Statistical aspects to epidemiological models

Geir Storvik, University of Oslo

May 29-30 2025

Overview

Introduction

Aim

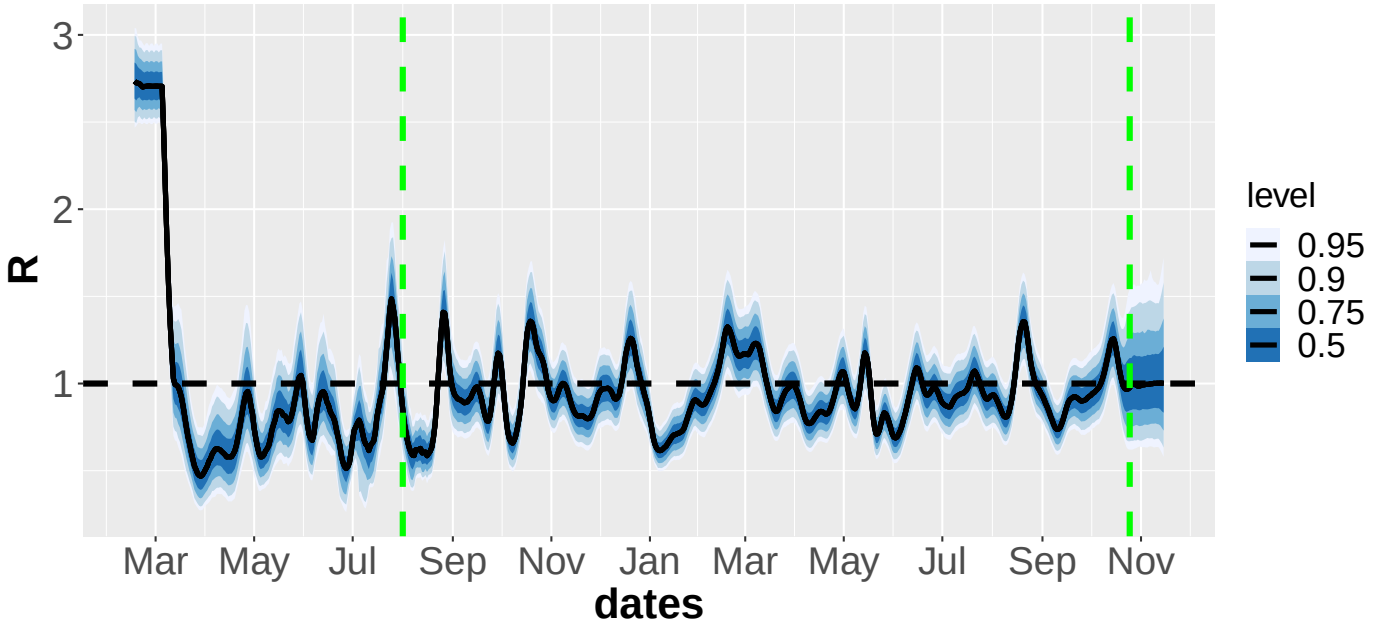
- Introduction to compartmental models for infectious diseases
  - Stochastic versions
- Data and state space models
- Inference
  - Bayesian framework
  - Computation by Monte Carlo
- Sequential Monte Carlo for infectious disease models
- Challenges

Why me?

- New nothing about infectious diseases before Covid pandemic
- The Norwegian Institute of Public Health had a model and data
  - but struggled with computation
- I was invited for implementing a Sequential Monte Carlo (SMC) method
  - included some extra modelling
- Ran the model every week during the pandemic
  - Results needed within 4 hours
- Published a paper in the end: Storvik et al. (2023): A sequential monte carlo approach to estimate a time-varying reproduction number in infectious disease models: the covid-19 case.

Practical task

- Estimate reproduction numbers based on available data sources



  - Daily numbers
  - Uncertainty quantification
  - Running each week
    - Four hours from receiving data to results should be available

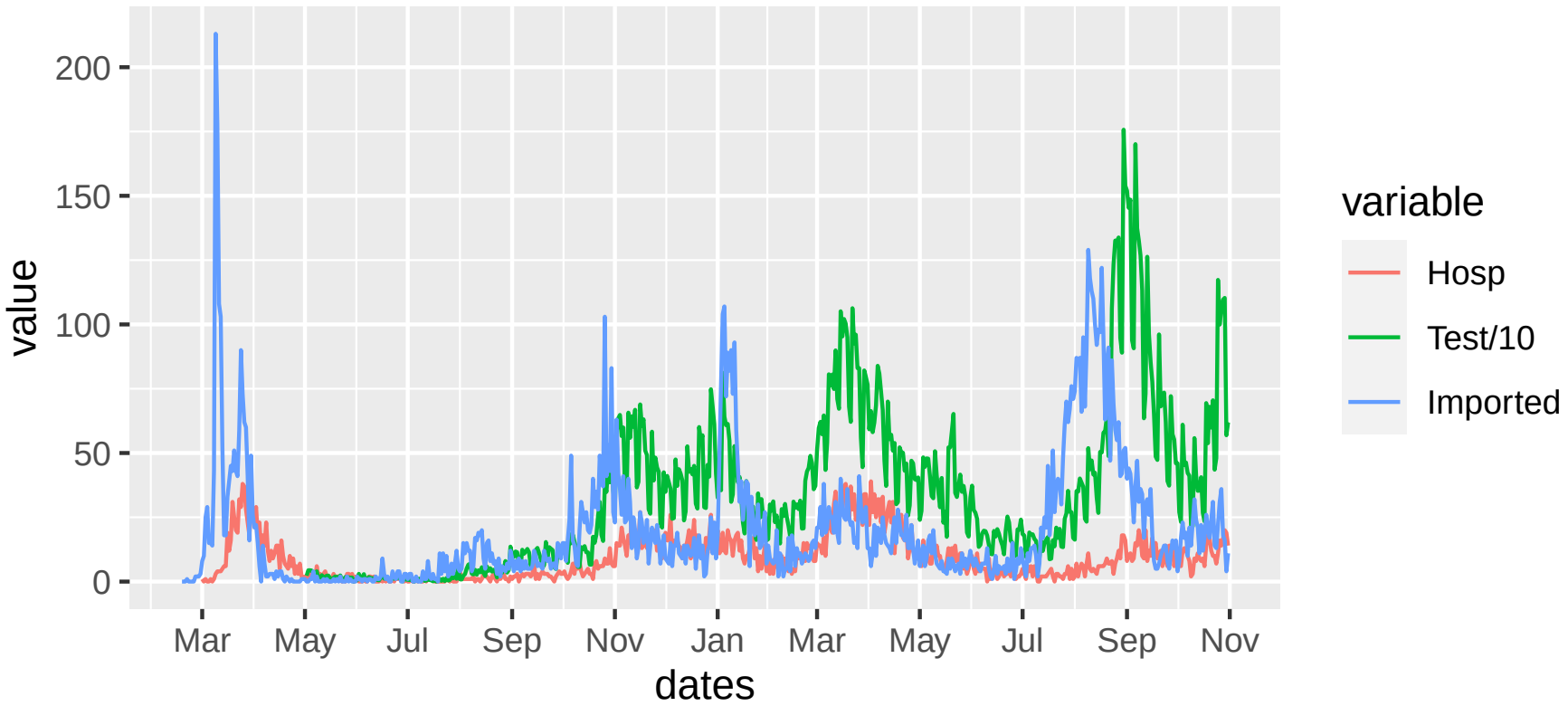
Multiple data sources

Output:

- Hospital prevalence
- Test positives

Input:

- Total number of tests
- Imported cases
- Mobility data

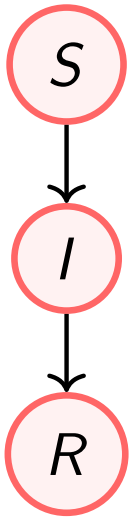


Models for infectious diseases

Compartmental models for spread of disease

- Individuals assigned to spesific compartments
- Simplest model: SIR

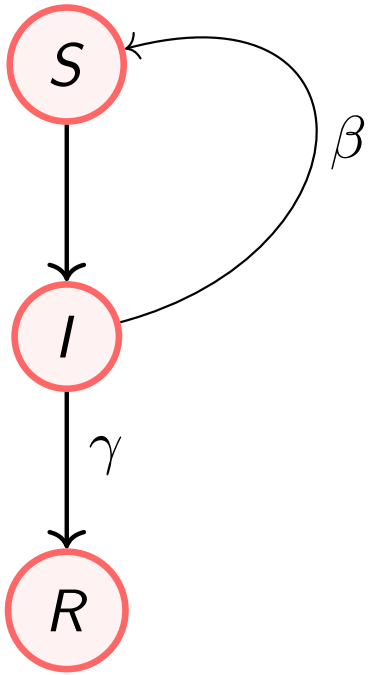
- S: Susceptible
- I: Infectious
- R: Recovered (Immune/dead)



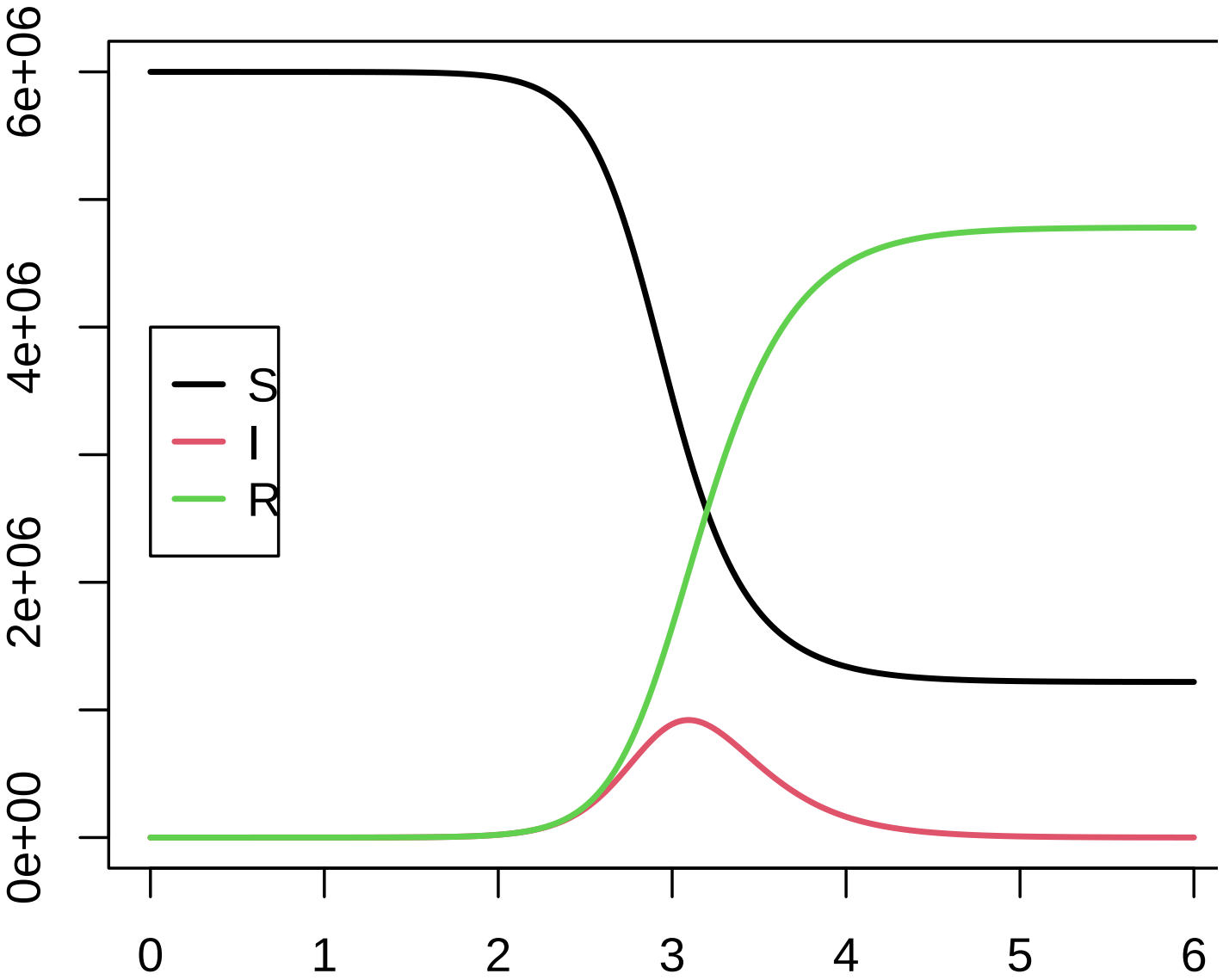
SIR - dynamics

- Model

$$\begin{aligned}
\frac{dS(t)}{dt} &= -\frac{\beta}{N} I(t)S(t) \\
\frac{dI(t)}{dt} &= \frac{\beta}{N} I(t)S(t) - \gamma I(t) \\
\frac{dR(t)}{dt} &= \gamma I(t) \\
R(t) &= N - S(t) - I(t)
\end{aligned}$$



Dynamics SIR



SIR-model - properties

- Purely deterministic model

$$S(\infty) = S(0) \exp\left(-\frac{\beta}{N\gamma} R(\infty)\right)$$

$$R(\infty) = N - S(\infty)$$

$$I(\infty) = 0$$

- S/I/R all continuous numbers.
- The basic reproduction number: $\mathcal{R}_0 = \frac{\beta}{\gamma}$

  The expected number of new infections from a single infection in a population where all subjects are susceptible

$$\mathcal{R}_0 > \frac{S(0)}{N} : \quad \text{Outbreak}$$

$$\mathcal{R}_0 < \frac{S(0)}{N} : \quad \text{No outbreak}$$

Discrete-time SIR model

- From continuous model:

$$\frac{S(t+h)-S(t)}{h} \approx -\frac{\beta(t)}{N} I(t)S(t)$$
$$\frac{I(t+h)-I(t)}{h} \approx -\frac{\beta(t)}{N} I(t)S(t) - \gamma(t)I(t)$$

- Discrete-time version (Allen, 1994):

$$\begin{aligned} S_{t+1} &= S_t \left[1 - \frac{\beta_t h}{N} I_t\right] \\ I_{t+1} &= I_t \left[1 + \frac{\beta_t h}{N} S_t - \gamma_t h\right] \\ R_{t+1} &= N - S_{t+1} - I_{t+1} \end{aligned}$$

Stochastic models

- Four possible approaches (Allen, 2008):
  - Continuous time Markov chain (CTMC) models
    - Time continuous, state discrete
  - Discrete-time Markov chain (DTMC) models
    - Time discrete, state discrete
  - (SDE models)
    - Time continuous, state continuous
  - Chain-binomial models

CTMC of SIR model

- Stochastic process $\{(S(t), I(t), R(t)\}$
- $R(t) = N - S(t) - I(t)$, only necessary to consider $\{S(t), I(t))\}$.
- $I(t), S(t) \in \{0, ..., N\}$
- Assume Markov property:

$$\Pr[S(t_{n+1}), I(t_{n+1}) | S(t_0), ..., S(t_n), I(t_0), ..., I(t_n)] = \Pr[S(t_{n+1}), I(t_{n+1}) | S(t_n), I(t_n)]$$

- Infinitesimal transition probabilities
  - Assume time-interval h so small that only three possible changes are possible:

$$(s, i) \to \begin{cases} (s-1, i+1) & \text{One more infected} \\ (s, i-1) & \text{One more recovered} \\ (s, i) & \text{No change} \end{cases}$$

  - Assumes time-homogeneous transitions.

Continuous time Markov process

Deterministic model

$$\frac{dS(t)}{dt} = -\frac{\beta}{N} I(t)S(t)$$
$$\frac{dI(t)}{dt} = \frac{\beta}{N} I(t)S(t) - \gamma I(t)$$

Stochastic model	
$(\Delta S, \Delta I)$	$\Pr(\Delta S, \Delta I \mid S(t), I(t) = (s, i))$
$(-1, 1)$	$\frac{\beta}{N} ish + o(h)$
$(0, -1)$	$\gamma ih + o(h)$
$(0, 0)$	$1 - \left[\frac{\beta}{N} is + \gamma i\right] h + o(h)$
other	$o(h)$

Here:

- $\lim_{t\to\infty} \frac{o(h)}{h} = 0.$
- $\Delta S = S(t+h) - S(t)$, $\Delta I = I(t+h) - I(t)$

Finite state Continuous time Markov chains

- Define infinitisimal matrix $\boldsymbol{A}$ by

$$\boldsymbol{A} = \lim_{h \to 0} \frac{\boldsymbol{P}(h) - \boldsymbol{I}}{h} \qquad k \neq i$$

Define

$\boldsymbol{\Lambda}$ =Diagonal matrix with eigenvalues of $\boldsymbol{A}$ on diagonal

$\boldsymbol{U}$ =Eigenvectors (columns) of $\boldsymbol{A}$

Then

$$\boldsymbol{P}(t) = \boldsymbol{U} e^{t\boldsymbol{\Lambda}} \boldsymbol{U}^{-1}$$

- Note: Dimension of $\boldsymbol{P}(t)$: $|\{(s, i) : 0 \leq s + i \leq N\}|$
  - Computational challenge

Discrete time approximation

- Assume for small h:

$$\begin{aligned}\Pr((s-1,i+1)|(s,i)) &= \frac{\beta}{N} ish \\ \Pr((s,i-1)|(s,i)) &= \gamma ih \\ \Pr((s,i)|(s,i)) &= 1-\left(\frac{\beta}{N}s+\gamma\right) ih\end{aligned}$$

- For $t = nh$: $\boldsymbol{P}(t) = \boldsymbol{P}(h)^n$
- Need h small enough:

$$\begin{aligned}\frac{\beta}{N} ish \leq 1 \quad &\Rightarrow h \leq \frac{N}{\beta is} \quad & h \leq \frac{4}{N\beta} \\ \gamma ih \leq 1 \quad &\Rightarrow h \leq \frac{1}{\gamma i} \quad & h \leq \frac{1}{N\gamma} \\ 1-\left(\frac{\beta}{N}s+\gamma\right) ih \geq 0 \quad &\Rightarrow h \leq \frac{1}{(\beta s/N+\gamma)i} \quad & h \leq N\frac{\beta+\gamma}{2\beta}\end{aligned}$$

- Again: Dimension of $\boldsymbol{P}(t)$ is huge, also n large

Chain-Binomial model

- Assume all infections/recovered happen at the start of (small) interval
- Discrete time
- Chain-binomial model:

$$\begin{aligned} S_{t+1} &= S_t - I_t^{new} & I_{t+1}^{new} &\sim \text{Binom}(S_t, \beta_t I_t h) \\ I_{t+1} &= I_t + I_{t+1}^{new} - R_t^{new} & R_t^{new} &\sim \text{Binom}(I_t, \gamma_t h) \\ R_{t+1} &= R_t + R_{t+1}^{new} \end{aligned}$$

- Typically, $\gamma_t = \gamma$, need $h < 1/\gamma$
- β_t vary, difficult to get $\beta_t I_t h < 1$ in general.
- Alternative:

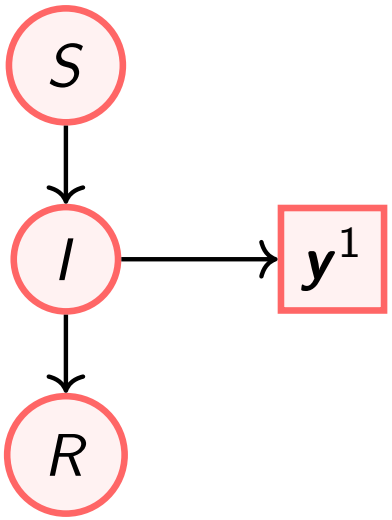
$$I_{t+1}^{new} \sim \text{Binom}(S_t, 1 - \exp(-\beta_t I_t h))$$

Data

Test positives

- Not all individuals get tested
- Delay from infected to tested/infectious
- Possible model:

$$y_t^1 \sim \text{Binomial}(I_{t-d_{test}}^{new}, q_1)$$



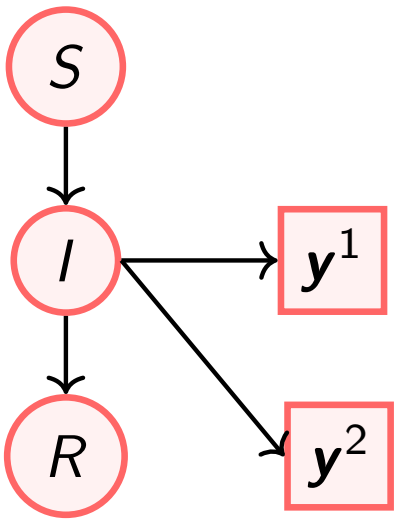
Challenges:

- q_1 not directly identifiable
  - Ideally: Controlled random sample
- q_1 may change over time
- Dependence
- Delays stochastic

Hospital data

- More reliable data
- More delays
- Possible model:

$$y_t^2 \sim \text{Binomial}(I_{t-d_{hosp}}^{new}, q_2)$$



Challenges:

- q_2 not directly identifiable
  - Ideally: Controlled random sample
- q_2 may change over time
  - Age distribution of infected
- Dependence
- Delays stochastic

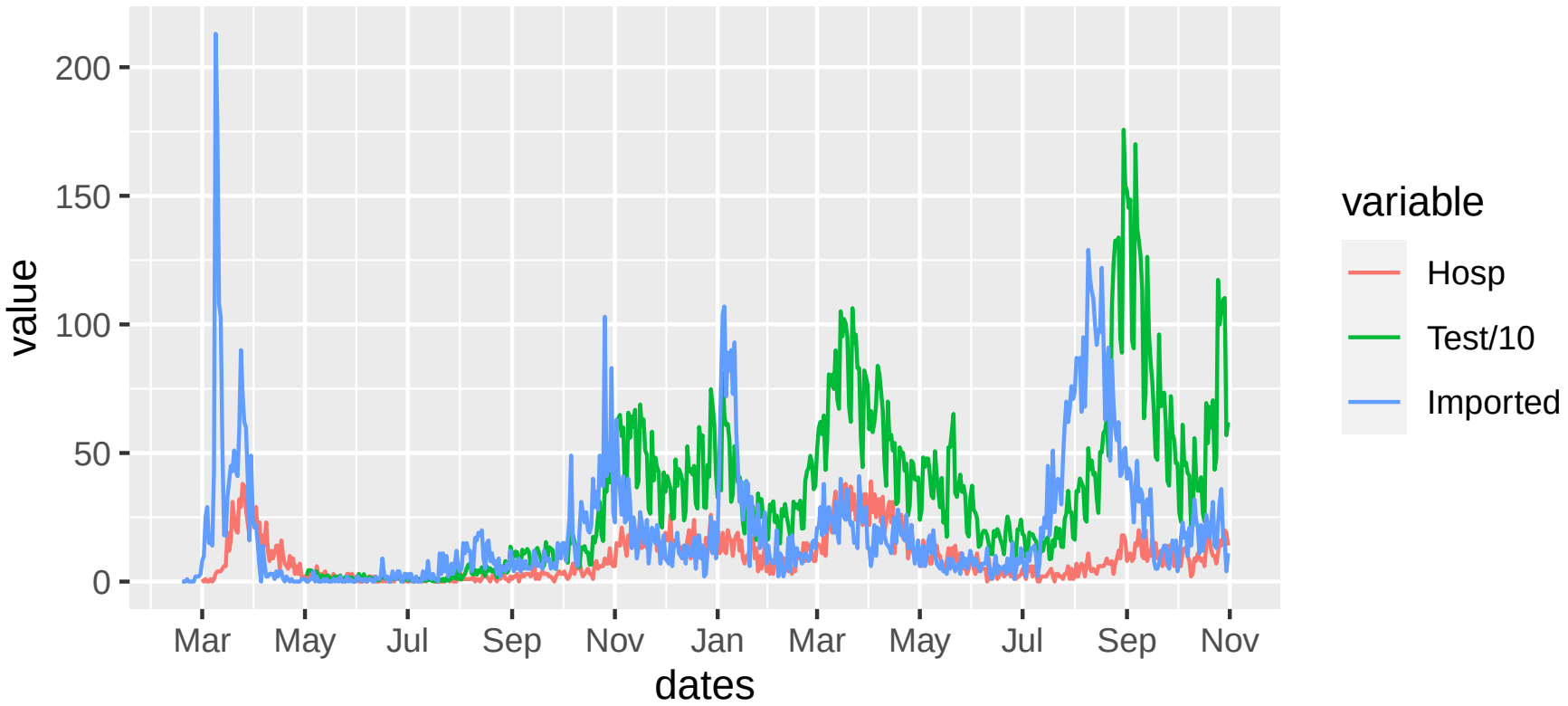
Data

Output:

- Hospital prevalence
- Test positives

Input:

- Total number of tests
- Imported cases
- Mobility data



Parameters

- Model

$$
\begin{aligned}
S_{t+1} &= S_t - I_t^{new} & I_{t+1}^{new} &\sim \text{Binom}(S_t, \beta_t I_t h) \\
I_{t+1} &= I_t + I_{t+1}^{new} - R_t^{new} & R_t^{new} &\sim \text{Binom}(I_t, \gamma_t h) \\
R_{t+1} &= R_t + R_{t+1}^{new}
\end{aligned}
$$

- Parameters: $\{\beta_t, \gamma_t\}$
  - Reasonable to assume $\gamma_t = \gamma$ (?)
  - Reasonable that β_t change with time
  - Define $r_t = \log(\beta_t/\gamma)$
  - Possible stochastic models:

$$
\begin{aligned}
r_{t+1} &= r_t + N(0, \sigma^2) && \text{RW-process} \\
r_{t+1} &= \mu + a(r_t - \mu) + N(0, \sigma^2) && \text{AR-process} \\
r_{t+1} &= \begin{cases} r_t & \text{with prob. } 1 - \phi_c \\ r_t + N(0, \sigma^2) & \text{with prob. } \phi_c \end{cases} && \text{Change-point}
\end{aligned}
$$

Inference

Inference

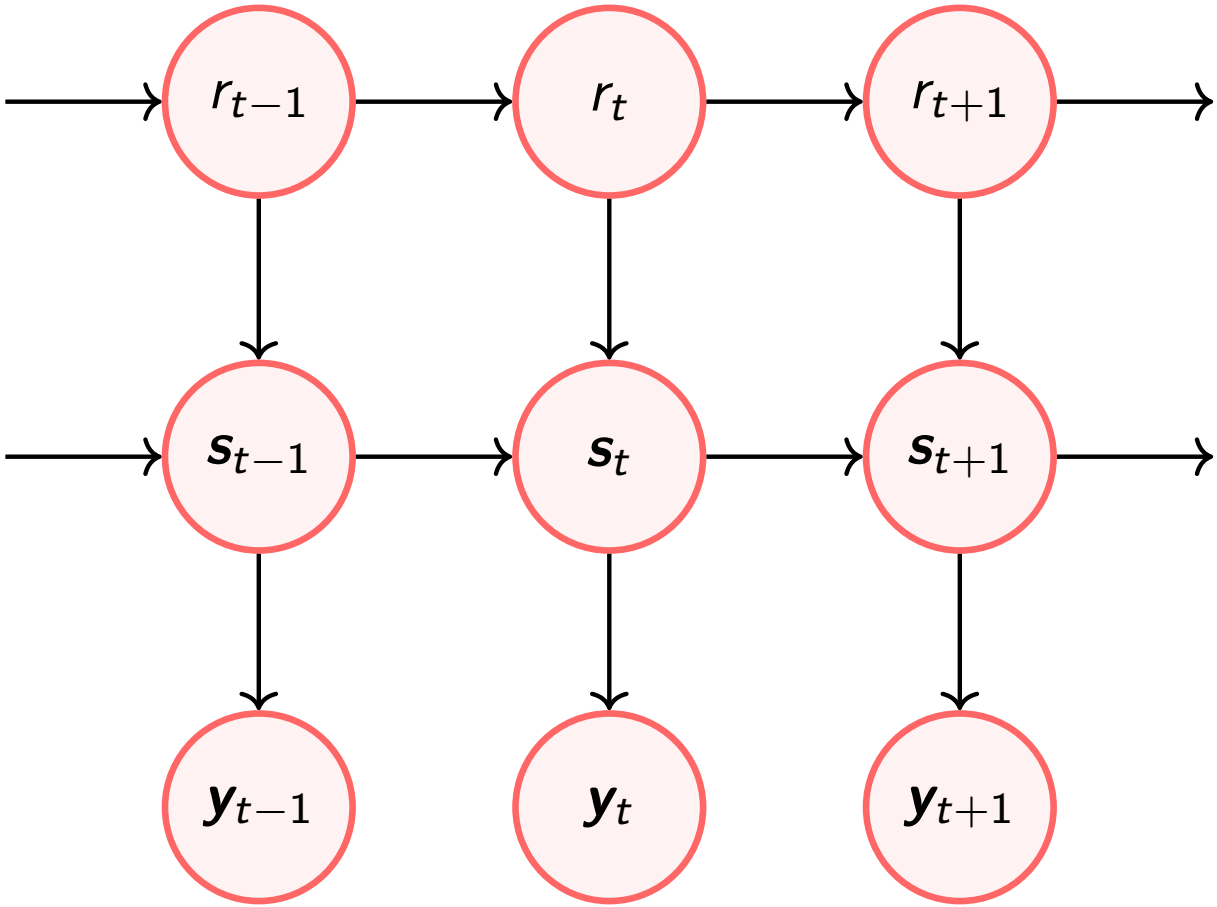
- Assume simplest stochastic SIR model
- Assume new infections are detected with probability π
- Assume $r_t = \log(\beta_t \gamma)$ with

$$r_{t+1} = r_t + N(0, \sigma^2) \qquad \text{Random walk}$$

- Unknowns:
  - Process variables (S_t, I_t, R_t)
  - Unknown "parameters" $\{r_t\}$
  - Unknown parameters (γ, σ^2)

State space formulation

- Denote $\boldsymbol{s}_t = (S_t, I_t, R_t)$
- y_t is the observed number of new infected at time n



General state space formulation

- Model

$$\boldsymbol{x}_t \sim p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1};\boldsymbol{\theta}) \qquad \text{State process}$$

$$\boldsymbol{y}_t \sim p(\boldsymbol{y}_t|\boldsymbol{x}_t;\boldsymbol{\theta}) \qquad \text{Observations}$$

- Markov structure

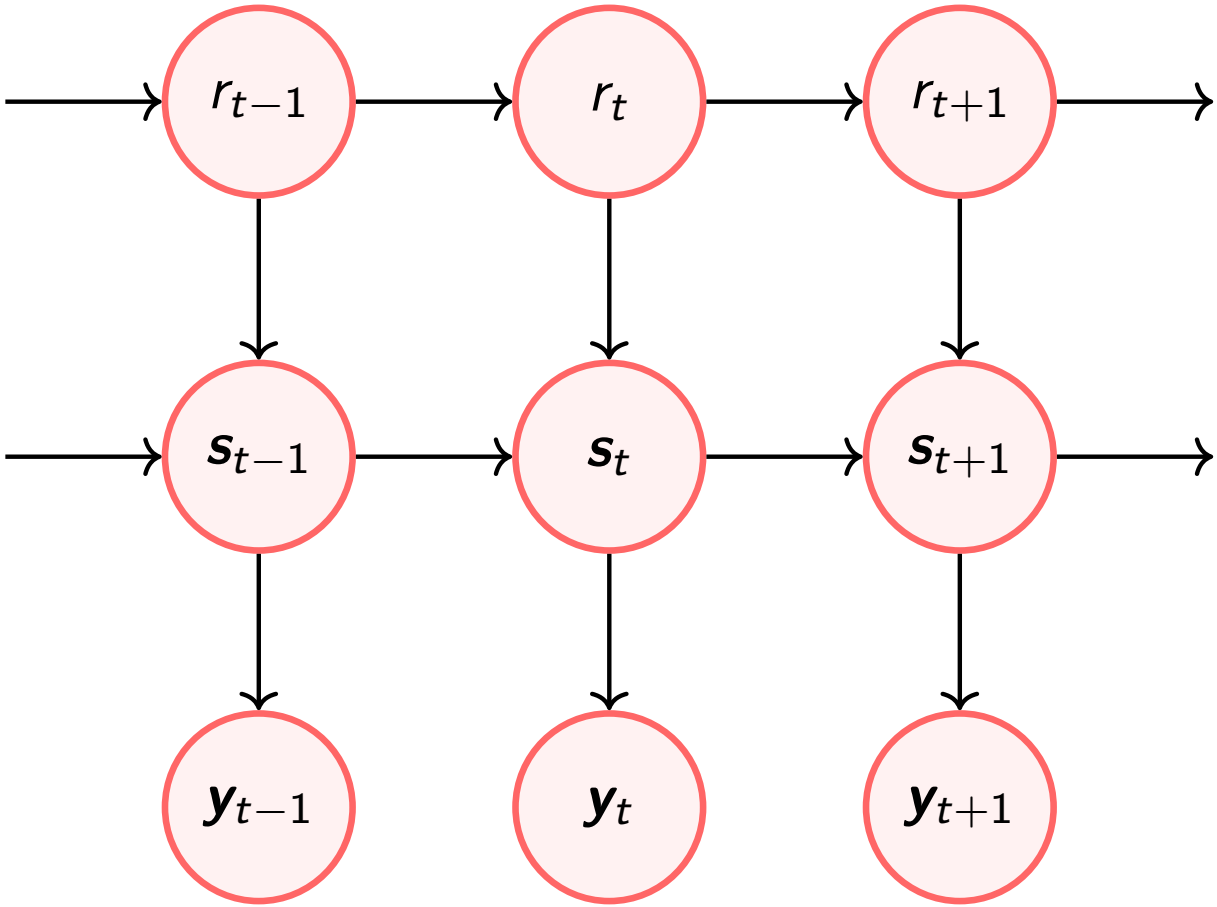
$$p(\boldsymbol{x}_{1:t};\boldsymbol{\theta}) = p(x_1;\boldsymbol{\theta}) \prod_{s=2}^{t} p(\boldsymbol{x}_s|\boldsymbol{x}_{s-1};\boldsymbol{\theta})$$

- Conditional independence

$$p(\boldsymbol{y}_{1:t}|\boldsymbol{x}_{1:t};\boldsymbol{\theta}) = \prod_{s=1}^{t} p(\boldsymbol{y}_s|\boldsymbol{x}_s;\boldsymbol{\theta})$$

State space formulation

- Denote $\boldsymbol{s}_t = (S_t, I_t, R_t)$
- y_t is the observed number of new infected at time n



Inference in state space models

- Filtering: $p(\boldsymbol{x}_t|\boldsymbol{y}_{1:t})$

$$p(\boldsymbol{x}_t|\boldsymbol{y}_{1:t}) = \int_{\boldsymbol{x}_{1:t-1}} p(\boldsymbol{x}_{1:t-1})p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})p(\boldsymbol{y}_{1:t}|\boldsymbol{x}_{1:t})d\boldsymbol{x}_{1:t-1}$$

- Smoothing: $p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t})$

$$p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t}) = \frac{p(\boldsymbol{x}_{1:t}, \boldsymbol{y}_{1:t})}{p(\boldsymbol{y}_{1:t})}$$

$$p(\boldsymbol{y}_{1:t}) = \int_{\boldsymbol{x}_{1:T}} p(\boldsymbol{x}_{1:T})p(\boldsymbol{y}_{1:T}|\boldsymbol{x}_{1:T})d\boldsymbol{x}_{1:T}$$

- Parameter estimation

$$L(\theta) = p(\boldsymbol{y}_{1:T}; \boldsymbol{\theta}) = \int_{\boldsymbol{x}_{1:T}} p(\boldsymbol{x}_{1:T}; \boldsymbol{\theta})p(\boldsymbol{y}_{1:T}|\boldsymbol{x}_{1:T}; \boldsymbol{\theta})d\boldsymbol{x}_{1:T}$$

- Computational challenge

Bayesian approach

- Priors on $\boldsymbol{\theta}$
  - From other data sources
  - From literature
- Extra computational challenge:

$$p(\boldsymbol{\theta}|\boldsymbol{y}_{1:T}) = \frac{p(\boldsymbol{\theta})p(\boldsymbol{y}_{1:T}|\boldsymbol{\theta})}{p(\boldsymbol{y}_{1:T})}$$

$$p(\boldsymbol{y}_{1:T}|\boldsymbol{\theta}) = \int_{\boldsymbol{x}_{1:T}} p(\boldsymbol{x}_{1:T};\boldsymbol{\theta})p(\boldsymbol{y}_{1:T}|\boldsymbol{x}_{1:T};\boldsymbol{\theta})d\boldsymbol{x}_{1:T}$$

$$p(\boldsymbol{y}_{1:T}) = \int_{\boldsymbol{\theta}} p(\boldsymbol{\theta})p(\boldsymbol{y}_{1:T}|\boldsymbol{\theta})d\boldsymbol{\theta}$$

Monte Carlo methods

- Aim: $E[g(\boldsymbol{\theta})|\boldsymbol{y}]$
- $\boldsymbol{\theta}^b \sim p(\boldsymbol{\theta}|\boldsymbol{y})$
- Monte Carlo approximation

$$\widehat{E[g(\boldsymbol{\theta})|\boldsymbol{y}]} = \frac{1}{B}\sum_{b=1}^{B} g(\boldsymbol{\theta}^b)$$

Markov chain Monte Carlo

- In principle possible
- Might work for simple models
- Struggle with complex models
  - Might not be possible to evaluate $p(\boldsymbol{x}_{t+1}|\boldsymbol{x}_t;\boldsymbol{\theta})$
  - Might be possible to simulate from $p(\boldsymbol{x}_{t+1}|\boldsymbol{x}_t;\boldsymbol{\theta})$

Approximate Bayesian computing

- Assumes possible to simulate $\boldsymbol{y}^*_{1:T} \sim p(\boldsymbol{y}_{1:T}|\boldsymbol{\theta})$

$$\boldsymbol{x}^*_t \sim p(\boldsymbol{x}_t|\boldsymbol{x}^*_{t-1};\boldsymbol{\theta})$$
$$\boldsymbol{y}^*_t \sim p(\boldsymbol{y}_t|\boldsymbol{y}^*_t;\boldsymbol{\theta})$$

- Simulate $(\boldsymbol{\theta}^*_b, \boldsymbol{y}^*_b)$
- Keep $\{\boldsymbol{\theta}^*_b)$ for which $\text{Dist}(\boldsymbol{y}, \boldsymbol{y}^*_b)$ is small
- Will give approximate samples from posterior
- Challenges
  - How to simulate $\boldsymbol{\theta}^*$
  - Distance measure $\text{Dist}(\boldsymbol{y}, \boldsymbol{y}^*_b)$
  - What is small?

Importance sampling

- Sampling from $p(\boldsymbol{x}|\boldsymbol{y};\boldsymbol{\theta})$
- Utilise the dynamic structure of the model
- Based on importance sampling

$$
\begin{aligned}
E[h(\boldsymbol{x})|\boldsymbol{y}] &= \int_{\boldsymbol{x}} h(\boldsymbol{x}) p(\boldsymbol{x}|\boldsymbol{y}) d\boldsymbol{x} \\
&= \int_{\boldsymbol{x}} h(\boldsymbol{x}) \frac{p(\boldsymbol{x}|\boldsymbol{y})}{q(\boldsymbol{x})} q(\boldsymbol{x}) d\boldsymbol{x} \\
&\approx \frac{1}{B} \sum_{b=1}^{B} h(\boldsymbol{x}^b) \frac{p(\boldsymbol{x}^b|\boldsymbol{y})}{q(\boldsymbol{x}^b)} \qquad \boldsymbol{x}^b \sim q(\boldsymbol{x})
\end{aligned}
$$

Sequential Monte Carlo

- Chopin and Papaspiliopoulos (2020): An Introduction to Sequential Monte Carlo
- Main idea:
  - Assume $\hat{p}(\boldsymbol{x}_t|\boldsymbol{y}_{1:t}) \approx \sum_{b=1}^{B} w_t^b \delta_{\boldsymbol{x}_t}(\boldsymbol{x}_t^b)$
  - Update to $\hat{p}(\boldsymbol{x}_{t+1}|\boldsymbol{y}_{1:t+1}) \approx \sum_{b=1}^{B} w_{t+1}^b \delta_{\boldsymbol{x}_{t+1}}(\boldsymbol{x}_{t+1}^b)$
- Desired properties:
  - Consistency: $\hat{p}(\boldsymbol{x}_t|\boldsymbol{y}_{1:t}) \to p(\boldsymbol{x}_t|\boldsymbol{y}_{1:t})$ as $B \to \infty$
  - Control of error: Error does not increase with t

Recursive formulas for state space models

- General state space model

$$\begin{aligned} \boldsymbol{x}_t|\boldsymbol{x}_{1:t-1} &\sim p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) \\ \boldsymbol{y}_t|\boldsymbol{x}_{1:t}, \boldsymbol{y}_{1:t-1} &\sim p(\boldsymbol{y}_t|\boldsymbol{x}_t) \end{aligned}$$

- Break large simulation from $p(\boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T})$ into smaller pieces
- Sequentially updating $\hat{p}(\boldsymbol{x}_{1:t-1}|\boldsymbol{y}_{1:t-1})$ to $\hat{p}(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t})$
- Recursive relation:

$$p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t-1}) = p(\boldsymbol{x}_{1:t-1}|\boldsymbol{y}_{1:t-1})p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) \quad \text{Forecast}$$

$$p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t}) = \frac{p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t-1})p(\boldsymbol{y}_t|\boldsymbol{x}_t)}{p(\boldsymbol{y}_t|\boldsymbol{x}_{1:t-1})} \quad \text{Update}$$

$$p(\boldsymbol{y}_t|\boldsymbol{x}_{1:t-1}) = \int_{\boldsymbol{x}_t} p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t-1})p(\boldsymbol{y}_t|\boldsymbol{x}_t)d\boldsymbol{x}_t$$

Sequential importance sampling

- From model:

$$p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t}) \propto p(\boldsymbol{x}_1)p(\boldsymbol{y}_1|\boldsymbol{x}_1)\prod_{s=2}^{t} p(\boldsymbol{x}_s|\boldsymbol{x}_{s-1})p(y_s|\boldsymbol{x}_s)$$

- Samples (simplest case):

$$q(\boldsymbol{x}_{1:t}) = p(\boldsymbol{x}_1)\prod_{s=2}^{t} p(\boldsymbol{x}_s|\boldsymbol{x}_{s-1})$$

- Importance weights:

$$w_t = \frac{p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t})}{q(\boldsymbol{x}_{1:t})} = \prod_{s=1}^{t} p(\boldsymbol{y}_s|\boldsymbol{x}_s) = w_{t-1}p(\boldsymbol{y}_t|\boldsymbol{x}_t)$$

- $\mathrm{Var}(w_t)$ will increase with $t \Rightarrow$ degeneracy
- Resampling avoids weight degeneracy

Algorithm

- Recursive relation:

$$p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t-1}) = p(\boldsymbol{x}_{1:t-1}|\boldsymbol{y}_{1:t-1})p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$$
$$p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t}) \propto p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t-1})p(\boldsymbol{y}_t|\boldsymbol{x}_t)$$

- Assume samples $\{\boldsymbol{x}_{t-1}^1, ..., \boldsymbol{x}_{t-1}^B\}$ from $p(\boldsymbol{x}_{1:t-1}|\boldsymbol{y}_{1:t-1})$
  1. Forecasting: Simulate $\tilde{\boldsymbol{x}}_t^b$ from $q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}^b)$, put $\tilde{\boldsymbol{x}}_{1:t}^b = (\boldsymbol{x}_{1:t-1}^b, \tilde{\boldsymbol{x}}_t^b)$
  2. Calculate weights

$$w_t^b = \frac{p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}^b)p(\boldsymbol{y}_t|\tilde{\boldsymbol{x}}_t)}{q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}^b)}, \quad W_t^b = w_t^b / \sum_{b'=1}^{B} w_t^{b'}$$

  3. Update

$$\hat{p}(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t}) = \sum_{b=1}^{B} W_t^b \delta_{\tilde{\boldsymbol{x}}_{1:t}^b}(\boldsymbol{x}_{1:t})$$

  4. Resample $\{(\boldsymbol{x}_{1:t}^1, ..., \boldsymbol{x}_{1:t}^B)\}$ from $\{(\tilde{\boldsymbol{x}}_{1:t}^1, ..., \tilde{\boldsymbol{x}}_{1:t}^B)\}$ with probabilities $\{W_t^b\}$

Algorithm - Bootstrap filter

- $q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1} = p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) \Rightarrow w_t = p(\boldsymbol{y}_t|\boldsymbol{x}_t)$
- Assume samples $\{\boldsymbol{x}_{t-1}^1, ..., \boldsymbol{x}_{t-1}^B\}$ from $p(\boldsymbol{x}_{1:t-1}|\boldsymbol{y}_{1:t-1})$
  1. Forecasting: Simulate $\tilde{\boldsymbol{x}}_t^b$ from $p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}^b)$, put $\tilde{\boldsymbol{x}}_{1:t}^b = (\boldsymbol{x}_{1:t-1}^b, \tilde{\boldsymbol{x}}_t^b)$
  2. Calculate weights

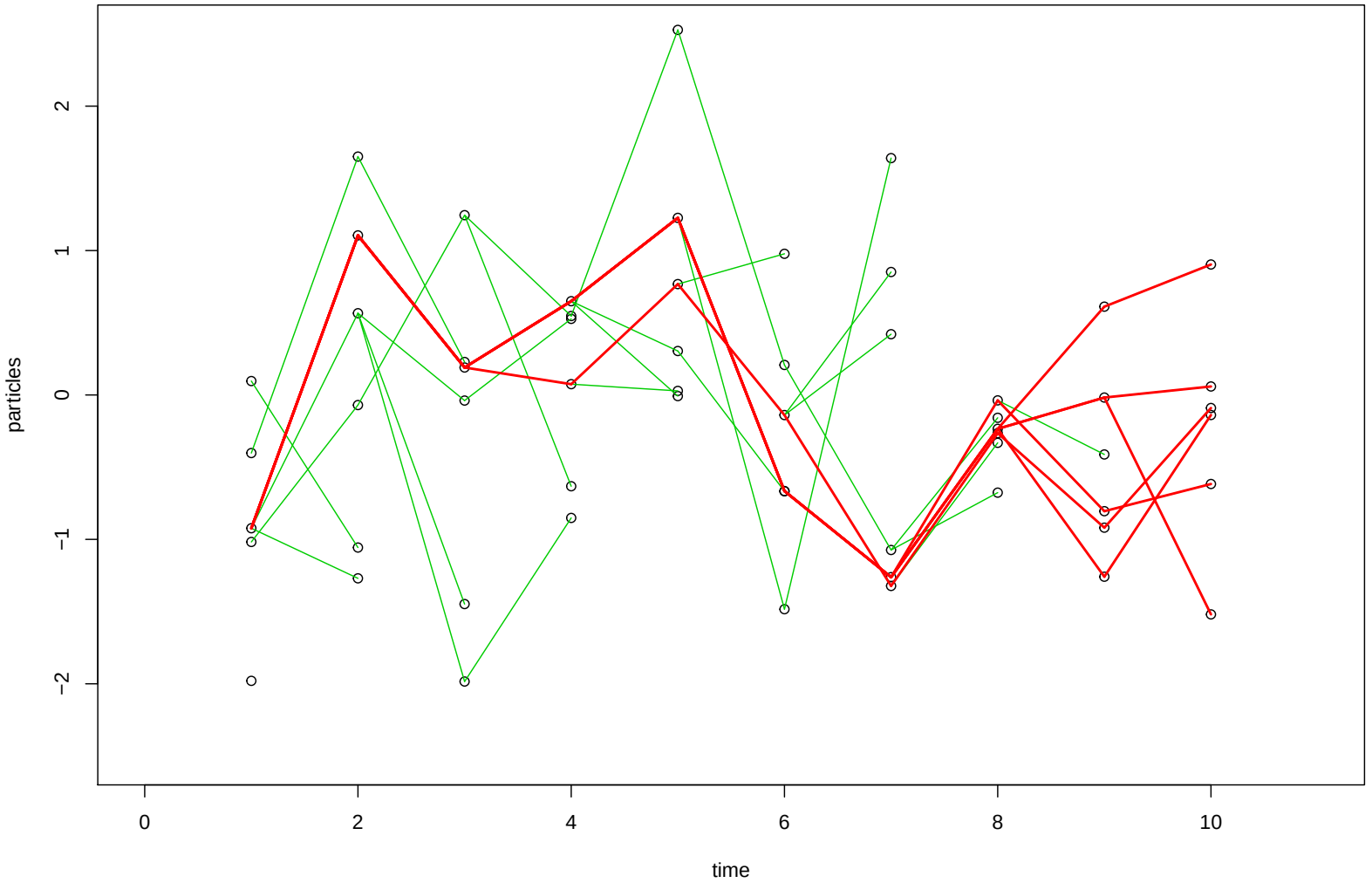
$$w_t^b = p(\boldsymbol{y}_t|\tilde{\boldsymbol{x}}_t), \quad W_t^b = w_t^b / \sum_{b'=1}^{B} w_t^{b'}$$

  3. Update

$$\hat{p}(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t}) = \sum_{b=1}^{B} W_t^b \delta_{\tilde{\boldsymbol{x}}_{1:t}^b}(\boldsymbol{x}_{1:t})$$

  4. Resample $\{(\boldsymbol{x}_{1:t}^1, ..., \boldsymbol{x}_{1:t}^B)\}$ from $\{(\tilde{\boldsymbol{x}}_{1:t}^1, ..., \tilde{\boldsymbol{x}}_{1:t}^B)\}$ with probabilities $\{W_t^b\}$

Sequential Monte Carlo



Simple example

- Model

$$x_t = a x_{t-1} + \varepsilon_t \qquad \varepsilon_t \sim N(0, \sigma_x^2)$$
$$y_t = x_t + \eta_t, \qquad \eta_t \sim N(0, \sigma_y^2)$$

- Aim: $p(x_t|\boldsymbol{y}_{1:t})$.
- Proposal distribution: $q(x_t|\boldsymbol{x}_{1:-1}) = p(x_t|x_{t-1})$
- Weight: $w_t = p(y_t|x_t)$
- Script: `SMC_AR.Rmd`

Algorithm

```
nT = length(y)
xsim = matrix(nrow=nT,ncol=B)
#Proposal at time 1
xsim[1,] = rnorm(B,0,sigma.x/sqrt(1-a^2))
w = dnorm(y[1],xsim[1,],sigma.y)
# Resample
ind = sample(1:B,B,prob=w,replace=T)
xsim[1,] = xsim[1,ind]
for(i in 2:nT)
{
 # Proposals
 xsim[i,] = rnorm(B,a*xsim[i-1,],sigma.x)
 w = dnorm(y[i],xsim[i,],sigma.y)
 # Resample
 ind = sample(1:B,B,prob=w,replace=T)
 xsim[1:i,] = xsim[1:i,ind]
 }
```

Theoretical properties

- As $B \to \infty$, for any t,

$$\hat{h}_t(\boldsymbol{x}_{1:t}) = \sum_{b=1}^{B} W_t^b h(\boldsymbol{x}_{1:t}^b) \to \mu_t(h) = E^{p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t})}[h(\boldsymbol{x}_{1:t})],$$

$$\sqrt{B}(\hat{h}_t(\boldsymbol{x}_{1:t}) - \mu_t(h)) \xrightarrow{d} N(0, V_t(h))$$

- For fixed B as t increases, $V_t(h)$ will in general increase
- For fixed B and if $h(\boldsymbol{x}_{1:t}) = h(\boldsymbol{x}_t)$, then $V_t(h)$ will be stable!
- Many more results, see Chopin and Papaspiliopoulos (2020); Moral (2004); Naesseth et al. (2019)

Parameter estimation with SMC

- So far: Simulation from $p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t}, \boldsymbol{\theta})$
- Aim now: $p(\boldsymbol{\theta}|\boldsymbol{y}_{1:T})$ or $p(\boldsymbol{\theta}, \boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T})$ or $p(\boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T})$
- Main challenge: $\boldsymbol{\theta}$ is static
- Several possibilities:
  - Make $\boldsymbol{\theta}$ dynamic (Liu and West, 2001):

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + N(0, \zeta_t)$$

  - Sufficient statistics approach (Storvik, 2002; Fearnhead, 2002): Simulate from $p(\boldsymbol{x}_t|\boldsymbol{x}_{1:t-1})$ through
    1. Simulate $\boldsymbol{\theta} \sim p(\boldsymbol{\theta}^*|\boldsymbol{x}_{1:t-1}) = p(\boldsymbol{\theta}^*|\boldsymbol{s}_t)$
    2. Simulate $\boldsymbol{x}_t \sim p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{\theta}^*)$
  - Particle MCMC (Andrieu et al., 2010)

Particle MCMC

- Likelihood $L(\boldsymbol{\theta}) = p(\boldsymbol{y}_{1:T}|\boldsymbol{\theta})$
- M-H:
  1. Propose $\boldsymbol{\theta}^* \sim q(\boldsymbol{\theta}^*|\boldsymbol{\theta})$
  2. Accept with probability

$$\alpha = \min\left\{1, \frac{p(\boldsymbol{\theta}^*)L(\boldsymbol{\theta}^*)q(\boldsymbol{\theta}|\boldsymbol{\theta}^*)}{p(\boldsymbol{\theta})L(\boldsymbol{\theta})q(\boldsymbol{\theta}^*|\boldsymbol{\theta})}\right\}$$

- Problem: $L(\boldsymbol{\theta})$ difficult to compute.
- Andrieu et al. (2009): Replace α by

$$\hat{\alpha} = \min\left\{1, \frac{p(\boldsymbol{\theta}^*)\hat{L}(\boldsymbol{\theta}^*)q(\boldsymbol{\theta}|\boldsymbol{\theta}^*)}{p(\boldsymbol{\theta})\hat{L}(\boldsymbol{\theta})q(\boldsymbol{\theta}^*|\boldsymbol{\theta})}\right\}$$

  where $E[\hat{L}(\boldsymbol{\theta})] = L(\boldsymbol{\theta})$
- Andrieu et al. (2010): Obtain $\hat{L}(\boldsymbol{\theta})$ through SMC

Algorithm with $\hat{L}(\theta)$

Require: $\theta^{(0)}$,
  Compute $\widehat{L}(\theta^{(0)})$ by SMC
  for $i = 1, ..., N$ **do**
    $\theta^* \sim q(\cdot|\theta^{(i-1)})$
    Compute $\widehat{L}(\theta^*)$ by SMC
    Compute $\hat{\alpha} = \min\left\{1, \frac{p(\theta^*)\hat{L}(\theta^*)q(\theta|\theta^*)}{p(\theta)\hat{L}(\theta^{(i-1)})q(\theta^*|\theta)}\right\}$
    if $unif(1) < \hat{\alpha}$ **then**
      $\theta^{(i)} = \theta^*$
    else
      $\theta^{(i)} = \theta^{(i-1)}$
      Store $L(\theta^{(i)}$
    end if
  end for

Simple example

- Reparametrize:

$$\theta_1 = \log \frac{1+a}{1-a}$$

$$\theta_2 = \log \sigma_x$$

$$\theta_3 = \log \sigma_y$$

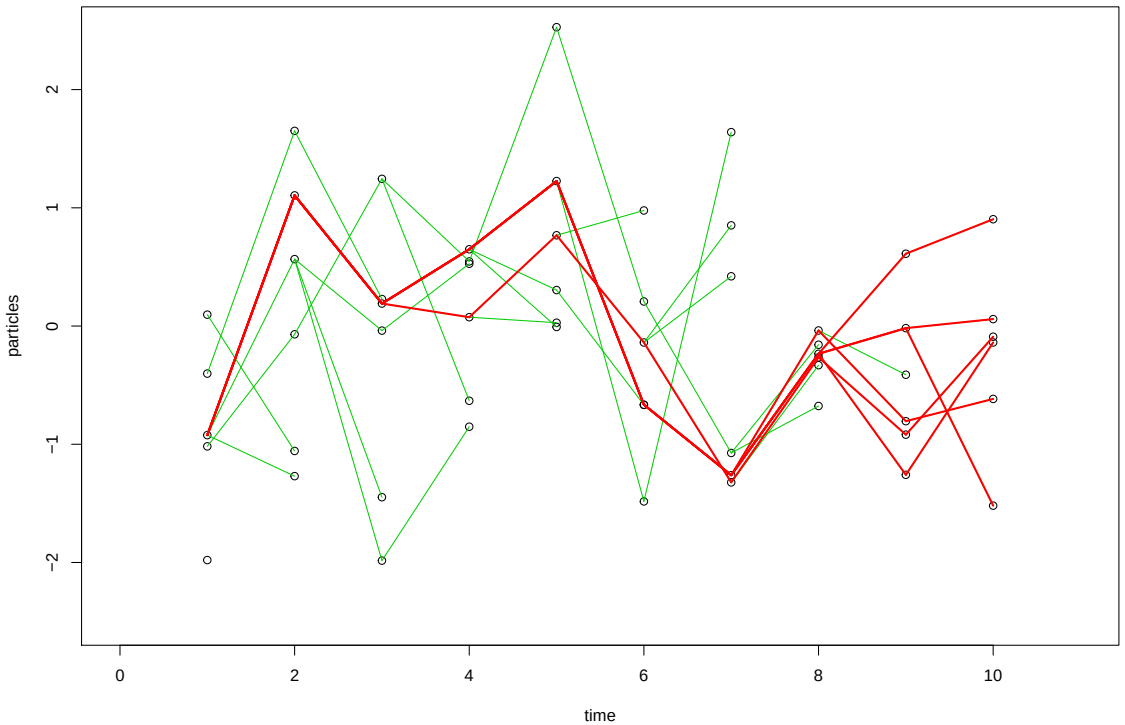
- Prior: Uniform on $\boldsymbol{\theta}$
- q: Random walk on $\boldsymbol{\theta}$
- Script: `SMC_AR.Rmd`

Smothing

- So far: Filtering $p(\boldsymbol{x}_t|\boldsymbol{y}_{1:t};\boldsymbol{\theta})$
- Sometimes of interest:
  - $p(\boldsymbol{x}_s|\boldsymbol{y}_{1:t};\boldsymbol{\theta})$ for $s \leq t$
  - $p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t};\boldsymbol{\theta})$ for $s \leq t$
- Several possible approaches
  - Resampling full path
  - Fixed lag smoothing: $p(\boldsymbol{x}_t|\boldsymbol{y}_{1:T}) \approx p(\boldsymbol{x}_t|\boldsymbol{y}_{1:t+s})$
  - Backwards sampling

Resampling full path

- By resampling $\boldsymbol{x}_{1:t}$ at all times, samples from $p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t})$ are obtained
- At time T: Samples from $p(x_t|\boldsymbol{y}_{1:T})$ for all t
- Degeneracy: Only a few unique values for small t



Backwards sampling

- We have

$$p(\boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T}) = p(\boldsymbol{x}_T|\boldsymbol{y}_{1:T}) \prod_{t=T-1}^{1} p(\boldsymbol{x}_t|\boldsymbol{x}_{t+1}, \boldsymbol{y}_{1:T})$$

- At time T: Samples from $p(\boldsymbol{x}_T|\boldsymbol{y}_{1:T})$
- We have

$$\begin{aligned} p(\boldsymbol{x}_t|\boldsymbol{x}_{t+1:T}, \boldsymbol{y}_{1:T}) &= p(\boldsymbol{x}_t|\boldsymbol{x}_{t+1:T}, \boldsymbol{y}_{1:t}) && \text{cond. ind} \\ &= p(\boldsymbol{x}_t|\boldsymbol{x}_{t+1}, \boldsymbol{y}_{1:t}) && \text{Markov} \\ &\propto p(\boldsymbol{x}_t|\boldsymbol{y}_{1:t}) p(\boldsymbol{x}_{t+1}|\boldsymbol{x}_t) && \text{Bayes} \end{aligned}$$

- Further:

$$p(\boldsymbol{x}_t|\boldsymbol{y}_{1:t}) \approx \sum_{b=1}^{B} w_t^b \delta(\boldsymbol{x}_t^b)$$

$$p(\boldsymbol{x}_t|\boldsymbol{x}_{t+1}, \boldsymbol{y}_{1:t}) \approx C \sum_{b=1}^{B} w_t^b p(\boldsymbol{x}_{t+1}|\boldsymbol{x}_t^b) \delta(\boldsymbol{x}_t^b)$$

- Simulate $\boldsymbol{x}_t$ from $\{\boldsymbol{x}_1^1, ..., \boldsymbol{x}_t^B\}$ with probabilities proportional to $w_t p(\boldsymbol{x}_t^{b'}|\boldsymbol{x}_{t+1})^b$
- Possible to construct versions that is $O(B)$!

SMC and infectious diseases

- Back to infectious diseases
- Chain-binomial model: State space model with some challenges:
  - Possible to sample from $p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$
  - Delays in data
    - Delays from infectious to test: Shift data by $d_{test} = 4$ days
    - Delays from infectious to hospital: Shift data by $d_{hosp} = 14$ days
  - Discretization: Run simulations with h corresponding to 6 hours (1/4 day)
  - Strong seasonality within week on tests: Smooth data
  - Infections from outside

Initialization/seeding

- Population starts with no one infected
- Infections from outside (abroad)
- Norway: Number of tested positive from abroad b_t
  - Not all registered:

$$B_t = b_t + \text{Poisson}((a-1) * b_t) \qquad a = 2.8$$
$$I_t = I_{t-1} + I_t^{new} + B_t - R_t^{new}$$

Estimating static parameters - simple approach

- Also unknown parameters $\boldsymbol{\theta}$.
- Bayesian approach:

$$p(\boldsymbol{\theta}, \boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T}) \propto p(\boldsymbol{\theta})p(\boldsymbol{x}_1|\boldsymbol{\theta})p(\boldsymbol{y}_1|\boldsymbol{x}_1, \boldsymbol{\theta})\times \prod_{t=2}^{T} p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{\theta})p(\boldsymbol{y}_t|\boldsymbol{x}_t, \boldsymbol{\theta})$$

- Simple approach: Define $\boldsymbol{\theta}_t = \boldsymbol{\theta}$

$$p(\boldsymbol{\theta}_{1:T}, \boldsymbol{x}_{1:T}|\boldsymbol{y}_{1:T}) \propto p(\boldsymbol{\theta}_1, \boldsymbol{x}_1)p(\boldsymbol{y}_1|\boldsymbol{x}_1, \boldsymbol{\theta}_1)\times \prod_{t=2}^{T} p(\boldsymbol{\theta}_t, \boldsymbol{x}_t|\boldsymbol{\theta}_{t-1}, \boldsymbol{x}_{t-1})p(\boldsymbol{y}_t|\boldsymbol{x}_t, \boldsymbol{\theta}_t)$$

Same structure, can use SMC as before

- $p(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1}) = I(\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1})$
- Resampling: Only a few unique samples of $\boldsymbol{\theta}_t$ after some time

Estimating static parameters - sufficient statistics

- Idea: Marginalize out $\boldsymbol{\theta}$
  - Complication: $p(\boldsymbol{x}_t|\boldsymbol{x}_{1:t-1})$ complicated
- Assume
  - $p(\boldsymbol{y}_t|\boldsymbol{x}_t, \boldsymbol{\theta}) = p(\boldsymbol{y}_t|\boldsymbol{x}_t)$
  - $p(\boldsymbol{\theta}|\boldsymbol{x}_{1:t}) = p(\boldsymbol{\theta}|G(\boldsymbol{x}_{1:t}))$
  - $G(\boldsymbol{x}_{1:t})$ is easy to update from $G(\boldsymbol{x}_{1:t-1})$
  - $p(\boldsymbol{\theta}|G(\boldsymbol{x}_{1:t}))$ is easy to sample from
- Imply we can easily sample from $p(\boldsymbol{x}_t|\boldsymbol{x}_{1:t-1})$
  - Sample $\boldsymbol{\theta} \sim p(\boldsymbol{\theta}|\boldsymbol{x}_{1:t-1})$
  - Sample $\boldsymbol{x}_t \sim p(\boldsymbol{x}_t|\boldsymbol{\theta}, \boldsymbol{x}_{t-1})$
- Can then constuct SMC for $(\boldsymbol{x}_t, G(\boldsymbol{x}_{1:t}))$
- Fearnhead (2002); Storvik (2002):

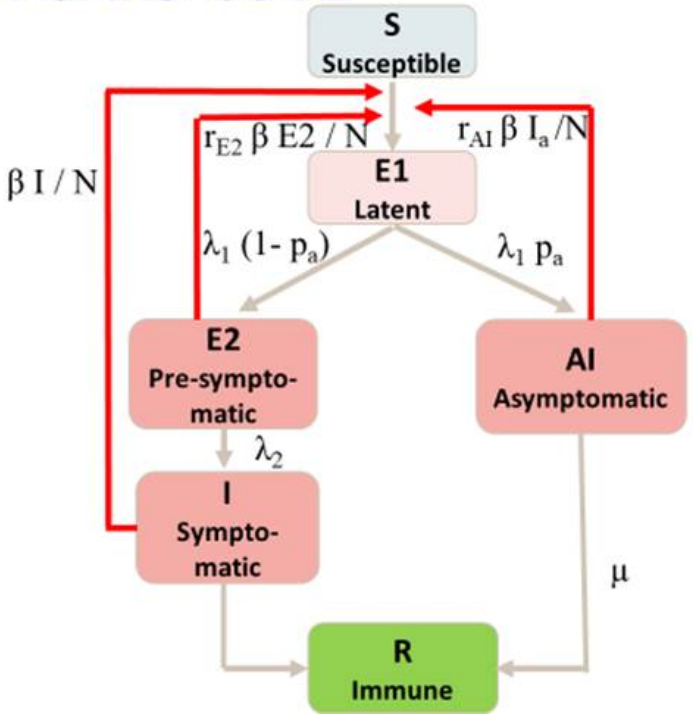
Extensions

Extensions of the SIR model

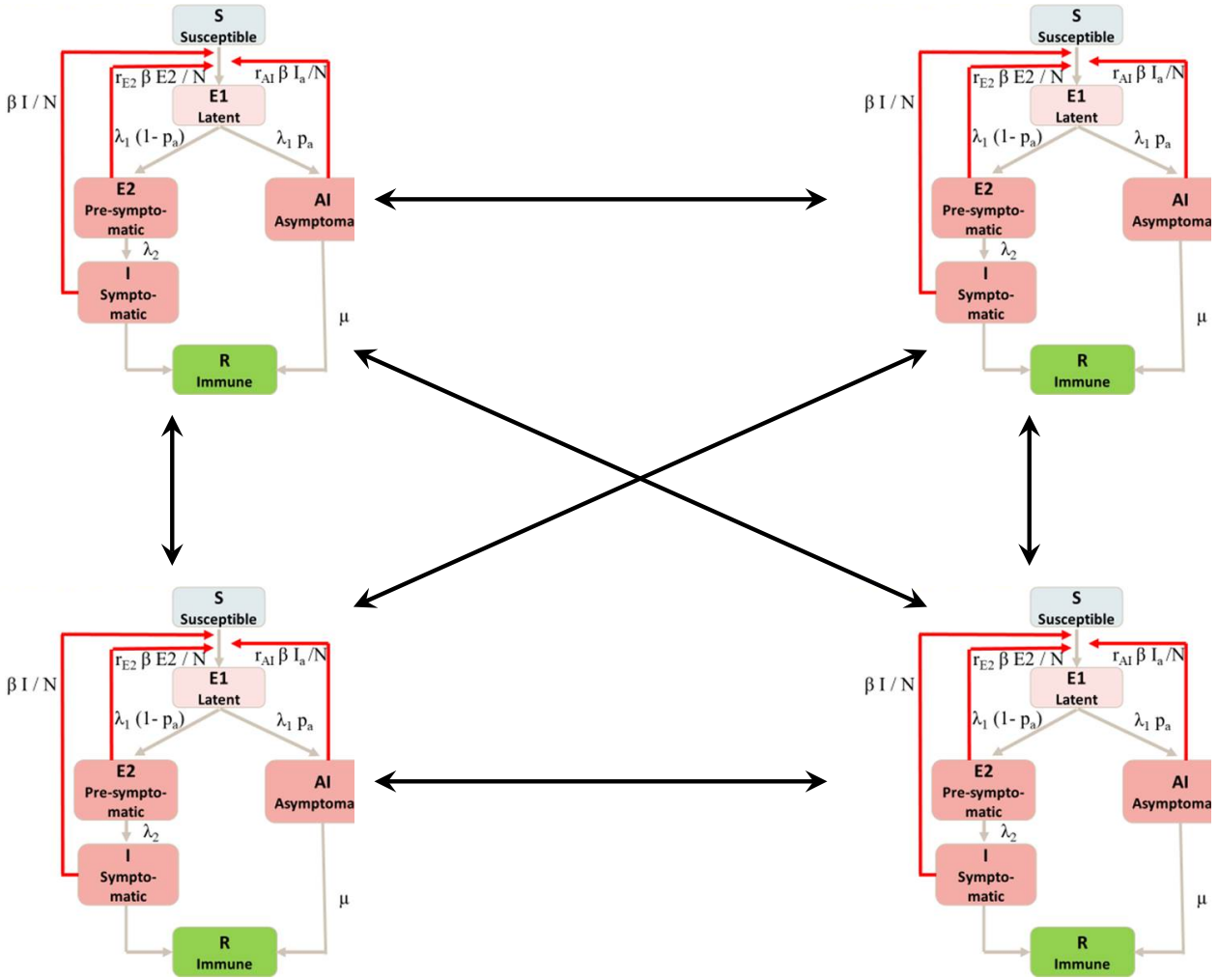
Transmisison model for SARS-CoV2

- SEIR-type model
- Duration
  - Latency period: 3 days (λ_1)
  - Sympt./Asympt. infection: 5 days (μ)
  - Presympt. infection: 2 days (λ_2)
- Relative infectiousness
  - Asymptomatic infection: 10% (r_{AI})
  - Presymptomatic infection: 125% (r_{E2})
- Proportion asymptomatic
  - p_a : 40% ()
- Feedback (red arrows)

  Force of infection (FOI_t) depends on the instantenous number of infected at time t

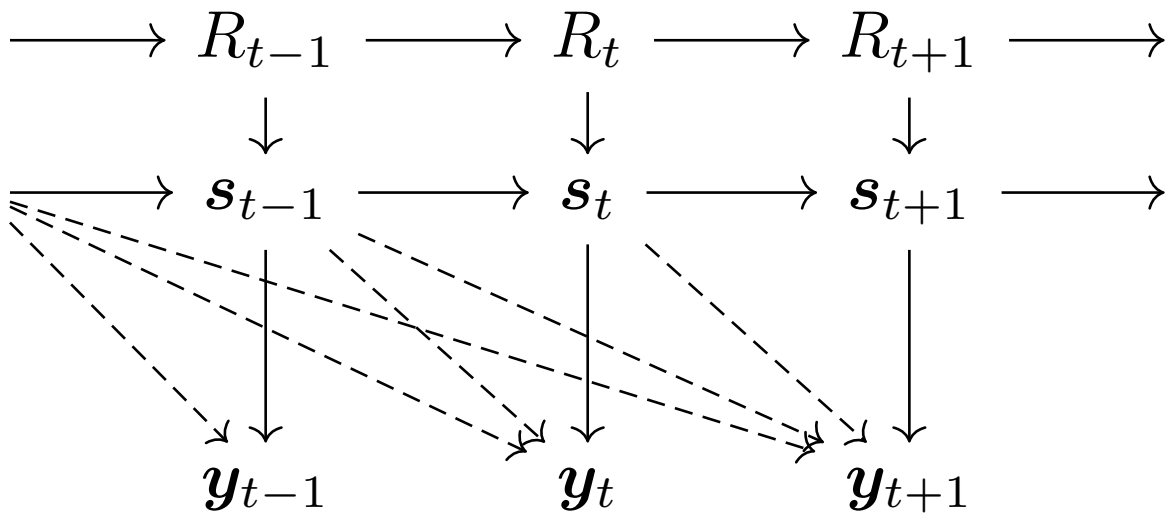


Regional model - mobility



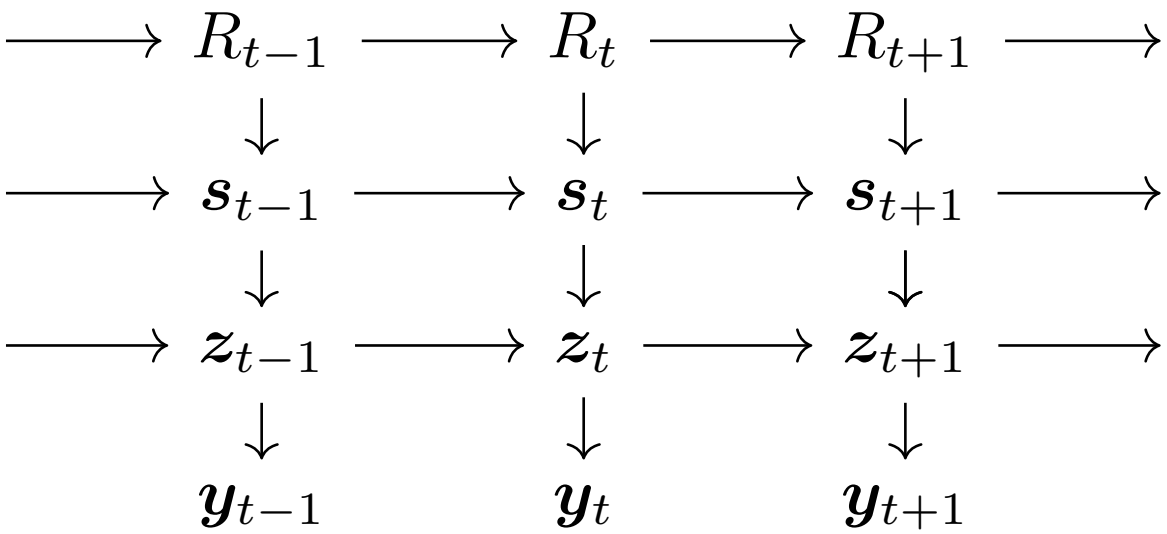
Stochastic delays

- Assumed fixed delay for hopsital/test data
- In practice: Individual differences
- Possible extension: Stochastic delays
- Observations: $\boldsymbol{y}_t$ related to $\boldsymbol{s}_{t-d:t}$



Stochastic delays - cont

- Include extra latent variables $\boldsymbol{z}_t$
  - $z_{t,s}$: Number of individuals infected at time $t-s$ but tested at time t



- Obtain Markov structure, but the dimension of latent variables increases

Conflicts

- Conflicts between prior (model) and data
- Conflicts between data sources
- SMC: Importance weights become extreme

Challenges

Challenges

- $\boldsymbol{x}_t = (R_t, \boldsymbol{s}_t, \boldsymbol{z}_t)$ is large
  - One R_t for each region
  - $S_t, E1_t, E2_t, I_t, Ia_t, R_t$ for each pairwise combination of regions (mobility!)
  - Some extra variables due to delay of data
- When observations are informative, a few weights will be large
  - If more infected than $S_{t,j}$, likelihood is zero!
  - Degeneracy: Distribution represented by very few samples
- Miller and Dunson (2018): Use $p(\boldsymbol{y}_t|\boldsymbol{x}_t)^\delta$
  - Approximation to $p(\boldsymbol{x}_{1:T}|d(\boldsymbol{y}_{1:T}, \boldsymbol{y}^*_{1:T}) < \varepsilon)$.

Algorithmic Challenges

- Algorithmic choice
  - Most simple (bootstrap) filter: Proposing new samples (particles) from $p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$
  - Many possible alternatives
    - Auxiliary particle filter
    - Nudging (importance sampling)
    - Twisting
    - Tempering
- Paradox: Struggle with high information in data
- Limitations (in our case)
  - Part of $p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$ only available through computer code
  - Flexibility with respect to model changes
  - Need results within a few hours
  - Limited number of cores (parallel computing)

Algorithmic "solutions

- Large number of particles (samples)
- Version of tempering
- Divide-and-conquer: Run models regionally, combine
- Better algorithms (Finke and Thiery, 2023)
- Online parameter estimation

Conflict measures

- Predictive checking: Comparing the simulation from the model with actual observations.
  - : Typically, summaries of data
- Difficult to specify summaries/discrepancy measures
- Possible to use importance weights?

Agent-based models

- Compartmental models assume all individuals have the same interaction pattern
- Many individual differences
- Agent-based models: Modelling individuals/households directly
- Many more unknowns
  - Parameters
  - Latent variables
- Statistical/computational challenge

Summary

- Stochastic models for infectious diseases are useful
- Bayesian approaches for
  - incorporating prior knowledge
  - quantifying uncertainty
- (Sequential) Monte Carlo methods are promising
- Challenges
  - Statistical
  - Computational
- The COVID pandemic:
  - Importance of data analysis
  - Importance of stochastic modelling
  - Interesting experience

References

Allen, L. J. (1994). Some discrete-time si, sir, and sis epidemic models. *Mathematical biosciences 124*(1), 83–105.

Allen, L. J. (2008). An introduction to stochastic epidemic models. In *Mathematical epidemiology*, pp. 81–130. Springer.

Andrieu, C., A. Doucet, and R. Holenstein (2010). Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 72*(3), 269–342.

Andrieu, C., G. O. Roberts, et al. (2009). The pseudo-marginal approach for efficient Monte Carlo computations. *The Annals of Statistics 37*(2), 697–725.

Chopin, N. and O. Papaspiliopoulos (2020). *An introduction to sequential Monte Carlo*. Springer Series in Statistics. Springer.

Fearnhead, P. (2002). Markov chain Monte Carlo, sufficient statistics, and particle filters. *Journal of Computational and Graphical Statistics 11*(4), 848–862.

Finke, A. and A. H. Thiery (2023). Conditional sequential monte carlo in high dimensions. *The Annals of Statistics 51*(2), 437–463.

Liu, J. and M. West (2001). Combined parameter and state estimation in simulation-based filtering. *Sequential Monte Carlo methods in practice*, 197–223.

Miller, J. W. and D. B. Dunson (2018). Robust bayesian inference via coarsening. *Journal of the American Statistical Association 114*(527), 1113–1125.

Moral, P. (2004). *Feynman-Kac formulae: genealogical and interacting particle systems with applications*. Springer.

Naesseth, C. A., F. Lindsten, T. B. Schön, et al. (2019). Elements of sequential monte carlo. *Foundations and Trends® in Machine Learning 12*(3), 307–392.

Storvik, G. (2002). Particle filters for state-space models with the presence of unknown static parameters. *IEEE Transactions on signal Processing 50*(2), 281–289.

Storvik, G., A. Diz-Lois Palomares, S. Engebretsen, G. Ø. I. Rø, K. Engø-Monsen, A. B. Kristoffersen, B. F. De Blasio, and A. Frigessi (2023). A sequential monte carlo approach to estimate a time-varying reproduction number in infectious disease models: the covid-19 case. *Journal of the Royal Statistical Society Series A: Statistics in Society 186*(4), 616–632.