

The Particle Filter

Connie Trojan February 23, 2022

Overview

This report introduces and discusses **particle methods** for inference on noisy or partially observed data. One application where these methods are useful is in analysing epidemic data - in this context, the spread of a disease can be modelled as a random process which we can only partially observe. The so-called **hidden state** of interest is the true number of active cases, which cannot be observed directly since not all infected individuals can be tested for the disease and tests are not 100% accurate. We might also be interested in estimating unknown **parameters** of the random process, like the transmission rate of the disease. There has been much recent interest in using particle methods in this context, as estimating true case numbers and monitoring the transmission rate of COVID-19 has been critical in informing government policy on managing its spread.

One main problem of interest in this setting is in estimating the **filtering distribution**, a probability distribution representing our knowledge (and uncertainty) about the hidden states given our sequence of observations. We can use this distribution to compute a best guess for the hidden states or to estimate the probability that they took a certain value. It is important to consider this as a probability distribution, since when there is randomness in the observation process it is not possible to be completely certain of what the hidden states actually were.

The key issue when performing this kind of analysis is that the filtering distribution often cannot be calculated exactly. In this case, we can use **particle filtering** to approximate it. This works by simulating many possibilities for the hidden states (known as **particles**) and determining how likely each one is given the data we observed. This can be used as a sub-algorithm in methods for analysing unknown parameters, such as **Markov chain Monte Carlo**.

This report will use a simple simulated epidemic dataset as a running example to illustrate the methods discussed. We will start by introducing **hidden Markov models** and defining the **filtering distribution** in Section 1. Section 2 will introduce the **bootstrap filter**, a particle filtering algorithm for approximating the filtering distribution. We will discuss methods for dealing with unknown model parameters in Section 3, focusing on the use of **particle Markov chain Monte Carlo** for performing inference jointly on the hidden states and parameters.

1 Introduction

This report will discuss the use of particle methods for inference on **hidden Markov models** (HMMs). Using the notation of Doucet et al. (2001), we will assume that we wish to perform inference on some discrete-time Markov process x_t (taking values in some state space \mathcal{X}) with initial distribution $p(x_1)$ and transitions described by $p(x_t | x_{t-1})$, subject to the restriction that we cannot observe x_t directly. Our only information about the **hidden state** x_t comes from a noisy or partial sequence of **observations** y_t with marginal distribution $p(y_t | x_t)$. The y_t are assumed to be independent given x_t .

The main problems of interest in this setting are in estimating the **filtering distribution** $p(x_t | y_{1:t})$, as well as to perform inference on any unknown parameters θ of the transition process $p(x_t | x_{t-1})$ and observation process $p(y_t | x_t)$. In the latter case we can target the joint posterior distribution $p(\theta, x_{1:t} | y_{1:t})$ to perform inference on both θ and $x_{1:t}$. The methods described in this report will be illustrated using a simple epidemiology example to be described in Section 1.2, where the hidden Markov process models the propagation of some disease through a population, and only a small fraction of the total number of infections is detected.

1.1 The Filtering Distribution

Particle filtering algorithms perform inference on hidden states by targeting the filtering distribution $p(x_t | y_{1:t})$. Assuming a HMM with known dynamics as described above, we can formulate this recursively - supposing we know the filtering distribution for time $t - 1$, we can obtain the filtering distribution for time t as:

$$p(x_t | y_{1:t}) = \frac{p(y_t | x_t) p(x_t | y_{1:t-1})}{p(y_t | y_{1:t-1})}, \quad (1)$$

where the predictive distribution $p(x_t | y_{1:t-1})$ is given by integrating over possibilities for x_{t-1} :

$$p(x_t | y_{1:t-1}) = \int_{\mathcal{X}} p(x_t | x_{t-1}) p(x_{t-1} | y_{1:t-1}) dx_{t-1}, \quad (2)$$

and the normalising constant by integrating over x_t :

$$p(y_t | y_{1:t-1}) = \int_{\mathcal{X}} p(y_t | x_t) p(x_t | y_{1:t-1}) dx_t. \quad (3)$$

Note that once this normalising constant has been computed, the marginal likelihood for y , $p(y_{1:t})$ can also be obtained recursively:

$$p(y_{1:t}) = p(y_{1:t-1}) p(y_t | y_{1:t-1}). \quad (4)$$

This result will be necessary for parameter inference, discussed in Section 3. Unfortunately, the integrals in (2) and (3) typically cannot be solved analytically, unless the HMM has a finite discrete state space or is a linear Gaussian model. We will overcome this problem by using **sequential Monte Carlo** (SMC) to obtain weighted samples of possible hidden states from $p(x_{1:t})$ and estimating the $p(x_t | y_{1:t-1})$ by the **empirical distribution function** (EDF).

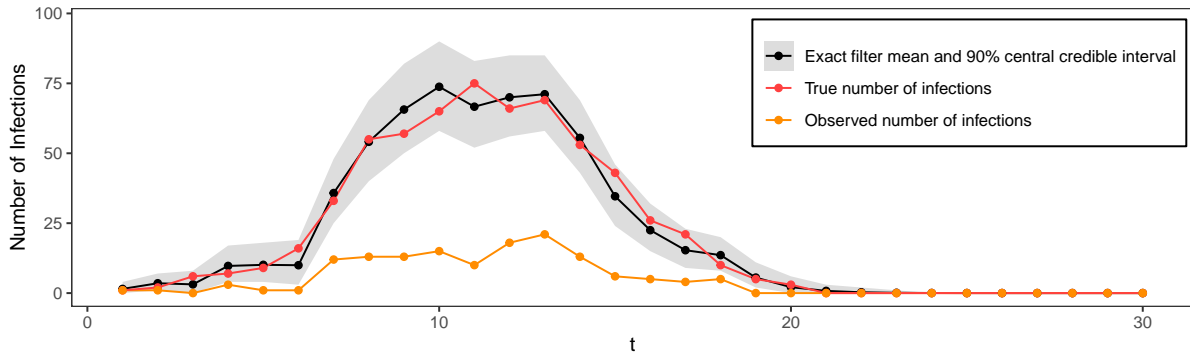


Figure 1: Simulated epidemic dataset and filtering distribution.

1.2 Example: Epidemic Modelling

Particle methods lend themselves to use in epidemiology, as the spread of a disease can be modelled by a stochastic process which is difficult to observe directly (Endo et al. (2019)). We will use the discrete time **Reed-Frost model** described by Abbey (1952) as a simple running example in this report. This is a **compartmental** model, where we categorise a closed population of fixed size N as **susceptible** to (S), **infected** by (I) or **recovered** from (R) a disease. It is assumed that the infectious period of the disease is short compared to its incubation period, so that individuals infected at time t will infect others at time $t+1$ and then recover.

We will denote the number of susceptible individuals at time t by S_t and the number of new infections by I_t , so that the hidden state in the process is $x_t = (S_t, I_t)$ and R_t can be recovered as $N - S_t - I_t$. All susceptible individuals have probability p of being infected by any given infectious individual, so that the probability of any one susceptible individual escaping infection at time $t+1$ is $(1-p)^{I_t}$. This gives rise to a binomial number of infections at each timestep, $I_{t+1} | (S_t, I_t) \sim \text{Binom}(S_t, 1 - (1-p)^{I_t})$, with $S_{t+1} = S_t - I_{t+1}$. We will assume that the epidemic is triggered by the arrival of a single ‘patient zero’, so that $I_1 \sim \text{Binom}(N, p)$ and $S_1 = N - I_1$. Finally, we will assume a fixed probability p_{obs} of detecting any given infection, so that the observation process is given by $y_t | x_t \sim \text{Binom}(I_t, p_{obs})$. See Figure 1 for a realisation of this model terminating at time $T = 30$ with $p = 0.0015$, $p_{obs} = 0.2$ and $N = 1000$, which will be used as a running example. Note that since the population is closed, the state space for this model is finite with N^2 possibilities for (S_t, I_t) . This means that we can compute the exact filtering distribution (also shown in Figure 1), although this can become prohibitively computationally expensive for large populations - in our small example this took six hours to calculate.

In light of the COVID-19 pandemic, there has been much recent interest in using particle methods in this context. The compartmental models used are usually time continuous and can take additional categories (e.g. asymptomatic, vaccinated) and varying transmission rates into account. For example, Romero-Severson et al. (2020) modeled COVID-19 case numbers as a partially observed Markov process with varying transmission rate. Their model was fitted with the iterated filtering method mentioned in Section 3 to analyse changing transmission rates.

2 Particle Filtering

2.1 Sequential Importance Sampling

One simple way of producing Monte Carlo samples from the filtering distribution is by **sequential importance sampling** (SIS) (Doucet et al. (2001)), a recursive version of the standard Monte Carlo importance sampling technique. We start by sampling m **particles** from $p(x_1)$ and iteratively propagating them forwards according to the HMM dynamics. The importance weight $w_t^{(i)}$ for each particle is updated at each timestep by multiplying $w_{t-1}^{(i)}$ by $p(y_t | x_t^{(i)})$ and re-normalising. At each timestep we estimate the filtering distribution by the weighted EDF of the particles, $\hat{p}(x_t | y_{1:t}) = \sum_{i=1}^m w_t^{(i)} \delta(x_t^{(i)} - x_t)$, where δ is the Dirac delta function.

Algorithm 1: Sequential Importance Sampling

Result: Estimate $\hat{p}(x_T | y_{1:T})$ of filtering distribution

Require: $y_{1:T}$, m , $p(x_t | x_{t-1})$, $p(y_t | x_t)$, $p(x_1)$

Initialise: For $i = 1, \dots, m$, sample $x_1^{(i)} \sim p(x_1)$ and set $w_1^{(i)} = \frac{p(y_1 | x_1^{(i)})}{\sum_j p(y_1 | x_1^{(j)})}$;

for $t = 2, \dots, T$ **do**

Propagate: for $i = 1, \dots, m$, sample $x_t^{(i)} \sim p(x_t | x_{t-1}^{(i)})$;
 Reweight: for $i = 1, \dots, m$, set $w_t^{(i)} = \frac{p(y_t | x_t^{(i)}) w_{t-1}^{(i)}}{\sum_j p(y_t | x_t^{(j)}) w_{t-1}^{(j)}}$;

end

Set $\hat{p}(x_T | y_{1:T}) = \sum_{i=1}^m w_T^{(i)} \delta(x_T^{(i)} - x_T)$.

Unfortunately, SIS typically fails as t increases as the importance weights become increasingly concentrated on a few particles with most falling in areas where the filtering distribution has low density, corresponding to possible hidden states that are very unlikely given the observed data. In fact, the number of particles required to produce good estimates increases exponentially in t (Doucet and Johansen (2009)). In our epidemiology example (see Figure 2a which shows $\mathbb{E}(I_t | y_{1:t})$ as well as the 5% and 95% quantiles), SIS did well initially but the shrinking credible intervals after $t = 10$ indicate that the sample weights are becoming increasingly concentrated on just a few trajectories.

2.2 The Bootstrap Filter

The **bootstrap filter** or **sequential importance resampling** filter was first proposed by Gordon et al. (1993) as a way of sequentially generating samples from $p(x_{1:t})$ that are more concentrated in areas of high density of $p(y_t | x_t)$. Instead of continually updating the weights, the bootstrap filter uses them to resample the particles, creating a new sample where all particles are given equal weight. This allows particles with negligible weight to be eliminated, while we simulate several **descendants** of the particles with higher weights. At each timestep of SIS, we sample the next generation of particles from a categorical distribution with outcomes $\{x_{t-1}^{(i)}\}_{i=1}^m$ and probabilities determined by their weights $\{w_{t-1}^{(i)}\}_{i=1}^m$.

Algorithm 2: The Bootstrap Filter

Result: Estimate $\hat{p}(x_T | y_{1:T})$ of filtering distribution, estimate $\hat{p}(y_{1:T})$ of data likelihood

Require: $y_{1:T}$, m , $p(x_t | x_{t-1})$, $p(y_t | x_t)$, $p(x_1)$

Initialise: For $i = 1, \dots, m$, sample $x_1^{(i)} \sim p(x_1)$ and set $w_1^{(i)} = \frac{p(y_1 | x_1^{(i)})}{\sum_j p(y_1 | x_1^{(j)})}$;

for $t = 2, \dots, T$ **do**

Resample: for $i = 1, \dots, m$, sample $\tilde{x}_{t-1}^{(i)} \sim \text{Categorical}(\{x_{t-1}^{(i)}\}_{i=1}^m, \{w_{t-1}^{(i)}\}_{i=1}^m)$;

Propagate: for $i = 1, \dots, m$, sample $x_t^{(i)} \sim p(x_t | \tilde{x}_{t-1}^{(i)})$;

Reweight: for $i = 1, \dots, m$, set $w_t^{(i)} = \frac{p(y_t | x_t^{(i)})}{\sum_j p(y_t | x_t^{(j)})}$;

 Set $\hat{p}(y_{1:t}) = \hat{p}(y_{1:t-1}) \hat{p}(y_t | y_{1:t-1}) = \hat{p}(y_{1:t-1}) \frac{1}{m} \sum_{i=1}^m p(y_t | x_t^{(i)})$;

end

Set $\hat{p}(x_T | y_{1:T}) = \sum_{i=1}^m w_t^{(i)} \delta(x_T^{(i)} - x_T)$.

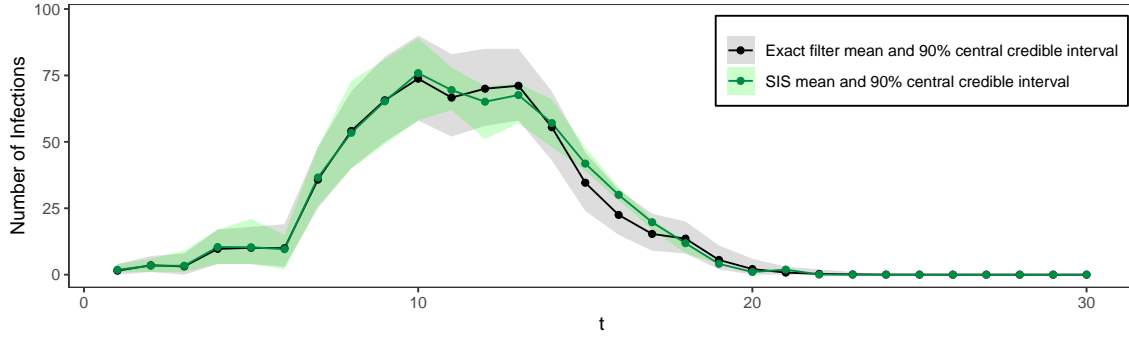
See Figure 2b for the performance of the bootstrap filter on our running example - its output with $m = 100$ is a reasonable approximation for the exact filter mean and quantiles, and for $m \geq 1000$ the two are virtually indistinguishable. so a very close approximation to the filtering distribution can be computed in seconds.

The likelihood estimation step in Algorithm 2 is optional for filtering and as discussed in Section 1.1 comes from estimating the normalising constant $p(y_t | y_{1:t-1})$ of the filtering distribution. The resulting estimate of $p(y_{1:T})$ is in fact unbiased (Fearnhead and Künsch (2018)) and will be the key output of this algorithm for the parameter estimation in Section 3.

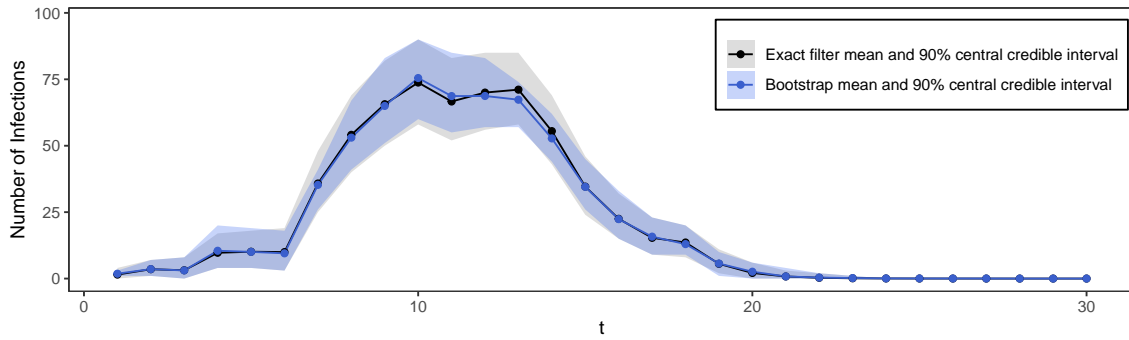
2.3 Improved Resampling Schemes

One issue with the bootstrap filter mentioned by Gordon et al. (1993) is that, if the region where the likelihood takes significant values is small, then some particles will be resampled many times - if this region is also in an area that is unlikely in the propagation step, then all of the resamples may be descended from a single particle, which will result in the same problem encountered with SIS if the propagation steps do not sufficiently restore particle diversity. Gordon et al. (1993) suggest that this could be alleviated by introducing random noise perturbation after resampling.

The resampling step in bootstrap, while alleviating the key issue with SIS, introduces extra variability in the estimates of the filtering distribution. There are other resampling schemes with lower variance that can still produce an unbiased estimate of the filtering density - one example is stratified sampling, which is used by Carpenter et al. (1999) in their **improved particle filter**, which they also showed to be less prone to the sample impoverishment problem discussed above.

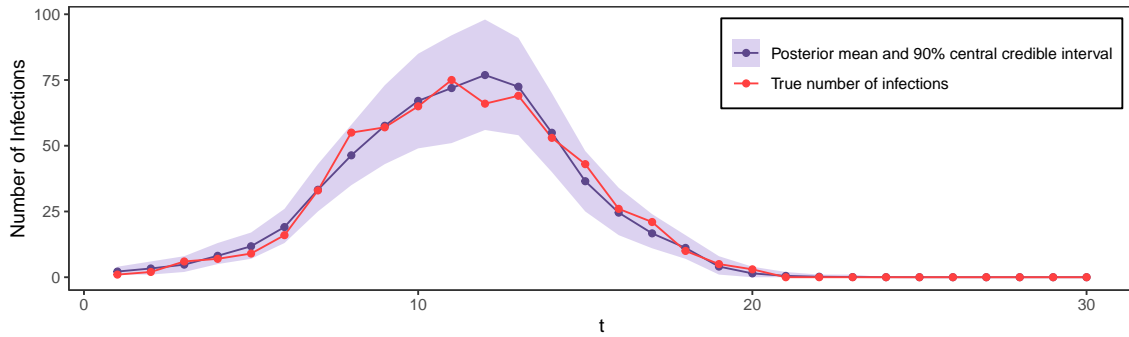


(a) Sequential importance sampling, $m = 100$ particles.

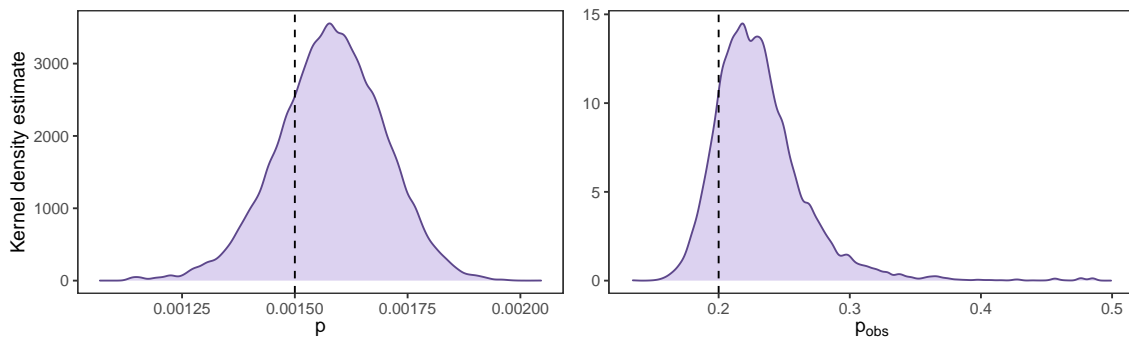


(b) The bootstrap filter, $m = 100$ particles.

Figure 2: Particle Filtering (Section 2)



(a) Estimated posterior marginal distribution for $I_{1:T}$.



(b) Estimated posterior marginals for p and p_{obs} . Dashed lines indicate true parameter values.

Figure 3: Particle MCMC (Section 3)

3 Particle Methods for Parameter Inference

So far, we have treated model parameters as known constants. This is often not the case, and parameter inference is often just as important as inference on the hidden states. We saw earlier that the data likelihood $p(y_{1:T})$ can be estimated using a particle filter. For the purposes of parameter inference, we consider this to also be a function of the unknown parameters θ , $p(y_{1:T} | \theta)$ which can be estimated for any given value of θ by using a particle filtering algorithm.

This can be used for maximum likelihood estimation, although challenges arise from the fact that this estimate of the likelihood function is noisy, and discontinuous even when common random numbers are used between evaluations (Kantas et al. (2015)). One solution is the **iterated filtering** method (Ionides et al. (2006)), which involves allowing θ to vary with t as a random walk. This means θ can be considered a hidden state in the problem formulation, so that particle filtering can be used to estimate the filter means $\mathbb{E}(\theta_t | y_{1:t})$, which are then used to iteratively update an estimate for the MLE of θ as the random walk variance is reduced.

The next section will focus on Bayesian parameter inference, since this is a natural way of capturing uncertainty about parameters and the resulting inference on the hidden states.

3.1 Particle Markov Chain Monte Carlo

The **particle marginal Metropolis-Hastings** sampler was first proposed by Andrieu et al. (2010) as a method for sampling from the joint posterior distribution $p(\theta, x_{1:T} | y_{1:T})$. It is an approximation to the usual Metropolis-Hastings sampler where a particle filtering algorithm is used to estimate the likelihood term in the acceptance probability. The filter is also used to propose samples of $x_{1:t}$, by sampling a single particle's trajectory from the distribution defined by the final filter weights $\{w_T^{(i)}\}_{i=1}^m$.

Algorithm 3: Particle Marginal Metropolis-Hastings

Result: samples from $p(\theta, x_{1:T} | y_{1:T})$

Require: prior p on θ , proposal distribution q , particle filtering algorithm

Initialise: sample $\theta_1 \sim p(\theta)$;

Set $L_1 = \hat{p}(y_{1:T} | \theta_1)$ and sample $x_{1:T}^{(1)}$;

for $n = 2, \dots$ **do**

 Propose new sample $\theta_n^* \sim q(\theta | \theta_{n-1})$;

 Set $L_n^* = \hat{p}(y_{1:T} | \theta_n^*)$ and sample $x_{1:T}^*$;

 Accept θ_n^* with probability $\min \left\{ 1, \frac{L_n^* p(\theta_n^*) q(\theta_{n-1} | \theta_n^*)}{L_{n-1} p(\theta_{n-1}) q(\theta_n^* | \theta_{n-1})} \right\}$;

 If θ_n^* was accepted, set $(\theta_n, L_n, x_{1:T}^{(n)}) = (\theta_n^*, L_n^*, x_{1:T}^*)$.

 Otherwise, set $(\theta_n, L_n, x_{1:T}^{(n)}) = (\theta_{n-1}, L_{n-1}, x_{1:T}^{(n-1)})$;

end

We'll now apply this to our epidemic example - for simplicity, uniform priors on the parameters and a normal proposal distribution with mean 0 were chosen. Estimates of the likelihood and samples from $p(x_{1:T} | y_{1:T})$ were obtained using the bootstrap filter described in Section 2.2, with $m = 500$ particles. Figure 3a shows the estimated means (and 90% credible intervals) of the marginal distributions of the I_t . This was close to the true number of infections at each time point but had a different shape to the filtering distribution since the marginal distribution $p(x_t | y_{1:T})$ also takes future observations into account. The estimated marginals for the parameters p and p_{obs} are shown in Figure 3b. The posterior mean for p was 0.00158 (with 90% central credible interval (0.00138, 0.00177)) and the posterior mean for p_{obs} was 0.231 (90% central credible interval (0.187, 0.293)).

Note that this is an **offline** method - we need to have observed all of the data up to time T before starting, and would have to repeat the analysis from scratch if a new datapoint arrived. This is in contrast to the particle filtering algorithms discussed, which are **online** methods and hence easy to update with new observations. Online methods do exist for particle MCMC but struggle with the particle degeneracy problem (Kantas et al. (2015)).

References

- Abbey, H. An examination of the Reed-Frost theory of epidemics. *Human Biology*, 24(3):201–233, 1952.
- Andrieu, C., Doucet, A., and Holenstein, R. Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(3):269–342, 2010.
- Carpenter, J., Clifford, P., and Fearnhead, P. Improved particle filter for nonlinear problems. *IEEE Proceedings - Radar, Sonar and Navigation*, 146:2–7(5), February 1999.
- Doucet, A. and Johansen, A. A tutorial on particle filtering and smoothing: Fifteen years later. *Handbook of nonlinear filtering*, 12(656-704):3, 2009.
- Doucet, A., Freitas, N., and Gordon, N. *Sequential Monte Carlo Methods in Practice*. Statistics for Engineering and Information Science. Springer New York, New York, NY, 2001.
- Endo, A., van Leeuwen, E., and Baguelin, M. Introduction to particle Markov-chain Monte Carlo for disease dynamics modellers. *Epidemics*, 29:100363, 2019.
- Fearnhead, P. and Künsch, H. Particle filters and data assimilation. *Annual Review of Statistics and Its Application*, 5(1):421–449, 2018.
- Gordon, N., Salmond, D., and Smith, A. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEEE Proceedings F. Radar and Signal Processing*, 140(2):P.107, 1993.
- Ionides, E. L., Bretó, C., and King, A. A. Inference for nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 103(49):18438–18443, 2006.
- Kantas, N., Doucet, A., Singh, S., Maciejowski, J., and Chopin, N. On Particle Methods for Parameter Estimation in State-Space Models. *Statistical Science*, 30(3):328 – 351, 2015.
- Romero-Severson, E., Hengartner, N., Meadors, G., and Ke, R. Change in global transmission rates of COVID-19 through May 6 2020. *PLOS ONE*, 15(8):1–12, 08 2020.