Sequential Monte Carlo Methods for United Kingdom Population Dynamics

Zikai Liu

April 5, 2024

Abstract

Accurate analysis of human population dynamics presented a significant challenge, often relying on models that can disentangle underlying demographic processes from inherent stochasticity and observational error. This study investigated how Sequential Monte Carlo Methods (SMC) could refine parameter estimation and state prediction within state-space models for time series analysis. We applied these methods along with Kalman filters to data from the Office of National Statistics (ONS) for England & Wales (1839-2021), using both frequentist and Bayesian approaches. Our findings demonstrated the superiority of a dynamic SMC bootstrap filter, where each sub-demographic process drove the state transition equation. This approach achieved superior predictive accuracy compared to the traditional bootstrap filter and Kalman filter, highlighting the potential of SMC methods for analyzing human population dynamics. Future research directions include exploring advanced parameter estimation techniques like particle MCMC and iterated filtering for improved performance.

Contents

1 Introduction

The study of population dynamics, particularly in the context of human populations, is a critical area of research with significant implications for policy and planning. Traditionally, deterministic models like logistic growth (Newman et al., 2014) have been employed. More recently, increasingly sophisticated methods are being used, such as analyzing time series of population counts through autoregressive-integrated moving average (ARMA) processes (Harvey, 1990), which can capture both the inherent trends and the stochastic fluctuations present in population data (Cappuccino and Price, 1995). However, these approaches present certain challenges. Changes in observed population size reflect a combination of demographic variability and measurement error, making it difficult to accurately discern underlying demographic trends (Clark and Bjørnstad, 2004). In this context, state space models (SSMs) have become increasingly popular for fitting population dynamics. These models offer a flexible framework for modeling time series data, accommodating (in this case) the intrinsic variability in population processes and the observation error inherent in population counts. The state space approach separates the true population state, which is often unobservable, (see Table 1) from the observation process, allowing for a more nuanced understanding of population dynamics. In the context of human population dynamics, SSMs can account for complexities such as international migration, age structure, and spatial variation, and distinguish these factors from observation deviations such as census error (Newman et al., 2014).

In this study, we applied three distinct state space models, delineating our analysis into two distinct inferential paradigms: frequentist inference and Bayesian inference. The frequentist approach is exemplified by our first model, which employs Kalman filters. Kalman filter is a special case of linear SSMs with Gaussian errors and initial states, enabling an efficient recursive solution to the filtering problem (Harvey, 1990). Unlike Bayesian methods that integrate prior knowledge (of parameters and initial states) into analysis, the Kalman filter adheres to the principles of maximum likelihood estimation (MLE), focusing on estimating the hidden states of the model given a sequence of noisy observations. Although the two-stage predictive and update steps can be linked to the posterior update of Bayesian inference, the Kalman filter contrasts the posterior density approach embodied by subsequent (Bayesian) models.

Term	**Definition**
Conditional Likelihood	Conditional distribution of the states given the observations and the fixed (or estimated) parameter values
Filtering	$p(x_t\|y_{1:t})$ use observations to time t, $y_{1:t}$, to estimate x_t
Forecasting	$p(x_t\|y_{1:t-s})$ use observations s time steps before t to predict x_t
Joint Likelihood L_j	The probability of observing the entire sequence of data (observations) and the hidden states
Marginal Likelihood L_m	The likelihood of the parameters given the observed data, obtained by integrating out the hidden states
Observation	The observed measurements y_t at time t
Observation Equation	$g(y_t\|x_t)$ transition function between the hidden states and the observed measurements, probabilistic as $p(y_t\|x_t)$
Observation Error	The gap between the observed data and the true state, often modeled as noise in the observation equation.
Online Estimation	Continually estimate x_t based on information available at time t
Process Equation	$f(x_t\|x_{t-1})$ represents the probabilistic transition dynamics $p(x_t\|x_{t-1})$ of the x_t in the system, capturing the temporal dependencies from one state to the next
Process Error	The discrepancy between the predicted state and the actual hidden state, often modeled as noise in the process equation
Smoothing	$p(x_t\|y_{1:T})$ use all T the observations to estimate the x_t
State	x_t represents the unobserved state (hidden states) of the system at time t, which is the quantity of interest
State Space Model	A statistical model for capturing system dynamics through **process equations** that describe state evolution and **observation equations** that link hidden states to observed data

Table 1: Definitions of terms used in state-space models

Our second and third models adopt a Bayesian approach, employing Sequential Monte Carlo (SMC) methods – namely, particle filter – to approximate posterior distributions (typically) in non-linear models with non-Gaussian error structures (Doucet et al., 2001). Sequential Importance Sampling (SIS) (Doucet et al., 2001), the foundation of Sequential Monte Carlo methods, generates N weighted particle trajectories to approximate the state distribution. The bootstrap filter (Gordon et al., 1993) extends SIS by incorporating a resampling step, which discards low-weight particles and replicates high-weight ones, thus improving computational efficiency and emphasizing the most probable state trajectories. These models allowed for vague prior (e.g., Jeffrey's or uniform prior) distributions on parameters like birth and death rates, or informative ones if we have substantive prior knowledge about them. The second model utilized a bootstrap filter within the `nimbleSMC` packages in R to simulate population dynamics, capturing birth, death, and immigration processes. We further explored integrating Particle Markov Chain Monte Carlo (PMCMC) methods (Andrieu et al., 2010) alongside the bootstrap filter to simultaneously infer hidden states and parameters. MCMC, or Markov Chain Monte Carlo, refers to a set of algorithms that approximate (in this case) the posterior distribution by constructing a Markov chain that converges to the target (Lee, 2012). PMCMC leverages the likelihood estimates from the bootstrap filter and constructs an MCMC algorithm for effective parameter estimation (Andrieu et al., 2010). Our third model refined this approach, incorporating dynamic birth and death rates within a hard-coded bootstrap filter algorithm. This model accounted for the stochasticity of demographic events, offering a nuanced understanding of population changes over time. These methods are crucial for accurately estimating measurement error, uncovering hidden demographic processes, and ultimately, determining population size with precision.

For the research presented in this dissertation, we have used extensive data from the UK National Statistics, covering the period from 1839 to 2021. This data comprised detailed records of births, deaths, population estimates, and migration records[1] in England and Wales. By employing state space models in conjunction with Kalman filter and particle filter techniques, our objective was to construct a robust model that effectively captures the nuances of population dynamics in the UK. This model aimed to disentangle the complexities of demographic factors such as birth rates, death rates, and migration trends, to provide a clearer understanding of population fluctuations over the past 150 years.

[1]The migration data used in this study only includes the years 1964 to 2021.

2 Methods

2.1 Data Sources

2.1.1 Population Data

The data source compromised national and sub-national population estimates for the UK and its constituent countries, the number of births and deaths, and the number of international and internal migrants. The data were collected and published by the Office for National Statistics (ONS). More data were available from the National Records of Scotland (NRS) and the Northern Ireland Statistics and Research Agency (NISRA). They included all usual residents regardless of nationality, incorporating international migrants, with specific historical adjustments (for National Statistics, 2022).

2.1.2 Migration Data

The net migration was calculated as the difference between the inflow and outflow of international migrants. The migration data published by ONS, which only dated back to 1964 for England and Wales, were collected from various sources and only includes long-term migration due to the lack of a population registration system in the UK (for Demography, 2005).

2.2 Inference Methods for State-Space Models

In the context of State Space Models (SSMs), the unobserved states are denoted as x_t, which follow a Markov Process [2] initiated from x_0 and adhere to a process equation $p(x_t|x_{t-1})$ for $t \geq 1$ (Harvey, 1990). It can be written as:

$$x_t = f(x_{t-1}, \eta_t) \tag{1}$$

Where f is a function that describes the state transition, and η_t is the process noise.

Similarly, the observations y_t are conditionally independent given x_t with observation equation as $p(y_t|x_t)$ for $t \geq 1$, which can be written as:

$$y_t = g(x_t, \epsilon_t) \tag{2}$$

[2] A Markov process is a stochastic process where the future state depends solely on the present state, not the entire history of the process (Harvey, 1990).

Where g is a function that describes how the state is observed, and ϵ_t is the observation noise. Together, they describe how the system evolves and how observations relate to the underlying state.

To facilitate inference, we define $x_{1:t} \equiv \{x_1, \ldots, x_t\}$ and $y_{1:t} \equiv \{y_1, \ldots, y_t\}$ respectively as a set of states and observations at time t. As mentioned earlier, we can fit the SSMs with both frequentist and Bayesian inference frameworks.

2.2.1 Frequentist Approach

For the frequentist approach, such as the Kalman filter, we estimate the hidden states using the conditional distribution of the states given the observations and the fixed (or estimated) parameter values θ. The joint likelihood L_j for the states $x_{1:T}$ given a length T observations $y_{1:T}$ is defined as:

$$L_j(x_{1:T}|y_{1:T}) = \prod_{t=1}^{T} g(x_t, \epsilon_t; \theta) f(x_{t-1}, \eta_t; \theta), \tag{3}$$

For our online estimation setting, to estimate the state with a known parameter value $\hat{\theta}$ (or given maximum likelihood estimator), the probability distribution of states can be represented as a filtering problem:

$$\hat{p}(x_{1:t}|y_{1:t}, \hat{\theta}) = \frac{L_j(x_{1:t}|y_{1:t}, \hat{\theta})}{\int L_j(x_{1:t}|y_{1:t}, \hat{\theta})\, dx_{1:t}} \tag{4}$$

Where the marginal likelihood, i.e. the integral in the denominator ensures that the probabilities sum to one.

In situations where the parameters vector θ are unknown, a two-step approach is often employed. The first step involves the use of maximum likelihood estimation (MLE) to estimate the values of θ that maximize the likelihood of the observed data, i.e. the marginal likelihood $L_m(\theta|y_{1:t})$ for θ defined as (Åkesson et al., 2008; de Valpine, 2012):

$$\begin{aligned} L_m(\theta|y_{1:t}) &= \int L_j(\theta, x_{1:t}|y_{1:t})\, dx_{1:t} \\ &= p(y_{1:t}|\theta) \\ &= p(y_1|\theta) \prod_{t=2}^{T} p(y_t|y_{1:t-1}, \theta) \end{aligned} \tag{5}$$

This is achieved by integrating out the hidden states $x_{1:t}$ from the joint likelihood L_j defined in (3) (Auger-Méthé et al., 2021). Once the parameters

are estimated, the second step involves using the estimated $\hat{\theta}$ to perform state estimation using the conditional distribution defined in (4).

The condition dependence of the (marginal) likelihood [3] enables us to update our (one-step) prediction of observation y (de Valpine, 2012) recursively at each step t as:

$$p(y_t|y_{1:t-1},\theta) = \int p(y_t|x_t,\theta) \times p(x_t|y_{1:t-1},\theta)dx_t \tag{6}$$

In the context of Kalman filtering in 2.3.1, the marginal likelihood, as shown in (6) for all T states, can be evaluated to find the maximum likelihood estimate of the parameters. Maximizing the marginal likelihood is equivalent to minimizing the negative log-likelihood due to the monotonic nature of the logarithmic function (de Valpine, 2012). Taking the negative logarithm of the likelihood function transforms the product of probabilities into a summation, which is numerically more stable. The log-likelihood function $\ell(\theta)$ for a vector of unknown parameters θ is computed as follows[4]:

$$\begin{aligned}\ell(\theta) &= \log\left(p(y_1|\theta)\prod_{t=2}^{T} p(y_t|y_{1:t-1},\theta)\right)\\ &= \log(p(y_1|\theta)) + \log\sum_{t=2}^{T} p(y_t|y_{1:t-1},\theta),\end{aligned} \tag{7}$$

Optimization Techniques for Parameter Estimation

Newton's method (Nocedal and Wright, 2006) is commonly used to optimize the log-likelihood function $\ell(\theta)$ as derived above. It iteratively updates parameter estimates using the first and second derivatives of the objective function, converging towards a local maximum. Newton's method employs the following update step:

$$x_{k+1} = x_k - [H_f(x_k)]^{-1}\nabla f(x_k) \tag{8}$$

where:

- x_k is the current estimate of the solution,
- $\nabla f(x_k)$ is the gradient of the function f at x_k,

[3] Refer to the likelihood of the observed data given parameters $p(y_{1:t}|\theta)$, distinct from the likelihood or evidence of data $p(y_{1:t})$ in Bayesian inference

[4] Detailed derivation of the Kalman filter MLEs equation in Appendix A

- $H_f(x_k)$ is the Hessian matrix of the function f at x_k.

BFGS (Broyden-Fletcher-Goldfarb-Shanno)(Nocedal and Wright, 2006), a quasi-Newton method, builds upon the principles of Newton's method but avoids the computationally expensive task of directly calculating and inverting the Hessian matrix $[H_f(x_k)]$ at each iteration. Instead, it iteratively constructs an approximation of the inverse Hessian $[B_f(x_k)]^{-1}$ based on gradient information, guiding the search for optimal H_t (Nocedal and Wright, 2006). We optimize the parameter θ, with initial values set to the logarithm of their defaults to ensure positivity (Auger-Méthé et al., 2021). The inverse Hessian $[B_f(x_k)]^{-1}$ was initialized as the identity matrix I. The search direction d_k was calculated as:

$$d_k = -B_k^{-1} \nabla f(x_k) \tag{9}$$

The algorithm then finds a step size α_k that minimizes $f(x_k + \alpha_k d_k)$ as defined (7) above and updates the positions $x_{k+1} = x_k + \alpha_k d_k$. We calculate change in parameter values x_k and gradient y_k as:

$$x_k = x_{k+1} - x_k = \alpha_k d_k, \; y_k = \nabla f(x_{k+1}) - \nabla f(x_k) \tag{10}$$

Finally, we update the inverse Hessian B_{k+1}^{-1} using the BFGS formula:

$$B_{k+1} = B_k + \frac{y_k y_k^T}{y_k^T x_k} - \frac{B_k x_k x_k^T B_k}{x_k^T B_k x_k} \tag{11}$$

2.2.2 Bayesian Approach

In contrast to the frequentist approach, where parameters are treated as fixed but unknown values, the Bayesian framework considers both states and parameters as random variables (Lee, 2012). In Bayesian inference, this involves specifying priors for demographic parameters of interest, such as birth and death rates, as well as the initial state x_0 (with distribution $p(x_0)$) if it is not assumed known (Newman et al., 2014). Our goal is to recursively estimate the posterior distribution $p(\theta, x_{1:t}|y_{1:t})$. If we have all T observations available, this is also known as the smoothing distribution, defined as $p(\theta, x_t|y_{1:T})$, the conditional probability of the state given all observations up to time T. We may also be interested in the filtering distribution defined as $p(\theta, x_t|y_{1:t})$ conditional only observation up to time t. Applying the Bayes theorem $p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)}$ (Lee, 2012), the posterior distribution of states and parameters given observations at time t is given by:

$$
\begin{aligned}
p(\theta, x_{1:t}|y_{1:t}) &= \frac{L_j(x_{1:t}, \theta|y_{1:t})p(\theta)}{\int_x \int_\theta L_j(x_{1:t}, \theta|y_{1:t})p(\theta)\, d\theta\, dx_{1:t}} \\
&= \frac{p(y_{1:t}|x_{1:t}, \theta)p(x_{1:t}|\theta)p(\theta)}{\int_x \int_\theta p(y_{1:t}|x_{1:t}, \theta)p(x_{1:t}|\theta)p(\theta)\, d\theta\, dx_{1:t}}
\end{aligned} \tag{12}
$$

Where θ represents the parameters, $p(\theta)$ is the prior distribution of parameters, $p(x_{1:t}|\theta)$ is the prior distribution of the states, and $p(y_{1:t}|x_{1:t}, \theta)$ is the likelihood of the data (joint likelihood in 3) given the states & parameters, and the denominator integral is the likelihood serves as a normalization factor (also denoted as $p(y_{1:t})$ independent of θ and x), ensuring the posterior distribution sums up to 1 (Michaud et al., 2021).

To simplify the estimation formula derivation, we assume here that the parameters θ are known and fixed, hence we can update the posterior recursively using the following steps:

$$
\begin{aligned}
p(x_{1:t+1}|y_{1:t+1}) &= \frac{p(x_{1:t+1}, y_{1:t+1})}{p(y_{1:t+1})} \\
&= \frac{p(x_{t+1}, y_{t+1}|x_{1:t}, y_{1:t})p(x_{1:t}, y_{1:t})}{p(y_{t+1}|y_{1:t})p(y_{1:t})} \\
&= \frac{p(y_{t+1}|x_{t+1}, x_{1:t}, y_{1:t})p(x_{t+1}|x_{1:t}, y_{1:t})p(x_{1:t}, y_{1:t})}{p(y_{t+1}|y_{1:t})p(y_{1:t})} \\
&= \frac{p(y_{t+1}|x_{t+1})p(x_{t+1}|x_t)p(x_{1:t}|y_{1:t})}{p(y_{t+1}|y_{1:t})} \\
&= \frac{p(y_{t+1}|x_{t+1})p(x_{t+1}|x_t)}{p(y_{t+1}|y_{1:t})} p(x_{1:t}|y_{1:t}) \\
&\propto p(y_{t+1}|x_{t+1})p(x_{t+1}|x_t)p(x_{1:t}|y_{1:t})
\end{aligned} \tag{13}
$$

We assume conditional independence such that y_{t+1} depends solely on the current state x_{t+1} and the Markov Chain property on x (Harvey, 1990), where the transition to the next state x_{t+1} relies only on the current state x_t. Hence, the posterior distribution $p(x_{1:t+1}|y_{1:t+1})$ can be expressed as the observation probability $p(y_{t+1}|x_{t+1})$ and the state transition probability $p(x_{t+1}|x_t)$ multiplied by the posterior distribution $p(x_{1:t}|y_{1:t})$ at time t.

2.2.3 Inference Methods

Evaluating the likelihood in (4) and (12) is challenging as it often involves a high-dimensional integration known proportionally to time t (Doucet et al.,

2001). For example, for an SSM with a 1-dimensional state at each time step. If we want to evaluate the posterior distribution at time $t = 100$, we would need to compute a 100-dimensional integral. This is because the denominator in (12), according to Bayes' theorem, involves an integral over all possible values of $x_{0:t}$, which is a t-dimensional integral. As t increases, the dimensionality of this integral also increases, making it more difficult to compute.

To address the 'intractable likelihood' problem, a variety of computational approaches have been developed. In the frequentist domain, numerical integration methods, like the Laplace approximation aim to represent the likelihood with a normal density function (Auger-Méthé et al., 2021). Specifically designed for linear Gaussian state-space models, the Kalman filter provides an analytical solution to compute the posterior distribution of states efficiently (Model 2.3.1) (Harvey, 1990). This filter operates under the premise that both the transition and observation models are linear, with Gaussian errors, enabling computationally efficient recursive calculations.

Under Bayesian settings, common approaches include Markov Chain Monte Carlo (MCMC) and Sequential Monte Carlo (SMC) (Auger-Méthé et al., 2021). These methods leverage Monte Carlo simulations to approximate the marginal likelihood, offering versatility in handling non-linear and non-Gaussian models. Particularly, the SMC employs algorithms like the Bootstrap filter (Models 2.3.2 and 2.3.3), which approximates the posterior distribution of states and parameters by sampling from a set of weighted particles. Each particle's weight represents its likelihood given the observed data (Doucet et al., 2001).

Monte Carlo Methods

For Bayesian parameter inference, similar to the frequentist factorization of likelihood in (5), the likelihood function $p(y_{1:t}|\theta)$, i.e. the likelihood function of θ given observations $y_{1:t}$ can be derived into a T dimensional integral as [5]:

$$p(y_{1:t}|\theta) = p(y_1|\theta) \prod_{t=2}^{T} p(y_t|y_{1:t-1}, \theta)$$

The condition dependence of y help us convert the intractable likelihood problem into probabilistic expectation (Lee, 2012), similar to (6) as:

$$p(y_t|y_{1:t-1}, \theta) = \int p(y_t|x_t, \theta) p(x_t|y_{1:t-1}, \theta) dx_t = \mathbb{E}(p(y_t|x_t, \theta)).$$

[5] Here we assume the initial state x_0 is fixed not a parameter to be estimated.

The Monte Carlo method is a set of approaches that samples directly from the posterior distribution and uses the sample average to approximate the expectation (Lee, 2012). A perfect Monte Carlo sampling with N independent and identically distributed (i.i.d) simulations $x_t^{(i)}$ from $p(x_t|y_{1:t-1}, \theta)$ can be used to estimate the expectation as:

$$\frac{1}{N}\sum_{i=1}^{N} p(y_t|x_t^i, \theta) \to \mathbb{E}(p(y_t|x_t, \theta)), \quad N \to \infty.$$

The variance of the estimated distribution can be calculated as:

$$\sigma_{MC}^2 = \frac{1}{N}\sum_{i=1}^{N}(p(y_t|x_t^i, \theta) - p(y_t|x_t, \theta))^2,$$

where $p(y_t|x_t^i, \theta)$ is the estimate from the i-th simulation, and $p(y_t|x_t, \theta)$ is the true value. To reduce the prediction error by a factor of 10, the number of simulations N must be increased by a factor of 100. Hence, the naive Monte Carlo method is not efficient for high-dimensional problems (for our state-space model) (Robert and Casella, 1999), as the number of simulations required to achieve a given level of accuracy increases exponentially with the dimensionality (Doucet et al., 2001). Therefore, we need to improve the sampling $x_t^{(i)}$ into high density region as evaluated by $p(y_t|x_t^{(i)}, \theta)$ to reduce the variance of the estimate.

Sequential Monte Carlo Methods

Sequential Monte Carlo (SMC) methods provide an alternative to traditional Monte Carlo techniques through the use of Sequential Importance Sampling (SIS) (Doucet et al., 2001). SMC methods aim to represent the state's posterior distribution at each time step by utilizing a set of weighted samples, also known as particles. These particles progress via the process model and are re-evaluated based on new observations using importance weight (Robert and Casella, 1999). We first need to specify a sampling distribution $q(x_t^{(i)}|x_{1:t-1}^{(i)}, y_{1:t})$, which for the bootstrap filter introduced below simplified to the process model $f(x_t|x_{t-1})$, to generate the particles. Using the recursive relation in (13), the importance weight $w_t^{(i)}$ for each particle can be calculated as (Doucet et al., 2001):

$$w_t^{(i)} \propto \frac{g(y_t|x_t^{(i)})f(x_t^{(i)}|x_{t-1}^{(i)})}{q(x_t^{(i)}|x_{1:t-1}^{(i)}, y_{1:t})} w_{t-1}^{(i)} = g(y_t|x_t^{(i)})w_{t-1}^{(i)} \tag{14}$$

Given the sampled particles $\{x_t^{(i)}\}_{i=1}^M$ and their corresponding weights $\{w_t^{(i)}\}_{i=1}^M$, we can then approximate the likelihood at each step $p(y_t|y_{1:t-1};\theta)$ and likelihood $p(y_{1:t}|\theta)$ (Michaud et al., 2021) as:

$$\tilde{p}(y_t|y_{1:t-1};\theta) = \frac{1}{N}\sum_{i=1}^{N} w_t^i, \quad \tilde{p}(y_{1:T}|\theta) = \prod_{t=1}^{T}\left(\frac{1}{N}\sum_{i=1}^{N} w_t^i\right). \tag{15}$$

Despite its effectiveness, the SIS approach is prone to weight degeneracy, a phenomenon where the particles' importance weights become unbalanced over time(i.e. the weight $w_{t-1}^{(i)}$ is close to zero in (14)), leading to a decrease in sampling efficiency (Robert and Casella, 1999). The bootstrap filter (Gordon et al., 1993) addresses this by incorporating a resampling step. This step refocuses the particle set towards regions with higher likelihood, maintaining diversity among the sample weights and offsetting the intrinsic weight degeneracy encountered in the SIS method (de Valpine et al., 2017; Doucet et al., 2001).

At each time step t, The bootstrap filter SMC algorithm employing M particles, proceed as follows:

1. **Sample** new particles from the state transition distribution: $\tilde{\mathbf{x}}_t^{(i)} \sim f(\mathbf{x}_t|\mathbf{x}_{t-1}^{(i)})$, $i = 1, ..., M$ where the proposal distribution $f(*)$ is the process (1) defined above.

2. **Weighting** by calculating importance weights for each particle: $w_t^{(i)} \propto g(\mathbf{y}_t|\tilde{\mathbf{x}}_t^{(i)})$, $i = 1, ..., M$, which here is the observation equation (2). The weights reflect how well each propagated particle explains the current observation y_t. Normalized the weights to $\pi_t^{(i)}$.

3. **Resample if ESS Low:** The effective sample size (ESS) is calculated as the inverse of the sum of the squared normalized weights (Gordon et al., 1993):

$$ESS = \frac{1}{\sum_{i=1}^{M} w_{t,i}^2} \tag{16}$$

   If it is below a certain threshold (i.e 80% suggested by Michaud et al. (2021)), indicating potential particle degeneracy, particles are resampled with replacement proportional to their weights $\pi_t^{(i)}$: $\mathbf{x}_t^{(i)} \sim \tilde{\mathbf{x}}_t^{(i)}$, $i = 1, ..., M$. After resampling, weights are reset to uniform.

4. **Approximates** the posterior $p(\mathbf{x}_t|\mathbf{y}_{1:t-1}, \theta)$ with the new set of particles $\{\mathbf{x}_t^{(i)}\}_{i=1}^M$

5. **Estimate likelihood** of the data at time t by the sum of the weights: $\tilde{p}(\mathbf{y}_t|\mathbf{y}_{1:t-1}) = \frac{1}{M}\sum_{i=1}^M w_t^{(i)}$. As the number of particles $M \to \infty$, this weighted approximation converges to the true likelihood function at time t (Doucet et al., 2001)

The likelihood $p(\mathbf{y}_{1:T})$ can be obtained by $\tilde{p}(\mathbf{y}_{1:T}) = \prod_{t=1}^T \tilde{p}(\mathbf{y}_t|\mathbf{y}_{1:t-1})$

Particle MCMC methods

As outlined in Michaud et al. (2021), the bootstrap particle filter provides an unbiased estimate of the marginal likelihood $\tilde{p}(y_{1:T}|\theta)$, under the assumption that the model parameters, θ, remain fixed. This estimate is essential for inference using Markov Chain Monte Carlo (MCMC) methods, such as the Metropolis-Hastings algorithm. MCMC methods[6] construct a Markov chain that gradually converges to the target distribution, which in this case is the joint posterior distribution $p(\theta, x_{1:T}|y_{1:T})$. To simultaneously infer hidden states and parameters, Particle Markov Chain Monte Carlo (PMCMC), particularly the Particle Marginal Metropolis-Hastings (PMMH) algorithm, utilizes the bootstrap filter to draw samples of the states while employing MCMC techniques to draw samples of the parameters (Dahlin and Schön, 2019).

In the context of the PMMH algorithm, the likelihood can be approximated using the value obtained from particle filters in (15) as follows:

$$p(y_{1:T}|\theta) \approx \prod_{t=1}^{T}\left(\frac{1}{N}\sum_{i=1}^{N} w_t^i\right),$$

In the Bayesian MCMC framework, this likelihood is used within the Metropolis-Hastings algorithm to obtain samples from the posterior (target) distribution $p(\theta, x_{1:T}|y_{1:T})$, updating our beliefs about θ in light of the observed data $y_{1:T}$. Given the bootstrap filter defined above, the PMCMC at each time step t with N iterations proceeds as follows (Michaud et al., 2021):

1. **Proposal:** Propose new parameters θ^* from a proposal distribution $q(\theta^*|\theta^{(i-1)})$, looping through the number of iteration $i = 1, 2, \ldots, N$.

2. **Particle Filter:** Run a bootstrap filter with the proposed parameters θ^* to obtain the marginal likelihood estimate $\tilde{p}(y_{1:T}|\theta^*)$. This marginal

[6] More detail on Markov Chain Monte Carlo on Peter M. Lee, *Bayesian Statistics: An Introduction, Fourth Edition* Chapter 9

likelihood is used in the next step to calculate the acceptance probability for θ^*. At the final step T of the bootstrap filter, generate particles from the smoothing distribution $x^*_{1:T} \sim p(x_{1:T}|y_{1:T}, \theta^*)$.

3. **Acceptance Probability:** Calculate the acceptance probability α using the Metropolis-Hastings rule:

$$\alpha(\theta^*, \theta^{(i-1)}) = \min\left(1, \frac{\tilde{p}(y_{1:T}|\theta^*)p(\theta^*)q(\theta^{(i-1)}|\theta^*)}{\tilde{p}(y_{1:T}|\theta^{(i-1)})p(\theta^{(i-1)})q(\theta^*|\theta^{(i-1)})}\right)$$

   Note: The likelihood $\tilde{p}(y_{1:T}|\theta)$ is obtained in the particle filter step.

4. **Update:** With probability α, accept the proposal: set $\theta^{(i)} = \theta^*$ and $x^{(i)}_{1:T} = x^*_{1:T}$ and $\tilde{p}^{(i)} = \tilde{p}(y_{1:T}|\theta^*)$. Otherwise, retain the previous values: $\theta^{(i)} = \theta^{(i-1)}$, $x^{(i)}_{1:T} = x^{(i-1)}_{1:T}$, and $\tilde{p}^{(i)} = \tilde{p}^{(i-1)}$

2.3 Model Specification

2.3.1 Linear Gaussian with Kalman Filter

In this model, we assumed that the population dynamics could be described by a linear Gaussian SSM, suitable to be fit with a Kalman filter using R packages `Fast Kalman filter`. The state-space model was defined by the following process and observation equations:

$$x_t = c_t + T_t \cdot x_{t-1} + \eta_t, \tag{17}$$

where $\eta_t \sim \mathcal{N}(0, H_t)$, c_t being a vector of annual net migration, η_t was assumed to be mean zero and variance H_t [7].

Observations y_t were modeled as random walks around the state variable with added noise to account for the uncertainty and potential inaccuracies in population data. The observation noise ϵ_t was assumed normal with mean zero and variance G_t:

$$y_t = x_t + \epsilon_t, \tag{18}$$

where $\epsilon_t \sim \mathcal{N}(0, G_t)$ [8].

- x_t is the state of the system at time t, with the starting value x_0 being the initial population count (in 1839).

[7] Deriving from the general form $x_t = c_t + T_t \cdot x_{t-1} + Q_t \cdot \eta_t$ as Q_t equaled the identity

[8] Deriving from $y_t = d_t + Z_t \cdot x_t + K_t \cdot \eta_t$ with d_t being zero, Z_t set to a vector of one and η_t was assumed to be mean zero and variance G_t, and K_t equaled the identity

- c_t is the number of net immigration count each year [9]
- T_t is the state transition (growth rate) parameter, describing how the state evolves from time $t-1$ to t, utilizing net effects of $\phi + \beta$ (mean death and birth rates) derived from the data.
- y_t is the observed population at time t.
- H_t and G_t are the process and observation noise variances, respectively. Both were assumed to be independent and identically distributed (i.i.d.) normal variables with a mean of zero (Harvey, 1990). The variance of H_t was estimated from the year-over-year population differences to capture stochasticity not explained by process (17). The variance of G_t was set based on 3% of the mean population count (arithmetic mean of all T population) based on the 2021 ONS census report suggestion from Powell and Racinskij (2022). The implications of these assumptions will be discussed in 3.1.

The Kalman filter outputs the analytical solutions recursively in two steps: prediction and update (Harvey, 1990). Given the initialization values x_0 and the variance of the estimation error P_0,

For each step at time t, where $t = 1, 2, \ldots, T$:

1. Given state mean x_{t-1} and variance P_{t-1}
2. **Prediction**
   (a) we predict the state as $\hat{x_t} = T_t x_{t-1} + c_t$
   (b) with the variance as $\hat{P}_t = T_t^2 P_{t-1} + H_t$ where $\hat{P}_t$ is the predicted variance at time t.
3. **Update**
   (a) We then calculate the Innovation (Prediction Error) as $e_t = y_t - \hat{x_t} = y_t - T_t \hat{x}_{t-1} - c_t$.
   (b) The Kalman Gain $K_t = \hat{P}_t + G_t$
   (c) We then update our estimation of x_t with a normal distribution of mean $x_t = \hat{x_t} + \hat{P}_t {K_t}^{-1} e_t$ and variance: $P_t = (I - \hat{P}_t {K_t}^{-1}) \hat{P}_t$ (Harvey, 1990).

[9] Available data from 1964 to 2023 and set to zero otherwise.

We iteratively applied the Kalman filter to update our population estimates. The predictive distribution at step (c) is the (one step ahead) filtering distribution in (6) (Durbin and Koopman, 2012). The filtering distribution $p(x_t|y_{1:t})$ provides a refined view of the population size, by adjusting the model's predictions with observed data. We also computed standard errors from the post-update variance P_t to construct 95% confidence intervals around the estimates. We employed the Fast Kalman Filter (`FKF`) package in R for model fitting. This package delivers a computationally efficient C-based implementation of the Kalman filter and smoother, supporting both univariate and multivariate linear models, making it well-suited for our Gaussian state-space model (Åkesson et al., 2008). Utilizing the `FKF` package's output log-likelihood value (given a parameter set), we could then obtain the maximum-likelihood parameter estimates through R's `optim` functions, employing non-linear optimization methods such as Newton's method or a Quasi-Newton method (i.e. BFGS) (Harvey, 1990).

Optimization for Estimating Standard Deviation

In the above example, we assumed that the parameters (e.g. the growth rate T_t) are known. However, when parameter θ - which may include H_t, G_t, among others - are unknown, we could maximize the marginal likelihood estimation outline in (6) to obtained the estimated parameters $\hat{\theta}$. While the Kalman filter assumes a constant growth rate, real-world population dynamics rarely exhibit the constant birth and death rates and immigration flows implied by the process model. Process variance, H_t, addresses this uncertainty, allowing the Kalman filter to prioritize recent measurements. A high H_t (relative to population) enables the model to closely track observations (Harvey, 1990), deviating from the linear trajectory suggested by the process equation. Similarly, observation variance, G_t, accounts for errors in population data collection, such as non-response and double-counting in censuses (Powell and Racinskij, 2022). For the parameter optimization, we fixed the observation error G_t to be 3% of the mean population count. To further refine the process error H_t, we constructed a function that takes G_t as its sole argument. This function is based on the negative log-likelihood function in (7). This optimization aimed to minimize the negative log-likelihood of the Kalman filter's output:[10]

$$\underset{H_t}{\text{minimize}} \; -\ell(H_t, \bar{G}_t, y_{1:T}) \tag{19}$$

We employed the BFGS algorithm mentioned in 2.2.1 to solve this optimiza-

[10] $\bar{G}_t$ represents the default value of G_t

tion problem. The final estimate for H_t was obtained by exponentiating the optimized results.

Model Evaluation

We could evaluate our model by comparing the predicted observations with the actual value through the likelihood $L(\theta)$. However, the absence of true population states x_t (Auger-Méthé et al., 2021) precluded direct comparison between predicted and actual states. This limited the applicability of traditional fit metrics like Mean Squared Error (MSE). When evaluating different process error variance H_t within Kalman filter models, traditional model selection criteria like the Akaike Information Criterion (AIC) is not informative (Auger-Méthé et al., 2021; Harvey, 1990) ($AIC = -2\ell(\theta) + 2n$). The AIC balances model fit (through the log-likelihood $\ell(\theta)$) with a penalty for complexity (based on the number of parameters n). In this context, where the number of parameters remains constant across models, the choice of H_t primarily influences how well the model captures the inherent variability of the data (Auger-Méthé et al., 2021).

Rather than directly assessing model fit, we could analyze the standardized residuals of the Kalman filter $e_t = y_t - \hat{x}_t$ for diagnostic purposes (Harvey, 1990). Under the assumption of Gaussian errors, a well-specified model should yield standardized residuals that approximate a standard normal distribution (Harvey, 1990).

2.3.2 Bootstrap Filter with Poisson-Lognormal Model

To address potential non-linearity and non-Gaussian in population dynamics (i.e. to improve the linear process model in (17)), we employed a Bayesian state-space model fitted with Sequential Monte Carlo (SMC) methods, specifically the bootstrap filter (Doucet et al., 2001). This particle filter allowed us to estimate the posterior distribution of a population with a non-linear process equation (23), where an analytical solution for the likelihood is unsolvable. Population dynamics are shaped by the interplay of surviving population from the previous year, new births in the current year, and immigration count in the current year (for National Statistics, 2022). The Birth, Survival, Immigration (BSI) model, derived from the BRS model (survival, growth, and birth) in ecology (Newman et al., 2014), is a state-space model that represents the dynamics of a population with two key stages: a survival process followed by a birth process. In this model, we assigned a constant death rate ϕ and birth rate β throughout, establishing an 'uninformative' prior (Lee, 2012) with a uniform distribution between 0 and 0.1 for both

parameters. This upper bound is five times greater than the most extreme value observed in the birth and death data, regularizing variation in a reasonable range (Doucet et al., 2001). The model was formulated as follows with the process equation:

$$x_t \sim Poisson(\lambda = x_{t-1} - d_t + b_t + I_t) \tag{20}$$

where x_t is the estimated population at time t, modeled as a Poisson process to account for the demographic stochasticity 'overdispersion' [11] and the variation in immigration data (Newman et al., 2014). However, the Poisson process might lead to 'overparameterization' [12] of the model (Lee, 2012), as we will see in the result section 3.2, which encouraged building on a more constrained model in 2.3.3. b_t and d_t were the estimated number of births and deaths each year, derived from the process below. The death process d_t was modeled as a binomial process (Newman et al., 2014) with death probability ϕ applied on the population size x_{t-1} from the previous year:

$$d_t \sim Binomial(x_{t-1}, \phi) \quad \text{(Binomial deaths)} \tag{21}$$

After the death process, the surviving population was updated as $x_t^1 = x_{t-1} - d_t$. A Poisson process is useful when modeling events that occur randomly and independently over time, such as births (Newman et al., 2014). The birth process b_t was modeled as a Poisson process with rate $\beta \times x_t^1$, based on the updated surviving population.

$$b_t \sim Poisson(\beta \times x_t^1) \quad \text{(Poisson births)} \tag{22}$$

The net immigration, I_t, was incorporated as an exogenous input, directly augmenting the population size. In Summary, $f(x_t|x_{t-1})$, the state transition equation is summarized as:

$$x_t = Poisson(x_{t-1} - Binomial(x_{t-1}, \phi) + Poisson(x_t^1 \times \beta) + I_t) \tag{23}$$

The observed population y_t was formulated as a Log-normal distribution to reflect the density-dependent variance (Hostetler and Chandler, 2015).

$$y_t \sim Lognormal(\log(x_t), \log(G_t)) \tag{24}$$

where G_t is the observation error variance.

[11] Overdispersion means the data variance are larger than the model assumption

[12] Overparameterization indicates a model with more parameters than the data support

To maintain consistency between the model's initial state x_0 representation and its process dynamics, the initial population x_0 was modeled with a Poisson distribution (rate parameter $\lambda = x_0$), while its corresponding observation y_0 was set as a log-normal distribution with mean equal to the log of x_1 and variance as the observation error (3% of the mean population) (Powell and Racinskij, 2022). The model was then fitted with a bootstrap filter to estimate state sequences, and equally weighted posterior samples of state variables were used to generate time series estimates of the population states (Fig.6). As we shall discuss in the result section 3.2, the fitting revealed model misspecification. This finding, along with the constant parameter assumption within the bootstrap filter (Michaud et al., 2021), motivated the refined model (Section 2.3.3), which incorporated stochastic rate particles and error terms to improve model adaptability. Moreover, we will see that the Poisson-Lognormal model suffer from 'particle degeneracy' issues (Doucet et al., 2001) where only a few particles have significant weights, leading to a poor approximation of the posterior distribution.

As mentioned earlier, we could incorporate the PMMH algorithm which utilizes the Metropolis-Hastings (MH) algorithm for parameter generation and a bootstrap filter for latent state estimation. The method uses the approximate likelihood from the bootstrap filter to sample the vector of model parameters ϕ and β. We initialized with 5000 particles, with a multivariate random walk proposal within the particle filter block sampler.

We used `NimbleSMC`, an R package designed to analyze SSMs using SMC methods, to implement the Bootstrap filter. It works within the broader NIMBLE framework, allowing users to define models using BUGS-like language and compiled through C++ for efficient computation (Michaud et al., 2021). We implemented the PMMH approach directly within `NimbleSMC` by configuring a PMCMC block sampler for joint parameter vector estimation. See Section B.2.1 for the NIMBLE BUGS code setup.

2.3.3 Bootstrap Filter with Dynamic Birth and Death Rates

To address the limitations of deterministic birth & death rates in the basic bootstrap filter (section 2.3.2), we enhanced the algorithm by introducing a time-varying birth and death rate process model. Furthermore, we rectified the model misspecification issue identified in the process equation (20).

The new state space model process equation at time t is defined as:

$$x_t = x_{t-1} - Binomial(x_{t-1}, \phi_{t-1}) + Poisson(x_{t-1} \cdot \beta_{t-1}) + I_t$$

where the observation model is:

$$y_t \sim Lognormal(\log(x_t), \log(G_{t,t})),$$

and the variance of the observation error at time t, $G_{t,t}$, was dynamically recalculated at each time step as 3% of the population of the previous year, y_{t-1}, contrast to a fixed value G_t in the Kalman filter example (section 2.3.1) and NimbleSMC example (section 2.3.2) (Powell and Racinskij, 2022).

Birth and death rate particles i were initialized from a uniform [0, 0.1] distribution, reflecting our prior belief about the plausible range of these rates. For each subsequent time step $t > 0$, we updated the particles to incorporate new population change information. The rate particles i for year t were modeled using an autoregressive (AR(1)) model (Harvey, 1990), which captured the temporal dependence of rates across consecutive years:

$$\beta_{t,i} \sim \max(\mathcal{N}(\bar{\beta}_{t-1}, 0.01), 0),$$
$$\phi_{t,i} \sim \max(\mathcal{N}(\bar{\phi}_{t-1}, 0.01), 0),$$

where $\bar{\beta}_{t-1}$ and $\bar{\phi}_{t-1}$ are the posterior mean estimated birth and death rates from the previous year, respectively. The max function ensured that the rates remained non-negative after the perturbation.

Given the previous year's population, x_{t-1}, and a death rate particle, $\phi_{t,i}$, deaths for year t were simulated using a binomial distribution. The surviving population then interacts with a birth rate particle, $\beta_{t,i}$, to simulate births using a Poisson distribution. The estimated population for year t, $x_{t,i}$, is the sum of the surviving population, births, and immigration count I_t.

The particle weights were dynamically updated by comparing the observed population, y_t, against $\mathrm{p}(y_t|x_{t,i})$ denotes the density of y_t under a log-normal distribution with mean $\log(x_{t,i})$ and standard deviation $\log(G_{t,i})$. The updated weight for particle i at time t is given by:

$$w_{t,i} = w_{t-1,i} \cdot \mathrm{p}(y_t|x_{t,i}),$$

The effective sample size (ESS) was calculated according to Equation (16). If it is below a threshold of 80% (Michaud et al., 2021), particles were resampled.

We visualized results using the posterior mean as a point estimator and the 95% posterior density interval to represent estimation uncertainty. The dynamic particle setting allowed us to visualize time-varying birth and death rates (Fig.13). These can be compared against the exact rates (Fig.1) for model evaluation. The algorithm returned the posterior distributions of both the final states and parameters.

	Model 2: Sequential Monte Carlo with Bootstrap Filter	**Model 3: Particle Filter with Dynamic Birth and Death Rates**
Initialization	Fixed values for ϕ and β sampled from a uniform priors between 0 and 0.1; identical distributions for the initial state x_0 (observation y_0) and subsequent states (observation)	Initialized ϕ_t and β_t with uniform prior; subsequently updated through an AR(1) process incorporating normal error perturbations at each time step
Process Model	Poisson process equation with 2 sub-process - Binomial deaths and Poisson births, incorporating immigration as an exogenous input	Employed the same death, birth, and immigration sub-processes as Model 2, with stochastic rates derived from the dynamic particles.
Observation Model	Modelled the observation equations as Log-normal distributions with a fixed observation error variance G_t over time	Used the same Log-normal observation model but dynamically recalculated the variance of observation error $G_{t,t}$ at each time step
Resampling Strategy	Resampled at each time point proportional to particles' weights from the previous time step through `rankSample` function from `NimbleSMC` packages	Resampled if ESS falls below 80% threshold, with post-resampling weights reset to uniform to ensure equal confidence across particles

Table 2: Comparison between SMC Model 2 and Model 3

3 Results

3.1 Kalman Filter Results

Starting with initial parameters $T_t = 1.01, P_0 = 10^3, H_t = 1.61 \times 10^{11}$ and $G_t = 1.13 \times 10^6$, we achieved a close fit to the observations (Fig.3) with a log-likelihood of -2623.38. Optimization resulted in a slightly higher H_t estimate of 1.77×10^{11} and a new model likelihood of -2622.96. The distribution of (one-step ahead predictive) standardized residuals appeared to be symmetric around mean zero and within two standard deviations, suggesting a well-specified model (Fig.5). Upon optimization, the fit remained visually almost identical to the initial result (Fig.3). Consistent with findings in Harvey (1990), the initial variance P_0 has a marginal effect on the final likelihood value in a convergent filter. This is evidenced by the identical log-likelihood returned by setting P_0 to 10^3 or 10^6.

As the process and observation error variances approach zero, the Kalman filter remains valid (Harvey, 1990) but increasingly approximates a deterministic model with a near-fixed mean (Fig.4). When the process error was squared ($H_t = 4 \times 10^5$), the confidence interval significantly narrowed, resulting in more linear filtering and reduced responsiveness to new observations.

As mentioned in 2.3.1, the assumption of constant process error variance is questionable given the tripling of the population over 150 years. The variability in observation noise is also likely to change over time due to factors such as advancements in census methods, suggesting the need for a time-dependent H_t (and G_t) model in 2.3.3.

The fitting results (Fig.3) indicated a potential overestimation of the population from the 21st century onward. This overestimation could stem from the combined effects of added immigration and constant birth and death rate assumptions, despite a likely decline in birth rates in recent times (for National Statistics, 2022).

See the code in appendix: `kalman_filter/KalmanFilter-Model1.R`.

Model	**Process Error**	**Log-likelihood**
Initial Model	1.61×10^{11}	-2623.38
Optimized Model	1.77×10^{11}	-2622
Squared Model	4×10^5	-18821403

Table 3: Model Comparison with Process Error and Log-likelihood

3.2 Poisson-Lognormal Bootstrap Filter Results

The initial birth & death rate were set to their mean rates across the entire temporal dataset and the initial state is the first year (1838) population count. The model was then fitted in `NimbleSMC` Bootstrap Filter with 10^4 particles. The posterior samples' histogram (Fig.8) for the final states (posterior density of estimated population in 2021) following a normal density confirmed the convergence. However, the convergent estimate significantly diverged from the actual population observation of 59,660,524. The state estimations' time series (Fig.6), exhibited a narrow 95% posterior density interval, suggesting high confidence in the population estimates. Two factors might contribute. The increasing number of unique particles over time (Fig.7) suggested that the Poisson process specification (Section 2.3.2) might introduce excessive randomness, leading to model misspecification (Harvey, 1990). The elevated process noise might also contribute, causing the model to converge to a stable but inaccurate set of states.

The estimate of log-likelihood for the model is -3828.82, higher than that of the Kalman filter. The PMCMC algorithm was run for 5000 iterations. The trace plots for the parameters ϕ (death rate) and β (birth rate) indicated that the Markov chains have reached a stationary distribution, suggesting adequate mixing and convergence (Fig. 9). The histograms for the sampled parameters showed the estimated posterior distributions, which are notably uniform within the specified priors, suggesting the model misspecification or identifiability issues (Dahlin and Schön, 2019) (Fig.10).

See the code in the appendix: `particle_filter/SMC-PoisLogN.R`.

3.3 Dynamic Rate Bootstrap Filter Results

We set the birth & death rate as an AR(1) process for each time t and run the model with 10^4 particles. The resampling threshold was set to 80%, the default value suggested by Michaud et al. (2021). The filtering result (Fig.11), revealed that the estimated population closely tracks the actual observed data, especially when net immigration has been accounted for since the 1960s. The one-step-ahead predictive residuals (Fig.12) further supported this finding. The mean of these residuals across years closely aligned with the ideal model gap (Fig.2), which represents the population change not explained by births and deaths alone. The 95% posterior density became wider over time, caused by the proportional increase in the observation error variance, $G_{t,t}$, as the population grows. The consistent observation of both a high number of unique particles and a maximum effective sample size (ESS)

across each year suggested the absence of resampling events, attributable to the high diversity of particles and a uniform distribution of their weights across iterations. This phenomenon implied an efficient exploration of the state space, ensuring that each particle contributes almost equally to the posterior distribution's estimation (Doucet et al., 2001). However, the notably low log-likelihood value of -5361.017 indicated the introduction of significant perturbations at every iteration of the particle filter process. Such a scenario raised concerns about the potential loss of particle diversity, hindering the ability to accurately capture the posterior distribution (Gordon et al., 1993). This challenge in accurately estimating the underlying state dynamics suggested the potential benefit of exploring alternative SMC methods, such as iterated filtering (Ionides et al., 2015), which introduce more controlled perturbations.

Analysis of parameter inference (Fig.13) revealed consistent overestimation of both ϕ and β compared to their true values (Fig.1) over time. This discrepancy is further evident (Fig.14) at the posterior distributions for the final states $\phi_{t=183}$ and $\beta_{t=183}$, which exhibited a bias towards higher values than the true values. The discrepancy indicated potential missing covariates in the model, such as the effect of age structure on birth and death rates to be considered in future work.

See the code in the appendix: `particle_filter/BF-Coded.R`.

4 Discussion

4.1 Limitations and Future Work

In our future work, we aim to extend our current model to separately analyze the population dynamics in England, Scotland, Wales, and Northern Ireland. We will leverage the datasets provided by the National Records of Scotland (NRS) and the Northern Ireland Statistics and Research Agency (NISRA). These datasets include time series data on births, deaths, migration, and population estimates dating back to the mid-19th century. The data at the Least Unit of Administration (LUA) level enable the formulation of more complex state-space models by incorporating detailed age and sex structure.

In addition to the Sequential Monte Carlo and Kalman filter methods used in this study, we also plan to explore the use of Markov Chain Monte Carlo (MCMC) methods for obtaining smoothed state estimates. This approach will provide us with another perspective and potentially enhance our ability to capture the unique demographic trends and influences in each of these regions.

4.2 Conclusion

State-space models offer a versatile framework for analyzing time series data, effectively handling both observed and latent state variables. Frequentist approaches, such as the Kalman filter, provide computationally efficient parameter and state estimation using maximum likelihood estimation (MLE). Conversely, Bayesian approaches enable the incorporation of prior knowledge, quantifying uncertainty in parameter estimates. Particle Markov chain Monte Carlo (PMCMC) and dynamic sequential importance weighting facilitate accurate Bayesian inference of parameters and states. We demonstrated the application of these approaches through three illustrative models, highlighting the flexibility of Sequential Monte Carlo (SMC) methods. By employing a non-linear and non-Gaussian process model with a bootstrap filter for the Birth Survival and Immigration process, we surpassed the limitations of the traditional normal dynamic linear model (NDLM). This approach captured complex dynamics and provided plausible parameter estimates. Future work could further explore the use of advanced SMC methods, such as the Particle Metropolis-Hastings and iterated filtering, for improved performance in more complex state-space models.

A Proof of MLE for Kalman Filter

[13] Recall from (6), for the time series, a conditional probability density function is used to write the joint density function as

$$L_m(\theta|y_{1:t}) = p(y_{1:T}|\theta) = \prod_{t=1}^{T} p(y_t|y_{1:t-1}) \tag{25}$$

In section 2.3.1 we see that, after the prediction step, we have the conditional distribution of y_t being normal with a mean $\hat{x}_t + e_t$ and variance P_t

$$e_t = y_t - \hat{x}_t - \epsilon_t$$

where given G_t (observation error variance), the innovation variance P_t is:

$$P_t = \hat{P}_t + G_t \tag{26}$$

Assuming Gaussian noise, the PDF of the innovation at time t is:

$$p(e_t|y_1, ..., y_{t-1}, \theta) = \frac{1}{\sqrt{(2\pi)^t(P_t)}} \cdot \exp\left(-\frac{1}{2}P_t^{-1}e_t^2\right)$$

The joint likelihood function for all observations is the product of the individual:

$$p(y_{1:T}|\theta) = \prod_{t=1}^{T} p(e_t|y_1, ..., y_{t-1}, \theta) \tag{27}$$

We often work with the log-likelihood function for numerical stability:

$$\log p(y_{1:T}|\theta) = \sum_{t=1}^{T} \left[-\frac{1}{2}\log((P_t)) - \frac{1}{2}e_t^T P_t^{-1} e_t - \frac{N}{2}\log(2\pi)\right]$$

We are then able to use an optimization algorithm (Newton-Raphson or BFGS) to find the values of T_t, H_t, G_t that maximize the log-likelihood function.

[13] Adapted from *Forecasting, Structural Time Series Models and the Kalman Filter* Section 3.4 (Harvey, 1990)

B Supplementary Materials

B.1 Code Repository

The supplementary material extends to include additional models and scripts for data preparation, model fitting, and analysis, all of which are in the `Code Appendix` folder. Also available at GitHub Repository: https://github.com/kerryzl77/SMC-SSM

B.2 Additional SMC Model

B.2.1 BUGS Code of Poisson-Lognormal Model

```
stateSpaceModelCode <- nimbleCode({
    # Priors
    phi ~ dunif(0, 0.1)   # phi (death rate)
    beta ~ dunif(0, 0.1)  # beta (birth rate)

    # Initial state
    S[1] ~ dpois(lambda = N_initial)
    Y[1] ~ dlnorm(meanlog = log(S[1]), sdlog = log(sdo))

    # Process model
    for (t in 2:T) {
      # Binomial death process
      d[t] ~ dbin(size = S[t-1], prob = phi)

      # Surviving population after deaths but before births
      S_post_death[t] <- S[t-1] - d[t]

      # Poisson birth process on the surviving population
      b[t] ~ dpois(lambda = beta * S_post_death[t])

      # Update total surviving population including new births
      S[t] ~ dpois(lambda = S_post_death[t] + b[t] + I[t])

      # Observation process
      Y[t] ~ dlnorm(meanlog = log(S[t]), sdlog = log(sdo))
    }
  })
```

B.2.2 NimbleSMC Bootstrap Filter with Exact Rates

This model employed the exact death rate ϕ and birth rate β from the data, calculated as proportions of the annual total population corresponding to the number of deaths and births, respectively.

$$S_t = S_{t-1} - \text{death}_{t-1} + \text{birth}_{t-1} + \text{immigration}_{t-1} + \epsilon_t,$$
$$y_t = S_t + \eta_t,$$

where the process noise ϵ_t and the observation noise η_t are defined as:

$$\epsilon_t \sim \mathcal{N}(0, \sqrt{S_{t-1}}),$$
$$\eta_t \sim \mathcal{N}(0, G_t).$$

The result closely aligns the true population observation with a gap in net immigration. See particle_filter/NimbleSMC/SMC-Normal-Exact.R.

```
normalModelCode <- nimbleCode({
    # Initial state
    S[1] ~ dnorm(Population_initial, sd = 100000)

    # Process equation with exact phi and beta
    for (t in 2:T) {
        S[t] ~ dnorm((S[t-1] - death[t-1] + birth[t-1]
        + immigration[t-1]), sd = sqrt(S[t-1]))
    }
    # Measurement equation
    for (t in 1:T) {
        y[t] ~ dnorm(S[t], sd = sdo)
    }
})
```

B.2.3 NimbleSMC Bootstrap Filter with Dynamic Rates

We set up a dynamic death rate ϕ_t and birth rate β_t with a uniform prior at each time t, and fit using the NimbleSMC Bootstrap function. The result failed to converge due to the overstochasticity in the model, encouraging us to develop the hard-coded model in 2.3.3. See particle_filter/NimbleSMC/SMC_Bin_dynamic_rate.R.

```
stateSpaceModelCode <- nimbleCode({
  # Priors for time-varying rates
```

```
  for(t in 1:T) {
    phi[t] ~ dunif(0, 0.1)
    beta[t] ~ dunif(0, 0.1)
  }

  # Initial state
  S[1] ~ dpois(lambda = N_initial)
  Y[1] ~ dlnorm(meanlog = log(S[1]), sdlog = log(sdo))

  # Process model for subsequent years
  for(t in 2:T) {
    # Binomial birth process with time-varying rate
    b[t] ~ dbin(size = S[t-1], prob = beta[t-1])

    # Binomial death process with time-varying rate
    d[t] ~ dbin(size = S[t-1], prob = phi[t-1])

    # Update surviving population for the current year
    S[t] ~ dpois(lambda = S[t-1] + b[t] - d[t] + I[t])

    # Observation process
    Y[t] ~ dlnorm(meanlog = log(S[t]), sdlog = log(sdo))
  }
})
```

B.2.4 NimbleSMC Bootstrap Filter with Binomial Process

Rather than assuming a Poisson birth process dependent on the surviving population, we employed two independent binomial processes for births and deaths, maintaining the same parameterization as in 2.3.2. The results mirror the Poisson-Lognormal model result in 3.2, albeit with marginally higher population estimates, while achieving a comparable log-likelihood of -3828.82. See particle_filter/NimbleSMC/SMC-Bin.R.

B.3 Visualizations

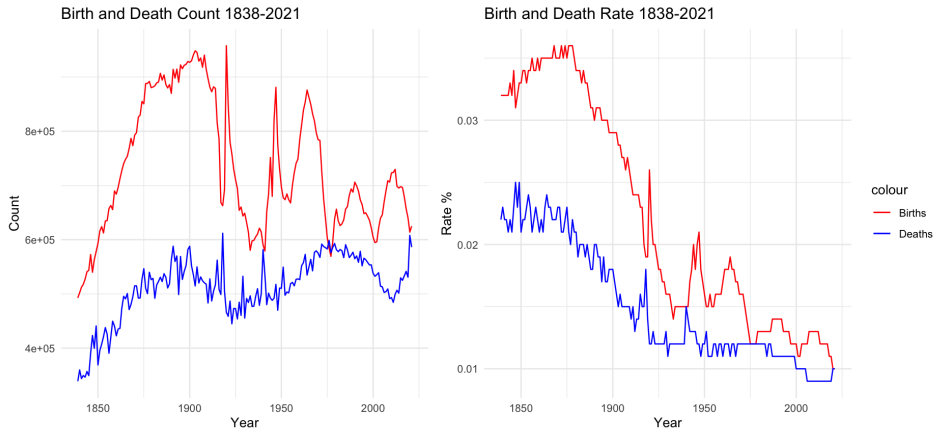

Figure 1: Birth and Death Count/Rate - 1838-2021

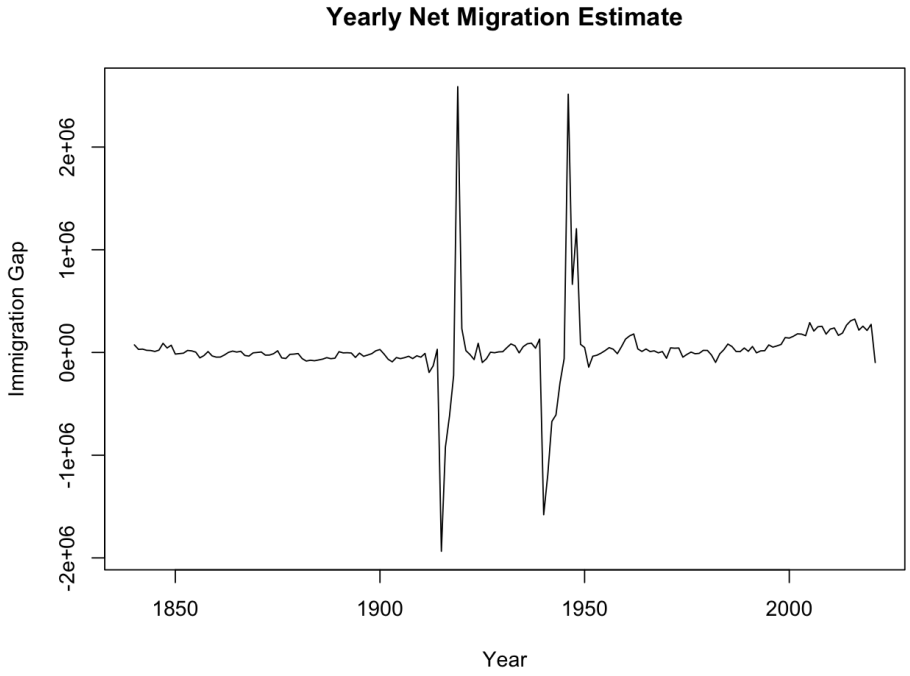

Figure 2: Yearly Net Migration Estimate for England & Wales from 1838 to 2021. The net migration estimate is determined by the discrepancy between the observed annual population change and the projected change derived from birth and death records

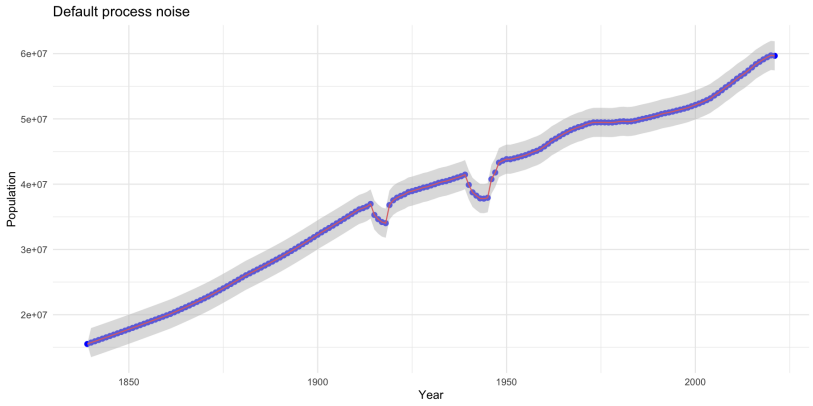

Figure 3: Model 1 - Kalman Filter Default Error. Note. Blue dots represent observed data points, red dots denote (Kalman filter) predictions, and the grey area indicates the 95% confidence interval

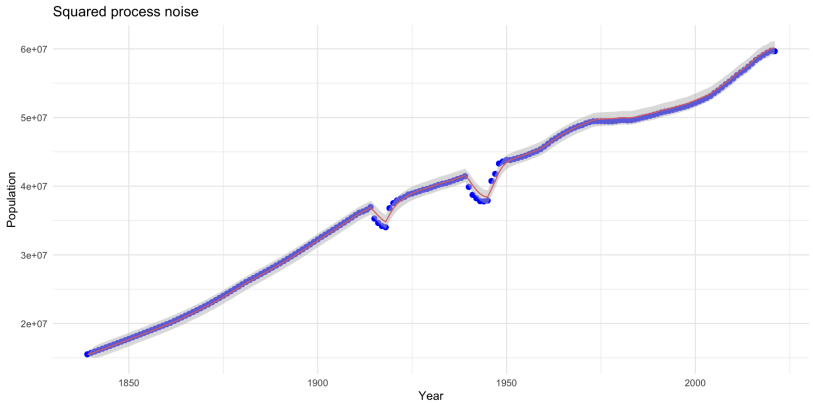

Figure 4: Model 1 - Kalman Filter Squared Error

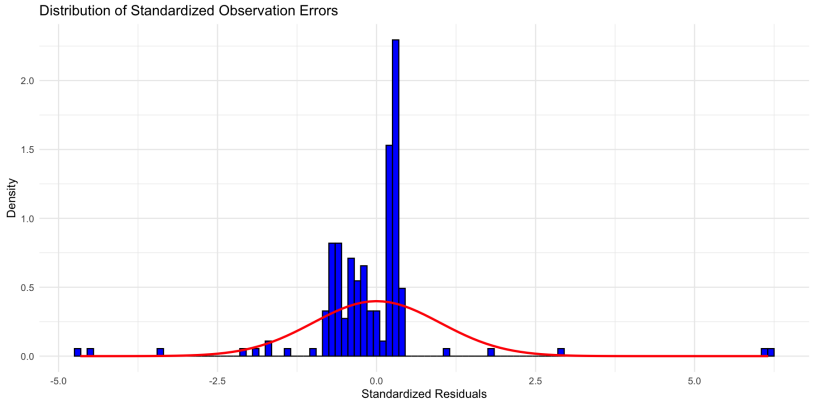

Figure 5: Model 1 - Kalman Filter Residuals

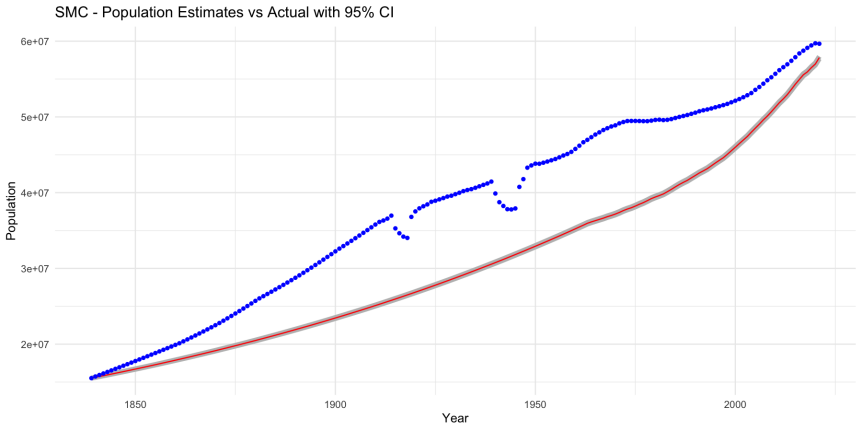

Figure 6: Model 2 - SMC Bootstrap Filter

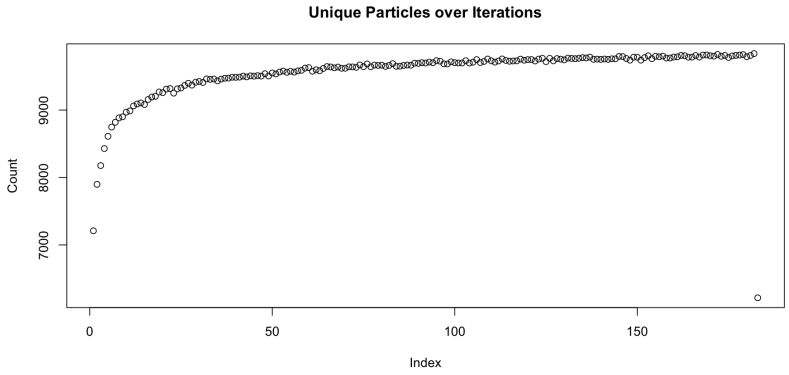

Figure 7: Model 2 - Unique Particles Numbers

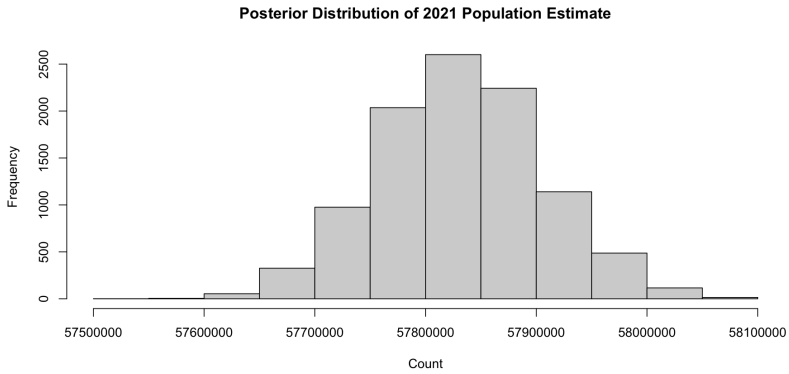

Figure 8: Model 2 - Posterior Density of Final State. Notably, the true population value of 59,660,524 lies outside the depicted posterior range

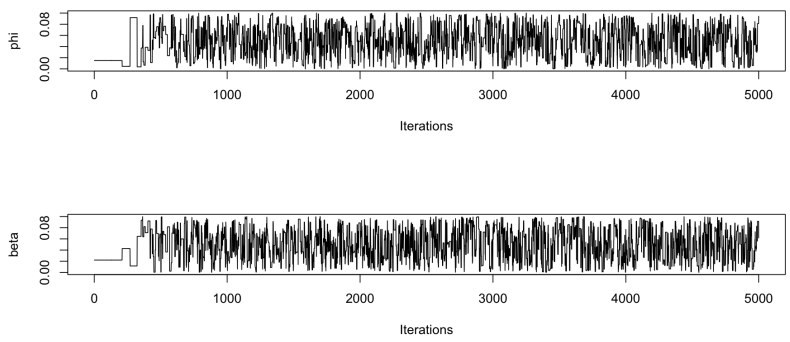

Figure 9: Model 2 - PMMH Trace Plots of parameters

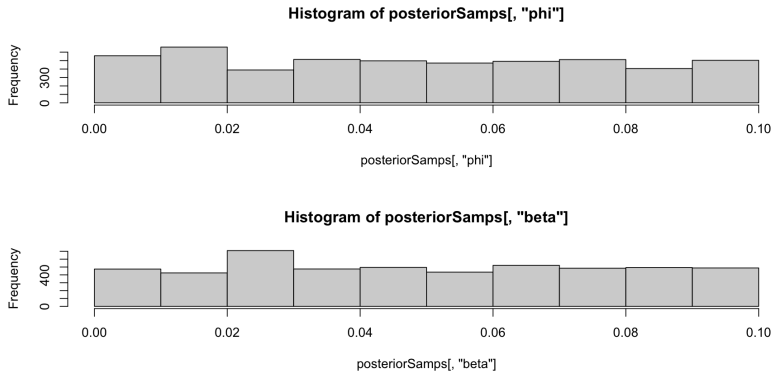

Figure 10: Model 2 - PMMH Posterior Density of parameters

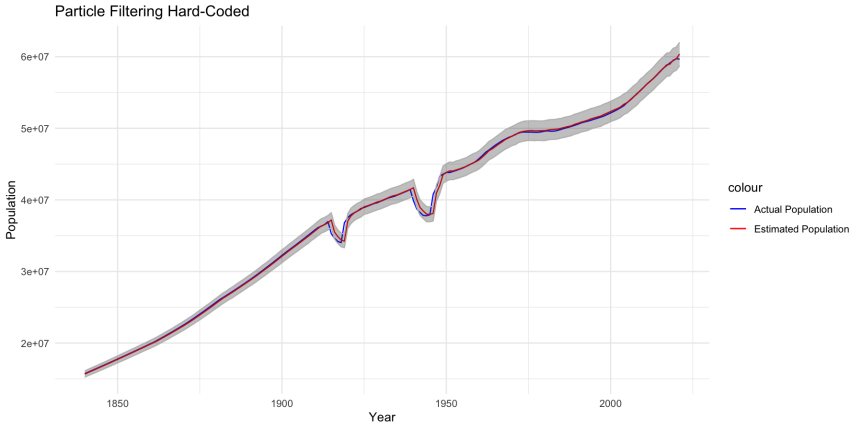

Figure 11: Model 3 - SMC Hard-Coded

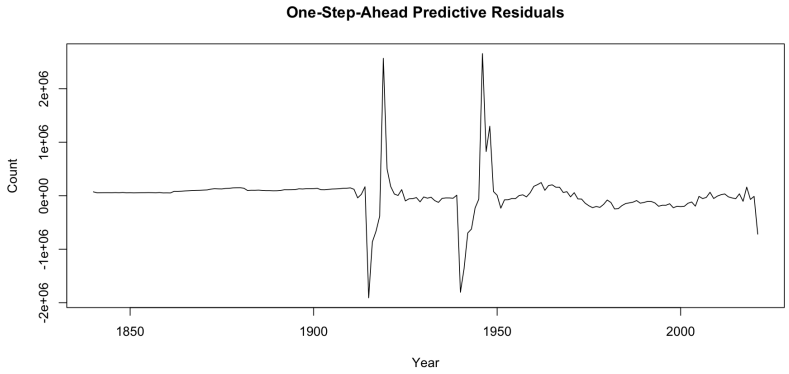

Figure 12: Model 3 - One Step Ahead Predictive Residuals

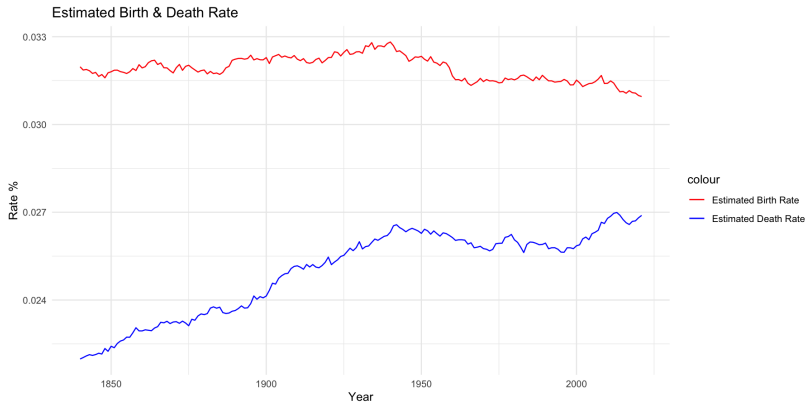

Figure 13: Model 3 - SMC Hard-Coded Birth & Death Rates

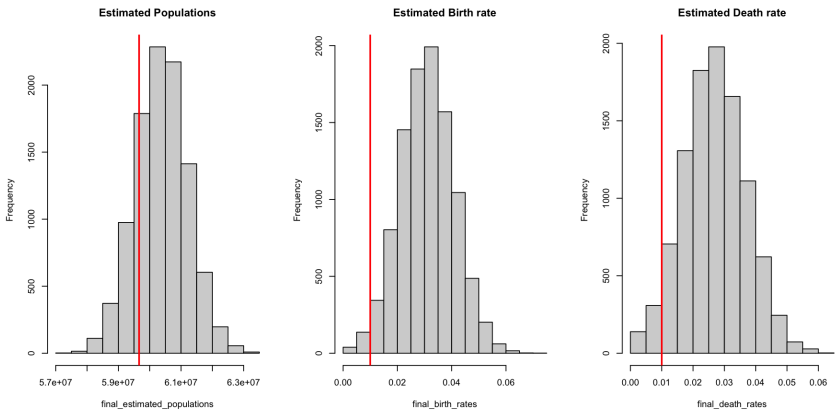

Figure 14: Model 3 - Final Year States & Parameters Distribution

References

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

Auger-Méthé, M., Newman, K., Cole, D., Empacher, F., Gryba, R., King, A. A., Leos-Barajas, V., Mills Flemming, J., Nielsen, A., Petris, G., and Thomas, L. (2021). A guide to state–space modeling of ecological time series. *Ecological Monographs*, 91(4):e01470.

Cappuccino, N. and Price, P. W. (1995). *Population Dynamics: New Approaches and Synthesis*. Academic Press, San Diego, CA.

Clark, J. S. and Bjørnstad, O. N. (2004). Population time series: Process variability, observation errors, missing values, lags, and hidden states. *Ecology*, 85:3140–3150.

Dahlin, J. and Schön, T. B. (2019). Getting started with particle metropolis-hastings for inference in nonlinear dynamical models. *Journal of Statistical Software, Code Snippets*, 88(2):1–41.

de Valpine, P. (2012). Frequentist analysis of hierarchical models for population dynamics and demographic data. *Journal of Ornithology*, 152(2):393–408.

de Valpine, P., Turek, D., Paciorek, C. J., Anderson-Bergman, C., Temple Lang, D., and Bodik, R. (2017). Programming with models: Writing statistical algorithms for general model structures with NIMBLE. *Journal of Computational and Graphical Statistics*, 26:403–413.

Doucet, A., de Freitas, N., and Gordon, N. (2001). *An Introduction to Sequential Monte Carlo Methods*, page 3–14. Springer, New York, NY.

Durbin, J. and Koopman, S. J. (2012). *Time Series Analysis by State Space Methods*. Oxford University Press.

for Demography, N. S. C. (2005). Estimating international migration for population estimates - an information paper. Technical report.

for National Statistics, O. (2022). Population estimates for the uk, england, wales, scotland and northern ireland: mid-2021.

Gordon, N. J., Salmond, D. J., and Smith, A. F. M. (1993). Novel approach to nonlinear/non-gaussian bayesian state estimation.

Harvey, A. C. (1990). *Forecasting, Structural Time Series Models and the Kalman Filter*. Cambridge University Press, Cambridge.

Hostetler, J. A. and Chandler, R. B. (2015). Improved state-space models for inference about spatial and temporal variation in abundance from count data. *Ecology*, 96(6):1713–1723.

Ionides, E. L., Nguyen, D., Atchadé, Y., Stoev, S., and King, A. A. (2015). Inference for dynamic and latent variable models via iterated, perturbed bayes maps. *Proceedings of the National Academy of Sciences*, 112(3):719–724.

Lee, P. M. (2012). *Bayesian Statistics: An Introduction*. Wiley Publishing, 4th edition.

Michaud, N., de Valpine, P., Turek, D., Paciorek, C. J., and Nguyen, D. (2021). Sequential monte carlo methods in the nimble and nimblesmc r packages. *Journal of Statistical Software*, 100(3):1–39.

Newman, K. B., Buckland, S. T., Morgan, B. J. T., King, R., Borchers, D. L., and Cole, D. J. (2014). *Modelling Population Dynamics: Model Formulation, Fitting and Assessment Using State-Space Methods*. Methods in Statistical Ecology. Springer, New York, NY, 1 edition.

Nocedal, J. and Wright, S. J. (2006). *Quasi-Newton Methods*, pages 135–163. Springer New York, New York, NY.

Powell, G. and Racinskij, V. (2022). Variance estimation for 2021 census population estimates. *ONS website*.

Robert, C. P. and Casella, G. (1999). *Monte Carlo Statistical Methods*. Springer, New York.

Åkesson, B. M., Jørgensen, J. B., Poulsen, N. K., and Jørgensen, S. B. (2008). A generalized autocovariance least-squares method for kalman filter tuning. *Journal of Process Control*, 18(7):769–779.