[1] Kingman (1982), "The coalescent", *Stochastic Processes and their Applications* 13(3):235-248

# Forward-in-time

Let $f(t)$ denote the incidence (new cases). The transmission rate is[2]

$$\lambda(t) = f(t)\frac{\binom{A_t}{2}}{\binom{X_t}{2}} \approx \binom{A_t}{2}\frac{2f(t)}{X_t^2}$$

In a birth-death model, $f(t) = \beta_t X_t$, so the transmission rate is

$$\lambda(t) \approx \binom{A_t}{2}\frac{2\beta_t}{X_t}.$$

[2] Volz *et al.* (2009), "Phylodynamics of infectious disease epidemics", *Genetics* 13(4):1421-1430

# Backward = Forward

Under some assumptions, coalescence events correspond to transmission events, i.e. backward-in-time mergers correspond to forward-in-time infections. Setting the coalescence rate equal to the transmission rate gives

$$\frac{1}{N_e(t)} = \frac{2\beta_t}{X_t}.$$

The probability of two lineages merging on day $n$ is

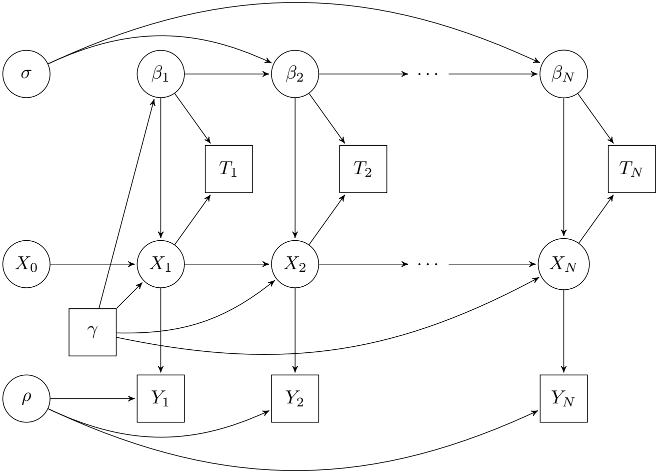
$$p_n = 1 - \exp\left(-\frac{2\beta_n}{X_n}\right).$$

# Prior for $\beta_t$

- $\beta_1 \sim \mathrm{Exp}(1/2\gamma)$
- For $n = 2, \ldots, N$, $\beta_n \mid \beta_{n-1} \sim \mathrm{Normal}(\beta_{n-1}, \sigma^2)$, truncated at 0

# State space model

Suppose the epidemic has been ongoing for $N$ days.



# Bayesian inference

Let $\theta = (\sigma, p, X_0)$ denote model parameters and $\beta = \beta_{1:N}$, $X = X_{1:N}$, $T = T_{1:N}$, $Y = Y_{1:N}$.

$$\begin{aligned} &p(\beta, \theta \mid \gamma, Y, T) \\ &\propto p(\theta) p(\beta, \gamma, Y, T \mid \theta) \\ &= p(\theta) \int p(\beta, \gamma, X, Y, T \mid \theta)\, dX \\ &= \underbrace{p(\theta)}_{\substack{\text{model}\\\text{parameters}}} \underbrace{p(\beta \mid \theta)}_{\substack{\text{birth}\\\text{rates}}} \int \underbrace{p(X \mid \beta, \gamma, \theta)}_{\substack{\text{latent}\\\text{epidemic}}} \underbrace{p(Y \mid X, \theta)}_{\substack{\text{observed}\\\text{epidemic}}} \underbrace{p(T \mid \beta, X)}_{\text{phylogeny}}\, dX \end{aligned}$$

Problem: Intractable likelihood :(

# Intractable likelihoods

Possible solutions:

1. Data augmentation
    - Dimension increases with the length of the time series
    - Time series variables are highly correlated
2. Pseudo-marginal MCMC
    - Inefficient
3. Particle marginal Metropolis–Hastings

# Particle marginal Metropolis–Hastings algorithm (PMMH)[3]

Basically Metropolis–Hastings, with a few key differences:

- Use an unbiased estimator of the likelihood instead of the true likelihood
- Idea: Get this estimator by sampling $K$ "particles" (i.e. $K$ trajectories for $\beta$ and $X$) and averaging over them
- Use sequential Monte Carlo (SMC) to generate the sampled trajectories

[3] Andrieu *et al.* (2010), "Particle Markov chain Monte Carlo methods", *J R Stat Soc Series B Stat Methodol*, 72(3):269-342

# SMC algorithm

Let $N$ denote the length of the epidemic and $K$ denote the number of particles. For $n = 1, \ldots, N$:

1. Sample: Draw $(\beta_n^k, X_n^k) \sim q_\theta(\cdot \mid \beta_{1:n-1}^k, X_{1:n-1}^k)$ for $k = 1, \ldots, K$.
2. Importance: Weight the pairs $(\beta_n^k, X_n^k)$ as

$$w_n^k = \frac{p_\theta(\beta_{1:n}^k, X_{1:n}^k, T_{1:n}, Y_{1:n})}{p_\theta(\beta_{1:n-1}^k, X_{1:n-1}^k, T_{1:n-1}, Y_{1:n-1}) q_\theta(\beta_n^k, X_n^k \mid \beta_{1:n-1}^k, X_{1:n-1}^k)}.$$

   Normalise $W_n^k = w_n^k / \sum_{j=1}^K w_n^j$.
3. Resample: Resample ancestors $A_n^{1:K}$ according to the normalised weights and keep pairs $(\beta_n^{A_n^{1:K}}, X_n^{A_n^{1:K}})$.

# Resampling causes problems...

Pros of resampling:

- Corrects proposals as you're building them
- Don't need as many particles to get a good estimate of the (log-)likelihood

Cons of resampling:

- Resampling introduces variance
- This causes path degeneracy

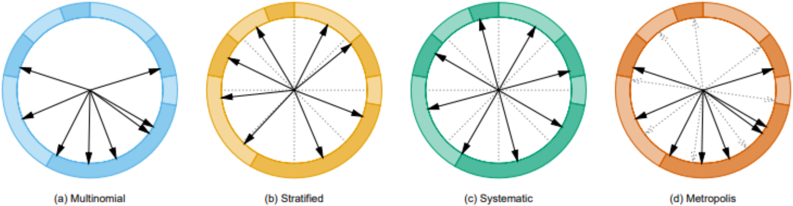
# Resample less: Adaptive resampling

Instead of resampling at every step, only resample if your weights degenerate.

$$\mathrm{ESS}(W^{1:K}) = \frac{1}{\sum_{k=1}^{K}(W_k)^2}.$$

If all particles have equal weight, then ESS$= K$. If one particle has all the weight, then ESS$= 1$. Conventionally, the resampling threshold is set to $K/2$.

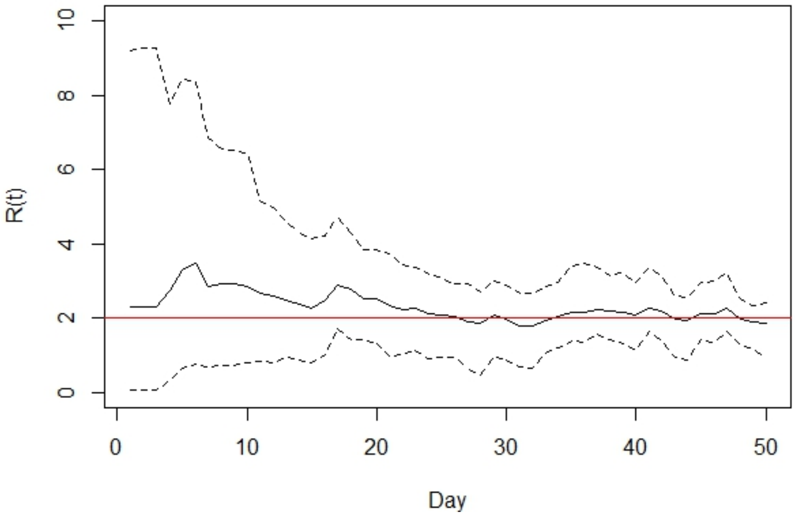
# Resample better: Systematic resampling



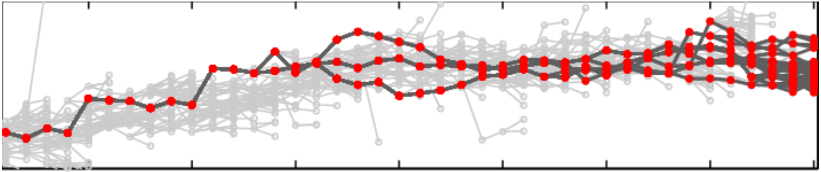
We use systematic resampling instead of multinomial resampling.

Murray (2012), "GPU acceleration of the particle filter: the Metropolis resampler", arXiv:1202.6163

# What does path degeneracy look like?



# Why does path degeneracy happen?



Svensson *et al.* (2015), "Nonlinear State Space Smoothing Using the Conditional Particle Filter", *IFAC-PapersOnLine*, 48(28):975-980

# Backward simulation[4]

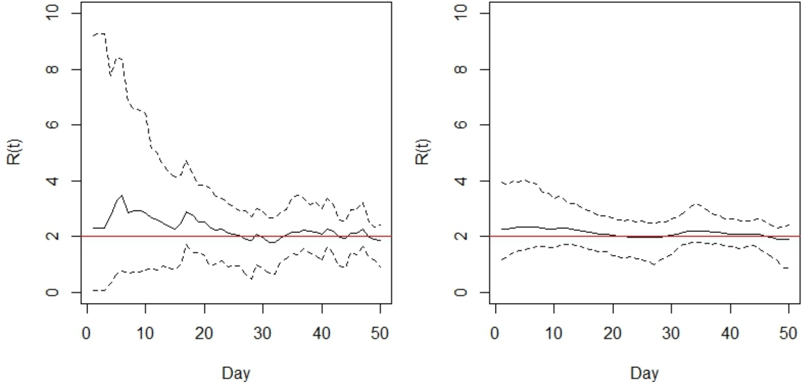
1. Run the SMC algorithm forward-in-time, storing all particles and weights in each generation, even those culled by resampling.
2. Set $j_N = k$ with probability $w_N^k / \sum_l w_N^l$.
3. For $n = N-1, \dots, 1$, compute the smoothing weights

$$w_{n|N}^k = \frac{w_n^k p(\beta_{n+1}^{j_{n+1}}, x_{n+1}^{j_{n+1}} \mid \beta_n^k, x_n^k)}{\sum_l w_n^l p(\beta_{n+1}^{j_{n+1}}, x_{n+1}^{j_{n+1}} \mid \beta_n^l, x_n^l)}.$$

4. Set $j_n = k$ with probability $w_{n|N}^k$.

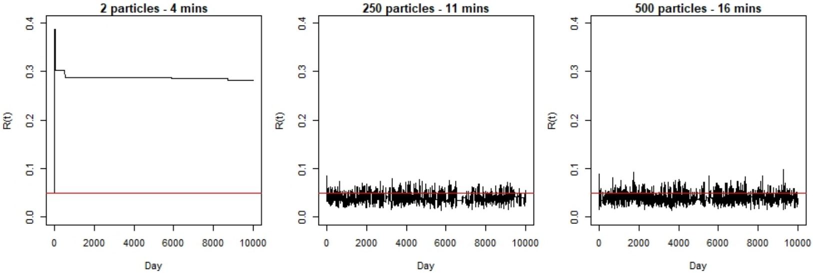
[4] Godshill *et al.* (2012), "Monte Carlo Smoothing for Nonlinear Time Series", *Journal of the American Statistical Association*, 99(465):156-168

# Phew!



# How many particles do you need?

Too few particles results in ‘sticky’ chains. Too many is inefficient.



# Choosing the optimal number of particles

We use the suggested guidance from Pitt *et al.* (2012)[5].

1. Run a short PMMH with a large number of particles to determine an approximate value for the posterior mean $\bar{\theta}$.
2. Run the SMC algorithm for several independent runs $R$ for a fixed value of particles $K_s$ and obtain an estimator of the likelihood $\hat{p}^i_{K_s}(y \mid \bar{\theta})$, $i = 1, \ldots, R$, for each.
3. Record the variance of the log-likelihood, $\hat{\sigma}^2(\bar{\theta}, K_s)$.
4. Choose the optimal number of particles

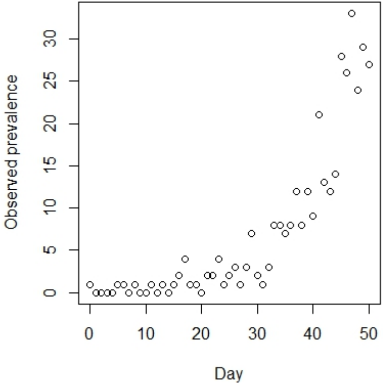
$$K_{opt} = K_s \times \frac{\hat{\sigma}^2(\bar{\theta}, K_s)}{0.92^2}.$$

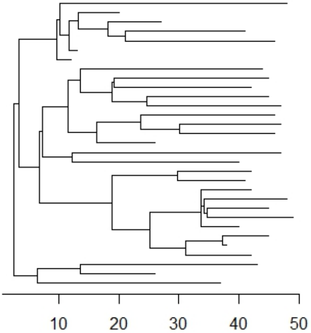
[5] Pitt *et al.* (2012), "On some properties of Markov chain Monte Carlo simulation methods based on the particle filter", *Journal of Econometrics*, 171(2):134-151

# Simulation: Constant $R(t)$

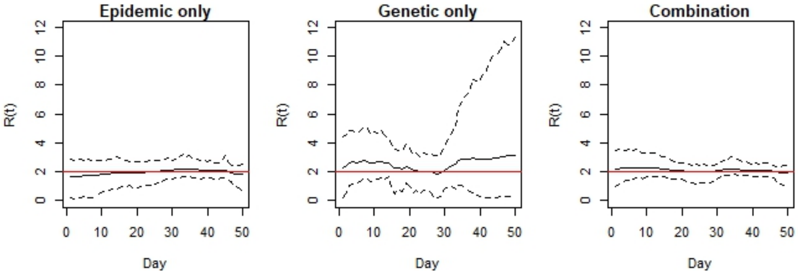
- Simulated a 50-day epidemic with $\beta_t = 0.2 \ \forall t$
- Tree generated from a random sample of 5% of past lineages - 31 tips
- Similarly, suppose a random sample of 5% of the epidemic observed
- Fixed and known death rate $\gamma = 0.1$
- 10,000 iterations
- 250 particles

# Data


Observed prevalence
0
5
10
15
20
25
30
0
10
20
30
40
50
Day


10
20
30
40
50

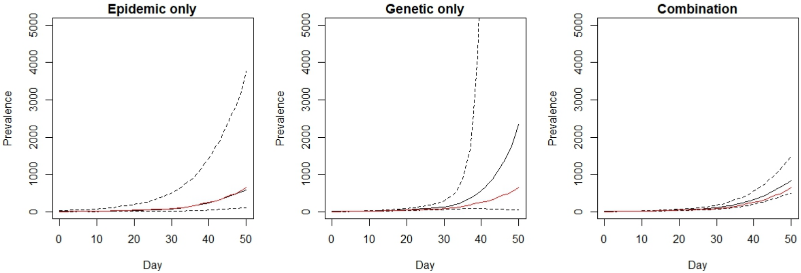
# $R(t)$ inference plot



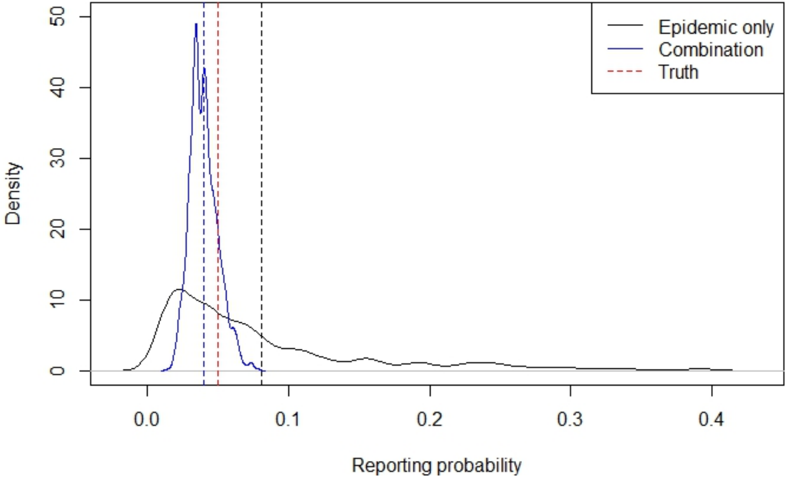
# $R(t)$ inference metrics

| Data | RMSE | Coverage | Mean CI width | Run time (mins) |
|---|---|---|---|---|
| Epi only | 0.16 | 100% | 1.8 | 11.2 |
| Gen only | 0.68 | 100% | 5.1 | 6.3 |
| Epi & gen | 0.15 | 100% | 1.4 | 11.1 |

# Prevalence inference plot



# Reporting probability



# Tree uncertainty

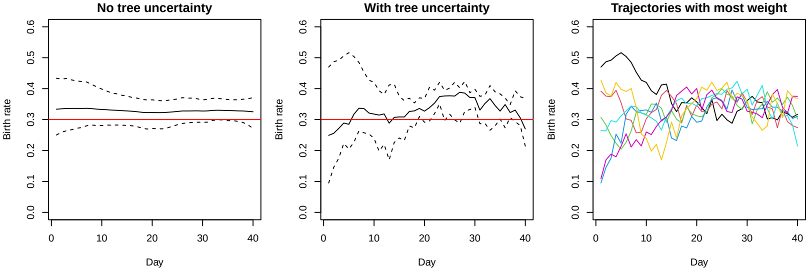
- Tim's idea: Generate trees and parameters given the sequence alignment data, then weight these according to the particle filter.
- Alicia's idea: Generate trees given the sequence alignment, run the PMMH algorithm on a sample of trees, then average the results.

# Tim's idea (1/3)

Generate trees and parameters given the sequence alignment data, then weight these according to the particle filter.

$$\begin{aligned} p(T, \theta \mid A, Y) &= \frac{p(A \mid T) p(Y \mid \theta, T) p(T \mid \theta) p(\theta)}{p(A, Y)} \\ &\propto \underbrace{\frac{p(A \mid T) p(T \mid \theta) p(\theta)}{p(A)}}_{\text{BEAST2}} \cdot \underbrace{\frac{\hat{p}(Y, T \mid \theta)}{p(T \mid \theta)}}_{\text{weights}} \end{aligned}$$

# Tim's idea (2/3)


No tree uncertainty
With tree uncertainty
Trajectories with most weight
Birth rate
0.0
0.1
0.2
0.3
0.4
0.5
0.6
Day
0
10
20
30
40

# Tim's idea (3/3)

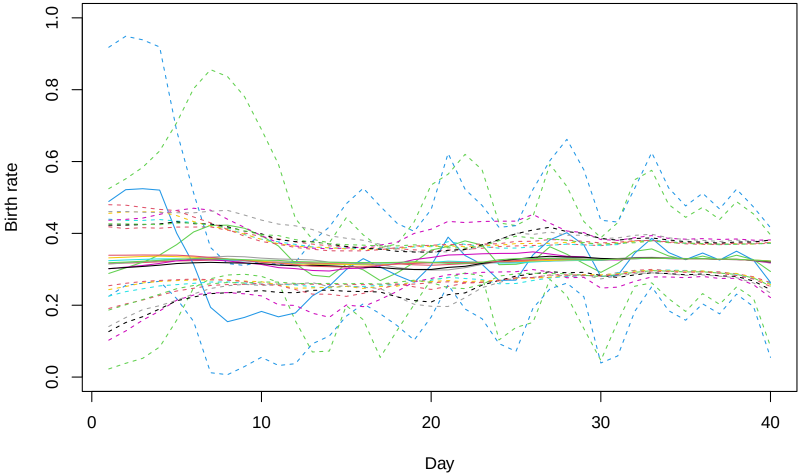
- Have to assume more parameters are known (i.e. $\rho$)
- Small number of trees carry the weight
- May need to run BEAST2 for more iterations and collect more samples

# Alicia's idea (1/4)

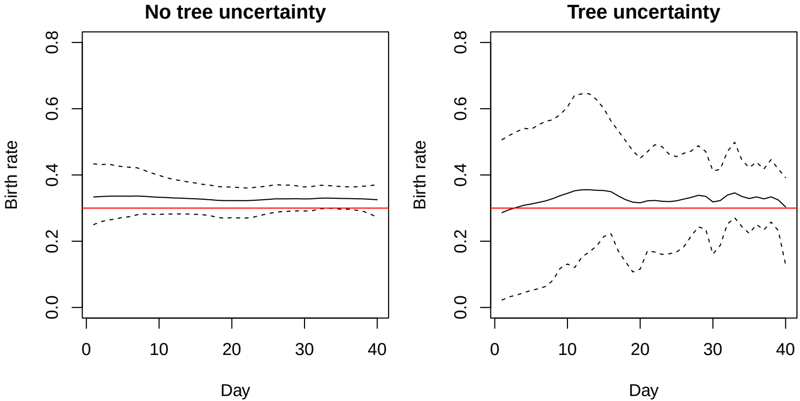
Generate trees given the sequence alignment, run the PMMH algorithm on a sample of trees, then average the results.

$$
\begin{aligned}
p(\beta, \theta \mid A, Y) &= \int \int p(\beta, \theta \mid Y, A, X, T) p(X, T \mid Y, A) \, dX \, dT \\
&= \int \int p(\beta, \theta \mid X, T) p(X \mid Y) p(T \mid A) \, dX \, dT \\
&\propto \int p(\beta, \theta \mid Y, T) p(T \mid A) \, dT \\
&\approx \frac{1}{M} \sum_{i=1}^{M} p(\beta, \theta \mid Y, T_i), \text{ where } T_i \sim p(\cdot \mid A).
\end{aligned}
$$

# Alicia's idea (2/4)



# Alicia's idea (3/4)



# Alicia's idea (4/4)

- Computationally intensive
- Assumes that the phylogeny $T$ is independent of the observed prevalence $Y$
- Model misspecification - generating birth-death trees and then evaluating them as coalescent trees

# Conclusion and limitations

Conclusions:

- Combining epidemic and genetic data seems to improve inference of $R(t)$ trajectory
- Also improves inference of other epidemiological parameters of interest, i.e. the reporting probability

Limitations:

- Simple epidemic model
- Computationally intensive
- Can incorporate phylogenetic uncertainty, but crudely

# I have a pre-print!

Alicia Gill, Jere Koskela, Xavier Didelot, Richard G. Everitt (2023), “Bayesian Inference of Reproduction Number from Epidemiological and Genetic Data Using Particle MCMC”, arXiv:2311.09838

https://arxiv.org/abs/2311.09838