Topics in Bayesian Inference and Model Assessment for Partially Observed Stochastic Epidemic Models

A Thesis Presented for

The Doctor of Philosophy

Degree

Faculty of Science

School of Mathematical Sciences

The University of Nottingham

Georgios Aristotelous

June 2020

Acknowledgements

First and foremost I would like to express my gratitude and appreciation to my supervisors, Professor Theodore Kypraios and Professor Philip O'Neill. Theo and Phil, there are numerous reasons to thank you for, both academic and personal, all of which I can not possibly list here. Theo, thank you for your positive and encouraging attitude along the way and thank you for being a friend. Phil, thank you for providing a role model for me to look up to and for all the invaluable advice, especially in the times that it was most needed. Thanks must go to my examiners, Professor Frank Ball (Internal) and Dr. Nikolaos Demiris (External), for their comments and suggestions. Frank, I additionally want to thank you for your valuable advice and recommendations as part of my annual reviews.

My deepest appreciation goes to my parents whose hard work, sacrifice and dedication provided me a life full of opportunities. I would also like to thank my brothers, Andreas and Nikolas, for their belief in me. Thanks go as well to the rest of my family, particularly to my cousin Nectarios who has always been a true friend and helped me in many important decisions in my life.

Special thanks go to my brotherly friends back home in Cyprus, Sotos, Kostas and Konstantinos, who have been next to me throughout all periods of my life. I will always be grateful to you for everything that you have done for me but I am especially grateful for your support during this challenging period of my life. I am proud to be

your friend.

Acknowledgement is due to my friends here in Nottingham too, together with which I shared the struggles and the joys of the PhD life. Rowland, thank you for the countless talks about research, culture, society and basically everything about life. Most of all thank you for being a good friend. Valon, I will always remember the days at the gym and the nights out! Most of all I will remember the long walks and our interesting talks. Thank you for pushing me to join you and play football! Yiannis and Eleni, thank you for helping me take my mind off my thesis although I could do with less pizza and desserts! You were always understanding without me needing to say much. I will always be grateful to you for helping me go through the writing-up period. This journey would be much harder for me without the friendship of all of you. I will definitely miss you all.

Thanks must go to Diana, whose presence in my life prompted me to move to the UK, in search of a better future, and to eventually pursue a PhD degree. Diana, thank you as without you I am not sure if I would have taken this path.

Finally, I would like to thank the University of Nottingham for providing me financial support during my PhD.

Abstract

Stochastic epidemic models can offer a vitally important public health tool for understanding and controlling disease progression. However, these models are of little practical use if they are not supported by data or are not applicable to efficient parameter inference methods. The peculiarities of the epidemic setting, where data are not independent and epidemic processes are rarely fully observed, complicate both model assessment and parameter inference for stochastic epidemic models. Methods for model assessment are not well-established and methods for inference, although more established, still remain inefficient for large-scale outbreaks.

This thesis is concerned with the development of methods for both model assessment and inference for stochastic epidemic models. The methods are illustrated on continuous time SIR (susceptible $\rightarrow$ infective $\rightarrow$ removed) models and it is assumed that the available data consist only of the removal times of infected individuals with their infection times being unobserved.

First, two novel model assessment tools are developed, based on the posterior predictive distribution of removal curves, namely the distance method and the position-time method. Both methods rely on the general idea of posterior predictive checking, where a model's fit is assessed by checking whether replicated data, generated under the model, look similar to the observed data. The distance method conducts the assessment by calculating distances between removal curves whereas the

position-time method conducts the assessment pointwise, at a sequence of suitably chosen time points. Both methods provide visual and quantitative outputs with meaningful interpretation. The performance of the methods benefits from the development and application of a time shifting intervention, that horizontally (time) shifts each replicated removal curve by an appropriately chosen constant, so that the stages of each replicated curve better correspond to those of the observed. Extensive simulation studies suggest that both the distance and the position-time methods can successfully assess the infectious period distribution assumption and the infection rate form assumption of stochastic epidemic models.

Then, the focus is placed on developing methods to assess the population mixing assumption of stochastic epidemic models, in the case that household information is available. To this end, a classical hypothesis test is developed for which the null hypothesis is that individuals mix in the population homogeneously. The test is based on household labels of individuals and relies on the idea that, in the presence of household effect, events of individuals belonging to the same household should occur closer in time rather than further apart. The key behind developing the test is that, under the null hypothesis of homogeneous mixing, the discrete random vector of household labels has a known sampling distribution that does not dependent on any model parameters. The test carries an ordinal interpretation, where the lower the observed value of the test statistic and its corresponding p-value are, the more the evidence against the null hypothesis and in favour of the hypothesis that there is a household effect in the spread of the outbreak. The test exhibits excellent performance when applied to both simulated data and to a widely studied real-life epidemic dataset.

In the remainder of the thesis, attention is turned from model assessment to Bayesian inference. The relevant aim is to develop Markov chain Monte Carlo (MCMC) algorithms that can conduct more efficient updating of the unobserved infection

times, than the currently existing algorithms. Initially, the problem of updating one infection time at a time is considered and a new 1-dimensional update algorithm is developed, namely the IS-1d MCMC algorithm. The main feature of the algorithm is the use of individual-specific parameters in the proposal distributions for the infection times. These parameters allow the proposal distributions to produce patterns of nonhomogeneity (among individuals) which are in some cases present in the target distribution. The IS-1d MCMC algorithm performs favourably when compared to currently existing 1-dimensional update algorithms. Subsequently, the more interesting problem of updating many infection times at a time is considered and a novel block update MCMC algorithm is developed, referred to as the DIS-block MCMC algorithm. Similar to the IS-1d algorithm, the proposal distributions of the DIS-block algorithm also have individual-specific parameters but they also have an additional parameter that induces dependency on the current state and makes the algorithm perform a dependent in nature exploration of the target space. The algorithm also benefits from another two features, parameter reduction and an automated method for optimally specifying the number of infection times to update. Simulation studies suggest that the DIS-block algorithm can offer a substantial improvement in mixing compared to the current optimally performing block update algorithm; for the considered datasets of the simulation study, the DIS-block algorithm is from 1.41 up to 6.57 times more efficient than its comparator, and 3.35 times on average.

Contents

List of Tables

List of Figures

List of Algorithms

Chapter 1

Introduction

1.1 Thesis motivation and aims

Stochastic epidemic models can offer a crucially important public health tool for understanding and controlling disease progression. More specifically, such models can be used to evaluate vaccination strategies (see e.g. Yuan et al. (2015); Nguyen and Carlson (2016)) or assess the effectiveness of proposed control measures (see e.g. Keeling (2001); Adrakey et al. (2017)). Also, it is typical that estimates of the parameters of these models correspond to estimates of quantities of epidemiological interest, such as the basic reproduction number, knowledge of which can determine if there is a positive probability for a major outbreak to occur. The great importance of these models is perhaps best highlighted by their role in the profound 2020 global COVID-19 pandemic, where the results of such epidemiological modelling have been used to estimate the basic reproduction number (see e.g. Kucharski et al. (2020b)), assess the effectiveness of disease-control control measures (see e.g. Kucharski et al. (2020a)) and inform policy making in the UK and other countries (see Ferguson et al. (2020)).

Nonetheless, these models have little practical use unless they allow efficient

estimation of their parameters and they provide adequate fit to real-life epidemic outbreaks. Unfortunately, neither parameter inference nor model assessment for stochastic epidemic models is straightforward. This is due to the fact that the epidemic setting is endowed with inherent difficulties, with epidemic data not being independent and epidemic processes being partially observed and realized once.

The biggest complication in parameter inference arises from the fact that epidemic processes are usually partially observed. It is typical in practice that infection times are not observed and that instead only case detection times are observed. This is almost always the case for human diseases, when the appearance of symptoms (case detection) is usually the first sign that an individual has been infected. Such type of partial observation typically leads to intractable model likelihoods which in turn handicap the ability to conduct inference for the model parameters of interest using conventional methods, such as maximum likelihood estimation. A way to overcome this problem is via data augmentation in a Bayesian framework, where the unobserved data are treated as additional unknown variables and inference is conducted by targeting the joint posterior distribution of model parameters and unobserved data, using Markov chain Monte Carlo (MCMC) methods (O'Neill and Roberts, 1999; Gibson and Renshaw, 1998). The challenge in the implementation of such methods comes from the typically high dimension of the space of unobserved data, which makes the inference procedure inefficient and affects its practical utility. Various MCMC algorithms have been employed to address these issues, making use of different ideas such as parameter reduction and non-centered parameterizations (see section 1.4.2 and the references therein). Although some of these MCMC algorithms have managed to mitigate the effect of the problem (see e.g. Neal and Roberts (2005); Kypraios (2007); Xiang and Neal (2014)), the fundamental issues of high-dimensionality still persist and more efficient algorithms are needed.

Even if parameter inference can be achieved for a model, moving then to assess the

adequacy of its fit to the data in question is a challenging task. For example, since epidemic data are not independent, simple and standard measures of model fit that are constructed to suit independent data settings (such as chi-squared goodness-of-fit statistics) are not directly usable. Also, the fact that epidemic processes are realized once, means that there is a lack of replication and that the variability of a fitted model can not be assessed. Due to such challenges, the area of model assessment of stochastic epidemic models is somewhat underdeveloped and there is significant scope and need for innovation (O'Neill, 2010; Gibson et al., 2018).

This thesis is concerned with the development of methods for both model assessment and inference, for stochastic epidemic models. Specifically, regarding model assessment, the aim is to develop new non-standard measures of model fit, that will be suited to the epidemic setting. As far as inference, the aim is to develop novel MCMC algorithms that can better address the issues of high-dimensionality, related to the unobserved data, and allow more efficient inference for the model parameters. The intent, is that the development of all methods is driven by practical utility and by acknowledging the peculiarities of the epidemic setting.

1.2 Thesis layout

This thesis is structured as follows. The remainder of chapter 1 collects the background related to the purposes of this thesis and reviews the relevant literature.

Chapter 2 is concerned with the development of two model assessment measures, based on the posterior predictive distribution of removal curves. The development procedure includes highlighting the peculiarities of the epidemic setting, describing a procedure for distinguishing between minor and major outbreak realizations and introducing a time shifting intervention to alleviate the undesired noise of simulated removal curves. The performance of the model assessment measures, in assessing the

infectious period, the infection rate form and the population mixing assumptions of SIR models, is examined via thorough simulation studies.

Chapter 3 develops a classical hypothesis test for assessing the population mixing assumption of epidemic models. The performance of the test is examined using an extensive simulation study and by applying it to a widely studied real-life dataset.

Chapter 4 is concerned with the development of MCMC algorithms for more efficient updating of unobserved data. The chapter consists of two main sections, section 4.2, which considers MCMC algorithms based on updating one unobserved data point at a time (in a 1-dimensional update step), and section 4.3, which considers MCMC algorithms based on updating many unobserved data points at a time (in a block update step). In both these sections, the layout is similar, with the development process being guided by acknowledging the limitations of existing algorithms, and the performance of the developed algorithms being compared to the existing ones via simulation studies.

Finally, chapter 5 concludes by summarizing the work of this thesis, discussing its general limitations and highlighting its main contribution.

1.3 Background

This section collects the background that is relevant to the purposes of this thesis.

1.3.1 Bayesian inference

This section describes the fundamentals of Bayesian inference. The literature on the topic is enormous and the purpose here is simply to outline the key aspects, similar

to Held et al. (2019, chapter 9). For a more detailed and rigorous approach the reader is directed to more comprehensive references such as Robert (2007) or Lee (2012).

1.3.1.1 The Bayesian approach

The Bayesian approach to statistical inference goes as follows. Suppose that some data have been observed, denoted as $\boldsymbol{y}$, and a sampling model $\pi(\boldsymbol{y} \mid \boldsymbol{\theta})$ has been assumed for that data, where $\pi(\boldsymbol{y} \mid \boldsymbol{\theta})$ is a probability density function (p.d.f.), referred to as the *sampling density* of $\boldsymbol{y}$ (if $\boldsymbol{y}$ is continuous), or, a probability mass function (p.m.f.), referred to as the *sampling mass* of $\boldsymbol{y}$ (if $\boldsymbol{y}$ is discrete), with parameter $\boldsymbol{\theta}$; both $\boldsymbol{y}$ and $\boldsymbol{\theta}$ are vectors in general. The inferential objective is to find plausible values for the model parameter given the observed data, i.e. to learn about $\boldsymbol{\theta}$ given $\boldsymbol{y}$. A fundamental feature of the Bayesian approach is that all unknown quantities are considered random and the uncertainty about them is described by probability distributions. Therefore, all Bayesian inference relies on the conditional distribution of $\boldsymbol{\theta}$ given $\boldsymbol{y}$, which is known as the *posterior distribution* of $\boldsymbol{\theta}$ given $\boldsymbol{y}$. The p.d.f. (if $\boldsymbol{\theta}$ is continuous) or p.m.f. (if $\boldsymbol{\theta}$ is discrete) of $\boldsymbol{\theta}$ given $\boldsymbol{y}$, denoted as $\pi(\boldsymbol{\theta} \mid \boldsymbol{y})$ and referred to as the *posterior density* (if $\boldsymbol{\theta}$ is continuous) or the *posterior mass* (if $\boldsymbol{\theta}$ is discrete) of $\boldsymbol{\theta}$ given $\boldsymbol{y}$, is given according to Bayes' theorem by

$$\pi(\boldsymbol{\theta} \mid \boldsymbol{y}) = \frac{\pi(\boldsymbol{y} \mid \boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\pi(\boldsymbol{y})} \propto \pi(\boldsymbol{y} \mid \boldsymbol{\theta})\pi(\boldsymbol{\theta}). \tag{1.1}$$

In the expression (1.1) above, $\pi(\boldsymbol{\theta})$ denotes the p.d.f. (if $\boldsymbol{\theta}$ is continuous) or p.m.f. (if $\boldsymbol{\theta}$ is discrete) of the *prior distribution* of $\boldsymbol{\theta}$ and it is known as the *prior density* (if $\boldsymbol{\theta}$ is continuous) or the *prior mass* (if $\boldsymbol{\theta}$ is discrete) of $\boldsymbol{\theta}$. The term $\pi(\boldsymbol{y} \mid \boldsymbol{\theta})$ is as above the sampling density (if $\boldsymbol{y}$ is continuous) or the sampling mass (if $\boldsymbol{y}$ is discrete) of $\boldsymbol{y}$ but is now regarded as a function of $\boldsymbol{\theta}$, therefore it is the *likelihood* of $\boldsymbol{\theta}$ given the observed data $\boldsymbol{y}$. Finally, $\pi(\boldsymbol{y})$ is a normalizing constant (i.e. does not depend on $\boldsymbol{\theta}$) that ensures that $\pi(\boldsymbol{\theta} \mid \boldsymbol{y})$ integrates (if $\boldsymbol{\theta}$ is continuous) or sums (if $\boldsymbol{\theta}$ is discrete) to 1, so that the posterior distribution is a probability distribution, and it is obtained

by $\pi(\boldsymbol{y}) = \int \pi(\boldsymbol{y} \mid \boldsymbol{\theta})\pi(\boldsymbol{\theta})d\boldsymbol{\theta}$ (if $\boldsymbol{\theta}$ is continuous) or $\pi(\boldsymbol{y}) = \sum \pi(\boldsymbol{y} \mid \boldsymbol{\theta})\pi(\boldsymbol{\theta})$ (if $\boldsymbol{\theta}$ is discrete), where the integration (or summation) is over the set of all the possible values of $\boldsymbol{\theta}$.

The prior distribution represents one's uncertainty about the parameters before observing the data while the likelihood represents the information from the observed data. Therefore, as evident from equation (1.1) above, the posterior distribution represents one's uncertainty about the parameters after combining the information from the prior distribution with the information from the observed data.

For ease of illustration, and since this thesis is mainly concerned with continuous data and model parameters, any results presented in the remainder of section 1.3.1 and in sections 1.3.2 and 1.3.3, are concerned with the continuous case, i.e. data, model parameters and any other random quantities are considered to take values in continuous spaces; where relevant the corresponding results for the discrete case can be obtained by replacing integration with summation.

1.3.1.2 Prior distributions

Since the prior distribution represents one's initial belief about $\boldsymbol{\theta}$, before seeing any data, it is inherently subjective and might be different for different users; in fact the choice of the prior distribution has drawn considerable attention in the Bayesian community (see e.g. Bernardo and Smith (1994)). In practice though, prior distributions can broadly be divided into two types, *informative* and *uninformative*. Informative prior distributions are typically used in situations where one may believe that certain values of $\boldsymbol{\theta}$ are more plausible than others. Such belief might be based on information from previous studies, expert opinion or biological factors. For example, if $\boldsymbol{\theta}$ represented the mean time that an individual remained infectious from measles and previous information on measles suggested a set of typical values of $\boldsymbol{\theta}$, $\pi(\boldsymbol{\theta})$ could be assigned to reflect that information accordingly. On the contrary, uninformative prior

distributions are typically used in situations where no information about $\boldsymbol{\theta}$ is available before observing the data. For instance, if $\boldsymbol{\theta}$ represented a proportion, a uniform prior distribution on $[0, 1]$ (i.e. any value of $\boldsymbol{\theta}$ being equally likely) would reflect one's lack of any information about $\boldsymbol{\theta}$ prior to observing any data. Uninformative prior distributions are also referred to as objective prior distributions in the sense that, since they contain very little information about $\boldsymbol{\theta}$, they yield posterior distributions for which the information about $\boldsymbol{\theta}$ is driven almost entirely by the information in the observed data rather than subjective prior beliefs.

1.3.1.3 Purpose and practical complications

Let $\boldsymbol{\theta}|\boldsymbol{y}$ be a random vector having the posterior distribution. Just about any aspect of the posterior distribution that may be of interest can be written as an integral of the form

$$\mathrm{E}(g(\boldsymbol{\theta}) \mid \boldsymbol{y}) = \int g(\boldsymbol{\theta})\pi(\boldsymbol{\theta} \mid \boldsymbol{y})d\boldsymbol{\theta}, \tag{1.2}$$

for some suitably chosen $\mathbb{R}$-valued function g. For example, considering a 1-dimensional $\boldsymbol{\theta}$, using $g(\boldsymbol{\theta}) = \boldsymbol{\theta}$ and $g(\boldsymbol{\theta}) = \boldsymbol{\theta}^2$ gives the mean and the variance of the posterior distribution. Also, using $g = \mathbb{1}_A$, where $\mathbb{1}_A$ denotes the indicator function of the event A, gives the probability of any desired event A with respect to the posterior distribution. That is, a probability of the type $P(\boldsymbol{\theta} \in A \mid \boldsymbol{y})$; for instance, considering a 2-dimensional $\boldsymbol{\theta} = (\theta_1, \theta_2)$, A could be such that $A = \{a < \theta_1 < b\}$ or $A = \{\theta_1 < \theta_2\}$ or $A = \{a < \frac{\theta_1}{\theta_2} < b\}$, for some $a, b \in \mathbb{R}$. Therefore, in most cases, the main purpose of Bayesian inference comes down to calculating integrals as such of equation (1.2).

Although in theory the above task might appear rather simple, several complications arise in practice. The first is that, as mentioned in section 1.3.1.1 above, the normalizing constant $\pi(\boldsymbol{y})$, required to know the posterior density $\pi(\boldsymbol{\theta} \mid \boldsymbol{y})$, is obtained by the integral $\pi(\boldsymbol{y}) = \int \pi(\boldsymbol{y} \mid \boldsymbol{\theta})\pi(\boldsymbol{\theta})d\boldsymbol{\theta}$, where the region of integration is the set of all possible values of the model parameter $\boldsymbol{\theta}$. This integral is quite often

analytically intractable, especially in problems where $\boldsymbol{\theta}$ is high-dimensional, making it impossible to derive an analytic expression for $\pi(\boldsymbol{\theta} \mid \boldsymbol{y})$ and in turn impossible to perform the integration of equation (1.2). The second complication is that the likelihood $\pi(\boldsymbol{y} \mid \boldsymbol{\theta})$, also needed to know $\pi(\boldsymbol{\theta} \mid \boldsymbol{y})$ and make the integration of equation (1.2) possible, might also be intractable (see section 1.3.5.2 below for an example). A third problem is that, even in the cases that an explicit expression for $\pi(\boldsymbol{\theta} \mid \boldsymbol{y})$ is available, this in itself might not be that useful as integrations of the type of equation (1.2) are still extremely challenging to perform, complicated by the form of $\pi(\boldsymbol{\theta} \mid \boldsymbol{y})$ or $g(\boldsymbol{\theta})$ or both.

1.3.2 Markov chain Monte Carlo methods

The present section describes the fundamentals of MCMC methods and reviews some well known MCMC algorithms most relevant to the purposes of this thesis. As for the material on Bayesian inference, the literature is vast and the purpose is simply to explain the key concepts. Hence the approach taken is an intuitive one, similar to Hoff (2009) and Held et al. (2019, chapter 9). For a more rigorous approach one is referred to more comprehensive textbooks such as Gilks et al. (1996) or Robert and Casella (2004).

1.3.2.1 Motivation

Let $\pi(\boldsymbol{x})$ be a p.d.f., corresponding to a random vector $\boldsymbol{x} \in \mathcal{X}$, and suppose that interest is in evaluating integrals of the form

$$\mathrm{E}(g(\boldsymbol{x})) = \int g(\boldsymbol{x})\pi(\boldsymbol{x})d\boldsymbol{x}, \tag{1.3}$$

where g is some $\mathbb{R}$-valued function, but analytic calculations are not possible. This is exactly the problem that arises in Bayesian inference, described in section 1.3.1.3, just formulated in a more general setting; the only difference being that the posterior density $\pi(\boldsymbol{\theta} \mid \boldsymbol{y})$ is replaced by a general p.d.f. $\pi(\boldsymbol{x})$. Suppose for a while that one was

able to directly and independently sample from $\pi(\boldsymbol{x})$. Then, given an independent and identically distributed (i.i.d.) sample, $\{\boldsymbol{x}_{\text{MC}}^{(1)}, \boldsymbol{x}_{\text{MC}}^{(2)}, \ldots, \boldsymbol{x}_{\text{MC}}^{(S)}\}$, from $\pi(\boldsymbol{x})$, the strong law of large numbers (SLLN) would ensure that for any integrable $\mathbb{R}$-valued function g, the sample mean of g would converge almost surely to the expected value of g with respect to $\pi(\boldsymbol{x})$, i.e. $\frac{1}{S}\sum_{s=1}^{S} g(\boldsymbol{x}_{\text{MCMC}}^{(s)}) \overset{\text{a.s.}}{\to} \text{E}(g(\boldsymbol{x})) = \int g(\boldsymbol{x})\pi(x)d\boldsymbol{x}$, as $S \to \infty$, provided that $\text{E}(g(\boldsymbol{x}))$ exists. Therefore, for large enough sample size S, one would be able to approximate, arbitrarily exactly, the required integral as

$$\text{E}(g(\boldsymbol{x})) = \int g(\boldsymbol{x})\pi(\boldsymbol{x})d\boldsymbol{x} \approx \frac{1}{S}\sum_{s=1}^{S} g(\boldsymbol{x}_{\text{MC}}^{(s)}). \tag{1.4}$$

Approximation (1.4) above is referred to as *Monte Carlo (MC) approximation* and is considered the 'gold standard' of this type of sample-based approximations in the sense that knowledge about $\pi(\boldsymbol{x})$ is represented by the most informative type of samples, an i.i.d. sample.

Unfortunately, in most practical cases, direct and independent sampling from $\pi(\boldsymbol{x})$ is not possible and this approximation cannot be performed. MCMC methods provide an alternative way for sampling from $\pi(\boldsymbol{x})$ and approximating integrals as the above.

1.3.2.2 Overview

The idea of MCMC methods dates back to 1953 and originated from the particle Physics literature and the work of Metropolis et al. (1953). It was later generalized in a statistical context by Hastings (1970). Nonetheless it was not until the work of Gelfand and Smith (1990) that the statistical community became aware of the potential of MCMC methods for Bayesian inference. Since then, the use of MCMC methods for applied statistical modelling has increased rapidly and has revolutionized the way statistical models are fitted and in the process, dramatically revised the scope of models which can be entertained.

MCMC methods are a collection of computational techniques (algorithms) for sampling from a non-normalized p.d.f. $\pi(\boldsymbol{x})$, typically referred to as the *target density*. More precisely, if $\pi(\boldsymbol{x})$ is a p.d.f. known up to proportionality, corresponding to a random vector $\boldsymbol{x} \in \mathcal{X}$, MCMC methods provide a way of obtaining a sample $\{\boldsymbol{x}_{\text{MCMC}}^{(1)}, \boldsymbol{x}_{\text{MCMC}}^{(2)}, \ldots, \boldsymbol{x}_{\text{MCMC}}^{(S)}\}$ from the target density $\pi(\boldsymbol{x})$. The main idea behind MCMC methods is based on ergodic Markov chain theory, and specifically on the fact that if a discrete-time Markov chain is ergodic (i.e. satisfies some desirable properties from an asymptotic standpoint; see e.g. Gilks et al. (1996, chapter 4) for details), it then converges in distribution to a unique probability distribution, known as its stationary distribution. MCMC methods utilize this fact as follows. Given a target density $\pi(\boldsymbol{x})$, corresponding to a random vector $\boldsymbol{x} \in \mathcal{X}$, they construct an ergodic discrete-time Markov chain having as state space $\mathcal{X}$ and as stationary distribution the distribution associated with the target density $\pi(\boldsymbol{x})$. If this chain is then simulated, and run long enough so that convergence to the stationary distribution can be assumed, the post-convergence chain's sample path $\{\boldsymbol{x}_{\text{MCMC}}^{(1)}, \boldsymbol{x}_{\text{MCMC}}^{(2)}, \ldots, \boldsymbol{x}_{\text{MCMC}}^{(S)}\}$ is, at least approximately, a sample from the target density $\pi(\boldsymbol{x})$. In addition, and most crucially, consistency of ergodic averages (see Gilks et al. (1996, theorem 4.3)) ensures that, although the sample $\{\boldsymbol{x}_{\text{MCMC}}^{(1)}, \boldsymbol{x}_{\text{MCMC}}^{(2)}, \ldots, \boldsymbol{x}_{\text{MCMC}}^{(S)}\}$ is by construction not independent (Markov property; each sampled value $\boldsymbol{x}_{\text{MCMC}}^{(s+1)}$ depends on the previously sampled value $\boldsymbol{x}_{\text{MCMC}}^{(s)}$), it can still be used as in the MC approximation case and perform the desired integral approximations. More specifically, it is true that for any integrable $\mathbb{R}$-valued function g, the sample mean of g converges almost surely to the expected value of g with respect to $\pi(\boldsymbol{x})$, i.e. $\frac{1}{S}\sum_{s=1}^{S} g(\boldsymbol{x}_{\text{MCMC}}^{(s)}) \overset{\text{a.s.}}{\to} \text{E}(g(\boldsymbol{x})) = \int g(\boldsymbol{x})\pi(x)d\boldsymbol{x}$, as $S \to \infty$, provided that $\text{E}(g(\boldsymbol{x}))$ exists. Therefore, for large enough sample size S, it is true that

$$\text{E}(g(\boldsymbol{x})) = \int g(\boldsymbol{x})\pi(\boldsymbol{x})d\boldsymbol{x} \approx \frac{1}{S}\sum_{s=1}^{S} g(\boldsymbol{x}_{\text{MCMC}}^{(s)}). \tag{1.5}$$

Approximation (1.5) above is referred to as *MCMC approximation* and it is the cornerstone of MCMC methods.

Before proceeding to the next section, a pause is taken to acknowledge how MCMC methods can solve the practical complications of Bayesian inference mentioned earlier in section 1.3.1.3. In a Bayesian context, the target density of interest $\pi(\boldsymbol{x})$ is the posterior density $\pi(\boldsymbol{\theta} \mid \boldsymbol{y})$. The first complication, namely the calculation of the normalizing constant $\pi(\boldsymbol{y})$ in Bayes' theorem (see equation (1.1)), is avoided since MCMC methods only require knowledge of the target density up to proportionality. In other words, to implement MCMC methods one only needs to be able to compute $\pi(\boldsymbol{y} \mid \boldsymbol{\theta})\pi(\boldsymbol{\theta})$ and not $\pi(\boldsymbol{y})$. The second complication, of an intractable likelihood, can be addressed by introducing additional variables in a way that the resulting likelihood becomes tractable. Then MCMC methods can be used to sample from the joint posterior distribution of model parameters and additional variables. It is often the case that such additional variables represent missing data, which are used to augment the observed data, and thus such an approach is called *data augmentation* and was originally developed by Tanner and Wong (1987); see sections 1.3.5.2 and 1.3.5.3 and relevant parts of sections 1.3.5.5 to 1.3.5.7 for examples on how data augmentation is employed in settings most relevant to the purposes of this thesis. The third complication, the calculation of integrals of the type of equation (1.2), is dealt by the MCMC approximation (approximation (1.5)), as explained in the preceding paragraph.

1.3.2.3 MCMC diagnostics

As in section 1.3.2.2 above, let $\pi(\boldsymbol{x})$, corresponding to a random vector $\boldsymbol{x} \in \mathcal{X}$, be the target density of interest. As already discussed in sections 1.3.2.1 and 1.3.2.2, the main task of MCMC methods (especially in the context of Bayesian inference) is to obtain a sample $\{\boldsymbol{x}^{(1)}_{\text{MCMC}}, \boldsymbol{x}^{(2)}_{\text{MCMC}}, \ldots, \boldsymbol{x}^{(S)}_{\text{MCMC}}\}$ from $\pi(\boldsymbol{x})$ (by constructing and simulating an ergodic discrete-time Markov chain with state space $\mathcal{X}$ and stationary distribution

$\pi(\boldsymbol{x})$) and use it to calculate integrals as such of equation (1.3) using approximation (1.5). Although the theory (see section 1.3.2.2) ensures that eventually, as $S \to \infty$, the chain will converge to its stationary distribution $\pi(\boldsymbol{x})$ and in turn approximation (1.5) will be exact, in practice the chain cannot be possibly run forever and the quality of the obtained MCMC sample, in relation to performing the aforementioned task, is not guaranteed. Therefore, the standard practice is to use diagnostic tools in order to assess the quality of an obtained MCMC sample. Specifically, there are two properties (not unrelated to each other) of an MCMC sample (chain) that determine its quality and require assessment, referred to as *stationarity* and *mixing*.

Stationarity A chain that has reached stationarity is a chain that has converged to its stationary distribution. In practice checking for stationarity translates to checking if, from a chain iteration and onwards, the chain values can be assumed to be sampled from $\pi(\boldsymbol{x})$. To this end, it is typical for a user to discard the first S_B values of the chain, in the so-called *burn-in* period and start recording values from the $(S_B + 1)^{\text{th}}$ iteration; this is done as an attempt to allow the chain to be run long enough so that it 'forgets' its initial state and settles to stationarity. However, choosing the exact value of S_B, and ultimately concluding that a chain has reached stationarity, is fraught with epistemological problems. In general one cannot practically know for sure if the chain has indeed converged. Nevertheless, there could be evidence that the chain has not converged and thus one should at least investigate this latter possibility, i.e. check for evidence of non-stationarity.

The most common diagnostic for non-stationarity is the visual inspection of MCMC trace plots, that is, plots of some scalar function g of the MCMC sample against chain iteration (see e.g. Gilks et al. (1996) or Robert and Casella (2004)). For example, if for a given function g, chains that are initiated from different values of the state space produce materially different trace plots, then convergence cannot be assumed. Similarly, if for a given burn-in length S_B, and some function g, the trace plot is

evidently different along successive chain iterations (e.g. different for iterations 1 to 5000 compared to iterations 5001 to 10000) then again convergence cannot be assumed and the burn-in needs to be run for longer. All results reported in this thesis are based on chains that appear to have converged.

Mixing To understand the concept of chain mixing it might be helpful to think of the sampled sequence of chain values as the trajectory of a particle moving around the state space; a good mixing chain is one which the particle can quickly move between different regions of the space and a poor mixing chain is one which the particle gets stuck in some regions or moves very slowly. For example, a MC sampler for $\pi(\boldsymbol{x})$, that is a sampler that produces i.i.d. samples from $\pi(\boldsymbol{x})$, has perfect mixing as there is no correlation between the sampled values and thus it is possible to jump between any two different regions of the space in one step. This is not the case for an MCMC sampler, where by construction there is dependence among the simulated chain values. Even if the chain starts at stationarity, high correlation among the sampled values can cause the chain to have poor mixing as it will struggle to move around the state space. In practice, the correlation among the sampled values can be reduced by storing only every L^{th} iteration of the chain, while discarding the rest, in a procedure called *thinning*. Algorithms with good mixing properties are considered efficient, in the sense that they will require fewer number of iterations to adequately explore the entirety of the target distribution. The above ideas are made more formal by introducing measures that quantify and assess an algorithm's efficiency.

Suppose that interest is in evaluating $\mathrm{E}(g(\boldsymbol{x})) = \int g(\boldsymbol{x})\pi(\boldsymbol{x})d\boldsymbol{x}$, for some $\mathbb{R}$-valued function g. As already explained in section 1.3.2.2, the task of MCMC methods is to obtain a sample $\{\boldsymbol{x}^{(1)}_{\text{MCMC}}, \boldsymbol{x}^{(2)}_{\text{MCMC}}, \ldots, \boldsymbol{x}^{(S)}_{\text{MCMC}}\}$ from $\pi(\boldsymbol{x})$ and use $\frac{1}{S}\sum_{s=1}^{S} g(\boldsymbol{x}^{(s)}_{\text{MCMC}})$ to approximate $\mathrm{E}(g(\boldsymbol{x})) = \int g(\boldsymbol{x})\pi(\boldsymbol{x})d\boldsymbol{x}$ (MCMC approximation (1.5)). To this end, the efficiency of an MCMC algorithm can be quantified by the precision of the above approximation, that is, by the variance of $\hat{f}_{\text{MCMC}} := \frac{1}{S}\sum_{s=1}^{S} g(\boldsymbol{x}^{(s)}_{\text{MCMC}})$ regarded as

an estimator of $\mathrm{E}(g(\boldsymbol{x})) = \int g(\boldsymbol{x})\pi(\boldsymbol{x})d\boldsymbol{x}$. Assuming that the sampled chain is in stationarity, a few lines of algebra (see e.g. Kypraios (2007, section 1.10)) yield that

$$\mathrm{var}(\hat{f}_{\mathrm{MCMC}}) = \mathrm{var}(\hat{f}_{\mathrm{MC}}) \left(1 + 2\sum_{k=1}^{S-1}\left(1 - \frac{k}{S}\right)\rho_k\right), \tag{1.6}$$

where $\rho_k = \mathrm{corr}\left(g(\boldsymbol{x}^{(1)}_{\mathrm{MCMC}}), g(\boldsymbol{x}^{(1+k)}_{\mathrm{MCMC}})\right)$ is the autocorrelation of the Markov chain at lag-k and $\mathrm{var}(\hat{f}_{\mathrm{MC}}) = \frac{\mathrm{var}(g(\boldsymbol{x}))}{S}$ is the variance of $\hat{f}_{\mathrm{MC}} := \frac{1}{S}\sum_{s=1}^{S} g(\boldsymbol{x}^{(s)}_{\mathrm{MC}})$, the estimator of $\mathrm{E}(g(\boldsymbol{x})) = \int g(\boldsymbol{x})\pi(\boldsymbol{x})d\boldsymbol{x}$ based on the 'gold-standard' MC approximation (1.4), with associated i.i.d. sample $\{\boldsymbol{x}^{(1)}_{\mathrm{MC}}, \boldsymbol{x}^{(2)}_{\mathrm{MC}}, \ldots, \boldsymbol{x}^{(S)}_{\mathrm{MC}}\}$. Equation (1.6) reveals that the variance of $\hat{f}_{\mathrm{MCMC}}$ is equal to the variance of $\hat{f}_{\mathrm{MC}}$ plus a generally positive term that depends on the autocorrelation among the sampled values of the Markov chain. This implies that the higher the autocorrelation in the chain, the larger the variance of $\hat{f}_{\mathrm{MCMC}}$ and the less precise the MCMC approximation is; notice that quantifying the efficiency of an MCMC algorithm via the variance of $\hat{f}_{\mathrm{MCMC}}$ is in line with the intuitive description of algorithm efficiency with respect to mixing, given in the paragraph above, in the sense that a chain with high autocorrelation not only yields large variance for $\hat{f}_{\mathrm{MCMC}}$ but also has poor mixing.

In practice, to measure the amount of autocorrelation there is in the chain one can estimate the autocorrelations at lag-k, ρ_k, by calculating the corresponding sample autocorrelations at lag-k, denoted as $\hat{\rho}_k$. Typically this information is then presented by a plot, referred to as an *autocorrelation function (ACF) plot*, of $\hat{\rho}_k$ against k. Another way to assess how much autocorrelation there is in the chain, is by the *effective sample size* of an MCMC chain, which following Robert and Casella (2004) and Gelman et al. (2013), is defined as $S_{\mathrm{eff}} = S\frac{\mathrm{var}(\hat{f}_{\mathrm{MC}})}{\mathrm{var}(\hat{f}_{\mathrm{MCMC}})}$ so that it is interpreted as the number of i.i.d. MC sampled values required to give the same precision as the MCMC sample in question. From equation (1.6), it can be seen that $S_{\mathrm{eff}} = \frac{S}{\tau}$, where $\tau = 1 + 2\sum_{k=1}^{S-1}\left(1 - \frac{k}{S}\right)\rho_k$, and in practice S_{eff} can be estimated as $\hat{S}_{\mathrm{eff}} = \frac{S}{\hat{\tau}}$, where $\hat{\tau}$

is an estimate of τ based on the obtained MCMC sample (see Gelman et al. (2013, section 11.5) for details on how $\hat{\tau}$ is calculated).

1.3.2.4 Algorithms

Recall from section 1.3.2.2 that, given a target density $\pi(\boldsymbol{x})$, corresponding to a random vector $\boldsymbol{x} \in \mathcal{X}$, the purpose of an MCMC algorithm is to construct an ergodic discrete-time Markov chain having as state space $\mathcal{X}$ and as stationary distribution the distribution associated with the target density $\pi(\boldsymbol{x})$. In general, having chosen an initial value for the chain, say $\boldsymbol{x}^{(1)}$, an MCMC algorithm is defined by specifying the mechanism with which the Markov chain transitions to the next state $\boldsymbol{x}^{(s+1)}$, given its current state $\boldsymbol{x}^{(s)}$, i.e. an MCMC algorithm is defined iteratively by specifying how $\boldsymbol{x}^{(s+1)}$ is generated from $\boldsymbol{x}^{(s)}$. The only requirement is that this is done is such a way that the constructed Markov chain is ergodic and has the desired stationary distribution. The two most commonly used MCMC algorithms, also the MCMC algorithms used in this thesis, are the *Gibbs sampler* and the *Metropolis-Hastings (MH) algorithm*, which are described right below.

Gibbs sampler

Idea and procedure Suppose that $\mathcal{X}$ is such that $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \ldots \times \mathcal{X}_n$ so that $\boldsymbol{x} \in \mathcal{X}$ is decomposed into n components as $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$, $x_k \in \mathcal{X}_k$, $k = 1, 2, \ldots, n$. In the simplest of cases each component x_k is 1-dimensional but in general x_k might itself be multidimensional. The idea behind Gibbs sampling requires that one is able to sample from the *full conditional distribution* of each component x_k, $k = 1, 2, \ldots, n$, that is, the conditional distribution of x_k given the values of all the other components. Let $\pi(x_k \mid x_1, x_2, \ldots x_{k-1}, x_{k+1}, \ldots, x_n)$ denote the p.d.f. associated with the full conditional distribution of x_k, $k = 1, 2, \ldots, n$. The Gibbs sampler iteratively generates the next state, $\boldsymbol{x}^{(s+1)}$, given the current state, $\boldsymbol{x}^{(s)}$, according to Algorithm 1.

Algorithm 1 Gibbs sampler

1. Suppose the current state is $\boldsymbol{x}^{(s)} = (x_1^{(s)}, x_2^{(s)}, \ldots, x_n^{(s)})$
2. Sample $x_1^{(s+1)} \sim \pi(x_1 \mid x_2^{(s)}, x_3^{(s)} \ldots, x_n^{(s)})$
3. Sample $x_2^{(s+1)} \sim \pi(x_2 \mid x_1^{(s+1)}, x_3^{(s)} \ldots, x_n^{(s)})$
4. $\vdots$
5.
6. Sample $x_n^{(s+1)} \sim \pi(x_n \mid x_1^{(s+1)}, x_2^{(s+1)} \ldots, x_{n-1}^{(s+1)})$
7. Set the next state as $\boldsymbol{x}^{(s+1)} = (x_1^{(s+1)}, x_2^{(s+1)}, \ldots, x_n^{(s+1)})$.

Why it works The intuitive explanation of why the Gibbs sampler constructs Markov chains that have the desired stationary distribution $\pi(\boldsymbol{x}) = \pi(x_1, x_2, \ldots, x_n)$, is based on the fact that knowledge of the conditional distributions is enough to determine a joint distribution (Casella and George, 1992). To see this more clearly consider the case that $\boldsymbol{x} = (x_1, x_2)$ is 2-dimensional and the desired stationary distribution is $\pi(\boldsymbol{x}) = \pi(x_1, x_2)$. Suppose that $(x_1^{(s)}, x_2^{(s)})$ is sampled from $\pi(x_1, x_2)$. To 'convince' ourselves that $\pi(x_1, x_2)$ is indeed the stationary distribution of the chain, $(x_1^{(s+1)}, x_2^{(s+1)})$ must also be sampled from $\pi(x_1, x_2)$. Since $(x_1^{(s)}, x_2^{(s)})$ is a value sampled from $\pi(x_1, x_2)$, the joint distribution of (x_1, x_2), $x_2^{(s)}$ can be seen as a value sampled from $\pi(x_2)$, the marginal distribution of x_2. The Gibbs sampler (see Algorithm 1) samples $x_1^{(s+1)}$ from $\pi(x_1 \mid x_2^{(s)})$, the full conditional distribution of x_1. Using the fact that $\pi(x_1, x_2)$ can be expressed as $\pi(x_1, x_2) = \pi(x_1 \mid x_2)\pi(x_2)$, the pair $(x_1^{(s+1)}, x_2^{(s)})$ can be seen as a value sampled from $\pi(x_1, x_2)$. In turn, $x_1^{(s+1)}$ can be seen as a sample from $\pi(x_1)$, the marginal distribution of x_1. The Gibbs sampler (see Algorithm 1) proceeds to sample $x_2^{(s+1)}$ from $\pi(x_2 \mid x_1^{(s+1)})$, the full conditional distribution of x_2. The fact that $\pi(x_1, x_2)$ can also be expressed as $\pi(x_1, x_2) = \pi(x_2 \mid x_1)\pi(x_1)$ implies that the pair $(x_1^{(s+1)}, x_2^{(s+1)})$ can indeed be seen as a value sampled from $\pi(x_1, x_2)$. For a more rigorous treatment one is referred to

the relevant ergodic results regarding the Gibbs sampler (see e.g. Robert and Casella (2004, chapter 10)).

Metropolis-Hastings algorithm

Idea and procedure The idea of the MH algorithm involves a *proposal distribution* according to which candidate moves are proposed. This proposal distribution generally depends on the current state (although this is not necessary; see the one after the next paragraph) and can be chosen arbitrarily as long as the constructed Markov chain is ergodic and has the desired stationary distribution. More specifically, the MH algorithm works as follows. Given a current state $\boldsymbol{x}^{(s)}$, the algorithm proposes a candidate next value $\boldsymbol{x}^*$ from a proposal density $q(\boldsymbol{x} \mid \boldsymbol{x}^{(s)})$. Then, with probability $1 \wedge r$, where $r = \frac{\pi(\boldsymbol{x}^*)}{\pi(\boldsymbol{x}^{(s)})} \times \frac{q(\boldsymbol{x}^{(s)}|\boldsymbol{x}^*)}{q(\boldsymbol{x}^*|\boldsymbol{x}^{(s)})}$ is the acceptance ratio and where $a \wedge b$ denotes the minimum of a and b, $\boldsymbol{x}^*$ is accepted and the next state becomes $\boldsymbol{x}^{(s+1)} = \boldsymbol{x}^*$, otherwise $\boldsymbol{x}^*$ is rejected and $\boldsymbol{x}^{(s+1)} = \boldsymbol{x}^{(s)}$, i.e. the chain does not move. Algorithm 2 collects the steps of the above procedure.

Algorithm 2 Metropolis-Hastings algorithm

1. Suppose the current state is $\boldsymbol{x}^{(s)}$
2. Generate $\boldsymbol{x}^* \sim q(\boldsymbol{x} \mid \boldsymbol{x}^{(s)})$
3. Compute the acceptance ratio $r = \frac{\pi(\boldsymbol{x}^*)}{\pi(\boldsymbol{x}^{(s)})} \times \frac{q(\boldsymbol{x}^{(s)}|\boldsymbol{x}^*)}{q(\boldsymbol{x}^*|\boldsymbol{x}^{(s)})}$
4. Set the next state as $\boldsymbol{x}^{(s+1)} = \boldsymbol{x}^*$ with probability $1 \wedge r$; otherwise set the next state as $\boldsymbol{x}^{(s+1)} = \boldsymbol{x}^{(s)}$.

Why it works An intuition on why the MH algorithm targets the density $\pi(\boldsymbol{x})$ as desired can be gauged from the expression of the acceptance ratio $r = \frac{\pi(\boldsymbol{x}^*)}{\pi(\boldsymbol{x}^{(s)})} \times \frac{q(\boldsymbol{x}^{(s)}|\boldsymbol{x}^*)}{q(\boldsymbol{x}^*|\boldsymbol{x}^{(s)})}$. The first factor in the expression of r, namely the posterior ratio $\frac{\pi(\boldsymbol{x}^*)}{\pi(\boldsymbol{x}^{(s)})}$, dictates that the more probable a proposed value is compared to the current value,

with respect to $\pi(\boldsymbol{x})$, the higher the probability of acceptance; therefore the sampler explores the desired regions of the state space. The second factor, namely the proposal ratio $\frac{q(\boldsymbol{x}^{(s)}|\boldsymbol{x}^*)}{q(\boldsymbol{x}^*|\boldsymbol{x}^{(s)})}$, can be seen as a correction factor adjusting for the fact that, according to the proposal distribution, some values might be more likely to be proposed than others. Just like in the case of the Gibbs sampler, one is referred to the relevant literature for a more rigorous description of the ergodic properties of the MH algorithm (see e.g. Robert and Casella (2004, chapter 7)).

Different versions of the Metropolis Hastings algorithm Different forms of proposal distributions lead to different versions of the MH algorithm. For example, the *Metropolis algorithm* is a special case of the MH algorithm for which the proposal distribution is symmetric, that is, the proposal distribution is such that $q(\boldsymbol{x}^* \mid \boldsymbol{x}^{(s)}) = q(\boldsymbol{x}^{(s)} \mid \boldsymbol{x}^*)$ for all $\boldsymbol{x}^{(s)}$ and $\boldsymbol{x}^*$. For the Metropolis algorithm the acceptance ratio reduces to $r = \frac{\pi(\boldsymbol{x}^*)}{\pi(\boldsymbol{x}^{(s)})}$. Another example is when the proposal distribution does not depend on the current value, in which case $q(\boldsymbol{x}^* \mid \boldsymbol{x}^{(s)}) = q(\boldsymbol{x}^*)$ for all $\boldsymbol{x}^{(s)}$ and $\boldsymbol{x}^*$, and the acceptance ratio is given by $r = \frac{\pi(\boldsymbol{x}^*)}{\pi(\boldsymbol{x}^{(s)})} \times \frac{q(\boldsymbol{x}^{(s)})}{q(\boldsymbol{x}^*)}$. This algorithm is called the *independent sampler* or the *independence Metropolis algorithm*.

Dependent and independent proposal distributions Although in principle the proposal distribution can be chosen arbitrarily, only subject to satisfying the required ergodic properties, in practice the specific choice of the proposal distribution will determine chain mixing and algorithm efficiency. The task of choosing a good proposal distribution (i.e. a proposal distribution that produces a good mixing chain and an efficient algorithm) is problem-specific and not always a straightforward task; in some instances designing a good proposal distribution is somewhat of an art form. Nonetheless, there are some generally desirable features, that a proposal distribution should have, which help guide this choice. These features are different for the case that the proposal distribution depends on the current state (as it is for the general MH algorithm and the Metropolis algorithm) and for the case that it does not (as it is

for the independence Metropolis algorithm). To simplify wording, for the remainder of this thesis, the former type of proposal distributions are referred to as *dependent proposals* and the latter as *independent proposals*.

Dependent proposals typically work by centering themselves around the current value $\boldsymbol{x}^{(s)}$ and proposing moves around it. The simplest example is a Normal distribution, $N(\boldsymbol{x}^{(s)}, \sigma^2)$, with mean $\boldsymbol{x}^{(s)}$ and some variance σ^2. As explained in Gilks et al. (1996), a dependent proposal should be designed so that the proposed steps are neither too small (a proposal distribution generating too small steps will typically have a high acceptance proportion but will nevertheless mix slowly as the chain will only transition between nearby states), nor too large (a proposal distribution generating too large steps will frequently get stuck in one location as it will often propose moves from the body to the tails of the target density, which will typically be rejected). To see this more clearly, consider the example of the Metropolis algorithm, where the acceptance ratio is given by $r = \frac{\pi(\boldsymbol{x}^*)}{\pi(\boldsymbol{x}^{(s)})}$. If the step size is too small, i.e. $\boldsymbol{x}^*$ is very close to $\boldsymbol{x}^{(s)}$, then $\pi(\boldsymbol{x}^*) \approx \pi(\boldsymbol{x}^{(s)})$ and $r \approx 1$. Conversely, if the step size is too large, such as when attempting to move from a state $\boldsymbol{x}^{(s)}$ near a mode to a state $\boldsymbol{x}^*$ near the tails, $\pi(\boldsymbol{x}^*) \ll \pi(\boldsymbol{x}^{(s)})$ and $r \approx 0$. The step size of a proposal distribution, or more generally its scaling, is often controlled by one of its parameters; for the aforementioned $N(\boldsymbol{x}^{(s)}, \sigma^2)$ example, the scaling parameter is the variance σ^2. Such parameters are referred to as *tuning parameters* as they essentially control the efficiency of the algorithm. Roberts and Rosenthal (2001) showed that, for a class of dependent proposal distributions (which include Gaussian proposal distributions), optimal algorithm efficiency can be achieved for a certain acceptance proportion; 0.44 for 1-dimensional and 0.234 for multidimensional proposal distributions. Utilizing this result, in practice, tuning parameters can be set so that the algorithm in question yields acceptance proportion close to the reference optimal proportion.

Being not dependent on (and in particular being not centered around) the current

value, independent proposals attempt to explore the target space in a fundamentally different manner than dependent proposals. Independent proposals are in a sense 'bolder' than dependent proposals and can end up working very well or very badly. More specifically, while a dependent proposal attempts a guided exploration of the target space, an independent proposal attempts to move to any region of the space in one jump ignoring where the chain is at the current time. This suggests that for an independent proposal to work well it should resemble the target distribution and in turn, since the acceptance ratio is given by $r = \frac{\pi(\boldsymbol{x}^*)}{\pi(\boldsymbol{x}^{(s)})} \times \frac{q(\boldsymbol{x}^{(s)})}{q(\boldsymbol{x}^*)}$, that it should have a high acceptance proportion; in fact the 'ideal' choice for an independent proposal would be the (unknown) target density $\pi(\boldsymbol{x})$ itself, in which case $r = 1$, and the algorithm would reduce to i.i.d. sampling from $\pi(\boldsymbol{x})$. However, as mentioned in Gilks et al. (1996), it is safer if, in addition to being similar to, the proposal distribution has heavier tails than the target distribution. To see this, consider the case that the proposal distribution has lighter tails than the target distribution. The first problem that might occur in such case, is that moves to the tails of the target distribution are not proposed, during the finite number of iterations that the chain is run; thus resulting to regions of the target space remaining unexplored. The second problem is that, if and when the chain does visit the tails of the target distribution it will be difficult for it to then leave; if the current state $\boldsymbol{x}^{(s)}$ is at the tails of $\pi(\boldsymbol{x})$, since a proposed value $\boldsymbol{x}^*$ will most likely not be at the tails of $\pi(\boldsymbol{x})$, it will be typical that $\frac{\pi(\boldsymbol{x}^*)}{q(\boldsymbol{x}^*)} \ll \frac{\pi(\boldsymbol{x}^{(s)})}{q(\boldsymbol{x}^{(s)})}$ giving acceptance ratio $r \approx 0$. Heavy-tailed independent proposals help to avoid such problems at the expense of a lower acceptance proportion.

General MCMC algorithm Although the Gibbs sampler and the MH algorithm are MCMC algorithms in their own right, in most practical cases they are used as building blocks of more general MCMC algorithms. A general procedure for MCMC implementation, followed typically in practice and for all MCMC inferences in this thesis, is as follows. First, the state space $\mathcal{X}$ is written as $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \ldots \times \mathcal{X}_n$ so that $\boldsymbol{x} \in \mathcal{X}$ is decomposed into n (in general multidimensional) components as

$\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$, $x_k \in \mathcal{X}_k$, $k = 1, 2, \ldots, n$. The specific way that $\boldsymbol{x}$ is divided into components is often naturally suggested by the model in question or motivated by reasons of computational convenience. Then, a Gibbs sampler framework is set up, so that in an MCMC iteration, components are updated one by one, according to their full conditional distributions. Components which have standard full conditional distributions are typically (although not necessarily) updated by directly sampling from their full conditional distribution, in a so-called *Gibbs step*. Otherwise, if the full conditional distribution is non-standard and direct sampling from it is not possible, a MH algorithm (or some other MCMC algorithm) is used to target the full conditional distribution and generate a value from it; when performed using a MH algorithm, such step is referred to as a *MH step*. All steps of the procedure are depicted in Algorithm 3. Note that, this general MCMC algorithm is the same as a Gibbs sampler but more general in the sense that individual components can be updated using any appropriate MCMC algorithm and not necessarily using a Gibbs step.

Algorithm 3 General MCMC algorithm

1. Suppose the current state is $\boldsymbol{x}^{(s)} = (x_1^{(s)}, x_2^{(s)}, \ldots, x_n^{(s)})$
2. Generate $x_1^{(s+1)}$ according to $\pi(x_1 \mid x_2^{(s)}, x_3^{(s)} \ldots, x_n^{(s)})$ using a Gibbs step or a MH step (or some other MCMC algorithm)
3. Generate $x_2^{(s+1)}$ according to $\pi(x_2 \mid x_1^{(s+1)}, x_3^{(s)} \ldots, x_n^{(s)})$ using a Gibbs step or a MH step (or some other MCMC algorithm)

$\vdots$

4.
5. Generate $x_n^{(s+1)}$ according to $\pi(x_n \mid x_1^{(s+1)}, x_2^{(s+1)} \ldots, x_{n-1}^{(s+1)})$ using a Gibbs step or a MH step (or some other MCMC algorithm)
6. Set the next state as $\boldsymbol{x}^{(s+1)} = (x_1^{(s+1)}, x_2^{(s+1)}, \ldots, x_n^{(s+1)})$.

Remarks This section concludes by collecting some useful remarks regarding MCMC algorithms.

There is a sense that Gibbs sampling is preferable to MH since it uses additional information about the target density, namely the full conditional distributions. However, it is not always possible to use Gibbs sampling since the full conditional distributions might be such that it is hard to sample from. Also, the Gibbs sampling does not allow much control over the mixing of the Markov chain, unlike the MH algorithm where the mixing can be adjusted by careful selection of tuning parameters. It is also worth mentioning that the Gibbs sampling can be seen as a special case of the MH algorithm where the proposal distribution is the full conditional distribution and the acceptance probability is always equal to 1 (see e.g. Hoff (2009, section 10.4)).

As mentioned in Gilks et al. (1996), when components of $\boldsymbol{x}$ are highly correlated in the target distribution, mixing can be slow. To this end, in an attempt to improve mixing, one might choose to block such components into one higher-dimensional component and update them together in a so-called *block update step*. It is also possible to repeat the update step of a slow mixing component several times in an MCMC iteration and only record the last value. This procedure is very similar in nature to thinning and it is again done with the purpose of improving slow mixing by allowing the chain to move around the target space.

When updating a component according to its full conditional distribution, as is done in the Gibbs sampling and the general MCMC algorithm described above, it is not necessary for its full conditional distribution to depend on all of the remaining parameters, i.e. it could be the case that the component in question is conditionally independent to some components given the rest of the components. For clarity, the approach taken throughout this thesis is to drop such nominal dependencies in the notation of full conditional distributions.

1.3.3 Posterior predictive checking

1.3.3.1 Rationale, procedure and implementation

An essential part of any responsible data analysis is the assessment of the model's fit to the data in question. An intuitive, natural and potentially very useful way to assess a model's fit, within a Bayesian framework, is via *posterior predictive checking*. The general idea of posterior predictive checking is that replicated data generated under the model should look similar to the observed data, i.e. the observed data should look plausible under the posterior predictive distribution (Gelman et al., 2013). Let $\boldsymbol{y}^{obs}$ denote the observed data, $\pi(\boldsymbol{y} \mid \boldsymbol{\theta})$ the sampling density of an assumed model with parameter $\boldsymbol{\theta}$ (both $\boldsymbol{y}^{obs}$ and $\boldsymbol{\theta}$ are vectors in general) and $\pi(\boldsymbol{\theta} \mid \boldsymbol{y}^{obs})$ the posterior density of $\boldsymbol{\theta}$. Formally, the replicated data $\boldsymbol{y}^{rep}$, are data that are generated from the sampling density of the model $\pi(\boldsymbol{y} \mid \boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is averaged over the posterior density $\pi(\boldsymbol{\theta} \mid \boldsymbol{y}^{obs})$; that is the *posterior predictive density*, given by

$$\pi(\boldsymbol{y}^{rep} \mid \boldsymbol{y}^{obs}) = \int \pi(\boldsymbol{y}^{rep} \mid \boldsymbol{\theta})\pi(\boldsymbol{\theta} \mid \boldsymbol{y}^{obs})d\boldsymbol{\theta}. \tag{1.7}$$

The aspects of the data for which assessment is desired can be represented by test statistics. As in the classical case, a *test statistic* T is a scalar function of data (observed or replicated) $\boldsymbol{y}$. Plugging in $\boldsymbol{y} = \boldsymbol{y}^{obs}$ the test statistic assumes its observed value $T(\boldsymbol{y}^{obs})$. For $\boldsymbol{y} = \boldsymbol{y}^{rep} \sim \pi(\boldsymbol{y}^{rep} \mid \boldsymbol{y}^{obs})$ the test statistic is a random variable, under the posterior predictive distribution, called the replicated variable. To simplify notation, let $T^{obs} := T(\boldsymbol{y}^{obs})$ and $T^{rep} := T(\boldsymbol{y}^{rep})$. Assessment, for the aspect of the data represented by T, is conducted (quantitatively and visually) by comparing the posterior predictive distribution of T^{rep} to its observed value T^{obs}.

Quantitatively, one can calculate the *posterior predictive p-value* (ppp-value) which is the probability that the replicated variable T^{rep} is more extreme than the observed

value T^{obs} and it is given by

$$\begin{aligned} \text{ppp-value} &= P(T^{rep} \leq T^{obs} \mid \boldsymbol{y}^{obs}) \\ &= \mathrm{E}(\mathbb{1}_{\{T^{rep} \leq T^{obs}\}} \mid \boldsymbol{y}^{obs}) = \int \mathbb{1}_{\{T^{rep} \leq T^{obs}\}} \pi(\boldsymbol{y}^{rep} \mid \boldsymbol{y}^{obs}) d\boldsymbol{y}^{rep}. \end{aligned} \tag{1.8}$$

Extreme ppp-values (close to 0 or 1) imply evidence for lack of fit, whereas values near 0.5 indicate good fit, for the aspect of the data in context (Gilks et al., 1996; Gelman et al., 2013).

In most practical cases the posterior and the posterior predictive distributions are not known analytically (see section 1.3.1.3) and thus simulations are used to approximate the ppp-value. Suppose that a sample $\{\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, \ldots, \boldsymbol{\theta}^{(S)}\}$ has been drawn from the posterior distribution (using a method such as MCMC). For each posterior value $\boldsymbol{\theta}^{(s)}$, a replicated dataset $\boldsymbol{y}^{rep^{(s)}}$ can be simulated from the sampling density of the model $\pi(\boldsymbol{y} \mid \boldsymbol{\theta}^{(s)})$, $s = 1, 2, \ldots, S$. It is clear from equation (1.7) that $\{\boldsymbol{y}^{rep^{(1)}}, \boldsymbol{y}^{rep^{(2)}}, \ldots, \boldsymbol{y}^{rep^{(S)}}\}$ constitutes a sample from the posterior predictive distribution; thus $\{T^{rep^{(1)}}, T^{rep^{(2)}}, \ldots, T^{rep^{(S)}}\}$ is a sample from the posterior predictive distribution of T^{rep}. Then the ppp-value can be calculated using MC approximation as:

$$\text{ppp-value} = \int \mathbb{1}_{\{T^{rep} \leq T^{obs}\}} \pi(\boldsymbol{y}^{rep} \mid \boldsymbol{y}^{obs}) d\boldsymbol{y}^{rep} \approx \frac{1}{S} \sum_{s=1}^{S} \mathbb{1}_{\{T^{rep^{(s)}} \leq T^{obs}\}}, \tag{1.9}$$

which is simply the proportion of simulations for which a sampled replicated value $T^{rep^{(s)}} := T(\boldsymbol{y}^{rep^{(s)}})$, $s = 1, 2, \ldots, S$, does not exceed the observed value T^{obs}. Visually, the observed value T^{obs} can be imposed on the histogram of sampled replicated values $\{T^{rep^{(1)}}, T^{rep^{(2)}}, \ldots, T^{rep^{(S)}}\}$; an observed value near the middle of the histogram would indicate good fit (Gilks et al., 1996).

1.3.3.2 Purpose and terminology

The purpose of doing model assessment using the posterior predictive distribution is to assess if the model fits the data (Gelman et al., 2013). More precisely, the interest is in assessing the practical usefulness of a model, by revealing settings where the model's predictions are inconsistent with observed data, and not testing whether the model is true or not; in practice all models are wrong (misspecified) and would be rejected given enough data (Gelman et al., 1996). Put simply, if a model is wrong, but has a genuine ability to fit the data, then it is not desirable for the test to reject (Gelman et al., 2013). Taking these into consideration, whenever terms such as 'test', 'accept', 'reject', 'power', 'type I error' and 'type II error' are used in the context of posterior predictive checking, they would not have the same meaning as in the classical setting; in the posterior predictive context, *test* would mean assess/assessment, *accept* would mean failure to detect lack of fit, *reject* would mean detect lack of fit, *power* would mean ability to detect lack of fit when there is lack of fit, *type I error* would mean detecting lack of fit when there is not lack of fit, and *type II error* would mean failure to detect lack of fit when there is lack of fit.

1.3.3.3 Test quantities and unknown variables

For some problems, it is difficult or unnatural to represent aspects of a model via functions of the data alone, and it would be preferable to use functions of both data and parameters (Gelman et al., 2013). In a Bayesian context, this is possible by allowing dependence on model parameters (and any unknown quantities such as unobserved data) under their posterior distribution; such functions are called *test quantities* and they are the Bayesian generalization of classical test statistics. The fact that test quantities allow dependence on any unknown quantity is very appealing in theory, especially when unobserved data are present. However, test quantities that depend largely on parameters or unobserved data are difficult to interpret and their

power is greatly reduced by the amount of information that is imputed from the model itself (Gelman, 2013).

1.3.3.4 Interpretating ppp-values

The interpretation of ppp-values follows from the purpose and philosophy of posterior predictive checking. That is, ppp-values are interpreted directly as probabilities over the posterior predictive distribution; although, as discussed in section 1.3.3.2 not as $P(\text{model is true} \mid \text{oberved data})$ (Gelman et al., 2013). Extreme ppp-values (close to 0 or 1) imply that the discrepancy, between model and data, can not be reasonably explained by chance and the model can not be expected to capture the aspect of the data in context (Gilks et al., 1996; Gelman et al., 2013). A close to optimal ppp-value (close to 0.5) should be interpreted as an indication that the model adequately captures the specific aspect of the data being investigated. Two important comments should be made regarding the last statement. First, it is not necessary that a good ppp-value implies goodness of fit for the aspect of the data under assessment. One such example is described by Gelman (2013), where the very noisy posterior predictive distribution undesirably traps the sampling distribution of ppp-values around the optimal value. In such cases, a good ppp-value is not an indication of a genuine ability of the model to capture the underlying aspect of the data but rather a reflection of the degraded power of the test to detect lack of fit. Second, even if a model fits a specific aspect of the data, that does not necessarily mean that the model is a good fit overall; there might be other aspects of the data, which might not even be assessed, that the model fails to capture.

1.3.4 Stochastic modelling of epidemic data

This section provides a brief overview (types of models, types of data and inference methods) of the area of infectious disease modelling while highlighting the specific assumptions formulating the framework adopted in this thesis. Since epidemic

modelling is a subject of an enormous and diverse literature, this overview is by no means extensive and mostly follows the review of O'Neill (2010); for a more comprehensive introduction see e.g. Daley and Gani (1999) or Andersson and Britton (2000).

1.3.4.1 Types of models

Terminology Epidemic models are concerned with a population consisting of individuals who can potentially transmit the disease to one another and are typically defined at an individual level, by describing the health of each individual with respect to the disease in question, using disease-development states. An individual who is at risk of contracting the disease is referred to as *susceptible*, one who has the disease but is not able to infect others (i.e. infected but not infectious) is called *exposed*, one who can transmit the disease is called *infective* and one who can no longer transmit the disease but is also not susceptible is called *removed*. The removal state might correspond to different things in practice such as immunity, death or the appearance of symptoms (case detection) which make an individual too ill to continue interacting with the population as usual. Nonetheless, the common characteristic of all removed individuals is that they play no part in the spread of the epidemic.

The basic states, susceptible (S), exposed (E), infective (I) and removed (R), along with the order in which individuals transition between them, are used in abbreviation to describe models. For example, an SIR ($S \to I \to R$) model assumes that individuals progress from being susceptible to being infective without an exposed period and upon the end of their infectious period they become removed. On the contrary, an SEIR ($S \to E \to I \to R$) model additionally assumes that an individual first goes through an exposed period, before being infectious, whereas an SIS ($S \to I \to S$) model is one which individuals are never removed and can become reinfected right after the end of their infectious period. In this thesis, only SIR models are considered although most of the methods developed could naturally be extended to SEIR models.

Deterministic and stochastic models Mathematical models for infectious disease transmission can broadly be divided into two categories, namely *deterministic models* and *stochastic models* (Andersson and Britton, 2000). Modelling epidemics deterministically, assumes that, given some initial conditions, the progression of the outbreak is determined (i.e. the progression is not random). Such an assumption might be useful to gain an understanding of the, loosely called, expected behaviour of an epidemic and could be particularly effective in instances of large population outbreaks, where, roughly put, the amount of stochasticity is reduced by law-of-large-number-type behaviour (see e.g. Andersson and Britton (2000, chapter 5)). However, as real-life outbreaks are inherently random, stochastic models are arguably more naturally suited to capture their features e.g. they allow for the possibility to have a minor outbreak, infecting only a few individuals, or a major outbreak, infecting a fairly large proportion of individuals, and the calculation of the probability that each of these two events occurs (see e.g. Andersson and Britton (2000, theorem 3.1)). This thesis focuses only on stochastic epidemic models.

Homogeneity, heterogeneity and other modelling assumptions The most basic models of disease transmission assume *homogeneity at the population level*, meaning that all individuals in the population mix together at random Andersson and Britton (2000, chapter 2). In terms of disease transmission, this means that a given infective is equally likely to infect any of the currently susceptible individuals. Such models typically assume *homogeneity at the individual level* as well, in the sense that there is no variation between individuals in, for example, the distribution of the time spent in the exposed or the infective state. In many contexts homogeneity assumptions are unrealistic and more elaborate models are used which introduce *heterogeneity* in the population. This can be done in various ways. For example, the population mixing structure can be modelled explicitly by dividing the population into groups such as households, schools or workplaces and assuming that individuals mix at different rates within and between groups (see e.g. Ball et al. (1997); Britton

et al. (2011)). Alternatively, heterogeneities might also be introduced by including individual-level covariate information (see e.g. Kypraios (2007); Jewell et al. (2009)) or by incorporating the spatial structure of the population (see e.g. Jewell et al. (2009); Retkute et al. (2018)).

Other assumptions include modelling in discrete time (for example using chain-binomial models; see e.g. Andersson and Britton (2000) and the references therein) or continuous time (see e.g. Bailey (1975); Becker (1989); Andersson and Britton (2000)), modelling the outbreak after it has ceased or while it is ongoing (see e.g. O'Neill and Roberts (1999)) and modelling closed or open populations, i.e. assuming whether or not individuals may enter or leave the population via e.g. births, deaths or migration (see e.g. O'Neill (1996); Clancy et al. (2001)). All models in this thesis are continuous time, deal with closed populations and ceased epidemics and are homogeneous at the individual level; in particular they make no use of covariates and the infectious periods of individuals are independent and identically distributed. Regarding the population mixing assumption, both homogeneous and heterogeneous models are considered.

Infectious period distribution Individual infectious periods are typically assumed to be independent and identically distributed. For the purposes of this thesis three choices are considered for the infectious period distribution, namely the Exponential distribution, the Gamma distribution and the constant distribution (meaning that infectious periods of individuals are assumed to be constant and of the same length). Along with the Weibull distribution (see e.g. Streftaris and Gibson (2004a)), these choices are the most commonly used in the literature; for the Exponential see e.g. O'Neill and Roberts (1999), for the Gamma see e.g. Xiang and Neal (2014) and for the constant see e.g. Clancy and O'Neill (2008).

Using exponentially distributed infectious periods essentially means that there is no

typical length for infectious periods; most of them are short, a few are long and fewer are extremely long. Although this assumption is not very realistic for many diseases, it still remains very appealing due to mathematical convenience e.g. it is a well known result that by assuming Exponential infectious periods, one can approximate the initial stages of a standard SIR model (see section 1.3.5.5 for the definition of the standard SIR model) using a linear birth-death process (Kendall, 1956). The Gamma distribution offers more flexibility and thus is more appropriate for most practical cases but the caveat is that (unless one of its parameters is assumed known), when fitted to data, it requires estimation of an additional parameter. Finally, the constant distribution is in most cases employed to reduce the computational cost of fitting the model; by assuming constant infectious periods, the unobserved infection times (see sections 1.3.4.2 and 1.3.4.3 for when infection times are not observed and how this complicates inference) are deterministically specified given removal times and a constant value for the infectious period.

1.3.4.2 Types of data

Temporal data and final size data Broadly speaking there are two types of epidemic data, namely *temporal data* and *final size data* (O'Neill, 2010; Britton et al., 2011). Final size data only contain snapshot information at the start and at the end of the epidemic but no temporal information regarding the disease propagation throughout it, i.e. they can provide knowledge of which (or how many) individuals were initially susceptible and which (or how many) of these individuals were infected by the end of the outbreak. Temporal data provide information on the state of individuals during the epidemic. Depending on the extent that they are observed and the assumptions made (see paragraph below) temporal data might correspond to infection times and/or removal times of individuals; data consisting of both infection and removal times are referred to as *complete temporal data* and data consisting only of removals times are referred to as *partial temporal data*. Typically, such data come in aggregated form, for instance by day or week (i.e. as time series consisting the

number of infections and/or removals per day or week) although for all methods developed in this thesis it makes no difference to treat these times as continuous.

Partial observation In practice, the actual process of disease transmission is rarely observed, which implies that infection times are typically not observed. More precisely, the most commonly encountered type of data in practice, are case detection times with the actual times of infection being unknown. In fact, this is almost always the case for human diseases, when the appearance of symptoms (case detection) is usually the first sign that an individual has been infected. By making the additional assumption, that symptomatic individuals are removed, either because they are too ill to continue interacting with the population, or via some other sort of isolation, case detection times correspond to removal times.

Considering that the interest in this thesis is in the temporal aspects of outbreaks (for both model assessment and inference purposes) and that such partially observed temporal data are common in practice, the approach taken throughout this thesis is to assume that observed data consist of removal times, with infection times missing, unless otherwise stated. It is noted that such a framework is very common in the stochastic epidemic modelling literature, being adopted in many classical references such as Becker (1989); Andersson and Britton (2000); O'Neill and Roberts (1999).

Not independent data As at any point in time the health state of any given individual depends on the health state of all other individuals in the population, epidemic data are highly dependent, and any fitted models should take this into account. Although one might consider any models that can incorporate dependencies (such as time series models) the clear advantage of using disease transmission models is that their parameters typically carry a meaningful epidemiological or biological interpretation. On this account, disease transmission models are used throughout this thesis.

1.3.4.3 Inference methods

For a stochastic epidemic model (as for any stochastic model fitted to data) most inferential methods rely on the likelihood.

Tractable likelihood In the cases of complete temporal data (i.e. both removal times and infection times are observed) it is typical that the model likelihood is *tractable*, meaning that it can be analytically derived (Rida, 1991). Given a tractable likelihood, inference can proceed along conventional lines, either frequentist or Bayesian, using tools such as maximum likelihood estimation and Markov chain Monte Carlo (MCMC) methods (see e.g. Andersson and Britton (2000, chapter 9) and Kypraios (2007)).

Intractable likelihood However, as discussed in section 1.3.4.2, in most practical cases, only case detection times are observed (corresponding to removal times) and infection times are missing. The complication that arises, in such a case of partial observation, is that typically the associated likelihood (i.e. the likelihood based on observing only removals and not infections) becomes *intractable* (see section 1.3.5.2 for more details).

Different remedies can be applied to the problem of intractable likelihoods, such as resorting to various simplifying model assumptions (e.g. SIR models assuming fixed and known infectious periods, so that removal data deterministically yield infection data) or using approximate Bayesian computation (ABC) methods, which are simulation-based methods that avoid likelihood calculation (see e.g. McKinley et al. (2009); Kypraios et al. (2017); McKinley et al. (2018)). Arguably though, the most well suited and widely used approach for dealing with intractable likelihoods, due to missing data, is via data augmentation in a Bayesian framework (see section 1.3.2.2 and the references therein). In the present epidemic context this approach is typically implemented by introducing the unobserved infection data as additional

variables, in such a way that Bayesian inference for the model parameters can be performed by computing an augmented likelihood based on removals and infections (see sections 1.3.5.2 and 1.3.5.3 and relevant parts of sections 1.3.5.5 to 1.3.5.7 for more details); for the development of the approach in the epidemic context see Gibson and Renshaw (1998); O'Neill and Roberts (1999) and for adaptations and extensions see e.g Streftaris and Gibson (2004b); Neal and Roberts (2005); Kypraios (2007); Jewell et al. (2009); Xiang and Neal (2014).

Under the partially observed temporal data framework adopted in this thesis (see section 1.3.4.2), all considered models have intractable likelihoods and all methods of inference for model parameters are conducted via data augmentation in a Bayesian framework, as outlined above.

1.3.5 Stochastic epidemic models considered in this thesis

This section introduces the stochastic epidemic models that are used in this thesis, namely the standard SIR model, the non-linear infection rate SIR model and the two-level-mixing SIR model. For clarity and ease of presentation the assumptions and features shared by all considered models, as highlighted throughout section 1.3.4, are collected to place the models under a common framework, i.e. a class of stochastic epidemic models is formulated, in which all three considered models belong. Initially, notation, model definition, likelihood derivation and information on Bayesian inference and MCMC methods, are provided under this general framework (i.e. for a general model of the class). Afterwards, the standard, the non-linear infection rate and the two-level-mixing SIR models are introduced as specific examples of the class; MCMC algorithms are provided for each model and some relevant (for the purposes of this work) model-specific features are presented.

It is noted that, the MCMC algorithms presented in this chapter are based on the

MCMC algorithms for epidemic models initially proposed in the literature, such as those in O'Neill and Roberts (1999); O'Neill and Becker (2001); Neal and Roberts (2004). All these algorithms share the common feature that, whenever relevant (that is whenever the infectious period distribution is not assumed to be constant in which case all the infection times are deterministically updated given a constant value for the infectious period), the infections are updated one at a time, typically using a MH step and a model-inspired independent proposal distribution. Alternative MCMC algorithms, having different proposal schemes for the infections, will be discussed in chapter 4.

1.3.5.1 Definition

Consider a closed population (i.e. individuals cannot enter or leave the population) consisting initially of N susceptible and m infectious individuals; for simplicity it is assumed that $m = 1$ but this assumption can easily be relaxed. The health of each individual, with respect to the disease in question, is described via a continuous time SIR model (see section 1.3.4.1). Specifically, at each time point t, any given individual belongs in one of three states, susceptible (S), infective (I) or removed (R), and can only transition from being susceptible to being infective (S $\rightarrow$ I) and from being infective to being removed (I $\rightarrow$ R); the transitions S $\rightarrow$ I and I $\rightarrow$ R are referred to as the *infection process* and the *removal process*, respectively. Let $\boldsymbol{\beta}$ and $\boldsymbol{\phi}$ be the parameter vectors associated with the infection process and the removal process, respectively. Let $\mathcal{X}_t$ and $\mathcal{Y}_t$ respectively be the set of susceptible and infective individuals in the population at time t. Consider also the number of susceptible and infective individuals at time t, denoted as X_t and Y_t, respectively. Finally, let $1, 2, \ldots, N+1$ be the labels of the $N+1$ individuals in the population. The model is defined by specifying the assumptions of the two possible transitions.

S $\rightarrow$ I Each individual k, $k = 1, 2, \ldots, N+1$, at each time point t, is subjected to contacts from the currently infective individuals $\mathcal{Y}_t$, at the time points of a non-homogeneous Poisson process of rate $h_k(t; \boldsymbol{\beta})$; $h_k(t; \boldsymbol{\beta})$ is the all-to-one infection rate. If a contacted individual is susceptible, at the time of contact, they instantly become infective. These Poisson processes are assumed to be mutually independent. The aggregation property of Poisson processes (see e.g. Ross (2009, proposition 5.4)) implies that overall infections occur according to a non-homogeneous Poisson process of rate $h(t; \boldsymbol{\beta}) = \sum_{k \in \mathcal{X}_t} h_k(t; \boldsymbol{\beta})$; $h(t; \boldsymbol{\beta})$ is the all-to-all infection rate.

I $\rightarrow$ R Upon infection an individual enters their infectious period, in which they remain until they becomes removed. The infectious periods of the individuals are assumed to be independent and identically distributed (i.i.d.) according to a random variable T_D, having distribution $D(\boldsymbol{\phi})$, with parameter vector $\boldsymbol{\phi}$. In principle, $D(\boldsymbol{\phi})$ can be any arbitrary but specified distribution.

All of the Poisson processes involved are independent of the infectious periods. The epidemic ends when no infectives are left in the population.

1.3.5.2 Likelihood

As mentioned in section 1.3.4.2, it is assumed that observed data consist only of removal times, with the infection times being unknown. It is also assumed (see section 1.3.4.1) that the epidemic has ceased and thus the total number of infections, denoted by n, is fixed and known and equals the total number of removals. The notation and derivation of the likelihood mostly follows Held et al. (2019, chapter 9) and Kypraios (2007) and is as follows. Individuals that are ultimately infected (ever-infected) are labelled $1, 2, \ldots, n$ and individuals that escape infection (never-infected) are labelled $n+1, n+2, \ldots, N+1$. Although not essential, the n ever-infected individuals are conveniently labelled according to the time-ordered removal times $r_1 < r_2 < \cdots < r_n$; so that individual with label 1 is removed first, individual with label 2 is removed

second and so on. For $k = 1, 2, \ldots, n$, let i_k be the (unknown) infection time of individual k, that is removed at time r_k and for $k = n+1, n+2, \ldots, N+1$, set $i_k = r_k = \infty$. Let α be the label of the initial infective so that $i_\alpha < i_k$ for all $k \neq \alpha$ and collect all removal times and all infection times, except i_α, in vectors as $\boldsymbol{r} = (r_1, r_2, \ldots, r_n)$ and $\boldsymbol{i} = (i_1, \ldots, i_{\alpha-1}, i_{\alpha+1}, \ldots, i_n) = (i_1, i_2, \ldots, i_n) \backslash \{i_\alpha\}$, respectively.

The purpose is to conduct Bayesian inference for the interesting parameters $\boldsymbol{\beta}$ and $\boldsymbol{\phi}$ (parameters controlling the infection and the removal process respectively), i.e. the object of interest is the posterior density $\pi(\boldsymbol{\beta}, \boldsymbol{\phi} \mid \boldsymbol{r}) \propto \pi(\boldsymbol{r} \mid \boldsymbol{\beta}, \boldsymbol{\phi})\pi(\boldsymbol{\beta}, \boldsymbol{\phi})$, where $\pi(\boldsymbol{r} \mid \boldsymbol{\beta}, \boldsymbol{\phi})$ is the model likelihood (based on observing removal times $\boldsymbol{r}$) and $\pi(\boldsymbol{\beta}, \boldsymbol{\phi})$ is the prior density. To this end, one would need to compute $\pi(\boldsymbol{r} \mid \boldsymbol{\beta}, \boldsymbol{\phi})$. However, as mentioned in section 1.3.4.3, this likelihood is in all, but the simplest of cases, intractable. To make this clearer consider the slightly simpler case, which α and i_α are known. Then the model likelihood is $\pi(\boldsymbol{r} \mid \boldsymbol{\beta}, \boldsymbol{\phi}, \alpha, i_\alpha) = \int \pi(\boldsymbol{r}, \boldsymbol{i} \mid \boldsymbol{\beta}, \boldsymbol{\phi}, \alpha, i_\alpha) d\boldsymbol{i}$ and this integral is numerically intractable for all practically interesting cases, due to the non-trivial region of integration; calculation of this integral requires integrating over all possible configurations of infection times that do not cause the epidemic to cease before r_n, and this task is prohibitive unless the dimension of the infections is unrealistically small.

Fortunately, by introducing the unobserved infection times $\boldsymbol{i}$ and the initial conditions α and i_α, one can construct and compute an augmented likelihood $\pi(\boldsymbol{r}, \boldsymbol{i} \mid \boldsymbol{\beta}, \boldsymbol{\phi}, \alpha, i_\alpha)$ (based on observing infection times $\boldsymbol{i}$ and removal times $\boldsymbol{r}$). This augmented likelihood, is constructed by considering the contribution of each individual (see e.g. Britton and O'Neill (2002); Neal and Roberts (2005); Kypraios (2007)) as follows. Since, unless already removed, individuals at any time point t are at risk of either being infected or being removed (but not both), $\pi(\boldsymbol{r}, \boldsymbol{i} \mid \boldsymbol{\beta}, \boldsymbol{\phi}, \alpha, i_\alpha)$ can be broken

into the product of two parts, the infection process part L_1 (describing the transition S $\rightarrow$ I) and the removal process part L_2 (describing the transition I $\rightarrow$ R).

Infection process part To construct the infection process part of the likelihood one needs to consider how ever-infected and never-infected individuals contribute to it. An ever-infected individual k, $k = 1, 2, \ldots, n$, $k \neq \alpha$, contributes by avoiding infection until time i_k^-, where t^- denotes the time just before t, and by getting infected at time i_k. A never-infected individual k, $k = n+1, n+2, \ldots, N+1$, contributes by avoiding infection throughout the course of the epidemic (i.e. until time r_n when the epidemic ceases). Using survival analysis methodology, the infection process part can then be written as

$$L_1 = \left(\prod_{k=1, k\neq\alpha}^{n} h_k \left(i_k^-; \boldsymbol{\beta} \right) \right) \times \exp \left(- \int_{i_\alpha}^{r_n} h \left(t; \boldsymbol{\beta} \right) dt \right), \tag{1.10}$$

where $\int_{i_\alpha}^{r_n} h(t; \boldsymbol{\beta}) dt$ is the total infection pressure applied, by infectives to susceptibles, throughout the course of the epidemic.

Removal process part Since it is assumed that individual infectious periods $r_k - i_k$ are i.i.d. according to a random variable T_D, having distribution $D(\boldsymbol{\phi})$, with parameter vector $\boldsymbol{\phi}$ (denoted as $r_k - i_k \overset{\text{i.i.d.}}{\sim} D(\boldsymbol{\phi})$), $k = 1, 2, \ldots, n$, the removal process part is:

$$L_2 = \prod_{k=1}^{n} f_{T_D}(r_k - i_k; \boldsymbol{\phi}), \tag{1.11}$$

where f_{T_D} is the probability density function (p.d.f.) (if T_D is continuous) or probability mass function (p.m.f.) (if T_D is discrete) of the random variable T_D.

As mentioned in section 1.3.4.1, three choices are considered for the infectious period distribution, namely Exponential, Gamma and constant. Depending on the choice of the infectious period, the removal process part of the likelihood (equation (1.11)) assumes a different form.

The Exponential distribution with rate parameter γ, denoted as Exp(γ), has a p.d.f. given by $f(x;\gamma) = \gamma\text{exp}(-\gamma x)$, $x \geq 0;\ \gamma > 0$, and therefore, if the infectious period distribution is assumed to be Exponential, $\boldsymbol{\phi} = \gamma$ and the removal process part is given by

$$L_2 = \gamma^n \text{exp}\left(-\gamma \sum_{k=1}^{n} (r_k - i_k)\right). \tag{1.12}$$

If a Gamma distribution is used, parametrized by shape ν and rate λ, denoted as Gamma(ν, λ), and described by its p.d.f. $f(x;\ \nu, \lambda) = \frac{\lambda^\nu}{\Gamma(\nu)} x^{\nu-1}\text{exp}(-\lambda x)$, $x \geq 0;\ \nu, \lambda > 0$, then $\boldsymbol{\phi} = (\nu, \lambda)$ and the removal process part becomes

$$L_2 = \left(\frac{\lambda^\nu}{\Gamma(\nu)}\right)^n \left(\prod_{k=1}^{n} (r_k - i_k)\right)^{\nu-1} \text{exp}\left(-\lambda \sum_{k=1}^{n} (r_k - i_k)\right). \tag{1.13}$$

Finally, if constant infectious periods are assumed, T_D is a point mass at c ($T_D \equiv c$), where $c > 0$ the length of the infectious periods, and thus $\boldsymbol{\phi} = c$ and the removal process part is given by

$$L_2 = \mathbb{1}_{\{r_k - i_k = c,\ k=1,2,\ldots,n\}}. \tag{1.14}$$

Augmented likelihood The augmented likelihood of the model, based on observing data $\boldsymbol{i}$ and $\boldsymbol{r}$, is given by multiplying the infection process part L_1 and the removal process part L_2 as follows

$$\begin{aligned}\pi(\boldsymbol{r},\boldsymbol{i}\mid\boldsymbol{\beta},\boldsymbol{\phi},\alpha,i_\alpha) &= L_1\times L_2\\ &= \left(\prod_{k=1,k\neq\alpha}^{n} h_k\left(i_k^-;\boldsymbol{\beta}\right)\right)\times\exp\left(-\int_{i_\alpha}^{r_n} h\left(t;\boldsymbol{\beta}\right)dt\right)\\ &\times\prod_{k=1}^{n} f_{T_D}(r_k-i_k;\boldsymbol{\phi}).\end{aligned}\tag{1.15}$$

1.3.5.3 Bayesian Inference and MCMC algorithm

Relying on the fact that the augmented likelihood $\pi(\boldsymbol{r},\boldsymbol{i}\mid\boldsymbol{\beta},\boldsymbol{\phi},\alpha,i_\alpha)$ can be computed, Bayesian inference for the interesting parameters $\boldsymbol{\beta}$ and $\boldsymbol{\phi}$ is performed by targeting the joint posterior distribution of $\boldsymbol{\beta}$, $\boldsymbol{\phi}$ and the augmented data $(\alpha,i_\alpha,\boldsymbol{i})$, via an MCMC algorithm. Thus the object of interest is the augmented posterior density

$$\pi(\boldsymbol{\beta},\boldsymbol{\phi},\alpha,i_\alpha,\boldsymbol{i}\mid\boldsymbol{r})\propto\pi(\boldsymbol{r},\boldsymbol{i}\mid\boldsymbol{\beta},\boldsymbol{\phi},\alpha,i_\alpha)\pi(\boldsymbol{\beta},\boldsymbol{\phi},\alpha,i_\alpha),\tag{1.16}$$

where $\pi(\boldsymbol{r},\boldsymbol{i}\mid\boldsymbol{\beta},\boldsymbol{\phi},\alpha,i_\alpha)$ is the augmented likelihood, constructed as above and given by equation (1.15), and $\pi(\boldsymbol{\beta},\boldsymbol{\phi},\alpha,i_\alpha)$ is the joint prior density of $\boldsymbol{\beta}$, $\boldsymbol{\phi}$, α and i_α.

MCMC algorithms for each considered model will be given separately (see relevant parts of sections 1.3.5.5, 1.3.5.6 and 1.3.5.7 for the standard SIR model, the non-linear infection rate SIR model and the two-level-mixing SIR model, respectively), nonetheless since the update step for the infection component $(\alpha,i_\alpha,\boldsymbol{i})$ is in most cases performed in the same manner (see the second paragraph in the beginning of section 1.3.5) it is described under this general setting (i.e. for a general model of the class). Recall that all MCMC algorithms used in this thesis follow the procedure of the general MCMC algorithm (Algorithm 3), described in section 1.3.2.4, where the vector of interest is decomposed into components which are updated in separate steps, according to their full conditional distributions. Suppose that the chain is transitioning from its s^{th} to its $(s+1)^{\text{th}}$ value and that the update steps for the rest

of the components, $\boldsymbol{\beta}$ and $\boldsymbol{\phi}$, have already been conducted so that $\boldsymbol{\beta}^{(s+1)}$ and $\boldsymbol{\phi}^{(s+1)}$ are the current values of $\boldsymbol{\beta}$ and $\boldsymbol{\phi}$. Following O'Neill and Becker (2001), $(\alpha, i_\alpha, \boldsymbol{i})$ can be updated using a MH step and a model-driven independent proposal distribution, as follows. First, choose one of the n ever-infected individuals, say k, according to a discrete uniform distribution on $\{1, 2, \ldots, n\}$, denoted as U$[1:n]$; where U$[1:n]$ is such that so that if $X \sim$ U$[1:n]$ then X has p.m.f. $f(x) = P(X = x) = \frac{1}{n}$, $x = 1, 2, \ldots, n$. Then, propose a candidate infection time for individual k, say i_k^*, by proposing an infectious period $r_k - i_k^* \sim D(\boldsymbol{\phi}^{(s+1)})$. Finally, calculate the MH acceptance ratio and accordingly accept or reject the proposed move. Algorithm 4 depicts the above procedure.

Algorithm 4 Update step for the infection component in an MCMC algorithm for a general SIR model

1. Suppose that the current value of the infection component is $(\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$ and that the update steps for $\boldsymbol{\beta}$ and $\boldsymbol{\phi}$, have already been conducted so that $\boldsymbol{\beta}^{(s+1)}$ and $\boldsymbol{\phi}^{(s+1)}$ are, respectively, their current values.

2. Generate $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)})$ according to $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \boldsymbol{\beta}^{(s+1)}, \boldsymbol{\phi}^{(s+1)})$ using a MH step as follows

   (a) Choose one of the n ever-infected individuals, say k, as $k \sim$ U$[1:n]$

   (b) Propose a candidate infection time for individual k, say i_k^*, as $r_k - i_k^* \sim D(\boldsymbol{\phi}^{(s+1)})$

   (c) Calculate the acceptance ratio $r = \frac{\pi(\alpha^*, i_\alpha^*, \boldsymbol{i}^* | \boldsymbol{r}, \boldsymbol{\beta}^{(s+1)}, \boldsymbol{\phi}^{(s+1)})}{\pi(\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)} | \boldsymbol{r}, \boldsymbol{\beta}^{(s+1)}, \boldsymbol{\phi}^{(s+1)})} \times \frac{q(r_k - i_k^{(s)})}{q(r_k - i_k^*)}$, where $q(x)$ is the p.d.f. of a random variable $X \sim D(\boldsymbol{\phi}^{(s+1)})$

   (d) Set $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}) = (\alpha^*, i_\alpha^*, \boldsymbol{i}^*)$ with probability $1 \wedge r$; otherwise set $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}) = (\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$

Some remarks are in order regarding the above update step. The proposal distribution is referred to as model-driven as it essentially proposes to update infectious periods according to the infectious period distribution of the assumed model; recall that the model assumes that $r_k - i_k \overset{\text{i.i.d.}}{\sim} D(\boldsymbol{\phi})$, $k = 1, 2, \ldots, n$.

In the above algorithm, note that, since only the infection time of individual k is attempted to be updated, $\boldsymbol{i}^{(s)} = (i_1^{(s)}, \ldots, i_{k-1}^{(s)}, i_k^{(s)}, i_{k+1}^{(s)}, \ldots, i_n^{(s)}) \backslash \{i_\alpha^{(s)}\}$ and $\boldsymbol{i}^* = (i_1^{(s)}, \ldots, i_{k-1}^{(s)}, i_k^*, i_{k+1}^{(s)}, \ldots, i_n^{(s)}) \backslash \{i_\alpha^*\}$ and thus (if $k \neq \alpha^{(s)} = \alpha^*$) the only difference between the vectors $\boldsymbol{i}^*$ and $\boldsymbol{i}^{(s)}$ is their k^{th} entry; with all the rest entries being the same. Note also that, when i_k^* is proposed, α^* and i_α^* are proposed by default (which could be the same as $\alpha^{(s)}$ and $i_\alpha^{(s)}$ or different), since, given a set of infection times, the minimum infection time and the label of the individual to which that corresponds to are deterministically specified. What these imply is that, after choosing the individual k whose infection time i_k is to be updated, the target density of the infection step reduces from $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \boldsymbol{\beta}^{(s+1)}, \boldsymbol{\phi}^{(s+1)})$ to $\pi(\alpha, i_\alpha, i_k \mid \boldsymbol{r}, \boldsymbol{\beta}^{(s+1)}, \boldsymbol{\phi}^{(s+1)}, \boldsymbol{i}_{[-k]}^{(s)})$, where if $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ is a vector with n entries, $\boldsymbol{x}_{[-k]}$ denotes the vector containing all entries of $\boldsymbol{x}$ except x_k, $k \in \{1, 2, \ldots, n\}$.

In an MCMC scheme, in addition to the explicit terms of the infection component, calculations also involve terms that are functions of the infection component. Such terms must also be informed accordingly at each update step of the infection component (reference to such terms will be made, when describing the MCMC schemes for each model below).

In practice, it is typical to repeat the update step for the infection component several times in an MCMC iteration. As already remarked in section 1.3.2.4, this is done to allow the chain to move around the target space and improve mixing; in this case,

if only one infection time is updated at each MCMC iteration, the full conditional distributions of the other components, $\boldsymbol{\beta}$ and $\boldsymbol{\phi}$, will not change appreciably and their sampled values will have high autocorrelation. There are different ways that this step can be repeated. For example, having decided on the number of repetitions, say M, one may choose to sample (e.g. uniformly at random, as above) the label of the individual, whose infection time is to be updated, at every repetition. Alternatively, one may choose to first sample (e.g. again uniformly at random) the labels of the M out of n individuals, to be updated, and then update their infection times according to their sampled order. In any case, the infection times are updated one by one. In this thesis, whenever the infection step is repeated, it is by using the latter of the two methods.

1.3.5.4 Remarks

As mentioned in the beginning of section 1.3.5, all models considered in this thesis (standard SIR model, non-linear infection rate SIR model and two-level-mixing SIR model) follow the general framework described above. To define a particular model one needs to explicitly specify the components of the infection process, $\boldsymbol{\beta}$, $h_k(t;\boldsymbol{\beta})$, $h(t;\boldsymbol{\beta})$, and the parameter of the removal process, $\boldsymbol{\phi}$. The removal process assumptions are identical for all three models, and as described in sections 1.3.5.1 and 1.3.5.2, and depending on whether $T_D \sim \text{Exp}(\gamma)$, $T_D \sim \text{Gamma}(\nu,\lambda)$ or $T_D \equiv c$, the removal process parameter $\boldsymbol{\phi}$ is specified as $\boldsymbol{\phi} = \gamma$, $\boldsymbol{\phi} = (\nu,\lambda)$ or $\boldsymbol{\phi} = c$, respectively. What distinguishes the three models are the differences in their corresponding infection process assumptions, which in turn yield different specifications for $\boldsymbol{\beta}$, $h_k(t;\boldsymbol{\beta})$ and $h(t;\boldsymbol{\beta})$; details follow in sections 1.3.5.5 to 1.3.5.7.

Notice that, although $\boldsymbol{i}$ is part of the data augmentation scheme, it does not require a prior distribution as evident from expression (1.16). Note also that, it is typical to assume that parameters are a priori independent (see e.g. O'Neill and Roberts (1999) or Held et al. (2019, chapter 9)) so that the joint prior density $\pi(\boldsymbol{\beta},\boldsymbol{\phi},\alpha,i_\alpha)$ is

expressed as $\pi(\boldsymbol{\beta})\pi(\boldsymbol{\phi})\pi(\alpha)\pi(i_\alpha)$ and the assignment of the prior distribution is done marginally for each one of $\boldsymbol{\beta}$, $\boldsymbol{\phi}$, α and i_α.

According to the assumed framework, given removal data $\boldsymbol{r} = (r_1, r_2, \ldots, r_n)$, the associated epidemic takes place in the time interval $[i_\alpha, r_n]$. As seen from expression (1.16), the unknown i_α is an additional model parameter (random variable) and its support is $(-\infty, r_1)$; this is because the initial infection i_α can only occur before the first removal r_1. To make the inference procedure consistent across removal data from different outbreaks, in the remainder of this thesis, the removal vector is always shifted to the left by r_1, i.e. given removal data $\boldsymbol{r} = (r_1, r_2, \ldots, r_n)$, r_1 is subtracted from each r_k, $k = 1, 2, \ldots, n$, so that $r_1 = 0$ and i_α has support $(-\infty, 0)$, always. Notice that such intervention has no impact on the inference for the parameters of interest, $\boldsymbol{\beta}$ and $\boldsymbol{\phi}$, as by adding (or subtracting) a constant to the removal times, one simply shifts the time interval in which the epidemic takes place without affecting the dynamics of the process.

Notice that, in the case that $T_D \equiv c$, where $c > 0$, although the length of the infectious period c is fixed between individuals (i.e. $r_k - i_k = c$ for all $k = 1, 2, \ldots, n$), it is still an unknown parameter to be estimated from the data.

In the case that $T_D \sim \text{Gamma}(\nu, \lambda)$, it is assumed throughout this thesis that the shape parameter ν is known and thus, from an inference standpoint, the removal process parameter reduces from $\boldsymbol{\phi} = (\nu, \lambda)$ to $\boldsymbol{\phi} = \lambda$. Although one can relax this assumption, and treat ν as an additional unknown parameter to be estimated from the data (see e.g. Xiang and Neal (2014)), it is more appropriate for some of the purposes of this work to treat ν as known (see for example section 2.7.1.3).

It is a well known fact that the $\text{Gamma}(\nu, \lambda)$ distribution reduces to an $\text{Exp}(\lambda)$ distribution in the case that its shape parameter is $\nu = 1$. This remark is particularly

relevant when it is of interest to distinguish between models having Exponential and Gamma infectious periods (see section 2.7).

1.3.5.5 Standard SIR model

The *standard SIR model* (Andersson and Britton, 2000) is a simple model for the spread of a disease, assuming homogeneity both in the population and the individual level (see section 1.3.4.1). The model, in the case that the infectious periods are assumed to be Exponential, is also referred to as the general stochastic epidemic, and it is arguably the most well known and widely used stochastic disease transmission model, originating from Bartlett (1949) and featuring in classical references such as Bailey (1975).

Definition Consider the notation of section 1.3.5.1. As explained in the first remark of section 1.3.5.4, to define the model one needs to describe the infection process ($\mathrm{S} \to \mathrm{I}$) assumptions and in turn explicitly specify the infection process parameter $\boldsymbol{\beta}$, the all-to-one infection rate $h_k(t;\boldsymbol{\beta})$ and the all-to-all infection rate $h(t;\boldsymbol{\beta})$.

S $\to$ I It is assumed that the population is homogeneously mixing (i.e. individuals mix together at random) in such a way that any infective individual makes contacts with any other individual at the time points of a homogeneous Poisson process of rate β; β is the one-to-one infection rate. This specifies $\boldsymbol{\beta}$ as $\boldsymbol{\beta} = \beta$. If a contacted individual is susceptible, at the time of contact, they instantly becomes infective. All Poisson processes describing infection contacts are assumed to be mutually independent and independent of the infectious periods. The aggregation property of Poisson processes specifies the all-to-one infection rate as $h_k(t;\beta) = \beta Y_t$, for individual k, $k = 1, 2, \ldots, N+1$, and the all-to-all infection rate as $h(t;\beta) = \beta X_t Y_t$, where Y_t and X_t are, as in section 1.3.5.1, the number of infectives and susceptibles at time t, respectively.

Likelihood and calculation of its terms Recall that in section 1.3.5.2 the likelihood was derived under a general framework, which all models in this thesis conform to. Recall also, from the first remark of section 1.3.5.4, that the models in this thesis only differ in the infection process assumptions. Thus the augmented likelihood of the standard SIR model is given by substituting its model-specific infection process components, $\boldsymbol{\beta} = \beta$, $h_k(t;\beta) = \beta Y_t$ and $h(t;\beta) = \beta X_t Y_t$, to equation (1.15), which is the equation of the augmented likelihood under the general framework. Specifically, using the same notation as in section 1.3.5.2, the augmented likelihood of the standard SIR model, based on observing data $\boldsymbol{i}$ and $\boldsymbol{r}$, and for general infectious period T_D, is given by

$$\begin{aligned}\pi(\boldsymbol{r},\boldsymbol{i} \mid \beta,\boldsymbol{\phi},\alpha,i_\alpha) = \;& L_1 \times L_2 \\ =\;& \left(\prod_{k=1,k\neq\alpha}^{n} \beta Y_{i_k^-}\right) \times \exp\left(-\int_{i_\alpha}^{r_n} \beta X_t Y_t dt\right) \\ &\times \prod_{k=1}^{n} f_{T_D}(r_k - i_k;\boldsymbol{\phi}),\end{aligned} \tag{1.17}$$

where, as in section 1.3.5.2, L_1 and L_2 are the infection process part and the removal process part, respectively, and $\int_{i_\alpha}^{r_n} \beta X_t Y_t dt$ is the total infection pressure applied, by infectives to susceptibles, throughout the course of the epidemic. For the purposes of this thesis, the standard SIR model is considered using all three infectious period distribution choices, namely Exponential, Gamma and constant and denoted as Exp-HM, Gamma-HM and Constant-HM, respectively. Note that, as described in section 1.3.5.2, the removal process part L_2 (last line in equation (1.17) above) is given by equation (1.12), (1.13) or (1.14), for each of the three aforementioned choices, respectively.

The above likelihood (equation (1.17)) can be evaluated in practice as follows. The removal process part L_2 is easy to deal with as, for any of the considered choices for the infectious period distribution, it comprises terms that are straightforward to

calculate (see equations (1.12), (1.13) and (1.14)). The infection process part requires calculating the term $Y_{i_k^-}$, for $k = 1, 2, \ldots, n$, $k \neq \alpha$, and the integral $\int_{i_\alpha}^{r_n} X_t Y_t dt$. Since $Y_{i_k^-}$ is the number of infectives in the population at time i_k^- (i.e. just before time i_k), $Y_{i_k^-}$ is calculated by noticing that an individual j, $j = 1, 2, \ldots, n$, $j \neq k$, is infective at time i_k^- if and only if $i_j < i_k < r_j$ and then counting the number of infectives at i_k^- as

$$Y_{i_k^-} = \sum_{j=1, j\neq k}^{n} \mathbb{1}_{\{i_j < i_k < r_j\}}. \tag{1.18}$$

The integral $\int_{i_\alpha}^{r_n} X_t Y_t dt$ can be calculated by observing that $\int_{i_\alpha}^{r_n} X_t Y_t dt$ is in fact the total time for which infection pressure is applied, by infectives to susceptibles, throughout the course of the epidemic. To see this, recall that $\int_{i_\alpha}^{r_n} \beta X_t Y_t dt$ is the total infection pressure applied, throughout the course of the epidemic, and that, for the standard SIR model, an infective individual k contacts (i.e. exerts infection pressure on) a susceptible individual j at a one-to-one rate β, which does not depend on the considered pair (k, j) (see the definition of the model in the relevant paragraph above). Considering that an initially susceptible individual j, $j = 1, 2, \ldots, N+1$, $j \neq \alpha$, receives infection pressure from an ever-infected individual k, $k = 1, 2, \ldots, n$, for a length of time $r_k \wedge i_j - i_k \wedge i_j$, the total time for which infection pressure is applied, is given by summing over k and j as

$$\int_{i_\alpha}^{r_n} X_t Y_t dt = \sum_{k=1}^{n} \sum_{j=1, j\neq\alpha}^{N+1} (r_k \wedge i_j - i_k \wedge i_j). \tag{1.19}$$

Bayesian inference and MCMC algorithm For the purposes of this thesis, MCMC inference is required for all three versions of the standard SIR model (see e.g. section 2.7). First, the algorithm for the Exp-HM model is described and details are provided where relevant. Then the algorithms for the Gamma-HM and the Constant-HM models are described by drawing comparisons with the algorithm for the Exp-HM model.

Exp-HM model Recall from section 1.3.5.3 that, for the class of considered models, the target posterior density of interest is given by expression (1.16). For the Exp-HM model, $\boldsymbol{\beta} = \beta$ (see the model definition at the beginning of section 1.3.5.5) and $\boldsymbol{\phi} = \gamma$ (see the description of the removal process part in section 1.3.5.2), and thus (see expression (1.16)) the target posterior density is

$$\pi(\beta, \gamma, \alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}) \propto \pi(\boldsymbol{r}, \boldsymbol{i} \mid \beta, \gamma, \alpha, i_\alpha)\pi(\beta, \gamma, \alpha, i_\alpha), \tag{1.20}$$

where $\pi(\boldsymbol{r}, \boldsymbol{i} \mid \beta, \gamma, \alpha, i_\alpha)$ is the augmented likelihood, and it is given by equation (1.17) (with the removal process part L_2 in equation (1.17) being given by equation (1.12)), and $\pi(\beta, \gamma, \alpha, i_\alpha)$ is the joint prior density of β, γ, α and i_α.

As remarked in section 1.3.5.4, it is assumed that parameters are a priori independent and the assignment of the prior distribution is done marginally for each one of β, γ, α and i_α. Following O'Neill and Roberts (1999) this is done as follows.

$$\begin{aligned}
\beta &\sim \text{Gamma}(\nu_\beta, \lambda_\beta) \\
\gamma &\sim \text{Gamma}(\nu_\gamma, \lambda_\gamma) \\
\alpha &\sim \text{U}[1 : n] \\
-i_\alpha &\sim \text{Exp}(\xi_{i_\alpha}).
\end{aligned}$$

Above, Gamma(ν, λ) and Exp(γ) denote the Gamma and the Exponential distribution respectively, both as described in section 1.3.5.2, and U$[1 : n]$ denotes the discrete uniform distribution on $\{1, 2, \ldots, n\}$, as described in section 1.3.5.3. The prior assignment of β and γ is to exploit a conjugacy result (see right below) while the prior assignment of α is saying that, before observing the data, the initial infective is equally likely to be any of the n ever-infected individuals. Note that the prior assignment of i_α ensures that i_α has support $(-\infty, 0)$, which is sensible as the initial infection must occur before the first removal, which is set to be 0 in all instances (see

relevant remark in section 1.3.5.4).

Following the procedure of the general MCMC algorithm (Algorithm 3), described in section 1.3.2.4, the vector of interest is decomposed into three components, namely β, γ and $(\alpha, i_\alpha, \boldsymbol{i})$, which are updated according to their full conditional distributions. A few lines of algebra reveal that β and γ have standard full conditional distributions, and can thus be updated using Gibbs steps. Specifically,

$$\begin{aligned} \pi(\beta \mid \boldsymbol{r}, \alpha, i_\alpha, \boldsymbol{i}) &\equiv \text{Gamma}\left(n - 1 + \nu_\beta, A + \lambda_\beta\right) \\ \pi(\gamma \mid \boldsymbol{r}, \alpha, i_\alpha, \boldsymbol{i}) &\equiv \text{Gamma}\Big(n + \nu_\gamma, B + \lambda_\gamma\Big) \end{aligned} \tag{1.21}$$

where $A = \int_{i_\alpha}^{r_n} X_t Y_t dt$ and $B = \sum_{k=1}^{n} (r_k - i_k)$. On the other hand, $(\alpha, i_\alpha, \boldsymbol{i})$ has a non-standard full conditional distribution given by

$$\begin{aligned} \pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \beta, \gamma) \propto \left(\prod_{k=1, k\neq\alpha}^{n} Y_{i_k^-}\right) \times \exp\left(-\beta A\right) \\ \times \exp\left(-\gamma B\right) \times \exp(\xi_{i_\alpha} i_\alpha) \mathbb{1}_{\{i_\alpha < 0\}}, \end{aligned} \tag{1.22}$$

and is updated using a MH step and a model-driven independent proposal distribution, as described in section 1.3.5.3 and illustrated in Algorithm 4; where $\boldsymbol{\beta} = \beta$, $\boldsymbol{\phi} = \gamma$ and the proposal distribution is $D(\boldsymbol{\phi}^{(s+1)}) = \text{Exp}(\gamma^{(s+1)})$. Note that the terms $Y_{i_k^-}$, A and B, appearing in the expressions above, all depend on the infection times and hence their values must be informed accordingly at each update step of the infection component. Algorithm 5 collects the steps for updating all three components.

Gamma-HM model The MCMC implementation steps for the Gamma-HM model are very similar to the Exp-HM model. The target posterior density of interest is given, in general form, by expression (1.16). For the Gamma-HM model, $\boldsymbol{\beta} = \beta$ (see the model definition at the beginning of section 1.3.5.5) and, since the shape

Algorithm 5 MCMC algorithm for the Exp-HM model

1. Suppose the current state is $(\beta^{(s)}, \gamma^{(s)}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$
2. Sample $\beta^{(s+1)} \sim \pi(\beta \mid \boldsymbol{r}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)}) \equiv \text{Gamma}\left(n - 1 + \nu_\beta, A^{(s)} + \lambda_\beta\right)$ using a Gibbs step
3. Sample $\gamma^{(s+1)} \sim \pi(\gamma \mid \boldsymbol{r}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)}) \equiv \text{Gamma}\Big(n + \nu_\gamma, B^{(s)} + \lambda_\gamma\Big)$ using a Gibbs step
4. Generate $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)})$ according to $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \beta^{(s+1)}, \gamma^{(s+1)})$ using a MH step as follows
   (a) Choose one of the n ever-infected individuals, say k, as $k \sim \text{U}[1:n]$
   (b) Propose a candidate infection time for individual k, say i_k^*, as $r_k - i_k^* \sim \text{Exp}(\gamma^{(s+1)})$
   (c) Calculate the acceptance ratio $r = \frac{\pi(\alpha^*, i_\alpha^*, \boldsymbol{i}^* \mid \boldsymbol{r}, \beta^{(s+1)}, \gamma^{(s+1)})}{\pi(\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)} \mid \boldsymbol{r}, \beta^{(s+1)}, \gamma^{(s+1)})} \times \frac{q(r_k - i_k^{(s)})}{q(r_k - i_k^*)}$, where $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \beta, \gamma)$ is given by expression (1.22) and $q(x)$ is the p.d.f. of a random variable $X \sim \text{Exp}(\gamma^{(s+1)})$
   (d) Set $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}) = (\alpha^*, i_\alpha^*, \boldsymbol{i}^*)$ with probability $1 \wedge r$; otherwise set $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}) = (\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$
5. Set the next state as $(\beta^{(s+1)}, \gamma^{(s+1)}, \alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)})$.

parameter ν is assumed to be known (see relevant remark in section 1.3.5.4), $\boldsymbol{\phi} = \lambda$. Therefore, the target posterior density for the Gamma-HM model is

$$\pi(\beta, \lambda, \alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu) \propto \pi(\boldsymbol{r}, \boldsymbol{i} \mid \beta, \lambda, \nu, \alpha, i_\alpha)\pi(\beta, \lambda, \alpha, i_\alpha), \tag{1.23}$$

where $\pi(\boldsymbol{r}, \boldsymbol{i} \mid \beta, \lambda, \nu, \alpha, i_\alpha)$ is the augmented likelihood, given by equation (1.17) (with the removal process part L_2 in equation (1.17) being given by equation (1.13)), and $\pi(\beta, \lambda, \alpha, i_\alpha)$ is the joint prior density of β, λ, α and i_α.

The prior assignment is nearly identical as for Exp-HM and it is done as follows.

$$\begin{aligned}
&\beta \sim \text{Gamma}(\nu_\beta, \lambda_\beta) \\
&\lambda \sim \text{Gamma}(\nu_\lambda, \lambda_\lambda) \\
&\alpha \sim \text{U}[1:n] \\
&- i_\alpha \sim \text{Exp}(\xi_{i_\alpha}).
\end{aligned}$$

Following the general MCMC algorithm (Algorithm 3), described in section 1.3.2.4, the vector of interest is decomposed into three components, namely β, λ and $(\alpha, i_\alpha, \boldsymbol{i})$, which are updated according to their full conditional distributions. Straightforward calculations yield that

$$\begin{aligned}
\pi(\beta \mid \boldsymbol{r}, \alpha, i_\alpha, \boldsymbol{i}) &\equiv \text{Gamma}\left(n - 1 + \nu_\beta, A + \lambda_\beta\right) \\
\pi(\lambda \mid \boldsymbol{r}, \nu, \alpha, i_\alpha, \boldsymbol{i}) &\equiv \text{Gamma}\Big(\nu n + \nu_\lambda, B + \lambda_\lambda\Big).
\end{aligned} \tag{1.24}$$

The standard form of the above full conditional distributions, allows β and λ to be updated using Gibbs steps. As for the Exp-HM model case, $(\alpha, i_\alpha, \boldsymbol{i})$ has a non-standard full conditional distribution given by

$$\begin{aligned}
\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu, \beta, \lambda) \propto \left(\prod_{k=1, k\neq\alpha}^{n} Y_{i_k^-}\right) \times \exp\left(-\beta A\right) \times \left(\prod_{k=1}^{n} (r_k - i_k)\right)^{\nu-1} \\
\times \exp\left(-\lambda B\right) \times \exp(\xi_{i_\alpha} i_\alpha) \mathbb{1}_{\{i_\alpha < 0\}},
\end{aligned} \tag{1.25}$$

and is updated using the MH step, described in section 1.3.5.3 and illustrated in Algorithm 4; where $\boldsymbol{\beta} = \beta$, $\boldsymbol{\phi} = \lambda$ and the proposal distribution is $D(\boldsymbol{\phi}^{(s+1)}) = \text{Gamma}(\nu, \lambda^{(s+1)})$. Algorithm 6 collects the steps of the above procedure.

Constant-HM model The biggest difference in the MCMC procedure between the Constant-HM model and the other two standard SIR models comes from the fact

Algorithm 6 MCMC algorithm for the Gamma-HM model

1. Suppose the current state is $(\beta^{(s)}, \lambda^{(s)}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$
2. Sample $\beta^{(s+1)} \sim \pi(\beta \mid \boldsymbol{r}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)}) \equiv \text{Gamma}\left(n - 1 + \nu_\beta, A^{(s)} + \lambda_\beta\right)$ using a Gibbs step
3. Sample $\lambda^{(s+1)} \sim \pi(\lambda \mid \boldsymbol{r}, \nu, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)}) \equiv \text{Gamma}\Big(\nu n + \nu_\lambda, B^{(s)} + \lambda_\lambda\Big)$ using a Gibbs step
4. Generate $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)})$ according to $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu, \beta^{(s+1)}, \lambda^{(s+1)})$ using a MH step as follows
   (a) Choose one of the n ever-infected individuals, say k, as $k \sim \text{U}[1:n]$
   (b) Propose a candidate infection time for individual k, say i_k^*, as $r_k - i_k^* \sim \text{Gamma}(\nu, \lambda^{(s+1)})$
   (c) Calculate the acceptance ratio $r = \frac{\pi(\alpha^*, i_\alpha^*, \boldsymbol{i}^* \mid \boldsymbol{r}, \nu, \beta^{(s+1)}, \lambda^{(s+1)})}{\pi(\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)} \mid \boldsymbol{r}, \nu, \beta^{(s+1)}, \lambda^{(s+1)})} \times \frac{q(r_k - i_k^{(s)})}{q(r_k - i_k^*)}$, where $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu, \beta, \lambda)$ is given by expression (1.25) and $q(x)$ is the p.d.f. of a random variable $X \sim \text{Gamma}(\nu, \lambda^{(s+1)})$
   (d) Set $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}) = (\alpha^*, i_\alpha^*, \boldsymbol{i}^*)$ with probability $1 \wedge r$; otherwise set $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}) = (\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$
5. Set the next state as $(\beta^{(s+1)}, \lambda^{(s+1)}, \alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)})$.

that, when the infectious periods are assumed to be constant, all infection times are automatically determined given a value c for the infectious period; because in such case $r_k - i_k = c$ for all $k = 1, 2, \ldots, n$. Therefore the update step for the infections is different. Substituting the infection and removal process parameters corresponding to the Constant-HM model, $\boldsymbol{\beta} = \beta$ (see the model definition at the beginning of section 1.3.5.5) and $\boldsymbol{\phi} = c$ (see the description of the removal process part in section 1.3.5.2), into equation (1.16) yields the target posterior density

$$\pi(\beta, c, \alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}) \propto \pi(\boldsymbol{r}, \boldsymbol{i} \mid \beta, c, \alpha, i_\alpha)\pi(\beta, c, \alpha, i_\alpha), \tag{1.26}$$

where $\pi(\boldsymbol{r}, \boldsymbol{i} \mid \beta, c, \alpha, i_\alpha)$ is the augmented likelihood, given by equation (1.17) (with the removal process part L_2 in equation (1.17) being given by equation (1.14)), and $\pi(\beta, c, \alpha, i_\alpha)$ is the joint prior density of β, c, α and i_α.

Assuming that β and (c, α, i_α) are a priori independent and noticing that, knowing c, implies that α and i_α are deterministically specified, the joint prior density can be written as $\pi(\beta, c, \alpha, i_\alpha) = \pi(\beta)\pi(c, \alpha, i_\alpha) = \pi(\beta)\pi(c)\pi(\alpha, i_\alpha \mid c) = \pi(\beta)\pi(c)$. Thus the prior assignment is fully specified by assigning marginal prior distributions for β and c. This is done as follows

$$\beta \sim \text{Gamma}(\nu_\beta, \lambda_\beta)$$
$$c \sim \text{Exp}(\psi_c).$$

As with all MCMC algorithms in this thesis, the updating scheme follows Algorithm 3. In this case, the vector of interest $(\beta, c, \alpha, i_\alpha, \boldsymbol{i})$ is decomposed into two components, β and $(c, \alpha, i_\alpha, \boldsymbol{i})$. This is done to utilize the aforementioned fact that, given a value for the infectious period, the infection times are specified by default; thus when a candidate infectious period value, say c^*, is proposed, candidate values, say α^*, i_α^* and $\boldsymbol{i}^*$, are automatically proposed for the infection variables. Identically to the other two standard SIR models, β has a standard full conditional distribution and is updated using a Gibbs step. Specifically,

$$\pi(\beta \mid \boldsymbol{r}, c, \alpha, i_\alpha, \boldsymbol{i}) \equiv \text{Gamma}\left(n - 1 + \nu_\beta, A + \lambda_\beta\right) \tag{1.27}$$

On the other hand, $(c, \alpha, i_\alpha, \boldsymbol{i})$ has a non-standard full conditional distribution given by

$$\pi(c, \alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \beta) \propto \left(\prod_{k=1, k\neq\alpha}^{n} Y_{i_k^-} \right) \times \exp\left(-\beta A\right) \times \mathbb{1}_{\{r_k - i_k = c,\ k=1,2,\ldots,n\}} \times \psi_c \exp(-\psi_c c), \tag{1.28}$$

and is updated using a MH step and a dependent proposal as follows. Given a current value of the infectious period, say $c^{(s)}$, a candidate value, say c^*, is proposed using a Normal proposal distribution centered around the current value $c^{(s)}$ (in log scale), as $\log(c^*) \sim N(\log(c^{(s)}), \sigma^2)$, where σ^2 is the variance and plays the role of a tuning parameter (see the relevant part of section 1.3.2.4 for more information on dependent proposals and tuning parameters). Algorithm 7 collects the steps required to update all components.

Algorithm 7 MCMC algorithm for the Constant-HM model

1. Suppose the current state is $(\beta^{(s)}, c^{(s)}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$
2. Sample $\beta^{(s+1)} \sim \pi(\beta \mid \boldsymbol{r}, c^{(s)}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)}) \equiv \text{Gamma}\left(n - 1 + \nu_\beta, A^{(s)} + \lambda_\beta\right)$ using a Gibbs step
3. Generate $(c^{(s+1)}, \alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)})$ according to $\pi(c, \alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \beta^{(s+1)})$ using a MH step as follows
   (a) Propose a candidate infectious period value, say c^*, as $\log(c^*) \sim N(\log(c^{(s)}), \sigma^2)$
   (b) Calculate the acceptance ratio $r = \frac{\pi(c^*\alpha^*, i_\alpha^*, \boldsymbol{i}^* \mid \boldsymbol{r}, \beta^{(s+1)})}{\pi(c^{(s)}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)} \mid \boldsymbol{r}, \beta^{(s+1)})} \times \frac{q(c^{(s)} \mid c^*)}{q(c^* \mid c^{(s)})}$, where $\pi(c, \alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \beta)$ is given by expression (1.28) and $q(x \mid y)$ is the p.d.f. of a random variable X such that $\log(X) \sim N(\log(y), \sigma^2)$
   (c) Set $(c^{(s+1)}, \alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}) = (c^*, \alpha^*, i_\alpha^*, \boldsymbol{i}^*)$ with probability $1 \wedge r$; otherwise set $(c^{(s+1)}, \alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}) = (c^{(s)}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$
4. Set the next state as $(\beta^{(s+1)}, c^{(s+1)}, \alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)})$.

The motivation for working in the log scale, rather than the actual scale (and propose values as $c^* \sim N(c^{(s)}, \sigma^2)$), is to avoid instances that negative values for c are proposed; since the length of the infectious period has to be positive, negative proposed values would automatically be rejected increasing the risk of poor chain mixing. Note that, for similar reasons, the tuning parameter σ^2 can be specified using burn-in so that the acceptance proportion in the update step in question is close to its optimal reference value of 0.44 (see the part regarding dependent proposals in section 1.3.2.4). One way to do this in practice is to split the burn-in iterations into batches, and increase or decrease the value of σ^2 accordingly at the end of each batch, so that the acceptance proportion gets closer to the optimal reference value; if the acceptance proportion is lower (higher) than desired, then the value of σ^2 should be decreased (increased) so that the proposed steps become smaller (larger) and the chances of acceptance higher (lower).

Threshold behaviour and the basic reproduction number R_0 Epidemic models frequently demonstrate threshold behaviour. Roughly speaking, this means that during the course of an epidemic, either a few individuals are infected (minor outbreak), or a fairly large number are infected (major outbreak). For the standard SIR model, in the case that the population size is large, this threshold phenomenon can be made precise, and the minor and major outbreak probabilities can be both defined and calculated, by using a branching process approximation at the initial stages of the epidemic; this result is typically referred to as the threshold limit theorem for epidemics (see Andersson and Britton (2000, theorem 3.1) and Ball and Donnelly (1995) for all the details). The intuition behind the branching process approximation, is that during the initial stages of an epidemic, in a large population, we would expect that, with high probability, individuals contacted by infectives are susceptible (i.e. $X_t \approx N$), so that the number of infectious individuals Y_t follows some sort of branching behaviour (Andersson and Britton, 2000); infections and removals in the epidemic correspond to births and deaths in the approximating branching process.

The key parameter associated with the threshold limit theorem is the basic reproduction number R_0. Specifically, according to the theorem, in a population of infinitely many susceptibles, the probability for a major outbreak to occur is positive if and only if $R_0 > 1$. Although the result is not directly applicable to finite populations (as is the case in real-life applications), it is still broadly true that the value of R_0, and in particular whether it is greater or smaller than 1, will accordingly indicate whether or not a major outbreak can occur. Consequently, R_0 is a parameter with great epidemiological interest and inference on R_0 can determine the implementation of disease control interventions and strategies.

Following Andersson and Britton (2000), the basic reproduction number R_0 for the standard SIR model is loosely defined as the average number of new infections caused by a typical infective, during the early stages of the epidemic, in a large population, and it is given by

$$R_0 = N\beta \mathrm{E}(T_D). \tag{1.29}$$

1.3.5.6 Non-linear infection rate SIR model

The *non-linear infection rate SIR model* (O'Neill and Wen, 2012) is an extension of the standard SIR model (see section above) obtained by relaxing the linearity assumption of the overall (all-to-all) infection rate. More precisely, whereas the standard SIR model assumes that infections occur according to an overall (all-to-all) rate of $\beta X_t Y_t$, with β, X_t and Y_t as in section 1.3.5.5 above, the non-linear infection rate model modifies the overall (all-to-all) rate to have the form $\beta X_t Y_t^p$, where $p \in [0, 1]$ is a power parameter controlling the level of exposure of susceptibles to infectives. The idea of modifying the infection rate in such a way dates back to the work of Severo (1969) and it is based on the reasoning that the rate of new infections need not, in all situations, simply increase linearly in X_t and Y_t (O'Neill and Wen, 2012). For example, it might be the case that as an epidemic progresses, susceptibles become

more aware of the risk of infection and adjust their behaviour accordingly or it could be the case that new infective individuals make increasingly less difference to the overall infection pressure due to saturation effects (O'Neill and Wen, 2012). Such type of phenomena can be captured by the introduction of the power parameter $p \in [0, 1]$, where the smaller the p the lesser the exposure of susceptibles to infectives.

Definition Using the notation of section 1.3.5.1 and following the remark of section 1.3.5.4, the model is defined by specifying the infection process assumptions ($\mathrm{S} \to \mathrm{I}$) and its components $\boldsymbol{\beta}$, $h_k(t; \boldsymbol{\beta})$ and $h(t; \boldsymbol{\beta})$.

S $\to$ I Each individual k, $k = 1, 2, \ldots, N + 1$, at each time point t, is subjected to contacts from the currently infective individuals $\mathcal{Y}_t$, at the time points of a non-homogeneous Poisson process of rate $h_k(t; \beta, p) = \beta Y_t^p$, where $\beta > 0$ and $p \in [0, 1]$. This specifies $\boldsymbol{\beta}$ as $\boldsymbol{\beta} = (\beta, p)$ and the all-to-one infection rate as $h_k(t; \beta, p) = \beta Y_t^p$, for individual k, $k = 1, 2, \ldots, N + 1$. If a contacted individual is susceptible, at the time of contact, they instantly become infective. All Poisson processes describing infection contacts are assumed to be mutually independent and independent of the infectious periods. The aggregation property of Poisson processes specifies the all-to-all infection rate as $h(t; \beta, p) = \beta X_t Y_t^p$.

Likelihood and calculation of its terms As for the standard SIR model (see the corresponding paragraph in section 1.3.5.5), the likelihood of the non-linear infection rate SIR model is given by substituting its model-specific infection process components, $\boldsymbol{\beta} = (\beta, p)$, $h_k(t; \beta) = \beta Y_t^p$ and $h(t; \beta) = \beta X_t Y_t^p$, to equation (1.15), which is the likelihood derived under a general framework, common for all considered models. Using the same notation as in section 1.3.5.2, the augmented likelihood of the non-linear infection rate SIR model, based on observing data $\boldsymbol{i}$ and $\boldsymbol{r}$, and for

general infectious period T_D, is given by

$$\begin{aligned}\pi(\boldsymbol{r}, \boldsymbol{i} \mid p, \beta, \boldsymbol{\phi}, \alpha, i_\alpha) = & \; L_1 \times L_2 \\ = & \left(\prod_{k=1, k\neq\alpha}^{n} \beta Y_{i_k^-}^p \right) \times \exp\left(- \int_{i_\alpha}^{r_n} \beta X_t Y_t^p dt \right) \\ & \times \prod_{k=1}^{n} f_{T_D}(r_k - i_k; \boldsymbol{\phi}).\end{aligned} \tag{1.30}$$

For the purposes of this thesis the non-linear infection rate SIR model is considered only for the case that infectious periods are Exponential (see section 2.8 for how this assumption serves the purposes of this thesis) and it is denoted as Exp-NL. Hence, the removal process part L_2 of the likelihood (last line in equation (1.30) above) is given by equation (1.12).

Evaluating the above likelihood in practice involves computing the terms $Y_{i_k^-}^p$, for $k = 1, 2, \ldots, n$, $k \neq \alpha$, and $\int_{i_\alpha}^{r_n} X_t Y_t^p dt$. The former term is simply calculated by first calculating $Y_{i_k^-}$ (as already described in the corresponding paragraph in section 1.3.5.5; see equation (1.18)) and then raising to the power of p. The latter term is a bit less straightforward. Recall that for the case of the standard SIR model, the method for calculating the corresponding term, $\int_{i_\alpha}^{r_n} X_t Y_t dt$, makes use of the fact that the model specifies the assumptions under which one-to-one contacts occur (see corresponding paragraph in section 1.3.5.5). However, the non-linear infection rate model is not specified in such a way (see the definition in the beginning of section 1.3.5.6) and therefore an alternative method is used to calculate $\int_{i_\alpha}^{r_n} X_t Y_t^p dt$. Specifically, by ordering all event (infection and removal) times as $i_\alpha = t_1 < t_2 < \cdots < t_{2n} = r_n$ and noticing that both X_t and Y_t are piecewise constant, changing values only at event times, the integral $\int_{i_\alpha}^{r_n} X_t Y_t^p dt$ can be calculated as

$$\int_{i_\alpha}^{r_n} X_t Y_t^p dt = \sum_{k=1}^{2n-1} X_{t_k} Y_{t_k}^p (t_{k+1} - t_k). \tag{1.31}$$

Bayesian inference and MCMC algorithm In this thesis, the power parameter p is treated as known, whenever the Exp-NL is fitted to data (see section 2.8.1.3 for more details on how such an approach serves the purposes of this thesis). Therefore, from an inference standpoint, the infection process parameter reduces from $\boldsymbol{\beta} = (\beta, p)$ to $\boldsymbol{\beta} = \beta$. Considering also that the removal process parameter is $\boldsymbol{\phi} = \gamma$ (see relevant paragraph of section 1.3.5.2), equation (1.16) implies that the target posterior density for the Exp-NL model is

$$\pi(\beta, \gamma, \alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, p) \propto \pi(\boldsymbol{r}, \boldsymbol{i} \mid p, \beta, \gamma, \alpha, i_\alpha)\pi(\beta, \gamma, \alpha, i_\alpha), \tag{1.32}$$

where $\pi(\boldsymbol{r}, \boldsymbol{i} \mid p, \beta, \gamma, \alpha, i_\alpha)$ is the augmented likelihood, and it is given by equation (1.30) (with the removal process part L_2 in equation (1.30) being given by equation (1.12)), and $\pi(\beta, \gamma, \alpha, i_\alpha)$ is the joint prior density of β, γ, α and i_α.

As can be gauged from the expressions of their respective target posterior densities (see equations (1.20) and (1.32)), the MCMC procedure for the Exp-NL model (when p is assumed to be known) is nearly identical to that of the Exp-HM model (see relevant part in section 1.3.5.5). Specifically, the prior assignment is exactly as for the Exp-HM model while the full conditional distributions for the three components, β, γ and $(\alpha, i_\alpha, \boldsymbol{i})$, are the same as for the Exp-HM model with the only difference being that the terms $A = \int_{i_\alpha}^{r_n} X_t Y_t dt$ and $Y_{i_k^-}$, are now replaced by $A_{NL} = \int_{i_\alpha}^{r_n} X_t Y_t^p dt$ and $Y_{i_k^-}^p$, respectively. That is to say that,

$$\begin{aligned} \pi(\beta \mid \boldsymbol{r}, p, \alpha, i_\alpha, \boldsymbol{i}) &\equiv \text{Gamma}\left(n - 1 + \nu_\beta, A_{NL} + \lambda_\beta\right) \\ \pi(\gamma \mid \boldsymbol{r}, \alpha, i_\alpha, \boldsymbol{i}) &\equiv \text{Gamma}\Big(n + \nu_\gamma, B + \lambda_\gamma\Big), \end{aligned} \tag{1.33}$$

and

$$\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, p, \beta, \gamma) \propto \left(\prod_{k=1, k\neq\alpha}^{n} Y_{i_k^-}^{p} \right) \times \exp\left(-\beta A_{NL}\right) \times \exp\left(-\gamma B\right) \times \exp(\xi_{i_\alpha} i_\alpha) \mathbb{1}_{\{i_\alpha < 0\}}, \tag{1.34}$$

Algorithm 8 gives the step-by-step procedure for conducing MCMC inference for the Exp-NL model.

Algorithm 8 MCMC algorithm for the Exp-NL model

1. Suppose the current state is $(\beta^{(s)}, \gamma^{(s)}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$
2. Sample $\beta^{(s+1)} \sim \pi(\beta \mid \boldsymbol{r}, p, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)}) \equiv \text{Gamma}\left(n - 1 + \nu_\beta, A_{NL}^{(s)} + \lambda_\beta\right)$ using a Gibbs step
3. Sample $\gamma^{(s+1)} \sim \pi(\gamma \mid \boldsymbol{r}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)}) \equiv \text{Gamma}\left(n + \nu_\gamma, B^{(s)} + \lambda_\gamma\right)$ using a Gibbs step
4. Generate $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)})$ according to $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, p, \beta^{(s+1)}, \gamma^{(s+1)})$ using a MH step as follows
   (a) Choose one of the n ever-infected individuals, say k, as $k \sim \text{U}[1:n]$
   (b) Propose a candidate infection time for individual k, say i_k^*, as $r_k - i_k^* \sim \text{Exp}(\gamma^{(s+1)})$
   (c) Calculate the acceptance ratio $r = \frac{\pi(\alpha^*, i_\alpha^*, \boldsymbol{i}^* \mid \boldsymbol{r}, p, \beta^{(s+1)}, \gamma^{(s+1)})}{\pi(\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)} \mid \boldsymbol{r}, p, \beta^{(s+1)}, \gamma^{(s+1)})} \times \frac{q(r_k - i_k^{(s)})}{q(r_k - i_k^*)}$, where $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, p, \beta, \gamma)$ is given by expression (1.34) and $q(x)$ is the p.d.f. of a random variable $X \sim \text{Exp}(\gamma^{(s+1)})$
   (d) Set $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}) = (\alpha^*, i_\alpha^*, \boldsymbol{i}^*)$ with probability $1 \wedge r$; otherwise set $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}) = (\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$
5. Set the next state as $(\beta^{(s+1)}, \gamma^{(s+1)}, \alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)})$.

Remarks When the population size goes to infinity, similar to the standard SIR model (see the part regarding threshold behaviour in section 1.3.5.5), the non-linear infection rate SIR model can be approximated by a simpler process. Specifically,

for the case that infectious periods are Exponential, the number of infectives in the population, for a non-linear infection rate SIR process, is approximated by a non-linear birth-death process (see O'Neill and Wen (2012) for more details). Although this approximation provides useful insights as far as the threshold effect of the model for the case that the population size is infinite, for finite populations (as it is the case in real-life applications) it is not obvious how any such effect can be characterized in terms of model parameters (O'Neill and Wen, 2012). In particular, there is not a natural way to define a basic reproduction type of parameter (i.e. a parameter which can characterize any threshold effect) for the non-linear infection rate SIR model.

In the case that the power parameter $p \in [0, 1]$ is set to 1, the non-linear infection rate SIR model reverts to the standard SIR model. This remark is particularly useful when it is of interest to distinguish between the two models (see section 2.8).

1.3.5.7 Two-level-mixing SIR model

The *two-level-mixing SIR model* (Ball et al., 1997) is a generalization of the standard SIR model (see section 1.3.5.5) obtained by relaxing the homogeneity assumption at the population level. More specifically, rather than assuming that individuals in the population mix homogeneously, as is the case for the standard SIR model (see section 1.3.5.5), the two-level-mixing model introduces heterogeneities in the population by partitioning the population into social groups, such as households, schools or workplaces and assuming that individuals mix at different rates within and between groups. The motivation for incorporating the population structure into the model is based on the idea that, given a population partitioned into groups, one should expect that the rate at which individuals contact each other will typically be much higher within a group rather than between groups and thus the structure of the population can play a crucial role in facilitating the spread of the infection (Ball et al., 1997).

Definition As in section 1.3.5.1 consider a closed population consisting of $C = N + 1$ individuals of which initially N are susceptible and 1 is infectious. Assume additionally that the population is partitioned into l households, labelled as $1, 2, \ldots, l$, with each household m consisting of C_m individuals, $m = 1, 2, \ldots, l$, so that $C = \sum_{m=1}^{l} C_m$. Similarly to the standard SIR model and the non-linear infection rate SIR model, the two-level-mixing SIR model follows the general framework of sections 1.3.5.1 to 1.3.5.4 and, as explained in the first remark of section 1.3.5.4, to define the model one needs to describe the assumptions of the infection process (S $\rightarrow$ I) and specify its parameter vector and the all-to-one and all-to-all infection rates. However, unlike the former two models, the two-level-mixing model does not assume that the population is homogeneously mixing and thus the infection process can not be described only at a population level. Specifically, the overall infection process is described by two independent infection processes, one modelling contacts at the population level (i.e. contacts between individuals in the population), referred to as the *global infection process* and one explicitly modelling contacts at the household level (i.e. within household contacts), referred to as the *local infection process*. Therefore, to define the model one needs to describe the assumptions and specify the components (the parameter and the all-to-one and all-to-all infection rates), of both the global and the local infection processes. Just like in section 1.3.5.1, let $\mathcal{X}_t$ and $\mathcal{Y}_t$ be the set and X_t and Y_t the number of susceptible and infective individuals in the population at time t, respectively. In addition, let $\mathcal{X}_t^{L,m}$ and $\mathcal{Y}_t^{L,m}$ be the set and $X_t^{L,m}$ and $Y_t^{L,m}$ the number of susceptible and infective individuals in household m, $m = 1, 2, \ldots, l$, at time t, respectively. The model is defined as follows.

S $\rightarrow$ I Any infective individual makes (global) contacts with any other individual in the population at the time points of a homogeneous Poisson process of rate β_G; β_G is the one-to-one-global infection rate. This specifies the parameter of the global infection process to be β_G. Additionally, an infective individual makes (local) contacts with any other individual in their household at the time points of a homogeneous

Poisson process of rate β_L; β_L is the one-to-one-local infection rate. This specifies the parameter of the local infection process to be β_L. If a contacted individual is susceptible, at the time of (global or local) contact, they instantly become infective. All Poisson processes describing (global or local) infection contacts are assumed to be mutually independent and independent of the infectious periods. The aggregation property of Poisson processes specifies all the required infection rates as follows. For the global infection process, the all-to-one and the all-to-all rate, referred to as all-to-one-global and all-to-all-global rate, is $h_k^G(t;\beta_G) = \beta_G Y_t$, for individual k, $k = 1, 2, \ldots, N+1$, and $h^G(t;\beta_G) = \beta_G X_t Y_t$, respectively. Similarly, for the local infection process, the all-to-one and the all-to-all rate, referred to as all-to-one-local and all-to-all-local rate, is $h_k^L(t;\beta_L) = \beta_L Y_t^{L,m_k}$, for individual k, $k = 1, 2, \ldots, N+1$, in household $m_k \in \{1, 2, \ldots, l\}$, and $h^L(t;\beta_L) = \sum_{m=1}^{l} \beta_L X_t^{L,m} Y_t^{L,m}$, respectively.

Notice that, the global infection process of the two-level-mixing SIR model is identical to the (overall) infection process of the standard SIR model (see the definition of the standard SIR model in section 1.3.5.5). From this point of view, it is easy to see how the two-level-mixing SIR model extends the standard SIR model by considering an additional infection process, the local.

To avoid any confusion, it is highlighted that the term *global infections* (contacts) refers to infections (contacts) occurring from the action of the global infection process (i.e. from the action of the Poisson process of one-to-one rate β_G or its aggregations) and the term *local infections* (contacts) refers to infections (contacts) occurring from the action of the local infection process (i.e. from the action of the Poisson process of one-to-one rate β_L or its aggregations). Note that, for a global infection (contact) the individual initiating the contact and the contacted individual could be in the same or different household, while for a local infection (contact) both individuals, the one initiating the contact and the contacted one, must be in the same household.

Likelihood and calculation of its terms To derive a likelihood it is possible to work as for the previous models (see the relevant parts in sections 1.3.5.5 and 1.3.5.6) and introduce the unobserved infection times $\boldsymbol{i} = (i_1, \ldots, i_{\alpha-1}, i_{\alpha+1}, \ldots, i_n)$ and the initial conditions α and i_α. However, it is preferable, from an MCMC inference standpoint (see the part on Bayesian inference and MCMC algorithm that follows), to additionally introduce the unobserved infection types $\boldsymbol{b} = (b_1, \ldots, b_{\alpha-1}, b_{\alpha+1}, \ldots, b_n)$, where $b_k = 1$ or $b_k = 0$, in the instance that the type of infection of ever-infected individual k, $k = 1, 2, \ldots, n$, $k \neq \alpha$, is global or local, respectively; simply put, $\boldsymbol{b}$ is an $(n-1)$-dimensional vector signifying the type of infection (global or local) for every (excluding the initial infective α) ever-infected individual k, $k = 1, 2, \ldots, n$, $k \neq \alpha$. Such data augmentation scheme consists of $\boldsymbol{i}$, α, i_α and in addition $\boldsymbol{b}$. The derivation of this further augmented likelihood follows along the lines of the general framework, as set in section 1.3.5.2, with the only difference being that instead of one infection process, there are two (independent) infection processes involved, the global and the local (see the model definition above). As a result, the likelihood is the product of three parts, the global infection process part, say L_1^G, the local infection process part, say L_1^L, and the removal process part L_2. The removal process part is given, as in all other cases, by equation (1.11). The infection process parts, L_1^G and L_1^L, both conform to the general infection process framework, described by equation (1.10), and as a result each one is given by substituting its particular components (parameter, all-to-one rate and all-to-all rate), specified in the definition section above, into equation (1.10). Thus the augmented likelihood of the two-level-mixing SIR model, based on

observing data $\boldsymbol{i}$, $\boldsymbol{b}$ and $\boldsymbol{r}$, and for general infectious period T_D, is given by

$$\begin{aligned}\pi(\boldsymbol{r},\boldsymbol{i},\boldsymbol{b}\mid\beta_G,\beta_L,\boldsymbol{\phi},\alpha,i_\alpha) = \; & L_1^G\times L_1^L\times L_2\\ = \; & \left(\prod_{k\in\mathcal{I}_G}\beta_G Y_{i_k^-}\right)\times\exp\left(-\int_{i_\alpha}^{r_n}\beta_G X_t Y_t dt\right)\\ & \times\left(\prod_{k\in\mathcal{I}_L}\beta_L Y_{i_k^-}^{L,m_k}\right)\times\exp\left(-\int_{i_\alpha}^{r_n}\sum_{m=1}^{l}\beta_L X_t^{L,m}Y_t^{L,m}dt\right)\\ & \times\prod_{k=1}^{n} f_{T_D}(r_k-i_k;\boldsymbol{\phi}),\end{aligned} \tag{1.35}$$

where $\mathcal{I}_G=\{k\in\{1,2,\ldots,n\},k\neq\alpha:b_k=1\}$ is the set of ever-infected individuals for which their infection type is global and similarly $\mathcal{I}_L=\{k\in\{1,2,\ldots,n\},k\neq\alpha:b_k=0\}$ is the set of ever-infected individuals for which their infection type is local. To serve the purposes of this thesis the two-level-mixing SIR model is considered using Exponential and constant infectious periods (see sections 3.3 and 2.9, respectively) and is denoted as Exp-2L and Constant-2L, respectively. For each of the aforementioned cases, the removal process part L_2 of the likelihood (last line in equation (1.35) above) is given by equation (1.12) and (1.14), accordingly.

Regarding the evaluation of the above likelihood in practice, recall (see the model definition above) that the global infection process part (second line in equation (1.35) above) is identical to the overall infection process part of the standard SIR model and thus its calculation is also identical; see the corresponding part of section 1.3.5.5 and equations (1.18) and (1.19). The local infection process part (third line in equation (1.35) above) can also be computed in a similar manner. More specifically, the terms $Y_{i_k^-}^{L,m_k}$ and $\int_{i_\alpha}^{r_n}X_t^{L,m}Y_t^{L,m}dt$ are calculated in the same way as $Y_{i_k^-}$ (see equation (1.18)), and $\int_{i_\alpha}^{r_n}X_tY_tdt$ (see equation (1.19)) respectively, with the difference that, instead of considering (the contribution of) all individuals in the population, one only considers (the contribution of) the individuals in the household in question.

As mentioned above, it is also possible to derive a likelihood for the model by augmenting only $\boldsymbol{i}$, α and i_α (and not $\boldsymbol{b}$). Specifically, using a similar line of arguments as above, one finds that the augmented likelihood of the two-level-mixing SIR model, based on observing data $\boldsymbol{i}$ and $\boldsymbol{r}$, and for general infectious period T_D, is given by

$$\begin{aligned}\pi(\boldsymbol{r},\boldsymbol{i} \mid \beta_G,\beta_L,\boldsymbol{\phi},\alpha,i_\alpha) = &\left(\prod_{k=1,k\neq\alpha}^{n}\left(\beta_G Y_{i_k^-} + \beta_L Y_{i_k^-}^{L,m_k}\right)\right)\\ &\times \exp\left(-\int_{i_\alpha}^{r_n}\beta_G X_t Y_t dt\right)\times\exp\left(-\int_{i_\alpha}^{r_n}\sum_{m=1}^{l}\beta_L X_t^{L,m}Y_t^{L,m}dt\right)\\ &\times\prod_{k=1}^{n} f_{T_D}(r_k - i_k;\boldsymbol{\phi}),\end{aligned} \tag{1.36}$$

The above likelihood (equation (1.36)), and the further augmented likelihood (equation (1.35)) can be substituted into equation $\pi(\boldsymbol{b} \mid \boldsymbol{r},\beta_G,\beta_L,\boldsymbol{\phi},\alpha,i_\alpha,\boldsymbol{i}) = \frac{\pi(\boldsymbol{r},\boldsymbol{i},\boldsymbol{b}|\beta_G,\beta_L,\boldsymbol{\phi},\alpha,i_\alpha)}{\pi(\boldsymbol{r},\boldsymbol{i}|\beta_G,\beta_L,\boldsymbol{\phi},\alpha,i_\alpha)}$, to identify that the full conditional distribution of $\boldsymbol{b}$, is that of an $(n-1)$-dimensional random vector, $\boldsymbol{u} = (u_1,\ldots,u_{\alpha-1},u_{\alpha+1},\ldots,u_n)$, where its components, u_k, are mutually independent Bernoulli random variables, taking values $u_k = 1$ or $u_k = 0$ in the instance that the type of infection of individual k is global or local, respectively, with $P(u_k = 1) = \frac{\beta_G Y_{i_k^-}}{\beta_G Y_{i_k^-}+\beta_L Y_{i_k^-}^{L,m_k}}$, $k = 1,2,\ldots,n$, $k \neq \alpha$; note that this result can equivalently be deduced using the aggregation property of Poisson processes (see e.g. Ross (2009, proposition 5.4)). The standard form of the full conditional distribution of $\boldsymbol{b}$ (and in particular, the fact that it is easy to sample from it) is utilized in the MCMC algorithm, that follows right below.

Bayesian inference and MCMC algorithm While both the Exp-2L and Constant-2L models are considered in this thesis, only the latter is used for the purposes of MCMC inference (see section 2.9). For the Constant-2L model, where the infection process parameter is $\boldsymbol{\phi} = c$ (see relevant paragraph of section 1.3.5.2),

the target joint posterior density of parameters and augmented data is expressed as

$$\pi(\beta_G, \beta_L, c, \alpha, i_\alpha, \boldsymbol{i}, \boldsymbol{b} \mid \boldsymbol{r}) \propto \pi(\boldsymbol{r}, \boldsymbol{i}, \boldsymbol{b} \mid \beta_G, \beta_L, c, \alpha, i_\alpha)\pi(\beta_G, \beta_L, c, \alpha, i_\alpha), \tag{1.37}$$

where $\pi(\boldsymbol{r}, \boldsymbol{i}, \boldsymbol{b} \mid \beta_G, \beta_L, c, \alpha, i_\alpha)$ is the augmented likelihood, based on observing data $\boldsymbol{i}$, $\boldsymbol{b}$ and $\boldsymbol{r}$, and given by equation (1.35) (with the removal process part L_2 in equation (1.35) being given by equation (1.14)), and $\pi(\beta_G, \beta_L, c, \alpha, i_\alpha)$ is the joint prior density of β_G, β_L, c, α and i_α.

Following along the same lines as for the Constant-HM model (see corresponding part of section 1.3.5.5), and assuming in addition a priori independence between β_G and β_L, the joint prior density can be written as $\pi(\beta_G, \beta_L, c, \alpha, i_\alpha) = \pi(\beta_G)\pi(\beta_L)\pi(c)$ and is specified as

$$\begin{aligned}
\beta_G &\sim \text{Gamma}(\nu_{\beta_G}, \lambda_{\beta_G}) \\
\beta_L &\sim \text{Gamma}(\nu_{\beta_L}, \lambda_{\beta_L}) \\
c &\sim \text{Exp}(\psi_c).
\end{aligned}$$

Following the general MCMC algorithm (Algorithm 3), described in section 1.3.2.4, the vector of interest is decomposed into three components, namely β_G, β_L and $(c, \alpha, i_\alpha, \boldsymbol{i}, \boldsymbol{b})$, which are updated according to their full conditional distributions. Similar calculations as for the previous MCMC schemes yield that β_G and β_L have standard full conditional distributions, and can thus be updated using Gibbs steps. Specifically,

$$\begin{aligned}
\pi(\beta_G \mid \boldsymbol{r}, c, \alpha, i_\alpha, \boldsymbol{i}, \boldsymbol{b}) &\equiv \text{Gamma}\left(n_G + \nu_{\beta_G}, A + \lambda_{\beta_G}\right) \\
\pi(\beta_L \mid \boldsymbol{r}, c, \alpha, i_\alpha, \boldsymbol{i}, \boldsymbol{b}) &\equiv \text{Gamma}\left(n_L + \nu_{\beta_L}, \sum_{m=1}^{l} A_{L,m} + \lambda_{\beta_L}\right),
\end{aligned} \tag{1.38}$$

where $A = \int_{i_\alpha}^{r_n} X_t Y_t dt$ is as in section 1.3.5.5, $A_{L,m} = \int_{i_a lpha}^{r_n} X_t^{L,m} Y_t^{L,m}$, and n_G and n_L are the numbers of local and global infections, respectively. On the other hand, $(c, \alpha, i_\alpha, \boldsymbol{i}, \boldsymbol{b})$ has a non-standard full conditional distribution given by

$$\begin{aligned}\pi(c, \alpha, i_\alpha, \boldsymbol{i}, \boldsymbol{b} \mid \boldsymbol{r}, \beta_G, \beta_L) \propto & \left(\prod_{k \in \mathcal{I}_G} Y_{i_k^-}\right) \times \exp\left(-\beta_G A\right) \\ & \times \left(\prod_{k \in \mathcal{I}_L} Y_{i_k^-}^{L,m_k}\right) \times \exp\left(-\beta_L \sum_{m=1}^{l} A_{L,m}\right) \\ & \times \mathbb{1}_{\{r_k - i_k = c,\ k=1,2,\ldots,n\}} \times \psi_c \exp(-\psi_c c).\end{aligned} \tag{1.39}$$

This component is updated using a MH step in a procedure that involves two parts. The first part, proposes values for $(c, \alpha, i_\alpha, \boldsymbol{i})$; this is done in an identical way to how values for $(c, \alpha, i_\alpha, \boldsymbol{i})$ are proposed in the MCMC scheme of the Constant-HM model (see algorithm 7 and relevant part of section 1.3.5.5). The second part, proposes values for $\boldsymbol{b}$, conditioned on the already proposed values for $(c, \alpha, i_\alpha, \boldsymbol{i})$. More precisely, given a current value of the infectious period, say $c^{(s)}$, a candidate value, say c^*, is proposed as $\log(c^*) \sim N(\log(c^{(s)}), \sigma^2)$; in turn, candidate values, say α^*, i_α^* and $\boldsymbol{i}^*$, are automatically proposed for the infection variables. Then, given c^*, α^*, i_α^* and $\boldsymbol{i}^*$, values for $\boldsymbol{b}$ are proposed, according to the full conditional distribution of $\boldsymbol{b}$, utilizing the fact that it has a standard form, making it easy to sample from (see discussion in the likelihood section right above). Note that, all of the terms $Y_{i_k^-}$, $Y_{i_k^-}^{L,m_k}$, A, $A_{L,m}$, n_G and n_L, appearing in the expressions above, depend on $(c, \alpha, i_\alpha, \boldsymbol{i}, \boldsymbol{b})$ and thus their values must be informed accordingly at each update step of $(c, \alpha, i_\alpha, \boldsymbol{i}, \boldsymbol{b})$. Algorithm 9 gives the step-by-step procedure for conducing MCMC inference for the Constant-2L model.

In the above algorithm, the tuning parameter σ^2 is specified, using burn-in, in the same way as for the Constant-HM model (see the relevant part of section 1.3.5.5).

Algorithm 9 MCMC algorithm for the Constant-2L model

1. Suppose the current state is $(\beta_G^{(s)}, \beta_L^{(s)}, c^{(s)}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)}, \boldsymbol{b}^{(s)})$

2. Sample $\beta_G^{(s+1)} \sim \pi(\beta_G \mid \boldsymbol{r}, c^{(s)}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)}, \boldsymbol{b}^{(s)})$
   $\equiv \text{Gamma}\left(n_G^{(s)} + \nu_{\beta_G}, A^{(s)} + \lambda_{\beta_G}\right)$ using a Gibbs step

3. Sample $\beta_L^{(s+1)} \sim \pi(\beta_L \mid \boldsymbol{r}, c^{(s)}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)}, \boldsymbol{b}^{(s)})$
   $\equiv \text{Gamma}\left(n_L^{(s)} + \nu_{\beta_L}, \sum_{m=1}^{l} A_{L,m}^{(s)} + \lambda_{\beta_L}\right)$ using a Gibbs step

4. Generate $(c^{(s+1)}, \alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}, \boldsymbol{b}^{(s+1)})$ according to $\pi(c, \alpha, i_\alpha, \boldsymbol{i}, \boldsymbol{b} \mid \boldsymbol{r}, \beta_G^{(s+1)}, \beta_L^{(s+1)})$ using a MH step as follows

   (a) Propose a candidate infectious period value, say c^*, as $\log(c^*) \sim N(\log(c^{(s)}), \sigma^2)$

   (b) Given c^*, α^*, i_α^* and $\boldsymbol{i}^*$, propose a candidate infection type vector, say $\boldsymbol{b}^*$, as $\boldsymbol{b}^* \sim \pi(\boldsymbol{b} \mid \boldsymbol{r}, \beta_G^{(s+1)}, \beta_L^{(s+1)}, c^*, \alpha^*, i_\alpha^*, \boldsymbol{i}^*)$

   (c) Calculate the acceptance ratio $r = \frac{\pi(c^*\alpha^*, i_\alpha^*, \boldsymbol{i}^*, \boldsymbol{b}^* \mid \boldsymbol{r}, \beta_G^{(s+1)}, \beta_L^{(s+1)})}{\pi(c^{(s)}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)}, \boldsymbol{b}^{(s)} \mid \boldsymbol{r}, \beta_G^{(s+1)}, \beta_L^{(s+1)})}$
   $\times \frac{q_1(c^{(s)} \mid c^*)}{q_1(c^* \mid c^{(s)})} \frac{q_2(\boldsymbol{b}^{(s)} \mid c^{(s)}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})}{q_2(\boldsymbol{b}^* \mid c^*, \alpha^*, i_\alpha^*, \boldsymbol{i}^*)}$, where $\pi(c, \alpha, i_\alpha, \boldsymbol{i}, \boldsymbol{b} \mid \boldsymbol{r}, \beta_G, \beta_L)$ is given by expression (1.39), $q_1(x \mid y)$ is the p.d.f. of a random variable X such that $\log(X) \sim N(\log(y), \sigma^2)$ and $q_2(\boldsymbol{b} \mid c, \alpha, i_\alpha, \boldsymbol{i}) = \pi(\boldsymbol{b} \mid \boldsymbol{r}, \beta_G^{(s+1)}, \beta_L^{(s+1)}, c, \alpha, i_\alpha, \boldsymbol{i})$

   (d) Set $(c^{(s+1)}, \alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}, \boldsymbol{b}^{(s+1)}) = (c^*, \alpha^*, i_\alpha^*, \boldsymbol{i}^*, \boldsymbol{b}^*)$ with probability $1 \wedge r$; otherwise set $(c^{(s+1)}, \alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}, \boldsymbol{b}^{(s+1)}) = (c^{(s)}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)}, \boldsymbol{b}^{(s)})$

5. Set the next state as $(\beta^{(s+1)}, c^{(s+1)}, \alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}, \boldsymbol{b}^{(s+1)})$.

If one performs the MCMC inference, not as above but instead using the likelihood of equation (1.36), that augments only α, i_α and $\boldsymbol{i}$, and not $\boldsymbol{i}$ (see e.g. Alharthi (2016)), the full conditional distributions of β_L and β_G do not assume a standard form (unlike above) and MH steps are required for their update, which invoke additional calculations for the acceptance ratios and require tuning for the proposal distributions.

Threshold behaviour and the basic reproduction number R_* As mentioned previously (see corresponding paragraph in section 1.3.5.5), it is typical for epidemic models to demonstrate threshold behaviour. For the two-level-mixing model, similar to the standard SIR model (see corresponding paragraph in section 1.3.5.5), in the case that the population size is large, this behaviour can be both made precise, by approximating the epidemic process with a suitable branching process, and also characterized, via a basic reproduction number type of parameter, denoted as R_*, so that a global outbreak occurs with positive probability if and only if $R_* > 1$ (see Ball et al. (1997) for all the details). Unlike the case of the standard SIR model, where the basic reproduction number R_0 is defined at the individual-to-individual level (see corresponding paragraph in section 1.3.5.5), for the two-level-mixing model it is defined at the household-to-household level. More precisely, R_* can be loosely defined as the average number of households infected by a typically infected household in a totally susceptible population.

In the particular case that all household are of equal size C_H, that is $C_m = C_H$ for all households m, $m = 1, 2, \ldots, l$, and the population size C becomes large in such a way that the number of households l becomes large but the household size C_H remains fixed, R_* has a simple expression given by

$$R_* = \mu R_G, \tag{1.40}$$

where μ is the expected number of ever-infected individuals, including the initial infective, of the within household epidemic (i.e. the household epidemic in which only local infections occur) and R_G is the basic reproduction number for the model which all households are of size 1 (i.e. the standard SIR model for which only global infections occur). In practice, μ can be computed by solving a system of triangular equations (see Ball (1986)), while R_G, being the basic reproduction number of a standard SIR model with one-to-one infection rate β_G, is calculated (using equation

(1.29)) as

$$R_G = N\beta_G \mathrm{E}(T_D). \tag{1.41}$$

Remarks In the case that $\beta_L = 0$ or $C_m = 1$ for all m, $m = 1, 2, \ldots, l$, the two-level-mixing SIR model reduces to the standard SIR model. This remark is particularly relevant when it is desired to distinguish between the two models (see section 2.9).

1.4 Literature review

This section reviews the relevant literature regarding both, model assessment methods and Bayesian inference methods, for stochastic epidemic models.

1.4.1 Model assessment methods for stochastic epidemic models

As mentioned in O'Neill (2010), the literature on model assessment methods for stochastic epidemic models is not extensive. For instance, two classical references for stochastic epidemic modelling, Bailey (1975); Andersson and Britton (2000), do not present a procedure for assessing the fit of models based on temporal outbreak data. An approach for doing so, based on model deviance, is given in another standard reference, Becker (1989). The drawback is that the data were modelled using generalized linear models, rather than stochastic epidemic models, and it was also assumed that the times of infection and removal were observed for all individuals, something not adopted in the context of this work where the more realistic case of observing only removal times is considered (see section 1.3.4.2). Existing approaches can broadly be divided into two (overlapping) main categories. The first, is based on the idea of analyzing some sort of residuals and the second is based on posterior predictive checking. The literature related to each of these type of approaches is reviewed below.

1.4.1.1 Residual tests

The residual technique revolves around the idea of constructing a set of stochastic residuals, whose sampling distribution is known and independent of model parameters, and then assessing the posterior distribution of the 'reconstructed' residuals for consistency under the reference sampling distribution.

Streftaris and Gibson (2012) used Bayesian latent residuals to assess the fit of an SEIR model (see section 1.3.4.1). The authors utilized the Sellke construction (Sellke, 1983), an alternative but equivalent procedure for defining stochastic epidemic processes, to formulate the model. That is, each of the N initially susceptible individuals is independently assigned a critical level of exposure to infection (threshold) $Q_k \sim \text{Exp}(1)$ and becomes infected at time $i_k = \inf\{t \geq 0 : Q_k = \int_0^t \beta Y_u du\}$, $k = 1, 2, \ldots, N$, where β is the one-to-one infection rate and Y_u the number of infective individuals at time u. The authors generalized the thresholds as $Q_k \sim \text{Weibull}(\nu, \lambda)$, to allow dependence on the history of accumulation of infectious pressure. Then the latent Bayesian residuals were defined as $\tau_k = F_Q(q_k)$, where F_Q is the (common) cumulative distribution function of Q_k, $k = 1, 2, \ldots, N$. Under the assumption of correct model specification, the sampling distribution of τ_k is uniform on $[0, 1]$, i.e. $\tau_k \sim \text{U}(0, 1)$, $k = 1, 2, \ldots, N$. As the residuals depended on unobserved quantities, the authors used MCMC methods to sample from their posterior distribution and tested if the distribution of $\boldsymbol{\tau}^{(s)} = (\tau_1^{(s)}, \tau_2^{(s)}, \ldots, \tau_N^{(s)})$, at each MCMC iteration s, was $\text{U}(0, 1)$ by conducting a Kolmogorov-Smirnov test and recording the associated p-value. The resulting distribution of p-values was used as the basis for assessing the model's fit, where the more the mass on smaller values the worse the fit. The idea of Bayesian residuals was also previously employed in the work of Gibson et al. (2006), where the spread of disease in plant populations was studied using percolation stochastic models.

A similar approach was taken by Jewell et al. (2009) to assess models applied to the 2001 Foot and Mouth epidemic data. The difference in this case, was that the residuals were defined by utilizing a non-centered reparameterization (Papaspiliopoulos et al., 2003) of the model. More specifically, after specifying the infectious period T_D, of each farm k, to be $T_D = r_k - i_k \sim \text{Gamma}(\nu, \lambda)$, where i_k and r_k the time of infection and notification and $k = 1, 2, \ldots, n$, where n the number of farms, the authors reparameterized the infectious period as $u_k = \lambda(r_k - i_k) \sim \text{Gamma}(\nu, 1)$ in order to break the a priori dependence between i_k and λ and thus to improve the efficiency of the MCMC algorithm. Throughout the analysis, a known shape parameter $\nu = 4$ was assumed yielding a known sampling distribution for the variables u_k, $k = 1, 2, \ldots, n$. Utilizing this fact, the authors defined u_k as the non-centered residuals and assessed the fit of the model by examining if the posterior distribution of the u_k, achieved using MCMC samples, was in line with the reference Gamma$(4, 1)$ distribution.

The residual technique was also extended to the spatio-temporal epidemic setting by Lau et al. (2014) in order to assess the fit of an SEIR model. In the spatio-temporal setting, interest is usually placed in the form of the spatial transmission kernel function, given its importance in designing ring-culling strategies (Gibson et al., 2018). Thus, the residual test was designed to detect misspecification of the kernel function. The procedure was similar as in the previously mentioned references, where the posterior distribution of the residuals, obtained using MCMC samples, was compared to a U$(0, 1)$ reference distribution. It should be noted that this example is of less relevance to our setting, as the models in this thesis do not assume a spatial structure.

Approaches based on residuals have computational appeal, since the residuals are calculated within each iteration of the MCMC algorithm with minimal additional cost. However they have the considerable drawback of heavily relying on information imputed from the model. More specifically, as discussed in section 1.3.3.3, since the

missing information is imputed from the model itself the tests have low power to detect lack of fit, as the imputed values reinforce the assumptions of the model being tested (Gelman, 2013; Gibson et al., 2018). Also, as with any approach that does not focus on the observed data, the choice of which residuals to use is somewhat arbitrary (O'Neill, 2010).

1.4.1.2 Posterior predictive checking

Many researchers have used the general notion of posterior predictive checking (see section 1.3.3 and the references therein) to assess the fit of epidemic models (Gibson et al., 2018).

Lekone and Finkenstädt (2006) used the final size as a test statistic, in their attempt to assess a discrete time SEIR model fitted to Ebola outbreak data in the Democratic Republic of Congo in 1995, while Gardner et al. (2011) concluded that the length of the epidemic (the time of the last removal time) was the preferred statistic to assess the fit of spatio-temporal individual-level models. However, it can be argued that neither of these statistics utilize all observed temporal data information and fail to consider all the dynamics of the process.

Potentially a more informative statistic is the infection curve (curve of the cumulative number of infections as a function of time) as it incorporates temporal aspects of the data. Posterior predictive checking using infection curves is conducted in the same manner as for removal curves (see section 2.2.4.2); visually, by benchmarking the observed infection curve on an envelope of infection curves simulated from the posterior predictive distribution of the model. For example, Parry et al. (2014) employed infection curves as a way of assessing the fit of a range of a spatio-temporal models for the spread of Huanglongbing (HLB) virus in citrus orchards. Although infection curves are an appealing way of assessing the fit of an epidemic model, they are of less interest under the framework of this thesis, where the more realistic case of

observing only removal times (with infection times being unobserved) is considered (see section 1.3.4.2).

An example of a statistic that is based on observing only removal times is given in the paper of Boys and Giles (2007). There, the authors fit an SEIR model with time-dependent individual removal rates to two datasets on outbreaks of smallpox and a respiratory disease. To argue for the importance of using time dependence, the authors contrasted the marginal posterior predictive distributions of the k^{th} (in time order) removal time, for some chosen values of k, of a time-dependent and a time-homogeneous model and compared them with the observed data. However, assessment was only visual (box plots) and not quantitative; although this might be suitable for the specific problem of deciding between two models with homogeneous and inhomogeneous removal rates, more informative quantitative measures of fit are necessary for general goodness of fit assessment.

More recent, and the closest to our framework, is the work of Alharthi (2016); similar to our work the approaches developed were based on partially observed data (removal data) and the emphasis was put on the use of the removal curve. Alharthi (2016) suggested a two stage procedure for assessing the goodness of fit of epidemic models, that applies posterior predictive checking on both the final size and the removal data. At the first stage, the final size is assessed by visually inspecting whether the observed final size falls within the high density regions of its posterior predictive distribution. If the fit of the final size is adequate, the procedure progresses to the second stage, where the temporal aspects of the data are assessed using different Bayesian model assessment tools. More precisely, conditioning on the same final size as the observed, replicated epidemic realizations are generated under the posterior predictive distribution of the model. Letting $\boldsymbol{\theta}$ be the model parameter vector and $\boldsymbol{r}^{obs} = (r_1^{obs}, r_2^{obs}, \ldots, r_n^{obs})$ and $\boldsymbol{r}^{rep} = (r_1^{rep}, r_2^{rep}, \ldots, r_n^{rep})$ denote the time-ordered, observed and replicated removal times respectively, the

employed tools were the Bayesian residuals (see e.g. Gilks et al. (1996); Gelman et al. (2013)), defined as $d_k = r_k^{obs} - \mathrm{E}(r_k^{rep})$, where the expectation is taken over the posterior predictive distribution, and two discrepancies (test quantities), namely the χ^2 discrepancy (see e.g. Gelman et al. (2013)) and the Freeman-Tukey discrepancy (see e.g. Freeman and Tukey (1950)), defined as $D_{\chi^2}(\boldsymbol{r}, \boldsymbol{\theta}) = \sum_{k=1}^{n} \frac{(r_k - \mathrm{E}(r_k|\boldsymbol{\theta}))^2}{\mathrm{var}(r_k|\boldsymbol{\theta})}$ and $D_{FT}(\boldsymbol{r}, \boldsymbol{\theta}) = \sum_{k=1}^{n} (\sqrt{r_k} - \sqrt{\mathrm{E}(r_k \mid \boldsymbol{\theta})})^2$, respectively, where the expectations and variances are taken over the sampling distribution of the model under $\boldsymbol{\theta}$, for data (observed or replicated) $\boldsymbol{r} = (r_1, r_2, \ldots, r_n)$. Then the fit is assessed visually, by imposing the observed removal curve on a pack of replicated removal curves (using additionally the mean removal curve and removal curves corresponding to the 2.5^{th} or the 97.5^{th} quantile of the posterior predictive distribution) and quantitatively using the sum of the squared Bayesian residuals (SSR) and the ppp-values associated with the discrepancy measures. The advantage of this approach is that it attempts to use all observed removal data information. Simulation studies showed that the method could distinguish the true model between epidemic models with different infection mechanisms. Nonetheless, the approach also has important drawbacks. For instance, the tools used were designed to be applied to independent data settings; as discussed in section 2.2.2 Gelman et al. (2013, chapter 6) defines Bayesian residuals in the context of regression modelling, as a generalization of classical residuals, and thus directly applying them to the highly correlated epidemic data is questionable. Also, calculation of the two discrepancy measures is computationally intensive; besides the cost associated with conditioning on the same final size as the observed when creating replicated epidemic realizations under the posterior predictive distribution, there is increased cost required for calculating the terms $\mathrm{E}(r_k \mid \boldsymbol{\theta})$ and $\mathrm{var}(r_k \mid \boldsymbol{\theta})$, as for each chosen posterior value $\boldsymbol{\theta}^{(s)}$ an additional collection of realizations must be simulated from the model, conditioning on the same final size as the observed. Moreover, although the SSR is a quantitative measure of model fit, it can only be used relatively and it is rather uninformative on its own. For example, suppose that two epidemic models, A and B, are considered and SSR for model A is 100 and for

model B is 200. We may conclude that model A is a better fit than model B but we can not be sure if any of the two models actually adequately fits the data, i.e. SSR appears to be a measure more suited for model comparison rather than model assessment. Lastly, a general limitation was that the simulations studies conducted, to investigate the performance of the approach, were not extensive.

Based on the literature, it is evident that there is significant scope for innovation and more informative measures of fit, that can be routinely used by practitioners in the field, are needed.

1.4.2 Bayesian inference methods for stochastic epidemic models

Bayesian inference methods have been widely used for statistical analysis of stochastic epidemic models. One of the main reasons for this is that the Bayesian framework, offers a natural way of dealing with the problem of partial observation (typically encountered in stochastic epidemic models) by treating unobserved data as additional unknown variables. Then Bayesian inference for the model parameters can be performed by sampling from a posterior distribution consisting of both model parameters and unobserved data, via an MCMC algorithm. As mentioned in section 1.1, the challenge in the implementation of such methods, and the relevant aim of this thesis, is to come up with MCMC algorithms that can efficiently update components consisting of unobserved data, as such components are typically of high dimension and thus quite prone to mixing issues. To this end, the proceeding literature review is focused on the different existing methods used to update unobserved data.

O'Neill and Roberts (1999) and Gibson and Renshaw (1998) were the first to use data augmentation MCMC methods in the context of stochastic epidemic modelling. In O'Neill and Roberts (1999), the methodology was developed and illustrated on

the standard SIR model with Exponential infectious periods, with the unobserved data being the infection times of individuals, a setting very similar as to the one of this thesis (see for example section 1.3.5.5 further above). To update the infection component, assuming the epidemic was still in progress, a MH step was used, proposing to perform one of the following three moves, with equal probability. One, to add an infection time. Two, to remove an infection time. Three, to change one of the infection times. For any one of these moves, the associated individual was chosen uniformly at random and for moves one and three, where an infection time must be proposed, the proposed infection time was chosen from a uniform distribution on the interval of all possible values. Note that, if the epidemic is known to have ceased, only move three is possible. Essentially the same method for updating was used in Gibson and Renshaw (1998), with the only minor difference being that it was developed for an SEIR model, although again illustrated on an SIR model. This proposal mechanism, or some variation of it, has since been employed in many cases in the relevant literature (see e.g. Auranen et al. (2000); Britton and O'Neill (2002); Cauchemez et al. (2004); Streftaris and Gibson (2004b); O'Neill and Wen (2012)).

O'Neill and Becker (2001) used an alternative procedure for proposing infection times, based on a model-driven independent proposal distribution. This procedure has already been described in detail for a general model of this thesis, in section 1.3.5.3 and Algorithm 4. In the relevant reference, the authors considered an SEIR model featuring fixed and known exposure periods, Gamma(ν, λ) distributed infectious periods and randomly varying heterogeneity among susceptibles. As in the setting of this thesis and that of O'Neill and Roberts (1999) (see previous paragraph), the unobserved data were the infection times of individuals. The infection times were updated by choosing one of the ever-infected individuals, say k, uniformly at random, and proposing a candidate infection time for k, say i_k^*, as $r_k - i_k^* \sim \text{Gamma}(\nu^{(s+1)}, \lambda^{(s+1)})$, where r_k the observed removal time of individual k and $\nu^{(s+1)}$ and $\lambda^{(s+1)}$ the current values of ν and λ in the MCMC algorithm. Such type

of proposal schemes, which update one unobserved data point at a time, using model-driven proposal distributions (i.e. using as proposal distributions the corresponding sampling distributions of the model), are very commonly employed in the stochastic epidemic context (see e.g. Neal and Roberts (2004); Knock and Kypraios (2014); Alharthi (2016); Stockdale et al. (2017); Kypraios and O'Neill (2018)). It is noted that, proposal schemes that update one unobserved data point at a time, such as the ones described in the present and previous paragraph, have the appeal of typically being relatively easy to perform in practice. However, it is not hard to see how they can become very inefficient in cases that the dimension of the unobserved data is large (O'Neill, 2010).

Neal and Roberts (2005); Kypraios (2007); Jewell et al. (2009) utilized non-centered parameterizations (Papaspiliopoulos et al., 2003) to update unobserved data components. In a non-centered parameterization, the model is reparameterized so that the a priori (structural) dependence between model parameters and unobserved data, typically encountered in epidemic models, is broken. The intention behind such an approach is to make the model parameters of interest, less susceptible to inheriting mixing issues from components consisting of unobserved data. For example, consider a standard SIR model with Gamma(ν, λ) distributed infectious periods, based on observing individual removal times and not infection times, and let r_k and i_k be the removal and infection time of individual k, $k = 1, 2, \ldots, n$. Unlike the usual (centered) parametrization, that expresses the infectious period of individual k, $k = 1, 2, \ldots, n$, as $r_k - i_k \sim$ Gamma(ν, λ), the non-centered parameterization (see e.g. Jewell et al. (2009)), expresses the infectious period of individual k as $r_k - i_k = u_k/\lambda$, or equivalently writes $u_k = \lambda(r_k - i_k)$, where $u_k \sim$ Gamma$(\nu, 1)$; thus breaking the structural dependence between i_k and λ. Under such formulation, all infection times are updated at once, at the update step of λ, as updating λ (typically via a MH step using a dependent Normal proposal distribution centred around the current value) automatically updates all the infection times as well; this is because

the model assumes that $r_k - i_k = u_k/\lambda$. However such an updating scheme maintains a given ratio between the infectious periods of individuals and therefore a partially non-centered algorithm, where only some of the infection times are updated with λ, was found to be more efficient (Neal and Roberts, 2005; Kypraios, 2007). This partially non-centered algorithm, at each MCMC iteration, randomly partitions the ever-infected individuals into two sets, say $\mathcal{U}$ and $\mathcal{C}$. Then, for any individual $k \in \mathcal{U}$, as for the non-centered algorithm, one writes $u_k = \lambda(r_k - i_k) \sim \text{Gamma}(\nu, 1)$ and updates i_k by updating λ, but for any individual $k \in \mathcal{C}$, one updates i_k according to the model-driven proposal distribution used in O'Neill and Becker (2001) (and described in section 1.3.5.3 and Algorithm 4) that proposes its candidate infection time i_k^* as $r_k - i_k^* \sim \text{Gamma}(\nu^{(s+1)}, \lambda^{(s+1)})$, where $\nu^{(s+1)}$ and $\lambda^{(s+1)}$ the current values of ν and λ in the MCMC algorithm. Although, as already mentioned, the partially non-centered algorithm was found to work well, as stated in Xiang and Neal (2014), it is difficult to tune optimally in terms of how to partition individuals into $\mathcal{U}$ and $\mathcal{C}$.

The MCMC algorithms in Neal and Roberts (2005); Kypraios (2007) also make use of parameter reduction. In this context, parameter reduction refers to, when possible, analytically integrating out parameters from the target posterior distribution in order to make the target space smaller and less difficult to explore. For example, in the case of the standard SIR model with $\text{Gamma}(\nu, \lambda)$ infectious periods, it is possible to integrate out the one-to-one infection rate parameter β (see e.g. Neal and Roberts (2005)) or β and in addition λ (see e.g. Xiang and Neal (2014)). The appeal of these techniques is that they are easy to implement; samples from the posterior distribution of an integrated out parameter can easily be achieved by sampling from its respective known form full conditional distribution, conditioned on the already sampled values of the remaining components. A drawback however, is that typically the only parameters that can be analytically integrated out are model parameters and not unobserved data. Therefore, the reduction in dimension is very small.

Xiang and Neal (2014) developed an MCMC algorithm that updates many infection times at a time, in a block update step. This algorithm plays an important role for the purposes of this thesis and all of its features will be described in detail in section 4.3.1. The fundamental idea according to which infection times are proposed is the same as that in O'Neill and Becker (2001) (see section 1.3.5.3 and Algorithm 4 for a detailed description) where infection times are proposed using a model-driven proposal distribution. The difference, is that in O'Neill and Becker (2001) infections are accepted or rejected one at a time, whereas in Xiang and Neal (2014) as a block. The algorithm of Xiang and Neal (2014) also makes use of parameter reduction and a tuning procedure for optimally choosing the block step size (i.e. the number of infections to block update). Arguably, the algorithm of Xiang and Neal (2014) has shown to be the most successful in mitigating the mixing issues related to unobserved data. However, the algorithm is not without limitations as the block step size is typically chosen to be relatively small. For example, when the algorithm was applied (Xiang and Neal, 2014) to a foot and mouth disease dataset of 1021 infections, optimal algorithm performance was achieved for block step sizes around 16 and no proposed moves were ever accepted for block step sizes larger than 64.

Overall, although some of the existing methods have managed to offer welcome improvements, the mixing issues caused by the high-dimensionality of the unobserved data still persist and more efficient MCMC algorithms are required.

Chapter 2

Posterior Predictive Checking for SIR Models Based on Removal Data

2.1 Introduction

2.1.1 Chapter motivation and aims

Posterior predictive checking (see section 1.3.3 and the references therein) is an intuitive, natural and potentially very useful way to assess a model's fit within a Bayesian framework. As seen in section 1.4.1.2, posterior predictive checking has not been employed to its full potential within the stochastic epidemic context and more informative measures of fit, that would allow routine use by practitioners in the field, are needed. The main aim of this chapter is to use the posterior predictive distribution and derive quantitative measures of fit based on partially observed data. The focal point of the methods developed are disease progression curves (removal curves), that utilize all the information in the observed data and are independent of unknown quantities (see section 2.2.4.2). More specifically, the goal is to define model assessment methods based on removal curves (the observed data) and examine

their performance via simulations. The intention is to maximize the power of these methods by acknowledging the peculiarities of the epidemic setting and tailoring the methods around it.

All runs and plots in this chapter are produced using the statistical programming language R Core Team (2019).

2.1.2 Chapter layout

Section 2.2 explains how the peculiarities of the epidemic setting can complicate posterior predictive checking and motivates the need for their consideration.

Section 2.3 describes a procedure for distinguishing between minor and major outbreak realizations, given the posterior predictive distribution of an epidemic model; a task that can potentially help in substantially reducing the computational cost when producing data from the posterior predictive distribution of the model.

Section 2.4 introduces an approach that aims to alleviate the undesired high stochasticity of simulated removal curves and improve the performance of any model assessent procedure that is based on removal curves. This is done by time shifting each replicated removal curve by a suitably chosen constant.

Sections 2.5 and 2.6 define two novel posterior predictive checking methods, based on removal curves. The first method (section 2.5) revolves around the natural idea of defining a distance between removal curves and using it as a test statistic. The second method (section 2.6) assesses the plausibility of the observed removal curve, under its posterior predictive distribution, pointwise, at suitably chosen time points.

Sections 2.7, 2.8 and 2.9 examine the performance of the methods in assessing the

infectious period, the infection rate form and the population mixing assumptions of SIR models, respectively, via the use of three extensive simulation studies.

Finally, section 2.10 highlights the main accomplishments of this chapter, gives the limitations and discusses general remarks and further work.

2.2 Preliminaries

Epidemic data are endowed with inherent difficulties which complicate any model assessment procedure. As a result, before any methods for posterior predictive checking of epidemic models are designed, implemented and interpreted, an underlying appreciation of the peculiarities of the setting should be developed. This section highlights these peculiarities and pinpoints, wherever relevant, how existing approaches fall short in acknowledging them.

2.2.1 Partial observation

Under the framework assumed in this thesis (see section 1.3.4.2) large parts of the epidemic process are not observed; removal times are observed while infection times remain unobserved. Given this, there are two possible routes one can follow. The first route is to choose to work only with the available removal data and use test statistics. The second route is, as discussed in section 1.3.3.3, to impute the unobserved infection times and use test quantities. Both of the approaches would inevitably be affected by power issues; the first due to restricting to the information available from the observed data while the second due to reinforcing the model being tested (Gibson et al., 2018). Gelman (2013) recommends that measures that depend on unknown quantities should be avoided unless the amount of imputed information contained in them is very small; he illustrates this by a toy example where the imputation of a high-dimensional unobserved variable completely discards any power from the test

quantity in question to assess the fit, making the test essentially unusable. Taking into account that, under the assumed framework, the infection times are of equal dimension as the removal times, the approach taken in this work is to avoid test quantities that depend on the unobserved infection times and restrict to the use of test statistics that depend only on the observed removal times.

2.2.2 Not independent data

Epidemic data are highly correlated (see section 1.3.4.2). This means that model assessment measures that are constructed to suit independent data settings are not directly usable in the epidemic context. For example, Alharthi (2016) employed two commonly used posterior predictive distribution measures as a way of assessing the fit of epidemic models; Bayesian residuals (see e.g. Gelman et al. (2013)) and an overall goodness of fit test quantity, the χ^2discrepancy (see e.g. Gelman et al. (1996)). Although these measures appear to work in practice, they are designed to be applied to independent data settings; Gelman et al. (2013, chapter 6) defines Bayesian residuals in the context of regression modelling, as a generalization of classical residuals. Hence directly applying them to epidemic data is fundamentally questionable.

2.2.3 Single realization

Real-life epidemics are realized once, i.e. there is lack of replication. Consequently, the variability of a fitted model can not be assessed; a model's data generating process could be more, less or as stochastic as the true data generating process but there is no possible way of assessing that aspect of the data. Although there is little one can do regarding this peculiarity, it is important to acknowledge the implications it can have in the assessment of other aspects of the data.

To illustrate this, a single realization example is considered in the simplest of settings. Let $N(\mu, \sigma^2)$ denote a Normal distribution with mean μ and variance σ^2. Suppose

that a single data point $\boldsymbol{y}^{obs}$ is observed, where the true data generating distribution is $N(10,1)$. Two models are considered, M_1 and M_2, such that $\boldsymbol{y}^{obs} \sim N(\mu, \sigma_k^2)$, where μ unknown and σ_k^2 known, and prior distribution $\mu \sim N(0, s_k^2)$, for each M_k, $k = 1, 2$. Simple analytic calculations reveal that the posterior predictive distribution of a replication $\boldsymbol{y}^{rep}$ under M_k, is $N(\eta, \tau_k^2)$, where $\eta = \frac{s_k^2}{s_k^2+\sigma_k^2}\boldsymbol{y}^{obs}$ and $\tau_k^2 = \sigma_k^2 + \frac{s_k^2\sigma_k^2}{s_k^2+\sigma_k^2}$, $k = 1, 2$. Suppose that the test statistic of choice is the sample mean, which in this case (single observation) reduces to the identity function $T(\boldsymbol{y}) = \boldsymbol{y}$ for data (observed or replicated) $\boldsymbol{y}$; hence the ppp-value for M_k (see equation (1.8)) is $\Phi(\frac{\boldsymbol{y}^{obs}-\eta}{\tau_k})$, $k = 1, 2$, where Φ denotes the cumulative distribution function of a random variable $Z \sim N(0, 1)$. Setting $\sigma_1^2 = s_1^2 = 1$ and $\sigma_2^2 = s_2^2 = 50$ yields that the posterior predictive distributions for M_1 and M_2 are $N(\frac{\boldsymbol{y}^{obs}}{2}, 1.5)$ and $N(\frac{\boldsymbol{y}^{obs}}{2}, 75)$ respectively and the corresponding ppp-values are $\Phi(\frac{\boldsymbol{y}^{obs}}{2\sqrt{1.5}})$ and $\Phi(\frac{\boldsymbol{y}^{obs}}{2\sqrt{75}})$. Comparing the posterior predictive distribution of the two models with the true data generating distribution one would perhaps expect that the ppp-values for the sample mean assessment would be the same; the models have the same posterior predictive mean. However, model M_1 has a ppp-value very close to 1 and model M_2 from 0.66 to 0.72 for all possible $\boldsymbol{y}^{obs}$ that can come from the true data generating distribution $N(10, 1)$. This is a typical example of what was discussed in section 1.3.3.4, that it is not necessary that a good ppp-value implies goodness of fit for the aspect of the data under assessment. More accurately, in this case the satisfactory ppp-value for M_2 is not a real reflection of adequate fit but rather an inability to claim lack of fit due to the high variance of the model; for the less variable model M_1 the test has enough power and detects lack of fit. This weakened interpretation of the ppp-value is a direct implication of the single observation setting; if instead of a single observation a sample of observations was available, one would be able to assess the sample variance also and then the lack of fit of M_2 would be exposed.

Transferring this information to the epidemic setting, it should be expected that misspecified models would be harder to discard the more stochastic they are (as long

as the stochasticity of the posterior predictive distribution is inherited by the test statistic in question).

2.2.4 Scalar and time-statistics

In the usual random variable setting one can work with test statistics (or test quantities) which are scalar functions. However epidemic processes are stochastic processes and hence it is sometimes more natural and informative to consider non-scalar statistics that are functions of time; we refer to those as *time-statistics*.

2.2.4.1 Final size and duration

Final size and duration are the most commonly used scalar test statistics in the literature (see e.g. Gardner et al. (2011); Lekone and Finkenstädt (2006)). Let $\boldsymbol{r} = (r_1, r_2, \ldots, r_n)$ denote an n-dimensional time-ordered vector of (observed or replicated) removal times. The *final size* of the epidemic is defined as the number of initially susceptible individuals that ultimately become infected, $T_{fs}(\boldsymbol{r}) = n - 1$ (Andersson and Britton, 2000, chapter 2). The *duration* of the epidemic is the time elapsed between the first infection and the last removal (Andersson and Britton, 2000, chapter 4); to avoid dependence on unobserved infection times, we modify its definition slightly to be the time between the first and the last observed event, $T_{dur}(\boldsymbol{r}) = r_n - r_1$. The obvious advantage of using these statistics (or any scalar statistic T such that $T : \boldsymbol{r} \in \mathbb{R}^n \mapsto T(\boldsymbol{r}) \in \mathbb{R}$) is that the usual posterior predictive checking procedure (as described in section 1.3.3.1) can be implemented, allowing both visual (histogram) and quantitative assessment (ppp-value). The drawback is that although these statistics represent useful aspects of the data, their non dependence on time prevents them from utilizing all the available temporal data information, such as the progression dynamics of the process.

This can be illustrated by fitting a homogeneous Poisson process (HPP) model

to removal data generated from a standard SIR model and conducting posterior predictive checking using the final size and the duration. Let ρ denote the rate of the HPP, according to which removal times are assumed to occur, and let $[T_{\text{on}}, T_{\text{off}}]$ denote the time window which the HPP takes place, and note that ρ, T_{on} and T_{off} are unknown parameters that are estimated from the data (the definition, the likelihood and the procedure to fit the HPP using MCMC methods is given in the Appendix B.1). Note that, the HPP is a highly misspecified model for such data and it is only used to make the point in question. More specifically, the HPP does not consider infection events and assumes that removals occur at a constant rate ρ, unlike the standard SIR model that assumes that removals occur at a rate that varies with time; infections occur at an overall (all-to-all) rate of $\beta X_t Y_t$, where β, X_t and Y_t as in section 1.3.5.5, and removals are an i.i.d. shift of the infections (see the remark in section 2.2.4.2 right below). Despite these facts, the HPP typically captures final size and duration quite accurately when fited to data generated from a standard SIR model (see figure 2.1 for an example). This clearly suggests that final size and duration are not adequate statistics for assessing overall lack of fit; if one was to restrict overall model assessment to the assessment of final size and duration, they would have no evidence at all to doubt the adequacy of the HPP fit to such data.

2.2.4.2 Removal curve

The *removal curve* is a frequently used time-statistic to assess disease progression dynamics, when only removal data are available (Gibson et al., 2018). It is defined, as the cumulative number of removals at time t, that is $z_t(\boldsymbol{r}) = \sum_{k=1}^{n} \mathbb{1}_{\{r_k \leq t\}}$, where $\boldsymbol{r}$, as above, is the removal times vector. The important thing to note is that, unlike final size and duration, z_t is not scalar. More precisely, $z_t : \boldsymbol{r} \in \mathbb{R}^n \mapsto z_t(\boldsymbol{r}) \in L$, where L the space of right continuous, non-decreasing, $\mathbb{Z}_{\geq 0}$-valued functions of $t \in \mathbb{R}$. Assessment based on removal curves can be conducted visually by imposing the observed removal curve on a pack of removal curves drawn from the posterior predictive distribution of a fitted model (Gibson et al., 2018). The advantage of

using the removal curve statistic is its higher power compared to scalar statistics. For instance, unlike final size and duration, removal curves clearly expose the fit of a HPP model to removal data for its inability to capture the disease progression dynamics (see figure 2.1 for an example). The caveat though is that, unlike $\mathbb{R}$, L is not an ordered space and a quantitative measure of fit is not obviously defined; in the scalar statistics case the ppp-value (tail-area probability) is defined by implicitly utilizing the order of $\mathbb{R}$ (see equation (1.8)).

Elaborating on the point regarding the information that is contained in the removal curve statistic, it is easy to see that given z_t, for $t \in \mathbb{R}$, one can fully retrieve the removal data vector $\boldsymbol{r}$; this is because the removal curve takes non-negative-integer values, only has positive jumps of size 1 at each removal time point and remains constant elsewhere. So the removal curve statistic can actually be seen as the data. From a stochastic process point of view this makes perfect sense as the removal curve z_t is in fact a realization (sample path) of the removal process $\{Z_t\}_{t\in\mathbb{R}}$, where for each t, Z_t is the number of removed individuals in the population at time t. This property of the removal curves is vitally important and is what essentially establishes them as the focal axis of the methods developed in this chapter.

A useful remark to be made concerns the relationship between the removal curve and the infection curve, defined as the cumulative number of infections as a function of time. Recall from section 1.3.5, that within our assumed framework, removal times are an i.i.d shift of (the unobserved) infection times, i.e. $r_k - i_k \overset{\text{i.i.d.}}{\sim} T_D$, $k = 1, 2, \ldots, n$, using the notation of section 1.3.5.2. This implies that the removal curve can be seen as an i.i.d shift (proxy) of the (unobserved) infection curve with the amount of noise that is introduced in the shift being higher, the more uncertain the infectious period distribution is.

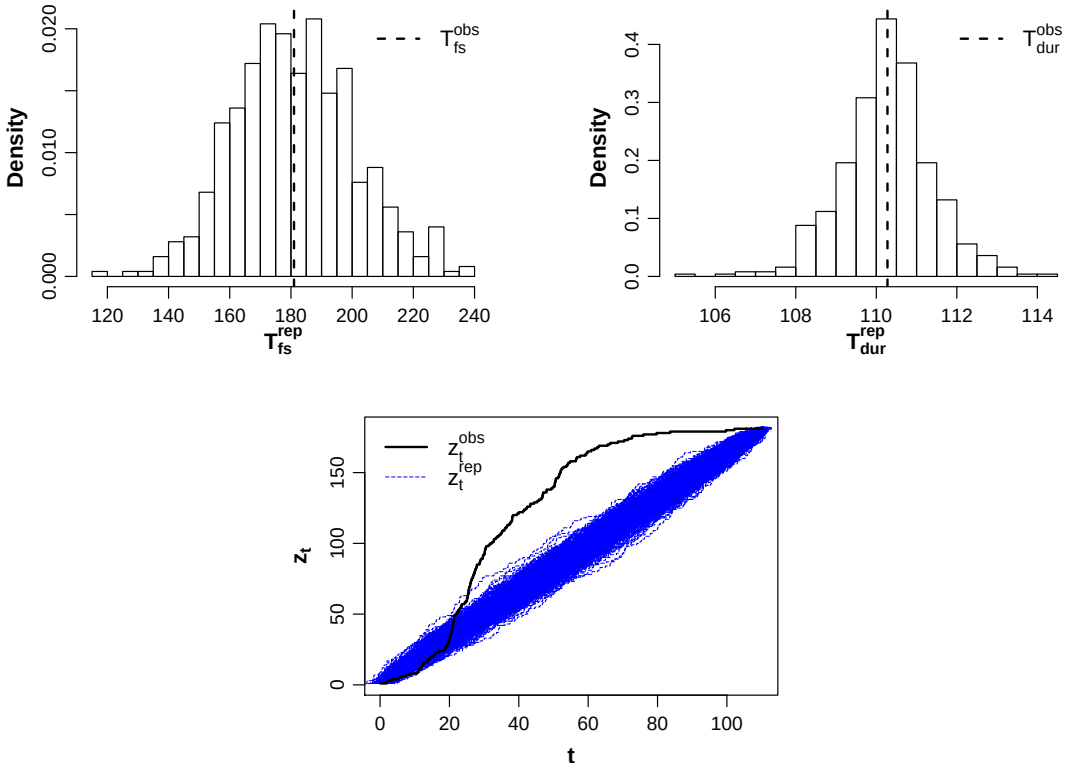


Figure 2.1: Example of posterior predictive checking where the final size and the duration fail to detect lack of fit and the removal curve does, for a clearly misspecified model. Fitted model is a HPP to removal data generated from an Exp-HM model ($N = 250$, $R_0 = 2$ and $\gamma = 0.1$). Top left plot is the histogram of 500 replications from the posterior predictive distribution of the final size T_{fs}^{rep} with the observed value of the final size $T_{fs}^{obs} = 181$ (black, dashed line) imposed. The ppp-value is 0.5. Top right plot is the histogram of 500 replications from the posterior predictive distribution of the duration T_{dur}^{rep} with the observed value of the duration $T_{dur}^{obs} = 110.3$ (black, dashed line) imposed. The ppp-value is 0.48. Bottom plot is the plot of 500 replications from the posterior predictive distribution of the removal curve (conditioned on having the same final size as the observed) z_t^{rep} with the observed removal curve z_t^{obs} (black, solid line) imposed.

2.2.5 Matched and unmatched removal curves

When employing removal curves for posterior predictive checking it is not obvious if one should allow replications of varying final size or condition on replications that have the same final size as the observed data; replications that are conditioned on having the same final size as the observed data are referred to as *matched* and replications of

varying final size as *unmatched*. The decision should be made based on practitioner preference and computational cost. For example, if a practitioner is interested in checking if a model can reproduce the observed disease progression dynamics based on the same final size, then matched removal curves should be used. Typically, this is what is done in the literature; such a decision can be accompanied by assessing the final size separately (see e.g. Alharthi (2016)). The drawback in this case is that creating matched replications is computationally expensive as before matching occurs many replications are discarded; to achieve matching rejection sampling is used. A different practitioner might be more inclined to check if a model can reproduce the observed disease progression dynamics while the final size is allowed to vary. The obvious appeal of this approach is that the computational cost of creating matched replications is avoided. Another potential advantage is that unmatched replicated removal curves could also incorporate information for the final size in the assessment and thus could be used as an omnibus goodness of fit statistic (that would assess the final size and disease progression dynamics simultaneously). The caveat though is that it is dubious if such information would be effectively incorporated; e.g. using unmatched removal curves could potentially decrease power from assessing disease progression dynamics.

Although the idea of using unmatched removal curves might appear too ambitious it has appealing reasons to be considered. It is also interesting to compare if and how any developed approaches differ when based on matched or unmatched removal curves. Thus the methods of this chapter are intended to be applicable to both matched and unmatched replications.

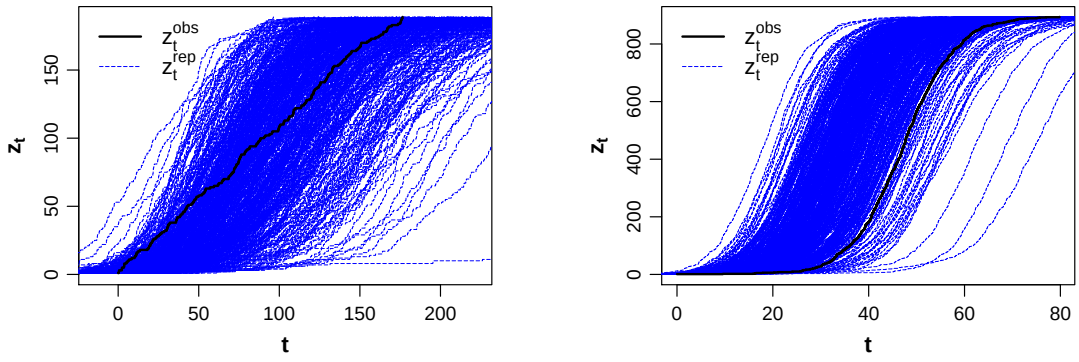
2.2.6 High stochasticity of removal curves

The intended derivation of assessment procedures based on removal curves is obstructed by a feature that they possess. Loosely, the removal curve is determined by

two components, its shape and its location in time. The issue arises from the fact that, for epidemic models, the latter component is 'very stochastic', to the point that the sampling or the posterior predictive distribution of the removal curve is clouded and monopolized by this type of noise. This downgrades the ability to extract meaningful conclusions in a visual or any potential quantitative assessment. More specifically, the power to detect lack of fit for a misspecified model is low since the pack of replicated removal curves is often too wide for the observed removal curve to appear implausible. The phenomenon is apparent even in instances of clear misspecification, such as fitting a standard SIR model to data generated from a HPP. Figure 2.2a illustrates one such example; in this case the replicated removal curves have a different shape than the observed (due to the misspecified process dynamics) but the (posterior predictive distribution) noise, around their location in time, prevents that aspect from being revealed. At the same time, this peculiarity has undesired implications in the cases that the epidemic model is correctly specified as well. More precisely, even for large datasets, it happens rather frequently that the fit of an epidemic model appears dubious when fitted to data generated from itself. One such example is given in figure 2.2b; in this instance the replicated removal curves have very similar shape as the observed (due to the correctly specified process dynamics) but the (sampling distribution) noise, around the location in time of the observed curve, places the observed removal curve on the tails of the replicated pack. Currently, no approaches in the literature acknowledge or adjust for this feature.

2.3 Cutoff for major outbreaks

A challenge in the attempt to use unmatched removal curves in posterior predictive checking arises from the fact that epidemic models frequently demonstrate threshold behaviour which, as already mentioned in section 1.3.5.5, roughly means that during the course of an epidemic, either a few individuals are infected (minor outbreak), or a fairly large number are infected (major outbreak). From an inference standpoint,



(a) (b)

Figure 2.2: Plots of 500 matched replications from the posterior predictive distribution of the removal curve (conditioned on having the same final size as the observed) z_t^{rep} with the observed removal curve z_t^{obs} (black, solid line) imposed. (a) Example where the (posterior predictive distribution) noise around the location in time of the replicated removal curves results in low power to detect a clearly misspecified model. Fitted model is a Gamma-HM model ($\nu = 10$) to data generated from a HPP ($\rho = 1$, $T_{\text{on}} = 0$, $T_{\text{off}} = 170$). (b) Example where the (sampling distribution) noise around the location in time of the observed removal curve results in doubting the fit of a correctly specified model. Fitted model is a Gamma-HM model ($\nu = 10$) to data generated from a Gamma-HM model ($N = 1000$, $R_0 = 2.5$, $\nu = 10$, $\lambda = 1$).

interest lies in major outbreaks and not epidemics that die out quickly, since for the latter parameter estimation is far less informative. As a result, one can not simply use the (unconditional) posterior predictive distribution when creating unmatched replicated removal curves and a way to condition on major outbreaks is desired. To this end, the purpose is to set a cutoff C such that replications with final size smaller than C are classified as minor outbreaks (and the remaining replications as major outbreaks). Explicitly defining minor and major outbreaks (and distinguishing between them) is not obvious. The task would perhaps be simpler if it was to be executed on the sampling distribution rather than the posterior predictive distribution. For example, for the standard SIR model, where minor and major outbreak probabilities can be both defined and calculated, using the threshold limit theorem (see section 1.3.5.5), a natural approach would be to choose C such that the

minor and major outbreak probabilities correspond to those implied by the theorem. However, similar explicit results are not available for all epidemic models (see the relevant remark in section 1.3.5.6 for the non-linear infection rate SIR model) and more importantly, any such results concern the sampling distribution of a model and not its posterior predictive distribution. Thus a different approach for choosing C is required.

Suppose that an SIR model has been fitted to removal data $\boldsymbol{r}^{obs}$ and a sample $\{T_{fs}^{rep^{(1)}}, T_{fs}^{rep^{(2)}}, \ldots, T_{fs}^{rep^{(S)}}\}$, from the posterior predictive distribution of its final size, has been achieved. Denote as $\hat{f}_{T_{fs}^{rep}}$ the empirical probability mass function (e.p.m.f.) associated with the above sample. What seems as a sensible way to choose C, is to start from the mode of the minor outbreak part at 0 and move to the right (i.e. make steps to the right) until the e.p.m.f. first starts strictly increasing. To account for the fact that in a given sample a strictly increasing step might occur by chance it is required for the e.p.m.f. to be strictly increasing for more than one consecutive steps; in practice two consecutive steps appear to suffice. That is, the cutoff is set at $C = \min\{t_{fs}^{rep} \in \{0, 1, \ldots, N\} : \hat{f}_{T_{fs}^{rep}}(t_{fs}^{rep} + 1) - \hat{f}_{T_{fs}^{rep}}(t_{fs}^{rep}) > 0 \text{ and } \hat{f}_{T_{fs}^{rep}}(t_{fs}^{rep} + 2) - \hat{f}_{T_{fs}^{rep}}(t_{fs}^{rep} + 1) > 0\}$. Then, unmatched (major outbreak) removal curves can be generated form the posterior predictive distribution, conditioning on them having final size greater or equal than C.

It is noted that, the same procedure can be repeated by requiring non-decreasing steps instead of strictly increasing steps, i.e. by setting the cutoff to be equal to $\min\{t_{fs}^{rep} \in \{0, 1, \ldots, N\} : \hat{f}_{T_{fs}^{rep}}(t_{fs}^{rep}+1) - \hat{f}_{T_{fs}^{rep}}(t_{fs}^{rep}) \geq 0 \text{ and } \hat{f}_{T_{fs}^{rep}}(t_{fs}^{rep}+2) - \hat{f}_{T_{fs}^{rep}}(t_{fs}^{rep}+1) \geq 0\}$. This latter approach places the cutoff at the right tail of the minor outbreak part, as opposed to the former approach, which places the cutoff at the left tail of the major outbreak. In cases where the minor and major outbreak parts are well separated there is not much difference, but in more 'difficult' cases, where the minor and major outbreak parts are less clearly separated, it is preferable to classify the 'in between'

replications as minor outbreaks, in order to avoid any chance of introducing noise in the assessment. For this reason, the first of the two cutoffs (i.e. the one requiring strictly increasing steps) is used in this thesis. A visual appreciation on where the two cutoffs typically lie, with respect to the posterior predictive distribution of the final size, is given by the example in figure 2.3.

Our procedure for choosing the cutoff was inspired by Demiris and O'Neill (2006). There, the authors used a similar way to separate minor from major outbreaks, in the sampling distribution of the final size, as an attempt to assess the accuracy of the branching process approximation (see section 1.3.5.5) regarding the calculation of the minor outbreak probability in standard SIR models. More specifically, starting from the mode of the minor outbreak part and moving to the right, the decision was to choose C as the first value such that the next value had probability less than some chosen $\epsilon > 0$ (the choice was $\epsilon = 10^{-3}$). To avoid the potential sensitivity in the choice of ϵ in datasets where the minor and the major outbreak part are not clearly separated, and allow routine use in simulation studies our approach was modified as described in the paragraph above.

2.4 Time shifting of removal curves

Given the issue with the high stochasticity around the location in time of a removal curve (see section 2.2.6) any assessment based on removal curves could be rather uninformative if not misleading. Thus, interventions capable of alleviating this problem are required.

2.4.1 Theoretical heuristics

Before deciding on any such intervention it is important to develop a heuristic understanding of the theory behind the cause of this feature. Consider the standard

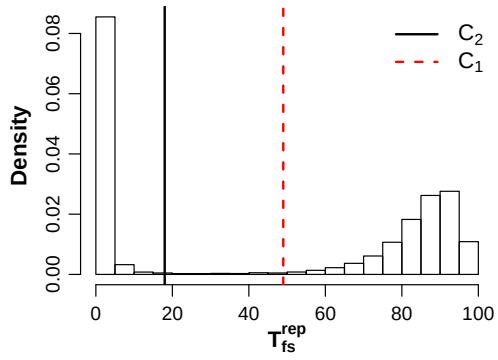


Figure 2.3: Example where cutoffs for major outbreaks are imposed on the histogram of 5000 replications from the posterior predictive distribution of final size T_{fs}^{rep}. Cutoff C_1 (red, dashed line) is given by $C_1 = \min\{t_{fs}^{rep} \in \{0, 1, \ldots, N\} : \hat{f}_{T_{fs}^{rep}}(t_{fs}^{rep} + 1) - \hat{f}_{T_{fs}^{rep}}(t_{fs}^{rep}) > 0$ and $\hat{f}_{T_{fs}^{rep}}(t_{fs}^{rep}+2) - \hat{f}_{T_{fs}^{rep}}(t_{fs}^{rep}+1) > 0\}$ and cutoff C_2 (black, solid line) by $C_2 = \min\{t_{fs}^{rep} \in \{0, 1, \ldots, N\} : \hat{f}_{T_{fs}^{rep}}(t_{fs}^{rep} + 1) - \hat{f}_{T_{fs}^{rep}}(t_{fs}^{rep}) \geq 0$ and $\hat{f}_{T_{fs}^{rep}}(t_{fs}^{rep} + 2) - \hat{f}_{T_{fs}^{rep}}(t_{fs}^{rep} + 1) \geq 0\}$. Fitted model is an Exp-HM model to data generated from an Exp-HM model ($N = 100$, $R_0 = 2.5$, $\gamma = 0.1$).

SIR model, as defined in section 1.3.5.5, and focus on the more interesting, from an inference standpoint, case of major outbreaks. Once more, the well-known branching process approximation result is called upon (see the relevant paragraph in section 1.3.5.5). As mentioned in section 1.3.5.5, this result tells us that for a population with a large number of initial susceptibles N, at the initial stages of a major epidemic, the number of infectives in the population is approximated by a branching process; infections and removals in the epidemic correspond to births and deaths in the branching process. Roughly stated, there is a time interval $[i_\alpha, t_{bp}]$ such that for $t \in [i_\alpha, t_{bp}]$, the number of infectives in the population Y_t is such that $Y_t \approx Y_t^*$ where Y_t^* denotes the number of individuals alive in the approximating branching process and t_{bp} is the time which the approximation breaks down. In turn, a standard result of branching process theory (see e.g. Haccou et al. (2005, theorem 6.3)) tells us that, for large t, $Y_t^* \approx We^{at}$, where $a > 0$ the Malthusian parameter and W a non-negative random variable which is trivial if and only if the branching process goes extinct; since the considered epidemic is a major outbreak then the approximating

branching process does not go extinct and thus W is non-trivial. Combining these two results (and noting that if N becomes large then t_{bp} becomes large also) we get that, for $t \in [i_\alpha, t_{bp}]$, $Y_t \approx We^{at}$. Now, consider the first time point $t_{\epsilon,N}$ such that ϵN infectives are present in the population (i.e. $Y_{t_{\epsilon,N}} = \epsilon N$), where $\epsilon > 0$ can be appropriately chosen so that $t_{\epsilon,N} \in [i_\alpha, t_{bp}]$. Then by taking logarithms and plugging in $t = t_{\epsilon,N}$ in the expression $Y_t \approx We^{at}$ we get, after rearranging, that $t_{\epsilon,N} = \frac{1}{a}(\log(\epsilon N) - \log(W))$. This implies that the time $t_{\epsilon,N}$ is random; this is because W is a non-trivial random variable. Loosely, this means that the initial stage of the epidemic (where the branching process approximation holds) will progress randomly and be completed at a random time, i.e. t_{bp} is random. As a result the epidemic will enter its following stage, where most of the events occur (the process takes off), at a random time; this fact is what causes the randomness around the location in time of the removal curves, that was discussed is section 2.2.6 and seen in figure 2.2.

2.4.2 Procedure and implementation

For the purposes of posterior predictive checking, where interest is in assessing the similarity between observed and replicated data, what would be more appropriate and informative is for all replicated removal curves, to exit their initial stage (and enter the following stage where most of the events occur) at approximately the same time as the observed; this can remove the undesired noise introduced from the initial stage and allow a more informative assessment. The way this is done is by horizontally (time) shifting each replicated removal curve by an appropriately chosen constant. A formal definition of what is meant by shifting a removal curve by a constant, is as follows. Let $\boldsymbol{r} = (r_1, r_2, \ldots, r_n)$ be a time ordered removal vector and $z_t(\boldsymbol{r})$ the corresponding removal curve. For a constant $c \in \mathbb{R}$, let $\boldsymbol{r} + c$ denote the vector $(r_1 + c, r_2 + c, \ldots, r_n + c)$, i.e. $\boldsymbol{r} + c := (r_1 + c, r_2 + c, \ldots, r_n + c)$. Then $\boldsymbol{r} + c$ is a shift by c of the removal vector $\boldsymbol{r}$ and accordingly the removal curve $z_t(\boldsymbol{r} + c)$ is a shift by c of the removal curve $z_t(\boldsymbol{r})$. To implement the time shifting intervention one needs to

specify the associated constant. Two different ways for doing so are described further below.

Before proceeding to describe how the time shifting constant is chosen, it must be noted that such an intervention is non-invasive for the purposes for which this assessment is conducted. Recall, from section 1.3.3.2 that, in posterior predictive checking, the concern is in obtaining a satisfactory posterior distribution. The parameters that are of interest are those that describe the dynamics of the process. For the general SIR model, as seen in section 1.3.5.5, these parameters are $\boldsymbol{\beta}$ (parameter controlling the infection process) and $\boldsymbol{\phi}$ (parameter controlling the removal process). Hence, interest is on the marginal posterior density $\pi(\boldsymbol{\beta}, \boldsymbol{\phi} \mid \boldsymbol{r})$. That being said, time shifting a replicated removal curve by a constant $c \in \mathbb{R}$ is equivalent to time shifting the observed removal curve by $-c$ and the marginal posterior density $\pi(\boldsymbol{\beta}, \boldsymbol{\phi} \mid \boldsymbol{r})$ is invariant to time shifting of the observed data (see the relevant remark in section 1.3.5.4), i.e. $\pi(\boldsymbol{\beta}, \boldsymbol{\phi} \mid \boldsymbol{r}) = \pi(\boldsymbol{\beta}, \boldsymbol{\phi} \mid \boldsymbol{r} + c)$ for any constant $c \in \mathbb{R}$.

2.4.2.1 Theoretical shifting

As mentioned in section 1.3.5.5, an outbreak from a standard SIR model, in a large population, can be approximated at the initial stages by a branching process. In particular, according to Ball and Donnelly (1995), the time until which a major outbreak grows like a branching process, denoted as $t_{\boldsymbol{i},\sqrt{N}}$, is about until $\sqrt{N}$ of the initial susceptibles become infected; that is $t_{bp} \approx t_{\boldsymbol{i},\sqrt{N}}$. Suppose temporarily that infection times $\boldsymbol{i}^{obs}$ were observed and that the focal point of the assessment was infection curves (curves of the cumulative number of infections as functions of time) instead of removal curves. Let $t_{\boldsymbol{i}^{obs},\sqrt{N}}$ be the time that $\sqrt{N}$ of the initial susceptibles become infected in the observed process. Suppose that a model has been fitted and a sample of size S has been drawn from the posterior predictive distribution of the model. Then one reasonable approach would be to pin all replicated infection curves

at the point $(t_{\boldsymbol{i}^{obs},\sqrt{N}}, \sqrt{N})$ so that the randomness of the initial stage is removed; that is to shift each replicated infection curve so that the time $t_{\boldsymbol{i}^{rep(s)},\sqrt{N}}$, for which $\sqrt{N}$ of the initial susceptibles become infected in replication s, is the same as that of the observed, $s = 1, 2, \ldots, S$. Of course, infection times are rarely observed and this can not be applied. Nonetheless a similar approach can be implemented on the removal curves. In general, since removal times $\boldsymbol{r}$ are an i.i.d. shift of the infection times $\boldsymbol{i}$ (see the relevant remark in section 2.2.4.2), it should be expected that the branching type growth in the removal curves breaks down about the time $t_{\boldsymbol{r},\sqrt{N}}$, where $\sqrt{N}$ of infective individuals become removed. Thus, what appears reasonable is to shift each replicated removal by a constant $c^{(s)}$ so that the time $t_{\boldsymbol{r}^{rep(s)},\sqrt{N}}$ for which $\sqrt{N}$ individuals are removed in replication s, is the same as the corresponding time point $t_{\boldsymbol{r}^{obs},\sqrt{N}}$ of the observed realization, $s = 1, 2, \ldots, S$. That is, to set $c^{(s)} = t_{\boldsymbol{r}^{obs},\sqrt{N}} - t_{\boldsymbol{r}^{rep(s)},\sqrt{N}}$, $s = 1, 2, \ldots, S$. We refer to this type of shifting as the *theoretical shifting*.

2.4.2.2 Distance shifting

In practice, pinning all removal curves at the time point $t_{\boldsymbol{r}^{obs},\sqrt{N}}$ appears to work reasonably well. However, due to the fact that the choice of the pinning point is based on an approximation, some information is still lost. More precisely, since the time of exit from the initial stage is approximate, in some replications it might occur at the time that some number around $\sqrt{N}$ of individuals becomes removed, and not necessarily at $t_{\boldsymbol{r}^{rep(s)},\sqrt{N}}$; so at time $t_{\boldsymbol{r}^{rep(s)},\sqrt{N}}$ the epidemic of replication s might still be in the initial phase or well into the next phase, $s = 1, 2, \ldots, S$. Another drawback is that SIR models with more general infection mechanisms do not exhibit the same behaviour as the standard SIR model and for those the choice of the 'pinning' point seems much less obvious; hence limiting the extensibility of the method to more general models. What appears as a more direct and natural way of maximizing the ability to compare disease progression dynamics, between any two removal curves, is to choose the shifting constant, so that some appropriately defined distance between them is minimized. Intuitively, if the two curves have similar (or different) shape, and

potentially different location in time, shifting one of them so that their distance is minimized would reveal their similarity (or difference) in shape. More precisely, the proposed approach shifts each replicated removal curve $z_t(\boldsymbol{r}^{rep^{(s)}})$ by a constant $c^{(s)}$, so that its distance d from the observed curve $z_t(\boldsymbol{r}^{obs})$ is minimized, i.e. it chooses $c^{(s)}$ so that $c^{(s)} = \arg\min_{c\in\mathbb{R}} d(z_t(\boldsymbol{r}^{obs}), z_t(\boldsymbol{r}^{rep^{(s)}} + c))$, for $s = 1, 2, \ldots, S$. Since this type of shifting is based on the use of some distance function it is called the *distance shifting*. The discussion regarding the specific choice of the distance function is postponed until section 2.5 where a quantitative model assessment method, based on calculating the distance between removal curves, is developed.

Algorithm 10 conveniently collects the steps for applying both of the proposed time shifting interventions.

Algorithm 10 Scheme for applying time shifting

Let $\boldsymbol{r}^{obs}$ be the time-ordered observed removal data and $\{\boldsymbol{r}^{rep^{(1)}}, \boldsymbol{r}^{rep^{(2)}}, \ldots, \boldsymbol{r}^{rep^{(S)}}\}$ a sample of replicated time-ordered removal data from the posterior predictive density of the model $\pi(\boldsymbol{r}^{rep} \mid \boldsymbol{r}^{obs})$.

1. - For theoretical shifting: Calculate $t_{\boldsymbol{r}^{obs},\sqrt{N}} = \min\{t \in \mathbb{R} : z_t(\boldsymbol{r}^{obs}) = \lceil\sqrt{N}\rceil\}$ and $t_{\boldsymbol{r}^{rep^{(s)}},\sqrt{N}} = \min\{t \in \mathbb{R} : z_t(\boldsymbol{r}^{rep^{(s)}}) = \lceil\sqrt{N}\rceil\}$, where N the number of initial susceptibles, $s = 1, 2, \ldots, S$. Calculate $c^{(s)} = t_{\boldsymbol{r}^{obs},\sqrt{N}} - t_{\boldsymbol{r}^{rep^{(s)}},\sqrt{N}}$, $s = 1, 2, \ldots, S$.
   - For distance shifting: Calculate $c^{(s)} = \arg\min_{c\in\mathbb{R}} d(z_t(\boldsymbol{r}^{obs}), z_t(\boldsymbol{r}^{rep^{(s)}} + c))$, $s = 1, 2, \ldots, S$.
2. Apply the time shifting on each replication by setting $\boldsymbol{r}^{rep^{(s)}} = \boldsymbol{r}^{rep^{(s)}} + c^{(s)}$, $s = 1, 2, \ldots, S$.

2.4.2.3 Examples

To appreciate the effect of time shifting, it is applied, following Algorithm 10, to the examples of section 2.2.6, figure 2.2. Recall that these examples highlighted why such an intervention was needed. It is noted that for visual aid, the mean removal curve,

under the posterior predictive distribution, is also imposed. Similarly to the choice of distance function, the mean removal curve is an integral part of the method that is developed in section 2.5 and thus its definition is deferred.

Figure 2.4 shows that for the example of the correctly specified model, the application of time shifting effectively removes the undesired noise around the time location of the observed removal curve. This allows for the true ability of the model to capture the data to be revealed. For both of the shifting applications the observed removal curve is placed in the middle of the pack of replicated removal curves with the imposed mean removal curve being on top of the observed. For the example where the model is clearly misspecified, both shifting types increase the power to detect the misspecification (see figure 2.4). However the power is higher for the distance shifting than the theoretical; under the latter the pack of replicated removal curves is still quite wide and the observed curve lies on top of it, while under the former it is placed on the tails and even outside the pack.

The favourable effect of the time shifting interventions is further illustrated in the simulation study of section 2.7.1 (where in addition the two time shifting methods are compared). From this point on, all methods developed in this chapter make use of time shifting.

2.5 Distance method

2.5.1 Rationale and procedure

The introduction of time shifting increases the amount of information in removal curves but still assessment is only visual and quantitative metrics need to be developed. A natural choice for a statistic on the space of removal curves is some sort of distance function. Intuitively, if the distance function is efficiently carrying

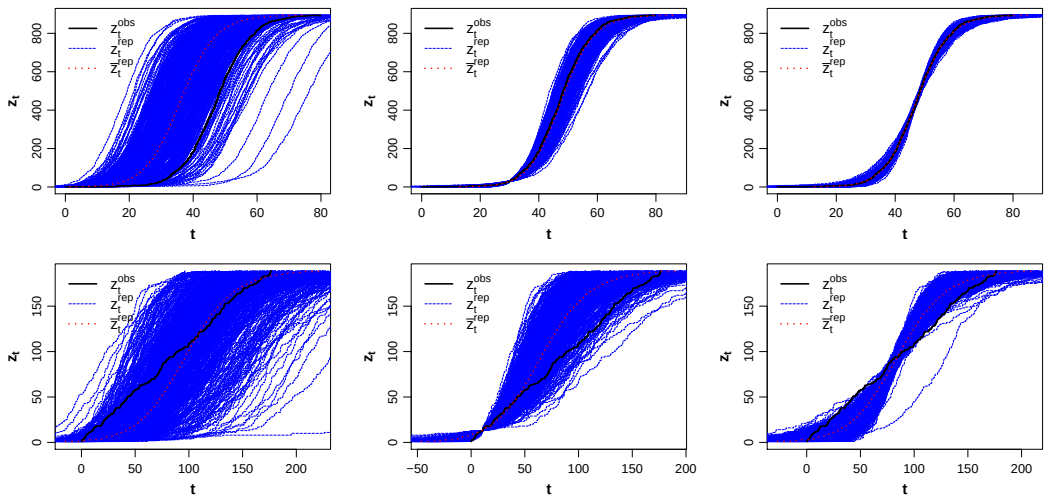


Figure 2.4: Plots of 500 matched replications from the posterior predictive distribution of the removal curve (conditioned on having the same final size as the observed) z_t^{rep} with its mean removal curve $\bar{z}_t^{rep}$ (red, dotted line) and the observed removal curve z_t^{obs} (black, solid line) imposed. For top row fitted model is a Gamma-HM model ($\nu = 10$) to data generated from a Gamma-HM model ($N = 1000$, $R_0 = 2.5$, $\nu = 10$, $\lambda = 1$); correctly specified model. For bottom row fitted model is a Gamma-HM model ($\nu = 10$) to data generated from a HPP ($\rho = 1$, $T_{\text{on}} = 0$, $T_{\text{off}} = 170$); clearly misspecified model. Left, middle and right columns correspond to applying no shifting, theoretical shifting and distance shifting, respectively.

the information contained in removal curves, then smaller distance between removal curves would imply higher similarity between the aspects of the data that are represent by removal curves.

Let $\boldsymbol{r}^{obs}$ denote the n^{obs}-dimensional, time-ordered, observed removal times and $\boldsymbol{r}^{rep}$ the n^{rep}-dimensional, time-ordered, replicated removal times, generated under the posterior predictive density $\pi(\boldsymbol{r}^{rep} \mid \boldsymbol{r}^{obs})$ of an assumed model; note that $n^{rep} - 1$ has the final size posterior predictive distribution. Then $z_t(\boldsymbol{r}^{obs})$ is the observed removal curve and $z_t(\boldsymbol{r}^{rep})$ has the posterior predictive distribution of the removal curve; to ease notation let $z_t^{obs} := z_t(\boldsymbol{r}^{obs})$ and $z_t^{rep} := z_t(\boldsymbol{r}^{rep})$. Also, let $\mathbb{E}z_t^{rep}$ denote the mean of z_t^{rep} (the definition of $\mathbb{E}z_t^{rep}$ follows in section 2.5.4). Define a scalar statistic T_d on the space of removal curves L such that $T_d(z_t) = d(z_t, \mathbb{E}z_t^{rep})$, where

L, as in section 2.2.4.2, is the space of right continuous, non-decreasing, $\mathbb{Z}_{\geq 0}$-valued functions of $t \in \mathbb{R}$. More precisely, $T_d : z_t \in L \mapsto d(z_t, \mathbb{E}z_t^{rep}) \in \mathbb{R}_{\geq 0}$, where d is a distance function on L (the choice of d follows in section 2.5.3), i.e. d is such that $d : z_t \times z_t^* \in L \times L \mapsto d(z_t, z_t^*) \in \mathbb{R}_{\geq 0}$. Plugging in $z_t = z_t^{obs}$ the test statistic assumes its observed distance $T_d(z_t^{obs}) = d(z_t^{obs}, \mathbb{E}z_t^{rep})$, which is the distance of the observed removal curve z_t^{obs} from the mean $\mathbb{E}z_t^{rep}$. For $z_t = z_t^{rep} \sim \pi(\boldsymbol{r}^{rep} \mid \boldsymbol{r}^{obs})$ the test statistic $T_d(z_t^{rep}) = d(z_t^{rep}, \mathbb{E}z_t^{rep})$ is a random variable, having the posterior predictive distribution of replicated distance, which is the distance of z_t^{rep} from the mean $\mathbb{E}z_t^{rep}$; for simplicity let $T_d^{obs} := T_d(z_t^{obs})$ and $T_d^{rep} := T_d(z_t^{rep})$. Then assessment is conducted, quantitatively and visually, in the usual fashion of posterior predictive checking (see section 1.3.3.1) by calculating the tail-area probability $P(T_d^{rep} \leq T_d^{obs})$ and by imposing T_d^{obs} on a histogram of sampled replicated values of T_d^{rep}. The idea is that if a model fits the data, z_t^{obs} must not be further from the mean $\mathbb{E}z_t^{rep}$ than z_t^{rep} is, i.e. T_d^{obs} must look plausible under T_d^{rep}. Due to its relation to distances, this method is referred to as the *distance method*.

2.5.2 Folded ppp-value and the assumption of symmetry

The distance method revolves around the idea of calculating distances from the posterior predictive mean removal curve $\mathbb{E}z_t^{rep}$. This section illuminates the precise involvement that $\mathbb{E}z_t^{rep}$ has in the procedure and the assumptions that are imposed.

Just like in section 2.2.3, revert to the simplest of settings where one single realization $\boldsymbol{y}^{obs}$ is observed (as it is the case for epidemic data) and suppose that all random quantities are random variables. Let $\boldsymbol{y}^{rep}$ be a random variable having the posterior predictive distribution of a posited model that is fit to $\boldsymbol{y}^{obs}$. Then, using as test statistic the identity function $T(\boldsymbol{y}) = \boldsymbol{y}$, the ppp-value (see equation (1.8)) is given by ppp-value $= P(\boldsymbol{y}^{rep} \leq \boldsymbol{y}^{obs})$. Now, define also the *folded ppp-value* as fppp-value $= P(|\boldsymbol{y}^{rep} - \mathrm{E}(\boldsymbol{y}^{rep})| \leq |\boldsymbol{y}^{obs} - \mathrm{E}(\boldsymbol{y}^{rep})|)$; owing its name to the fact that

the random variable $|\boldsymbol{y}^{rep} - \mathrm{E}(\boldsymbol{y}^{rep})|$ is a fold of the random variable $\boldsymbol{y}^{rep}$ at its mean $\mathrm{E}(\boldsymbol{y}^{rep})$. Then, under the assumption that $\boldsymbol{y}^{rep}$ has a symmetric distribution, it is easy to see that fppp-value $= 2|\text{ppp-value} - 0.5|$; visually this means that the position of $|\boldsymbol{y}^{obs} - \mathrm{E}(\boldsymbol{y}^{rep})|$ on the histogram of $|\boldsymbol{y}^{rep} - \mathrm{E}(\boldsymbol{y}^{rep})|$ implies two possible positions for $\boldsymbol{y}^{obs}$ on the histogram of $\boldsymbol{y}^{rep}$, which have the same distance from the mean $\mathrm{E}(\boldsymbol{y}^{rep})$. The simple deterministic relationship that connects the folded ppp-value with the ppp-value, under the assumption of symmetry, implies that the former can also be used as a sensible and interpretable measure of model fit. More specifically, folded ppp-values near 0 indicate goodness of fit (since they correspond to ppp-values near 0.5) while extreme folded ppp-values near 1 imply evidence of lack of fit (since they correspond to ppp-values near 0 or 1).

In settings as the above (where all random quantities are random variables and the ppp-value is clearly defined by utilizing the order of $\mathbb{R}$) working with the folded ppp-value and requiring symmetry for the posterior predictive distribution seems redundant. However, as discussed in section 2.2.4.2, in unordered spaces, such as the space of removal curves L or the space of n-dimensional removal vectors $\mathbb{R}^n$, the ppp-value is neither defined nor it is obviously extendable. In these cases it appears more straightforward to extend the definition of the folded ppp-value by replacing the absolute value distance with a distance function d and by requiring for the posterior predictive distribution to be symmetric (in some sense) around its mean. Specifically, for the space of removal curves L, the extended folded ppp-value is given by $P\left(d\left(z_t^{rep}, \mathbb{E}z_t^{rep}\right) \leq d\left(z_t^{obs}, \mathbb{E}z_t^{rep}\right)\right)$ where, as in section 2.5.1 d is a distance function on L and $\mathbb{E}z_t^{rep}$ a suitably defined mean of z_t^{rep}. From the definition of T_d above (see section 2.5.1), it is clear that the extended folded ppp-value coincides with the tail area probability $P(T_d^{rep} \leq T_d^{obs})$ of a posterior predictive check using the statistic T_d. That is, the tail-area probability $P(T_d^{rep} \leq T_d^{obs})$ is actually a folded ppp-value and, under the assumption that the posterior predictive distribution of the removal curve is symmetric, can be interpreted as above; the closer the values are to

0 the higher the indication of good fit and the closer the values are to 1 the more the evidence for lack of fit.

The connection between the (usual) ppp-value and the folded ppp-value in $\mathbb{R}$ is achieved in the context of observing a single realization and for test statistic the identity function. A subtle point that needs to be highlighted is that, when extending to L, these conditions are not violated. More precisely, the observed data are still a single realization and the test statistic is still the identity function, as the observed removal curve can be seen as the observed data (see discussion in the last paragraph of section 2.2.4.2).

For the folded ppp-value $P(T_d^{rep} \leq T_d^{obs})$ to be interpretable it is required for the posterior predictive distribution to be symmetric. In the space of removal curves L it is not obvious how to explicitly define and verify symmetry; even if a definition is derived, the posterior predictive distribution does not have a closed form and thus analytically checking if it is symmetric might not be possible. A pragmatic approach is taken and the assumption of symmetry is assessed by visually inspecting whether the mean removal curve lies in the center of the pack of sampled replicated removal curves.

2.5.3 Distance function

2.5.3.1 Distances on L

Let z_t and z_t^* be any two removal curves (i.e. $z_t, z_t^* \in L$ where L is defined in section 2.2.4.2) with corresponding time-ordered removal vectors $\boldsymbol{r} = (r_1, r_2, \ldots, r_n) \in \mathbb{R}^n$ and $\boldsymbol{r}^* = (r_1^*, r_2^*, \ldots, r_m^*) \in \mathbb{R}^m$, where n not necessarily equal to m (i.e. the removal curves z_t and z_t^* are not necessarily matched). A natural way to calculate the distance, between two removal curves, is to calculate the area between them. That is, define the distance function $d_{L_1}(z_t, z_t^*) = \int_{r_1 \wedge r_1^*}^{r_n \vee r_m^*} |z_t - z_t^*| dt$, where $a \wedge b$ and $a \vee b$ denote the

minimum and maximum of a and b, $a, b \in \mathbb{R}$, respectively. The apparent advantage of this choice is its intuitive interpretation. Another obvious choice is the Euclidean distance on L, given by $d_{L_2}(z_t, z_t^*) = \left(\int_{r_1 \wedge r_1^*}^{r_n \vee r_m^*} (z_t - z_t^*)^2 \, dt \right)^{\frac{1}{2}}$. The appeal in using d_{L_2}, over d_{L_1}, is that it integrates over squared differences between removal curves (rather than absolute differences as d_{L_1}) and thus removal curves that are not consistently close will be 'penalized' more; this could be more informative in the goal of assessing disease progression dynamics.

It must be noted that for both, d_{L_1} and d_{L_2}, the region over which integration is taken is chosen to be $[r_1 \wedge r_1^*, r_n \vee r_m^*]$. This is the region over which not both z_t and z_t^* are identically constant; $z_t = z_t^* = 0$ for $t \leq r_1 \wedge r_1^*$ and $z_t = n, z_t^* = m$ for $t \geq r_n \vee r_m^*$. In the case of matched removal curves, that is when $n = m$, it is worth mentioning that even if the region of integration is chosen to be the real line $\mathbb{R}$, it still reduces to $[r_1 \wedge r_1^*, r_n \vee r_m^*]$ as $z_t = z_t^* = 0$ for $t \leq r_1 \wedge r_1^*$ and $z_t = z_t^* = n$ for $t \geq r_n \vee r_m^*$. However, in the instance that removal curves are not matched, that is when $n \neq m$, choosing to integrate over $\mathbb{R}$ would cause the integral to become infinity as $z_t = n \neq m = z_t^*$ for $t \geq r_n \vee r_m^*$. This fact, besides being mathematically undesirable, is also counterintuitive as it would imply that any two removal curves that have different final size are given an infinite distance 'penalty'.

2.5.3.2 Distances on $\mathbb{R}^n$

Recall from the last paragraph of section 2.2.4.2 that one can choose to see the data as a removal curve or as a removal vector. The mindset under which the approaches of this work are developed, is choosing the former way. Nonetheless, it is useful to consider the latter route also. More precisely, by choosing to work with removal vectors, one can employ the Euclidean distance between vectors, denoted by d_{l_2}. This distance is the default choice in vector settings and often used in the epidemic

literature, in the context of approximate Bayesian computation (see e.g. Kypraios et al. (2017)). Hence, it would be very interesting to see how this choice compares with d_{L_1} and d_{L_2}. Letting $\boldsymbol{r} = (r_1, r_2, \ldots, r_n) \in \mathbb{R}^n$ and $\boldsymbol{r}^* = (r_1^*, r_2^*, \ldots, r_n^*) \in \mathbb{R}^n$, distance d_{l_2} is defined as $d_{l_2}(\boldsymbol{r}, \boldsymbol{r}^*) = \left(\sum_{k=1}^{n} (r_k - r_k^*)^2\right)^{\frac{1}{2}}$.

Note that the procedure for conducting the distance method (as described in section 2.5.1) remains unchanged, if one chooses to work with removal vectors rather than removal curves. The difference is that z_t^{obs} and z_t^{rep} are replaced by $\boldsymbol{r}^{obs}$ and $\boldsymbol{r}^{rep}$ respectively, where $z_t^{obs}, z_t^{rep}, \boldsymbol{r}^{obs}, \boldsymbol{r}^{rep}$ are as in section 2.5.1 with the restriction that the dimension of $\boldsymbol{r}^{rep}$ must equal the dimension of $\boldsymbol{r}^{obs}$, i.e. $n^{rep} = n^{obs} = n$. Then, instead of defining T_d as $T_d : z_t \in L \mapsto d(z_t, \mathbb{E}z_t^{rep}) \in \mathbb{R}_{\geq 0}$, where d such that $d : z_t \times z_t^* \in L \times L \mapsto d(z_t, z_t^*) \in \mathbb{R}_{\geq 0}$ and $\mathbb{E}z_t^{rep}$ a suitably defined mean of z_t^{rep}, one defines T_d to be $T_d : \boldsymbol{r} \in \mathbb{R}^n \mapsto d(\boldsymbol{r}, \mathbb{E}\boldsymbol{r}^{rep}) \in \mathbb{R}_{\geq 0}$, where d such that $d : \boldsymbol{r} \times \boldsymbol{r}^* \in \mathbb{R}^n \times \mathbb{R}^n \mapsto d(\boldsymbol{r}, \boldsymbol{r}^*) \in \mathbb{R}_{\geq 0}$ and $\mathbb{E}\boldsymbol{r}^{rep}$ a suitably defined mean of $\boldsymbol{r}^{rep}$.

A limitation of working with removal vectors is that the usual distances on the space of removal vectors are only defined for vectors of the same dimension, that is, of the same final size (hence the imposed restriction $n^{rep} = n^{obs} = n$ above). As a result these distances (for example d_{l_2}) can not be applied to the computationally cheaper case of unmatched data. Conversely, distances defined on the space of removal curves (for example d_{L_1} and d_{L_2}) are defined for both matched and unmatched data, as seen earlier.

2.5.4 Mean removal curve

2.5.4.1 Matched case

In the case of matched replicated removal curves, an obvious choice for defining the mean replicated removal curve is to evaluate the expected value of each time-ordered replicated removal time, under its marginal posterior predictive distribution

(see e.g. Alharthi (2016)). In more detail, let $\boldsymbol{r}^{obs} = (r_1^{obs}, r_2^{obs}, \ldots, r_{n^{obs}}^{obs})$ and $\boldsymbol{r}^{rep} = (r_1^{rep}, r_2^{rep}, \ldots, r_{n^{obs}}^{rep})$ respectively denote the observed and replicated n^{obs}-dimensional, time-ordered removal data. Note that the mean removal curve is defined after time shifting has been applied i.e. $\boldsymbol{r}^{rep}$ are the replicated removal data after time shifting. The posterior predictive mean removal curve $\mathbb{E}z_t^{rep}$ is defined as follows. First define the posterior predictive mean removal vector, denoted as $\mathbb{E}\boldsymbol{r}^{rep}$, by setting its components to be the expected values of the time-ordered replicated removal times, under their respective marginal posterior predictive distributions, i.e. as $\mathbb{E}\boldsymbol{r}^{rep} := (\mathrm{E}(r_1^{rep}), \mathrm{E}(r_2^{rep}), \ldots, \mathrm{E}(r_{n^{obs}}^{rep}))$, where $\mathrm{E}(r_k^{rep}) = \int r_k^{rep} \pi(r_k^{rep} \mid \boldsymbol{r}^{obs}) dr_k^{rep}$, $k = 1, 2, \ldots, n^{obs}$. Then the posterior predictive mean removal curve $\mathbb{E}z_t^{rep}$ is simply the removal curve that results when the removal curve statistic z_t is evaluated at the mean removal vector $\mathbb{E}\boldsymbol{r}^{rep}$, i.e. $\mathbb{E}z_t^{rep} := z_t(\mathbb{E}\boldsymbol{r}^{rep}) = \sum_{k=1}^{n} \mathbb{1}_{\{\mathrm{E}(r_k^{rep}) \leq t\}}$. It is easy to see that, as constructed above, $\mathbb{E}z_t^{rep} \in L$ and thus the required distances for implementing the distance method (see section 2.5.1) are well defined.

2.5.4.2 Unmatched case

The above definition for the mean replicated removal curve does not extend to the case of unmatched removal curves because the replicated removal data, unlike the matched case, are of varying dimension and thus it is not meaningful to work with the expected value of each time-ordered replicated removal time. Instead what appears as a sensible alternative is to define the mean removal curve by taking the expectation, of the posterior predictive distribution of the removal curve, pointwise, in an appropriately chosen interval. More specifically, let $\boldsymbol{r}^{obs} = (r_1^{obs}, r_2^{obs}, \ldots, r_{n^{obs}}^{obs})$ denote the n^{obs}-dimensional, time-ordered, observed removal times and $\boldsymbol{r}^{rep} = (r_1^{rep}, r_2^{rep}, \ldots, r_{n^{rep}}^{rep})$ the n^{rep}-dimensional, time-ordered, replicated removal times, as in section 2.5.1. Note that, as in the matched case, the mean removal curve is defined after time shifting has been applied i.e. $\boldsymbol{r}^{rep}$ are the replicated removal data after time shifting. The posterior predictive mean removal curve $\mathbb{E}z_t^{rep}$ is defined as follows. First observe that any removal curve $z_t \in L$, with corresponding time-ordered

removal vector $\boldsymbol{r} = (r_1, r_2, \ldots, r_n)$, is such that it is 0 before a time point and plateaus after some other. That is, there exist time points $t_L^{z_t}$ and $t_R^{z_t}$ such that $t_L^{z_t} = \min\{t \in \mathbb{R} : z_t > 0\}$ and $t_R^{z_t} = \min\{t \in \mathbb{R} : z_s = z_t \text{ for every } s > t\}$; it is easy to see that $t_L^{z_t} = r_1$ and $t_R^{z_t} = r_n$. The intention is for the mean removal curve $\mathbb{E}z_t^{rep}$ to mimic this behaviour; thus time points $t_L^{\mathbb{E}} = \min\{t \in \mathbb{R} : \mathbb{E}z_t^{rep} > 0\}$ and $t_R^{\mathbb{E}} = \min\{t \in \mathbb{R} : \mathbb{E}z_s^{rep} = \mathbb{E}z_t^{rep} \text{ for every } s > t\}$ need to be specified. Let $t_L^{z_t^{rep}} = \min\{t \in \mathbb{R} : z_t^{rep} > 0\}$ and $t_R^{z_t^{rep}} = \min\{t \in \mathbb{R} : z_s^{rep} = z_t^{rep} \text{ for every } s > t\}$ be the corresponding time points for z_t^{rep}; again note that $t_L^{z_t^{rep}} = r_1^{rep}$ and $t_R^{z_t^{rep}} = r_{n^{rep}}^{rep}$. Then the default choice for specifying $t_L^{\mathbb{E}}$ and $t_R^{\mathbb{E}}$ seems to be $t_L^{\mathbb{E}} := \mathrm{E}(t_L^{z_t^{rep}}) = \mathrm{E}(r_1^{rep})$ and $t_R^{\mathbb{E}} := \mathrm{E}(t_R^{z_t^{rep}}) = \mathrm{E}(r_{n^{rep}}^{rep})$. To complete the construction one needs to define the values that $\mathbb{E}z_t^{rep}$ takes in the interval $[t_L^{\mathbb{E}}, t_R^{\mathbb{E}}]$. This is done pointwise. Fix a $t \in [t_L^{\mathbb{E}}, t_R^{\mathbb{E}}]$ and note that for that fixed t, z_t^{rep} is a random variable. Then define the value of the mean removal curve at t to be the expected value of the random variable z_t^{rep}, i.e. $\mathbb{E}z_t^{rep} := \mathrm{E}(z_t^{rep})$.

The unmatched mean removal curve, as constructed above, is rather artificial in the sense that it does not behave exactly like a removal curve; instead of being a step function that jumps by 1 at some time points, it can assume real values. From a practical point of view, this is not a problem as the task assigned to the mean removal curve is to provide a reference point so that the plausibility of the observed removal curve, with respect to the replicated, can be assessed; as long as the posterior predictive distribution of removal curves appears to be symmetric around the mean removal curve then the approach can be carried out. However, reasons for concern arise from a technical standpoint as the unmatched removal curve is not strictly an element of L; although it is right continuous and non-decreasing, it is $\mathbb{R}_{\geq 0}$-valued rather than $\mathbb{Z}_{\geq 0}$-valued. Thankfully, this can be addressed rather easily. To accommodate for the unmatched removal curve, L can be extended to a more general space, denoted as $\tilde{L}$, that additionally allows for $\mathbb{R}_{\geq 0}$-valued functions. Similarly, the region of integration for the distance functions on the space can be generalized.

Specifically, if z_t and z_t^* are any two removal curves on the extended space of removal curves $\tilde{L}$, $t_L^{z_t}$ and $t_R^{z_t}$ are as above (with $t_L^{z_t^*}$ and $t_R^{z_t^*}$ the corresponding points for z_t^*), then the interval of integration is chosen to be $[t_L^{z_t} \wedge t_L^{z_t^*}, t_R^{z_t} \vee t_R^{z_t^*}]$ and thus distances involving the unmatched removal curve are well defined; in the cases that neither of z_t or z_t^* is the unmatched mean removal curve (and thus $t_L^{z_t} = r_1$, $t_R^{z_t} = r_n$, $t_L^{z_t^*} = r_1^*$ and $t_R^{z_t^*} = r_n^*$) the interval $[t_L^{z_t} \wedge t_L^{z_t^*}, t_R^{z_t} \vee t_R^{z_t^*}]$ reduces to the originally defined interval of integration $[r_1 \wedge r_1^*, r_n \vee r_m^*]$ (see section 2.5.3.1).

For clarity, all required distance calculations, for both matched and unmatched case, are gathered in Algorithm 11.

2.5.5 Implementation

Having defined the distance function and the mean removal curve, all necessary components are in order for implementing the distance method. Algorithms 12 and 13 describe the implementation steps, for the matched and the unmatched case, respectively. For illustration purposes the method is applied to an example dataset. Data are generated from an Exp-HM model and two models, namely, the Exp-HM model (correctly specified model) and the Gamma-HM model with shape parameter fixed at $\nu = 10$ (misspecified model) are fitted (as described in Algorithms 5 and 6, respectively) and assessed. The output of the assessment is given in figure 2.5. For the matched case, the method behaves reasonably by yielding a large folded ppp-value (0.95) for the misspecified model and a low one for the model that is specified correctly (0.23); recall from section 2.5.2 that the closer to 0 the folded ppp-value is, the stronger the indication of good fit, and the closer to 1 it is, the stronger the evidence for lack of fit. In the unmatched case both folded ppp-values are lower, 0.71 and 0.09 respectively. This is a combined result of the facts that the posterior predictive distribution is more uncertain, because the final size is allowed to vary (see discussion in section 2.2.3), and that both models capture the final size quite

Algorithm 11 Scheme for calculating distances

- **Case 1:** $z_t, z_t^* \in L$

  Let removal curves $z_t, z_t^* \in L$ with corresponding time-ordered removal vectors $\boldsymbol{r} = (r_1, r_2, \ldots, r_n) \in \mathbb{R}^n$ and $\boldsymbol{r}^* = (r_1^*, r_2^*, \ldots, r_m^*) \in \mathbb{R}^m$, where n not necessarily equal to m, i.e. data not necessarily matched.

  1. Sort the elements of the set $R = \{r_1, r_2, \ldots, r_n, r_1^*, r_2^*, \ldots, r_n^*\}$ in increasing order as $r_1 \wedge r_1^* = t_L^{z_t} \wedge t_L^{z_t^*} = t_1 \leq t_2 \leq \cdots \leq t_K = t_R^{z_t} \vee t_R^{z_t^*} = r_n \vee r_m^*$, where $K = m+n$.
  2. Calculate the required distance as:
     - $d_{L_1}(z_t, z_t^*) = \int_{t_1}^{t_K} |z_t - z_t^*| dt = \sum_{k=1}^{K-1} |z_{t_k} - z_{t_k}^*|(t_{k+1} - t_k) \overset{m=n}{=} \sum_{k=1}^{n} |r_k - r_k^*|$, where the last equality only holds for $m = n$, i.e. for matched data.
     - $d_{L_2}(z_t, z_t^*) = \left(\int_{t_1}^{t_K} (z_t - z_t^*)^2 \, dt \right)^{\frac{1}{2}} = \left(\sum_{k=1}^{K-1} \left(z_{t_k} - z_{t_k}^* \right)^2 (t_{k+1} - t_k) \right)^{\frac{1}{2}}$.
     - $d_{l_2}(\boldsymbol{r}, \boldsymbol{r}^*) \overset{m=n}{=} \left(\sum_{k=1}^{n} (r_k - r_k^*)^2 \right)^{\frac{1}{2}}$, where d_{l_2} is only defined for $m = n$, i.e. for matched data.

- **Case 2:** $z_t, z_t^* \in \tilde{L}$ and at least one of them $\notin L$

  Let removal curves $z_t, z_t^* \in \tilde{L}$ and at least one of them $\notin L$.

  1. Discretize the interval $[t_L^{z_t} \wedge t_L^{z_t^*}, t_R^{z_t} \vee t_R^{z_t^*}]$ by choosing a collection of equally spaced points as $t_L^{z_t} \wedge t_L^{z_t^*} = t_1 \leq t_2 \leq \cdots \leq t_K = t_R^{z_t} \vee t_R^{z_t^*}$ with K large enough so that the numerical approximation is accurate.
  2. Calculate the required distance using numerical approximation as:
     - $d_{L_1}(z_t, z_t^*) = \int_{t_1}^{t_K} |z_t - z_t^*| dt \approx \sum_{k=1}^{K-1} |z_{t_k} - z_{t_k}^*|(t_{k+1} - t_k)$.
     - $d_{L_2}(z_t, z_t^*) = \left(\int_{t_1}^{t_K} (z_t - z_t^*)^2 \, dt \right)^{\frac{1}{2}} \approx \left(\sum_{k=1}^{K-1} \left(z_{t_k} - z_{t_k}^* \right)^2 (t_{k+1} - t_k) \right)^{\frac{1}{2}}$.

accurately (see figure A.1 in Appendix). Again this behaviour of the method is desirable in the sense that in the unmatched case the final size seems to be effectively incorporated in the assessment. More specifically, for the matched case assessment is solely on disease progression dynamics, thus the 0.95 folded ppp-value for the Gamma-HM model is interpreted as strong evidence of inability of the model to reproduce the observed disease progression dynamics. On the contrary, recall from section 2.2.5 that for the unmatched case assessment is both on disease dynamics and the final size,

thus the lower folded ppp-value is a fair representation of the combined assessment of the model on these two aspects.

The results of this example suggest that the distance method might have the power to detect misspecification of the infectious period distribution, for standard SIR models. The validity of this speculation, as well as the performance of the method as a model assessment tool for more general epidemic models, are more thoroughly investigated in the simulation studies in sections 2.7.1, 2.8.1 and 2.9.1.

Algorithm 12 Scheme for implementing the distance method based on matched replications

1. **Sample from the posterior distribution:** Given time-ordered observed removal data $\boldsymbol{r}^{obs} = (r_1^{obs}, r_2^{obs}, \ldots, r_{n^{obs}}^{obs})$ fit an SIR model using MCMC methods to obtain a sample $\{\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, \ldots, \boldsymbol{\theta}^{(S')}\}$ from its posterior density $\pi(\boldsymbol{\theta} \mid \boldsymbol{r}^{obs})$, where $\boldsymbol{\theta}$ the model parameter vector.

2. **Sample from the matched posterior predictive distribution:** Choose (either by thinning or uniformly at random) $S \leq S'$ posterior values and under each chosen $\boldsymbol{\theta}^{(s)}$ simulate the model to generate matched (conditioning on $n^{rep^{(s)}} = n^{obs}$) replicated time-ordered removal data $\boldsymbol{r}^{rep^{(s)}} = (r_1^{rep^{(s)}}, r_2^{rep^{(s)}}, \ldots, r_{n^{rep^{(s)}}}^{rep^{(s)}})$ using rejection sampling, $s = 1, 2, \ldots, S$. Then $\{\boldsymbol{r}^{rep^{(1)}}, \boldsymbol{r}^{rep^{(2)}}, \ldots, \boldsymbol{r}^{rep^{(S)}}\}$ is a sample from the posterior predictive density of the model $\pi(\boldsymbol{r}^{rep} \mid \boldsymbol{r}^{obs})$ conditioned on $n^{rep^{(s)}} = n^{obs}$.

3. **Apply the time shifting:** Choose a shifting method and do as in Algorithm 10. (For distance shifting: if $d = d_{L_p}, p = 1, 2$, $c^{(s)}$ is calculated using numerical optimization and if $d = d_{l_2}$ as $c^{(s)} = \frac{1}{n^{obs}} \sum_{k=1}^{n^{obs}} (r_k^{obs} - r_k^{rep^{(s)}})$, $s = 1, 2, \ldots, S$). (To simplify notation let $z_t^{obs} := z_t(\boldsymbol{r}^{obs})$, $z_t^{rep} := z_t(\boldsymbol{r}^{rep})$ and $z_t^{rep^{(s)}} := z_t(\boldsymbol{r}^{rep^{(s)}})$, $s = 1, 2, \ldots, S$).

4. **Calculate the mean removal vector and the mean removal curve:** Calculate the expected value of each replicated removal time, under its marginal posterior predictive distribution, using Monte Carlo (MC) approximation as $\mathrm{E}(r_k^{rep}) = \int r_k^{rep} \pi(r_k^{rep} \mid \boldsymbol{r}^{obs}) dr_k^{rep} \approx \frac{1}{S} \sum_{s=1}^{S} r_k^{rep^{(s)}} = \bar{r}_k^{rep}$, $k = 1, 2, \ldots, n^{obs}$. Then the mean removal vector is approximated as $\mathbb{E}\boldsymbol{r}^{rep} := (\mathrm{E}(r_1^{rep}), \mathrm{E}(r_2^{rep}), \ldots, \mathrm{E}(r_{n_{obs}}^{rep})) \approx (\bar{r}_1^{rep}, \bar{r}_2^{rep}, \ldots, \bar{r}_{n^{obs}}^{rep}) =: \bar{\boldsymbol{r}}^{rep}$ and the mean removal curve as $\mathbb{E}z_t^{rep} := z_t(\mathbb{E}\boldsymbol{r}^{rep}) \approx z_t(\bar{\boldsymbol{r}}^{rep}) =: \bar{z}_t^{rep}$.

Algorithm 12 Scheme for implementing the distance method based on matched replications (continued)

5. **Calculate the required distances:** Set the distance d to be one of $d_{L_1}, d_{L_2}, d_{l_2}$:

   - For $d = d_{L_p}, p = 1, 2$: Calculate the (approximate) observed value of T_d as $T_d^{obs} := T_d(z_t^{obs}) = d(z_t^{obs}, \mathbb{E}z_t^{rep}) \approx d(z_t^{obs}, \bar{z}_t^{rep})$ and obtain an (approximate) sample $\{T_d^{rep^{(1)}}, T_d^{rep^{(2)}}, \ldots, T_d^{rep^{(S)}}\}$ from the posterior predictive distribution of $T_d^{rep} = T_d(z_t^{rep}) = d(z_t^{rep}, \mathbb{E}z_t^{rep})$, by calculating the (approximate) replicated distances as $T_d^{rep^{(s)}} := d(z_t^{rep^{(s)}}, \mathbb{E}z_t^{rep}) \approx d(z_t^{rep^{(s)}}, \bar{z}_t^{rep})$, $s = 1, 2, \ldots, S$.
   - For $d = d_{l_2}$: Do as above by replacing z_t^{obs} with $\boldsymbol{r}^{obs}$, z_t^{rep} with $\boldsymbol{r}^{rep}$, $z_t^{rep^{(s)}}$ with $\boldsymbol{r}^{rep^{(s)}}$, $\mathbb{E}z_t^{rep}$ with $\mathbb{E}\boldsymbol{r}^{rep}$, and $\bar{z}_t^{rep}$ with $\bar{\boldsymbol{r}}^{rep}$.

   (Distances are calculated as in Algorithm 11 Case 1).

6. **Assess the model:** Assess the model quantitatively by calculating the folded ppp-value using MC approximation as $P(T_d^{rep} \leq T_d^{obs}) \approx \frac{1}{S}\sum_{s=1}^{S} \mathbb{1}_{\{T_d^{rep^{(s)}} \leq T_d^{obs}\}}$ and visually by inspecting the histogram of replicated distances $\{T_d^{rep^{(1)}}, T_d^{rep^{(2)}}, \ldots, T_d^{rep^{(S)}}\}$ with T_d^{obs} imposed, as well as by inspecting the plot of replicated removal curves $\{z_t^{rep^{(1)}}, z_t^{rep^{(2)}}, \ldots, z_t^{rep^{(S)}}\}$ with z_t^{obs} and $\bar{z}_t^{rep}$ imposed.

Algorithm 13 Scheme for implementing the distance method based on unmatched (major outbreak) replications

1. **Sample from the posterior distribution:** Do as in step 1 of Algorithm 12.

2. **Sample from the posterior predictive distribution of the final size:** Choose (either by thinning or uniformly at random) $S'' \leq S'$ posterior values and under each chosen $\boldsymbol{\theta}^{(s)}$ simulate the model (unconditionally) to generate replicated time-ordered removal data $\boldsymbol{r}^{rep^{(s)}} = (r_1^{rep^{(s)}}, r_2^{rep^{(s)}}, \ldots, r_{n^{rep^{(s)}}}^{rep^{(s)}})$ and calculate the final size $T_{fs}(\boldsymbol{r}^{rep^{(s)}}) = n^{rep^{(s)}} - 1 =: T_{fs}^{rep^{(s)}}$, $s = 1, 2, \ldots, S''$. Then $\{T_{fs}^{rep^{(1)}}, T_{fs}^{rep^{(2)}}, \ldots, T_{fs}^{rep^{(S'')}}\}$ is a sample from the posterior predictive distribution of the final size.

3. **Calculate the cutoff for major outbreaks:** Calculate $C = \min\{t_{fs}^{rep} \in \{0, 1, \ldots, N\} : \hat{f}_{T_{fs}^{rep}}(t_{fs}^{rep} + 1) - \hat{f}_{T_{fs}^{rep}}(t_{fs}^{rep}) > 0 \text{ and } \hat{f}_{T_{fs}^{rep}}(t_{fs}^{rep} + 2) - \hat{f}_{T_{fs}^{rep}}(t_{fs}^{rep} + 1) > 0\}$ where $\hat{f}_{T_{fs}^{rep}}$ the e.p.m.f. corresponding to the sample $\{T_{fs}^{rep^{(1)}}, T_{fs}^{rep^{(2)}}, \ldots, T_{fs}^{rep^{(S'')}}\}$.

4. **Sample from the unmatched (major outbreak) posterior predictive distribution:** Choose (either by thinning or uniformly at random) $S \leq S'$ posterior values and under each chosen $\boldsymbol{\theta}^{(s)}$ simulate the model to generate unmatched (major outbreak) (conditioning on $n^{rep^{(s)}} \geq C + 1$) replicated time-ordered removal data $\boldsymbol{r}^{rep^{(s)}} = (r_1^{rep^{(s)}}, r_2^{rep^{(s)}}, \ldots, r_{n^{rep^{(s)}}}^{rep^{(s)}})$ using rejection sampling, $s = 1, 2, \ldots, S$. Then $\{\boldsymbol{r}^{rep^{(1)}}, \boldsymbol{r}^{rep^{(2)}}, \ldots, \boldsymbol{r}^{rep^{(S)}}\}$ is a sample from the posterior predictive density of the model $\pi(\boldsymbol{r}^{rep} \mid \boldsymbol{r}^{obs})$ conditioned on $n^{rep^{(s)}} \geq C + 1$.

5. **Apply the time shifting:** Do as in step 3 of Algorithm 12 (by excluding the choice $d = d_{l_2}$).

Algorithm 13 Scheme for implementing the distance method based on unmatched (major outbreak) replications (continued)

6. **Calculate the mean removal curve:**

   Approximate the mean removal curve as $\mathbb{E}z_t^{rep} \approx \bar{z}_t^{rep}$, where $\bar{z}_t^{rep}$ is constructed, using MC approximation, as follows. Choose $t_L^{\bar{z}_t^{rep}} = \min\{t \in \mathbb{R} : \bar{z}_t^{rep} > 0\}$ and $t_R^{\bar{z}_t^{rep}} = \min\{t \in \mathbb{R} : \bar{z}_s^{rep} = \bar{z}_t^{rep} \text{ for every } s > t\}$ to be $t_L^{\bar{z}_t^{rep}} := \frac{1}{S}\sum_{s=1}^{S} r_1^{rep^{(s)}} \approx \mathrm{E}(r_1^{rep}) = \mathrm{E}(t_L^{z_t^{rep}}) =: t_L^{\mathbb{E}}$ and $t_R^{\bar{z}_t^{rep}} := \frac{1}{S}\sum_{s=1}^{S} r_{n^{rep^{(s)}}}^{rep^{(s)}} \approx \mathrm{E}(r_1^{rep}) = \mathrm{E}(t_R^{z_t^{rep}}) =: t_R^{\mathbb{E}}$ respectively. Then fix a $t \in [t_L^{\bar{z}^{rep}}, t_R^{\bar{z}^{rep}}]$ and set the value of $\bar{z}_t^{rep}$ at t to be $\bar{z}_t^{rep} := \frac{1}{S}\sum_{s=1}^{S} z_t^{rep^{(s)}} \approx \mathrm{E}(z_t^{rep}) =: \mathbb{E}z_t^{rep}$.

7. **Calculate the required distances:** Do as in step 5 of Algorithm 12 (by excluding the choice $d = d_{l_2}$).

   (Distances are calculated as in Algorithm 11 Case 2).

8. **Assess the model:** Do as in step 6 of Algorithm 12.

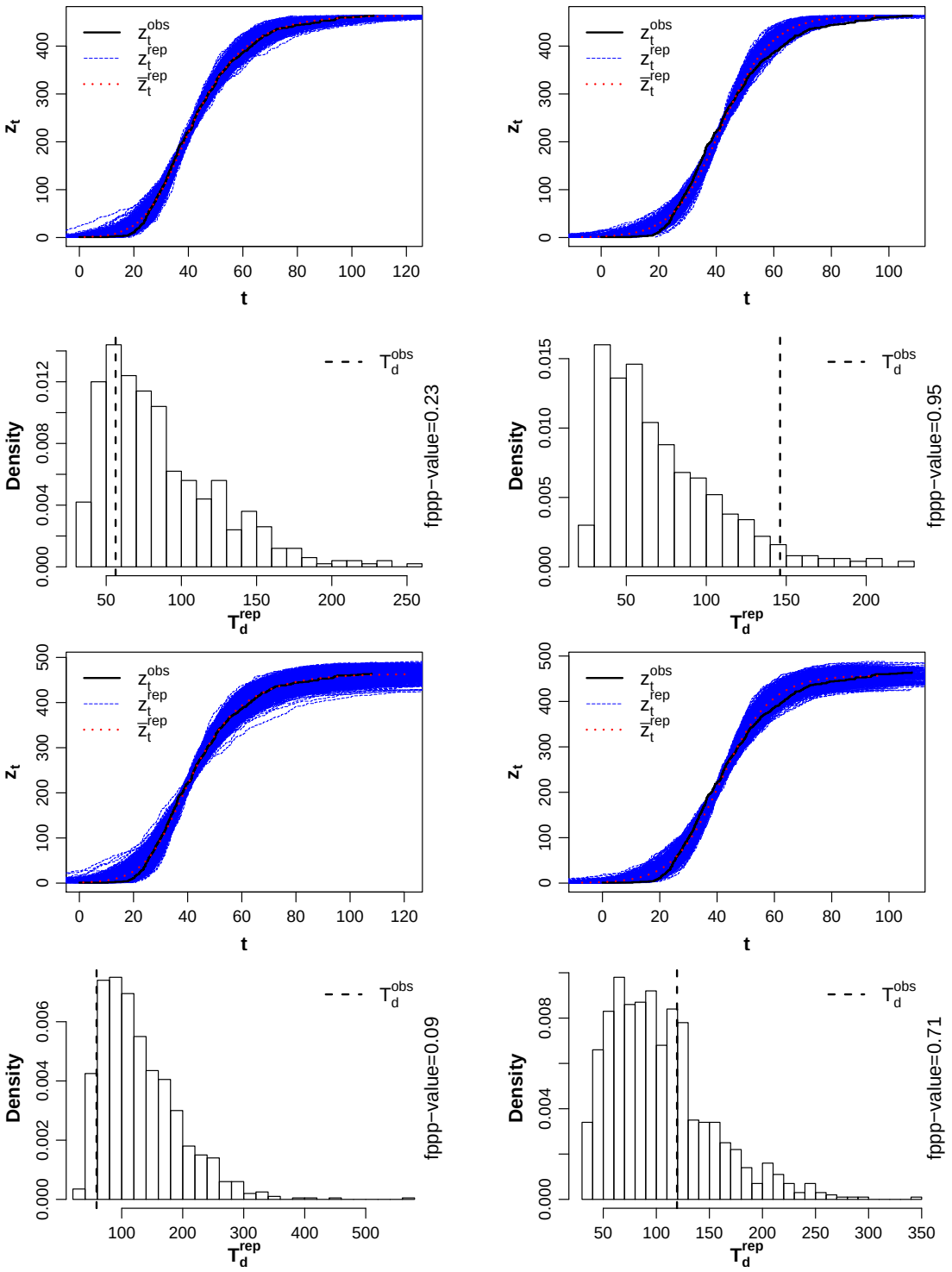


Figure 2.5: Example of posterior predictive checking using the distance method (d_{L_2} distance shifting and d_{L_2} distance function). Observed data are generated from an Exp-HM model ($N = 500$, $R_0 = 2.5$, $\gamma = 0.1$). Fitted models are the Exp-HM (left column) and the Gamma-HM ($\nu = 10$) (right column). Top two rows correspond to matched replications and bottom two to unmatched. Rows one and three are plots of 500 replications from the posterior predictive distribution of the removal curve z_t^{rep} with the mean removal curve $\bar{z}_t^{rep}$ (red, dotted line) and the observed removal curve z_t^{obs} (black, solid line) imposed. Rows two and four are histograms of 500 replications from the posterior predictive distribution of the distance T_d^{rep} with the observed distance T_d^{obs} (black, dashed line) imposed and the corresponding folded ppp-value stated.

2.6 Position-time method

2.6.1 Rationale and procedure

The distance method provides a quantitative measure of fit (the folded ppp-value) via the use of a distance function that attempts to efficiently summarize (integrate) over time. In a stochastic process setting though, it seems natural and informative to also consider quantitative measures of fit that are functions of time. Once more the goal is the same as in any posterior predictive check, to assess how plausible the observed data are under the posterior predictive distribution, only this time this is done at each time point t in an appropriately chosen time interval. More specifically, let $\boldsymbol{r}^{obs} = (r_1^{obs}, r_2^{obs}, \ldots, r_{n^{obs}}^{obs})$ denote the observed time-ordered removal vector, with associated removal curve z_t^{obs}. Suppose that a model has been fitted to $\boldsymbol{r}^{obs}$ and z_t^{rep} has the posterior predictive distribution of the removal curve. Then, the plausibility of z_t^{obs} under the distribution of z_t^{rep} can be assessed, pointwise, as follows. Fix a time point $t \in [r_1^{obs}, r_{n^{obs}}^{obs}]$, the 'interesting' interval in which z_t^{obs} is not identically constant, and note that for that fixed t, z_t^{rep} is a random variable. Specify the position of z_t^{obs} with respect to the distribution of z_t^{rep} by calculating the time-dependent (mid) ppp-value given by ppp-value$(t) = P(z_t^{rep} < z_t^{obs}) + \frac{1}{2}P(z_t^{rep} = z_t^{obs})$; the modification from the usual definition of the tail area probability (see equation (1.8)) is done to account for the fact that z_t^{rep} is a discrete random variable rather than continuous. Values closer to 0.5 would provide indication for goodness of fit (as the observed curve would lie in the middle of the pack of replicated curves) and values near 0 or 1 would imply evidence against the fit (as the observed curve would lie on the lower or upper tail of the pack of replicated curves, respectively). Owing to its nature, the method is referred to as the *position-time method*.

Having acquired the value of ppp-value(t) for $t \in [r_1^{obs}, r_{n^{obs}}^{obs}]$, there is flexibility for a range of visual and quantitative assessments. Visually, one can plot a histogram

of time dependent ppp-values calculated at a collection of equally spaced time points of $[r_1^{obs}, r_{n^{obs}}^{obs}]$; more mass near 0.5 would indicate better fit and mass near 0 and 1 a bad fit. Another idea, and arguably more informative, is to plot the function ppp-value(t) against time (a ppp-value(t) history plot); a good fit would be indicated in the cases that the function is consistently close to 0.5 and a lack of fit when the curve is consistently near 0 or 1. Quantitatively, statements for any interesting event, with respect to the posterior predictive distribution, can be made, by integrating the indicator function of the desired event over time e.g. what proportion of time does z_t^{obs} spend in a specified (inverse) quantile interval $[p_1, p_2]$ of z_t^{rep}, where $p_1, p_2 \in [0,1], p_1 \leq p_2$; for example, choosing $p_1 = 0.4$ and $p_2 = 0.6$ gives the proportion of time that z_t^{obs} spends in the (around the middle of the pack) interval $[0.4, 0.6]$. This is given analytically by $\frac{1}{r_{n^{obs}}^{obs} - r_1^{obs}} \int_{r_1^{obs}}^{r_{n^{obs}}^{obs}} \mathbb{1}_{\{\text{ppp-value}(t) \in [p_1, p_2]\}} dt$. A very informative summary follows by partitioning the space of (inverse) quantiles $[0,1]$ into intervals of length 0.1 and finding the proportion of time that z_t^{obs} spends in each of these, i.e. creating a table of quantile intervals and the corresponding proportions of time that z_t^{obs} spends in each interval.

2.6.2 Differences with the distance method

The main difference between the distance and the position-time methods is that in the latter there is no dimension reduction. Recall that in the distance method the information from the multidimensional space of removal curves L (or removal vectors $\mathbb{R}^n$) is compressed into the one-dimensional space $\mathbb{R}_{\geq 0}$ via the use of a distance statistic T_d (see section 2.5.1). The effectiveness of the method relies on how efficiently T_d can carry out this transferring of information and on the assumption that the posterior predictive distribution of the removal curve is symmetric around its mean (see section 2.5.2). Conversely, the position-time method does not use a statistic nor does it require an assumption of symmetry (in fact it does not even need a mean removal curve to be defined). Also, it allows for the possibility of determining whether the

observed curve lies on the lower or the upper tail (this corresponds to a ppp-value(t) near 0 or 1, respectively) and at which specific time points this happens; these types of information are not available with the distance method.

Another important difference between the two methods is how the information from each realization (observed and replicated) is handled. The distance method does not combine the information from different realizations; for each realization, a distance between the realization and the mean curve (or vector) is calculated, and then assessment is based on comparing these distances. Conversely, the position-time method is a pointwise approach and it gives the position of the observed curve, with respect to the pack of replicated curves, by combining the information from the replicated curves at each time point. The fact that the two methods manage this information differently is perceived as useful, as the mindset is not to choose one of the methods over the other, but rather, to utilize both in order to obtain complementary information.

2.6.3 Implementation

The position-time method can be implemented in practice, for matched and unmatched replicated data, as described by Algorithms 14 and 15, respectively. The method is exhibited on the same dataset as the distance method (see section 2.5.5) with the same two models being fitted (Exp-HM and Gamma-HM). Figure 2.6 and tables 2.1 to 2.4 present the results of the assessment. Similarly to the distance method, assessment appears to be reasonable. For the matched case, the Gamma-HM model (misspecified model) spends a proportion of 0.6 of its time at the lower tail interval $[0, 0.1]$, giving strong reasons to doubt the adequacy of its fit to the data (see table 2.2). On the contrary, the Exp-HM model (correctly specified model) spends a proportion of 0.8 of its time in the around the middle interval $[0.2, 0.8]$ (see table 2.1). In the unmatched case, both models appear to be slightly more plausible compared

to the matched case. This is in line with what occurred in the distance method and it is due to the fact that the final size is incorporated in the assessment, an aspect that both models capture (see figure A.1 in Appendix). This phenomenon is perhaps made clearer if one compares the ppp-value(t) history plots, between the matched and the unmatched cases, for both models. As seen in figure 2.6 the plots appear to be very similar up until around the later time stages where all curves plateau at a value equal to their final size (plus 1). The fact that the observed final size lies in the middle of the major outbreak part of the posterior predictive distribution of the final size, for both models (see figure A.1 in Appendix), implies that the observed removal curve lies in the middle of the pack of replicated removal curves at the later stages of the time period.

Just like the distance method, the performance of the position-time method in detecting misspecified infectious period distribution for standard SIR models, as well as in assessing the assumptions of more general epidemic models, is examined in the simulation studies in sections 2.7.1, 2.8.1 and 2.9.1.

Algorithm 14 Scheme for implementing the position-time method based on matched replications

1. **Sample from the posterior distribution:** Do as in step 1 of Algorithm 12.

2. **Sample from the matched posterior predictive distribution:** Do as in step 2 of Algorithm 12.

3. **Apply the time shifting:** Do as in step 3 of Algorithm 12.

4. **Calculate, pointwise, the position of the observed removal curve with respect to its posterior predictive distribution:** Discretize the interval $[r_1^{obs}, r_{n^{obs}}^{obs}]$ by choosing a collection of equally spaced points as $r_1^{obs} = t_1 \leq t_2 \leq \cdots \leq t_K = r_{n^{obs}}^{obs}$ with K large enough so that the numerical approximation is accurate. For each t_k calculate the time dependent ppp-value(t_k) using MC approximation as ppp-value$(t_k) = P(z_{t_k}^{rep} < z_{t_k}^{obs}) + \frac{1}{2}P(z_{t_k}^{rep} = z_{t_k}^{obs}) \approx \frac{1}{S}\sum_{s=1}^{S} \mathbb{1}_{\{z_{t_k}^{rep(s)} < z_{t_k}^{obs}\}} + \frac{1}{2S}\sum_{s=1}^{S} \mathbb{1}_{\{z_{t_k}^{rep(s)} = z_{t_k}^{obs}\}}$, $k = 1, 2, \ldots K$.

5. **Assess the model:** Assess the model quantitatively by calculating the proportion of time that z_t^{obs} spends in specified (inverse) quantile intervals $[p_1, p_2]$ of z_t^{rep}, where $p_1, p_2 \in [0, 1], p_1 \leq p_2$, using numerical approximation as $\frac{1}{r_{n^{obs}}^{obs} - r_1^{obs}} \int_{r_1^{obs}}^{r_{n^{obs}}^{obs}} \mathbb{1}_{\{\text{ppp-value}(t) \in [p_1, p_2]\}} dt \approx \frac{1}{K}\sum_{k=1}^{K} \mathbb{1}_{\{\text{ppp-value}(t_k) \in [p_1, p_2]\}}$ and visually by plotting the function ppp-value(t) against time t (a ppp-value(t) history plot), at the time points t_k, $k = 1, 2, \ldots, K$, as well as by inspecting the plot of replicated removal curves $\{z_t^{rep^{(1)}}, z_t^{rep^{(2)}}, \ldots, z_t^{rep^{(S)}}\}$ with z_t^{obs} imposed.

Algorithm 15 Scheme for implementing the position-time method based on unmatched (major outbreak) replications

1. **Sample from the posterior distribution:** Do as in step 1 of Algorithm 13.

2. **Sample from the posterior predictive distribution of the final size:** Do as in step 2 of Algorithm 13.

3. **Calculate the cutoff for major outbreaks:** Do as in step 3 of Algorithm 13.

4. **Sample from the unmatched (major outbreak) posterior predictive distribution:** Do as in step 4 of Algorithm 13.

5. **Apply the time shifting:** Do as in step 5 of Algorithm 13.

6. **Calculate, pointwise, the position of the observed removal curve with respect to its posterior predictive distribution:** Do as in step 4 of Algorithm 14.

7. **Assess the model:** Do as in step 5 of Algorithm 14.

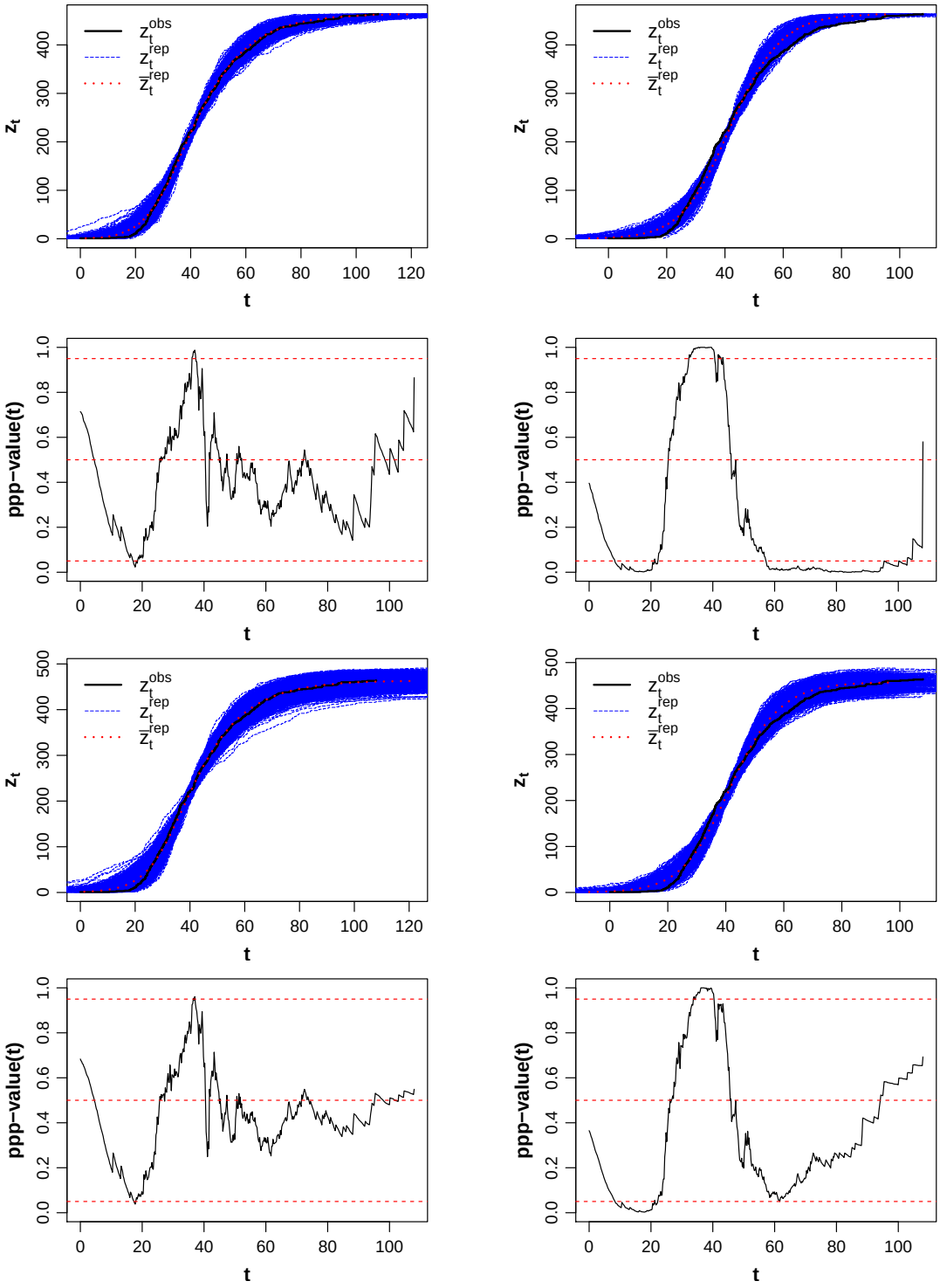


Figure 2.6: Example of posterior predictive checking using the position-time method (d_{L_2} distance shifting). Observed data are generated from an Exp-HM model ($N = 500$, $R_0 = 2.5$, $\gamma = 0.1$). Fitted models are the Exp-HM (left column) and the Gamma-HM ($\nu = 10$) (right column). Top two rows correspond to matched replications and bottom two to unmatched. Rows one and three are plots of 500 replications from the posterior predictive distribution (p.p.d.) of the removal curve z_t^{rep} with the mean removal curve $\bar{z}_t^{rep}$ (red, dotted line) and the observed removal curve z_t^{obs} (black, solid line) imposed. Rows two and four are history plots of the ppp-value(t) with the 0.1, 0.5 and 0.9 (inverse) quantiles (red,dashed lines) imposed. The proportion of time that z_t^{rep} spends at the (inverse) quantile intervals of z_t^{rep} is given in tables tables 2.1 to 2.4.

Table 2.1: Proportion of time that z_t^{rep} spends at the (inverse) quantile intervals of z_t^{rep} from the position-time method (d_{L_2} distance shifting) for the Exp-HM model, based on matched replications, for the example dataset of figure 2.6 (observed data are generated from an Exp-HM model ($N = 500$, $R_0 = 2.5$, $\gamma = 0.1$)).

quantile interval	[0,0.1]	(0.1,0.2]	(0.2,0.3]	(0.3,0.4]	(0.4,0.5]
time proportion	0.050	0.092	0.190	0.158	0.192
quantile interval	(0.5,0.6]	(0.6,0.7]	(0.7,0.8]	(0.8,0.9]	(0.9,1]
time proportion	0.148	0.084	0.032	0.036	0.018

Table 2.2: Proportion of time that z_t^{rep} spends at the (inverse) quantile intervals of z_t^{rep} from the position-time method (d_{L_2} distance shifting) for the Gamma-HM ($\nu = 10$) model, based on matched replications, for the example dataset of figure 2.6 (observed data are generated from an Exp-HM model ($N = 500$, $R_0 = 2.5$, $\gamma = 0.1$)).

quantile interval	[0,0.1]	(0.1,0.2]	(0.2,0.3]	(0.3,0.4]	(0.4,0.5]
time proportion	0.610	0.106	0.044	0.030	0.020
quantile interval	(0.5,0.6]	(0.6,0.7]	(0.7,0.8]	(0.8,0.9]	(0.9,1]
time proportion	0.016	0.018	0.014	0.040	0.102

Table 2.3: Proportion of time that z_t^{rep} spends at the (inverse) quantile intervals of z_t^{rep} from the position-time method (d_{L_2} distance shifting) for the Exp-HM model, based on unmatched replications, for the example dataset of figure 2.6 (observed data are generated from an Exp-HM model ($N = 500$, $R_0 = 2.5$, $\gamma = 0.1$)).

quantile interval	[0,0.1]	(0.1,0.2]	(0.2,0.3]	(0.3,0.4]	(0.4,0.5]
time proportion	0.050	0.062	0.060	0.252	0.274
quantile interval	(0.5,0.6]	(0.6,0.7]	(0.7,0.8]	(0.8,0.9]	(0.9,1]
time proportion	0.162	0.070	0.030	0.024	0.016

Table 2.4: Proportion of time that z_t^{rep} spends at the (inverse) quantile intervals of z_t^{rep} from the position-time method (d_{L_2} distance shifting) for the Gamma-HM ($\nu = 10$) model, based on unmatched replications, for the example dataset of figure 2.6 (observed data are generated from an Exp-HM model ($N = 500$, $R_0 = 2.5$, $\gamma = 0.1$)).

quantile interval	[0,0.1]	(0.1,0.2]	(0.2,0.3]	(0.3,0.4]	(0.4,0.5]
time proportion	0.236	0.128	0.216	0.070	0.038
quantile interval	(0.5,0.6]	(0.6,0.7]	(0.7,0.8]	(0.8,0.9]	(0.9,1]
time proportion	0.102	0.064	0.022	0.034	0.090

2.6.4 A scalar output for simulation studies

A desirable feature of the position-time method is that it is not restricted to a single quantitative output but it rather allows for a range of numerical summaries (see section 2.6.1 above). In simulation studies though, multiple datasets are considered as interest is in conducting investigations such as checking the effect of the dimension of the data, comparing results based on matched and unmatched data, comparing the

results of a model across different level of (mis)specification and comparing between the position-time and the distance methods. To conduct such investigations it appears necessary to restrict the output of the position-time method to a single scalar output. Note that restricting the output of the position-time method to a scalar is not a general suggestion and it is only employed for manipulating the results from multiple datasets in simulation studies; the rationale of the position-time method is based on the ability to provide non-scalar quantitative outputs and advocating the use of a single scalar output would be contradicting to the method itself.

Having clarified that, a sensible choice for a scalar output is to calculate the square root of the mean square error (MSE) of the collection of time-dependent ppp-values from the optimal value of 0.5. That is, to calculate $\sqrt{\text{MSE}} = \left(\frac{1}{K}\sum_{k=1}^{K}\left(\text{ppp-value}(t_k) - 0.5\right)^2\right)^{\frac{1}{2}}$, where t_k and ppp-value(t_k) are defined and calculated, respectively, as in step 4 of Algorithm 14, $k = 1, 2, \ldots, K$. Some useful reference values, that can help set a rough orientation, are $\sqrt{\frac{1}{12}} \approx 0.289$, the value of the $\sqrt{\text{MSE}}$ in the case that the collection of time-dependent ppp-values follows a uniform distribution in $[0, 1]$ (i.e. the observed curve assumes positions with respect to the replicated in a uniform way across time), and 0.5, the upper bound for the $\sqrt{\text{MSE}}$ that occurs when the time dependent ppp-value is fixed at the least optimal value of 0 or 1 (i.e. the observed curve lies outside of the pack of replicated removal curves at all time points).

For reference, the $\sqrt{\text{MSE}}$ values for the previous example (of section 2.6.3) are 0.22 and 0.19 for the Exp-HM model and 0.43 and 0.33 for the Gamma-HM model, for matched and unmatched case, respectively.

2.7 Application of the distance and the position-time methods for assessing the infectious period distribution assumption of SIR models

This section is concerned with the assessment of the infectious period distribution assumption of SIR models using the distance and the position-time methods. To conduct this assessment it makes sense to consider models that have the same infection process and differ only on the infectious period distribution assumption. Specifically, three widely used standard SIR models are considered, as defined and denoted in section 1.3.5.5, the Exp-HM, the Gamma-HM and the Constant-HM. That is to say, that the three considered choices for the infectious period T_D are Exponential ($T_D \sim \text{Exp}(\gamma)$), Gamma ($T_D \sim \text{Gamma}(\nu, \lambda)$) and constant ($T_D \equiv c$). Recall from sections 2.5.5 and 2.6.3 that the distance and the position-time methods showed promising results when used to detect misspecification of the infectious period. In this section, this speculation is further examined via an extensive simulation study, referred to as simulation study A, all details of which are described below.

2.7.1 Simulation study A

2.7.1.1 Purpose

The primary purpose of simulation study A is to examine the performance of the distance and the position-time methods as tools of assessing the infectious period distribution assumption for the three considered standard SIR models, the Exp-HM, the Gamma-HM and the Constant-HM. More precisely, interest is in investigating:

- The results of the methods when applied to matched and unmatched data under different simulation scenarios.
- The comparability of the results when methods are applied to matched and unmatched data. To make this comparison it is useful to examine the performance

of the models in capturing the final size (an examination interesting on its own). This is because when applied to matched data, the methods assess disease dynamics, whereas when applied to unmatched data, the methods simultaneously assess disease dynamics and the final size (see discussion in section 2.2.5). In other words, separately conducting posterior predictive checking for the final size can help explain any differences between matched and unmatched results.

- The comparability between the distance and the position-time methods.

In addition the following secondary investigations are conducted:

- The effect of time shifting, i.e. comparison of the performance of the methods when applying no shifting, theoretical shifting and distance shifting.
- The effect of the choice of the distance function on the distance method.

The sensible approach is to investigate all of the above under different cases of (mis)specification, i.e. both under the case that a model is correctly specified as well as when it is misspecified. Also, it is very interesting to examine if and how any trends are affected by the dimension of the observed data.

2.7.1.2 Simulation conditions

To address the tasks of the simulations study all three models are fitted to data generated under four simulation scenarios, for which the simulation conditions are summarized in table 2.5. In scenarios 1-3, data are generated from the Exp-HM, the Gamma-HM and the Constant-HM model, respectively. These three scenarios create the cases of correct specification and misspecification due to the infectious period and are tasked with addressing the main aim of the study, e.g. in scenario 1 where data are generated from the Exp-HM, the Exp-HM is a correctly specified model whereas the Gamma-HM and the Constant-HM are misspecified models. In scenario 4 data are generated from a HPP. This is a complementary scenario which creates a case of clear

model misspecification and serves for two purposes. First, it provides a way to check the credibility of the methods in the sense that the methods should be expected to detect lack of fit in such apparent cases of model misspecification before being considered for any further use. Second, it adds another case of misspecification under which the investigations of the simulation study can be conducted, therefore allowing for more informative conclusions e.g. when investigating the effect of the time shifting application it is interesting to do so not only under cases of correct specification and misspecification of the infectious periods but under cases of clear misspecification as well. We refer to the type of misspecification encountered in scenario 4 as *clear misspecification* to distinguish it from the (less clear) case of misspecification due to the infectious period distribution assumption, encountered in scenarios 1-3, which we refer to it simply as *misspecification*. To examine the effect of the dimension of the data, each scenario includes four rounds, where the number of initial susceptibles N is set at 100, 200, 500 and 1000, respectively. For each round 24 datasets are generated; this allows to capture sampling variability and to investigate the sampling properties of the model assessment measures.

In scenarios 1-3, in all rounds, the basic reproduction number R_0 is set at 2.5. The fixed value of R_0 allows for the investigation of the effect of the dimension of the data to be examined via N (common R_0 between rounds) and also ensures that datasets between scenarios are comparable, in the sense that they only differ in the infectious period distribution specification (common R_0 between scenarios). The mean infectious period $\mathrm{E}(T_D)$ is fixed at 10 between the three scenarios (again this is done for fair comparability reasons). This specifies $\gamma = 0.1$ and $c = 10$ for scenarios 1 and 3 respectively, while for scenario 2 the shape parameter ν is set at 10 (and thus the rate λ at 1) so that T_D in scenario 2 is 'different enough' from scenario 1; recall from section 1.3.5.4 that if $T_D \sim \mathrm{Gamma}(\nu, \lambda)$ with $\nu = 1$ then $T_D \sim \mathrm{Exp}(\lambda)$. Note that specification of R_0 and $E(T_D)$ determines β in all instances (see equation (1.29)). The 24 datasets of each round are generated conditioned on being major outbreaks (the

case of interest) using the approach described in section 2.3 for calculating the major outbreak cutoff; the only difference is that the approach is applied to the sampling distribution of the final size instead of its posterior predictive distribution.

In scenario 4, datasets are generated so that the number of events is around $0.85N$ (similar to scenarios 1-3). This is done by fixing, for all rounds, the HPP rate parameter as $\rho = 1$, the left time window as $T_{\text{on}} = 0$ and setting the right time window as $T_{\text{off}} = 0.85N$, while conditioning that the number of events can not exceed $N + 1$, as that would mean more events than the size of the population (see section B.1 in the Appendix for a more detailed description of the parameters of the HPP).

Table 2.5: Simulation conditions for simulation study A. Each simulation scenario consists of 4 rounds, where the number of initial susceptibles N is set at 100, 200, 500 and 1000, respectively. For each round 24 datasets are generated.

	Data generating process	**Parameter values**
Scenario 1	Exp-HM	$R_0 = 2.5, \gamma = 0.1$
Scenario 2	Gamma-HM	$R_0 = 2.5, \nu = 10, \lambda = 1$
Scenario 3	Constant-HM	$R_0 = 2.5, c = 10$
Scenario 4	HPP	$\rho = 1, T_{\text{on}} = 0, T_{\text{off}} = 0.85N$

2.7.1.3 Run conditions

The Exp-HM, Gamma-HM and Constant-HM models are fitted to each generated dataset and a sample of size 10000, after a burn-in of 1000, is achieved from the posterior distribution of the models using MCMC methods, following Algorithms 5, 6 and 7, respectively. Note that, in order to improve mixing, the infection update step is repeated as many times as the number of infections in each MCMC iteration, for all fitted models (see last paragraph of section 1.3.5.3). In all cases, the prior distribution assignment is done as in section 1.3.5.5, with the prior parameters being specified so that the uncertainty for all model parameters (except for the label of the

initial infective α, which is assigned a prior distribution as $\alpha \sim \mathrm{U}[1:n]$) is expressed via uninformative $\mathrm{Exp}(10^{-3})$ prior distributions. Next the distance and the position-time methods are applied to each model, using matched and unmatched removal data, under all possible permutations of choice of distance function and method of shifting (including no shifting), following the relevant Algorithms 12, 13, 14 and 15. In all instances, replicated datasets are created by choosing 500 posterior values using thinning (choosing every 20^{th} value). When creating matched replications (a computationally intensive process), a time limit of 15 hours is set; If 500 matched replications have not been achieved by 15 hours of runtime the methods are only applied to the unmatched case. All runs are conducted, in parallel for each dataset.

In all instances that the Gamma-HM model is fitted, the shape parameter ν is fixed at the value of 10. This serves two purposes. First, it clearly separates the cases for which the Gamma-HM model is correctly or wrongly specified and creates more interesting simulation scenarios. More precisely, in the case where the Gamma-HM model is fitted to data generated from the Exp-HM model (scenario 1), it prevents ν from being estimated close to 1 (and for Gamma-HM to revert to Exp-HM; see section 1.3.5.4) ensuring that the Gamma-HM model is misspecified; note that when data are generated from Gamma-HM (scenario 2) then it is ensured that Gamma-HM is correctly specified as ν is fixed at its true value. Second, it allows for a more accurate examination of the performance of the methods. More specifically, the Gamma-HM model has mixing issues when ν is an unknown parameter to be estimated from the data (see e.g. Kypraios (2007); Jewell et al. (2009); Alharthi (2016) where ν was also treated as known) and as a result the quality of the posterior sample and any posterior predictive check can be compromised. As interest in this simulation study is to check the performance of the model assessment methods, it is sensible to fix the shape parameter at a suitable value.

2.7.1.4 Results

Removal curve behaviour Before proceeding with the investigations it is highly important to acknowledge any similarities or differences between the removal curve behaviour of the three models. More specifically, Gamma-HM and Constant-HM produce very similar removal curves. This feature, also noticed and discussed by Alharthi (2016), can be visually appreciated in figure 2.7. Practically, the similarity between the removal curves of these two model means that the models are indistinguishable under any removal curve based assessment; in particular our methods cannot detect misspecification of the infectious period distribution when one of these models is fitted to data generated from the other. This can clearly be seen in figure 2.8; focusing on Gamma-HM (or Constant-HM) one notices that the sampling distribution of the folded ppp-values of the model is more or less the same irrespectively of which of the two models the observed data are generated from. To appreciate why this feature appears, consider the Gamma-HM model for which its infectious period T_D has a Gamma distribution with shape ν and rate λ (i.e. $T_D \sim \text{Gamma}(\nu, \lambda)$) and note that $\text{E}(T_D) = \nu/\lambda$ and $\text{var}(T_D) = \nu/\lambda^2$. Suppose that $\text{E}(T_D)$ is held fixed at a value, say c (i.e. $\text{E}(T_D) = \nu/\lambda = c$), as it is the case in the present simulation study where a common mean infectious period is set between scenarios, and, let $\nu \to \infty$. Since $\lambda = \nu/c$, then $\lambda \to \infty$ as well, and as a result $\text{var}(T_D) = \nu/\lambda^2 = c/\lambda \to 0$. This means that the Gamma(ν, λ) distribution will converge to a point mass at c (i.e. a constant distribution with parameter c) and, in particular, that the Gamma-HM model will reduce to the Constant-HM model. It appears that the value of $\nu = 10$, used in the present simulation study, is large enough to make this feature evident.

Conversely, the removal curve behaviour of the Exp-HM model is somewhat different. More precisely, it is typical that the Exp-HM model produces removal curves for which the time period, during which most of the events occur, lasts longer compared to the

Gamma-HM and the Constant-HM models; a visual appreciation of this feature is provided in the example of sections 2.5.5 and 2.6.3 (see the first row of figure 2.5 or 2.6).

In the light of these features, further investigations under the case of misspecification will refer to the instances when Exp-HM is fitted to data generated from Gamma-HM or Constant-HM and when Gamma-HM or Constant-HM are fitted to data generated from Exp-HM.

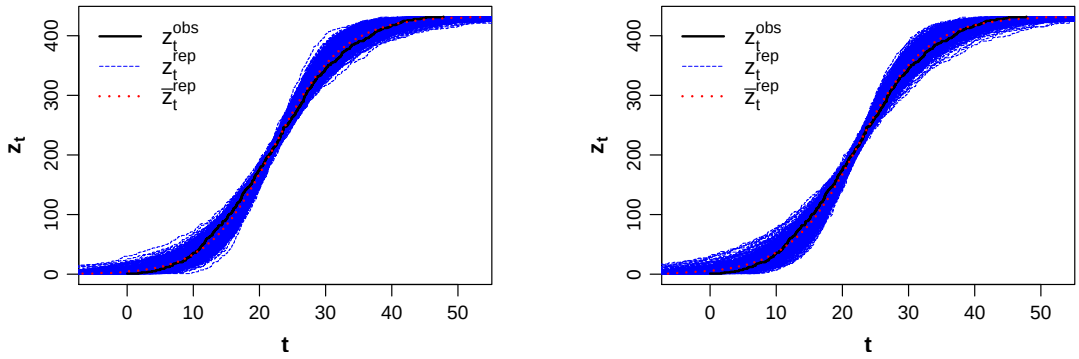


Figure 2.7: Plots of 500 matched replications from the posterior predictive distribution of the removal curve z_t^{rep} with the mean removal curve $\bar{z}_t^{rep}$ (red, dotted line) and the observed removal curve z_t^{obs} (black, solid line) imposed. Observed data is a typical dataset of round 3 ($N = 500$) in scenario 2 (data generated from a Gamma-HM) of simulation study A. Left and right plots correspond to the Gamma-HM and the Constant-HM models, respectively. For reference the folded ppp-value (d_{L_2} distance shifting, d_{L_2} distance function) and the $\sqrt{\text{MSE}}$ (d_{L_2} distance shifting) are $(0.38, 0.24)$ and $(0.37, 0.25)$ for the Gamma-HM and the Constant-HM models, respectively.

Matching and posterior predictive checking for the final size To assess a model's ability to capture the final size one can follow the usual procedure of posterior predictive checking, as described in section 1.3.3.1, and calculate the (mid) ppp-value of the final size, defined as $P(T_{fs}^{rep} < T_{fs}^{obs}) + \frac{1}{2}P(T_{fs}^{rep} = T_{fs}^{obs})$; the modification from the usual definition of the tail-area probability (see equation (1.8)) is to account for the fact that the final size is a discrete random variable rather than continuous. It

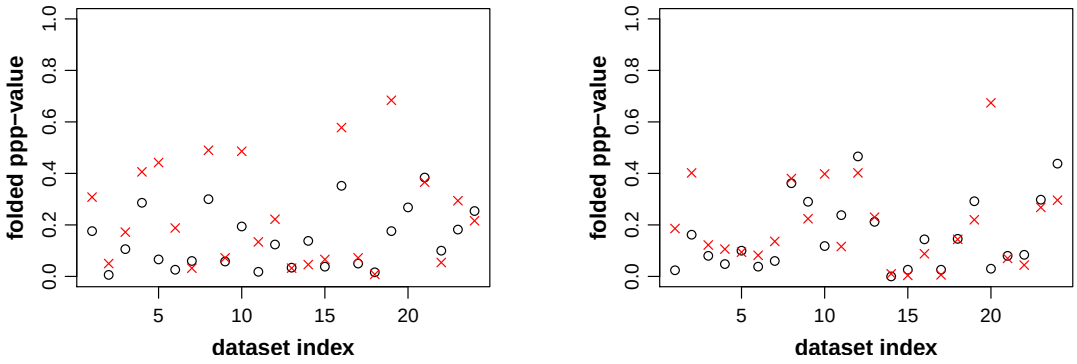


Figure 2.8: Folded ppp-value from the distance method (d_{L_2} distance shifting, d_{L_2} distance function), based on matched replications, against dataset index for the Gamma-HM (black circles) and the Constant-HM (red crosses) models for round 3 ($N = 500$) of simulation study A. Left and right plots correspond to data generated from the Gamma-HM and the Constant-HM model, respectively.

is noted that replications are conditioned on being major outbreaks, following the procedure described in section 2.3; if instead the unconditional posterior predictive distribution was used the final size ppp-value would be confounded by the mass on minor outbreaks.

Correct specification and misspecification There were 864 (3 scenarios $\times$ 4 rounds $\times$ 24 datasets $\times$ 3 fitted models) attempts to achieve matched replications (each allowed 15 hours), when standard SIR models were fitted to data generated from any standard SIR model, in scenarios 1-3. From those 864 attempts only 9 failed to complete the matching process. This is a reflection of the fact that standard SIR models accurately capture the final size, when fitted to data produced from a standard SIR model, even if the infectious period distribution is not correctly specified; this feature is also revealed in the work of other authors (see e.g. Alharthi (2016)). For example, in scenario 2 (data generated from Gamma-HM) the median (95% quantile interval) final size ppp-value (pooling all 96 datasets from the four rounds as trend between rounds was similar) was 0.56 (0.45, 0.67) for Exp-HM, 0.49 (0.41, 0.60) for Gamma-HM and 0.48 (0.35, 0.60) for Constant-HM. Clearly, final size ppp-values for

all three models are around, with high precision, the optimal value of 0.5; results were similar for scenarios 1 and 3.

Clear misspecification Conversely, when the three standard SIR models were fitted to data from the HPP (scenario 4) there were 171 failed attempts, out of 288 (1 scenario $\times$ 4 rounds $\times$ 24 datasets $\times$ 3 fitted models) for achieving matching. In this instance, this is a direct result of the inability of the models to capture the final size when fitted to data generated from a process that is distinctively different than an epidemic process. This time the final size ppp-values were much closer to the least optimal value of 1, revealing that typically the observed final size was situated on the right tail of its posterior predictive distribution. In addition, this effect becomes more and more apparent as N increases (for a visual appreciation see figure A.4 in the Appendix). For example, for the Exp-HM model the median (95% quantile interval) final size ppp-value was 0.69 (0.55, 0.92) for $N = 100$ and 1 (0.96,1) for $N = 1000$.

Detailed results, for all scenarios and rounds, of the matching procedure as well as median (95% quantile interval) ppp-values for the final size are given in tables A.1 to A.3 and tables A.4 to A.6 respectively, in the Appendix. Summarizing, simulations have shown that the final size has no power to detect misspecification of the infectious period distribution for standard SIR models. However, for clear misspecification cases, such as in scenario 4, the final size could reveal lack of fit, especially as the dimension of the data gets larger.

Effect of time shifting For investigating the effect of time shifting, the folded ppp-values from the distance approach (d_{L_2} distance function) are compared, for each dataset, under the application of no shifting, theoretical shifting and distance shifting (d_{L_2}). The effect is investigated under all cases of model (mis)specification. Under correct specification and under misspecification (i.e. scenarios 1-3), since matching was achieved for almost all instances, matched replicated data are used. In the case of

clear misspecification (scenario 4), where matching procedure was not completed for a lot of instances, unmatched replications are used. It is noted that, these numerical investigations were also conducted for all permutations of choice of distance function, matched or unmatched data, as well as for the position-time method, and conclusions were similar.

Correct specification Since in this case models are correctly specified, the desirable thing would be for the folded ppp-values to move closer to the optimal value of 0 when applying time shifting, i.e. under the application of time shifting, to move closer to the truth (reduce type I error). Indeed, both time shifting methods have this effect; less obvious for smaller datasets ($N = 100$ or $N = 200$) but more apparent for larger datasets ($N = 500$ or $N = 1000$). Figure 2.9 illustrates this effect for the rounds where $N = 1000$. The choice of the round is intentional to reiterate what was discussed in section 2.2.6, that if no shifting is applied the fit of an epidemic model can appear dubious even when fitted to data generated from itself and this phenomenon persists for large N. This is perhaps more obvious in the case of the Constant-HM model (right plot of figure 2.9) where, under no shifting application, the folded ppp-values are close to the least optimal value of 1 quite often. Comparing between the two shifting methods, it appears that the distance shifting performs better than the theoretical shifting, since the folded ppp-values are in general lower under the application of the former.

Clear misspecification In this case, models are clearly misspecified and so ideally the application of time shifting would move the folded ppp-values closer to the value of 1, i.e. increase power (reduce type II error). Once again, both shifting methods appear to have the desirable effect. It is noted that for larger values of N the misspecification becomes increasingly obvious, even without applying any time shifting and most folded ppp-values are already very close to 1. Hence the effect of time shifting is not apparent in these instances in the sense that there is not much

scope for improvement. The more interesting cases are for smaller values of N where the information is less and hence there is potential for the time shifting to increase power. Figure 2.10 illustrates this for the case of $N = 200$. As can be seen, if no time shifting is applied, quite frequently, the folded ppp-values are far from the desirable value of 1. As far as the comparison between the two shifting methods, the distance shifting seems to perform slighlty more favourable than the theoretical shifting.

Misspecification Similar to above, since models are misspecified, the desirable effect of the time shifting approaches would be to move the folded ppp-values closer to 1. Just like in the clear misspecification case this seems to be the overall effect; it is noted though that because the misspecification is less extreme in this case, the effect is more apparent for larger datasets. Figure 2.11 highlights this for $N = 1000$. Just like under the previous levels of (mis)specification, the distance-shifting application appears to move the folded ppp-values closer to the truth than the theoretical shifting.

Under all cases of (mis)specification the application of time shifting has the desirable effect; reducing type I and type II error accordingly. The two methods of shifting are comparable but the distance shifting performs slightly better under all cases. Considering also the fact that the distance shifting is readily extendable to more general epidemic models, unlike the theoretical shifting (see discussion in section 2.4.2), all further investigations are conducted under the application of the distance shifting.

Choice of distance function To compare the performance of the distance functions a similar approach to that of the time shifting examination was followed; folded ppp-values were compared, for each dataset, under each case of model (mis)specification, with values closer to the truth implying better distance function. Plots are given for the matched case, where comparison of all three distance functions is possible; recall from section 2.5.3 that d_{l_2} is only defined for matched datasets,

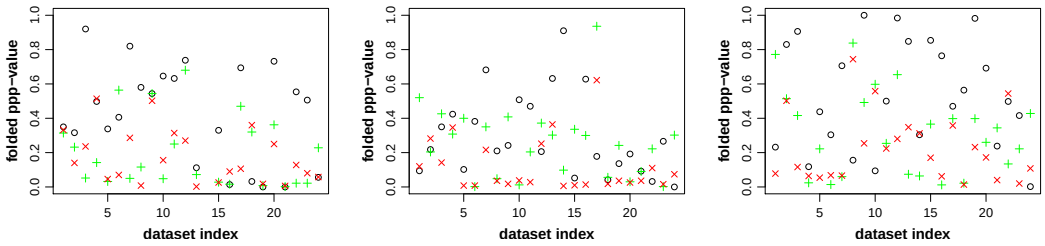


Figure 2.9: Folded ppp-value from the distance method (d_{L_2} distance function), based on matched replications, against dataset index using no shifting (black circles), theoretical shifting (green pluses) and distance shifting (d_{L_2}) (red crosses), under correct specification, for round 4 ($N = 1000$) of simulation study A. Data are generated from the fitted model. Left, middle and right plots correspond to the Exp-HM, the Gamma-HM and the Constant-HM models, respectively.

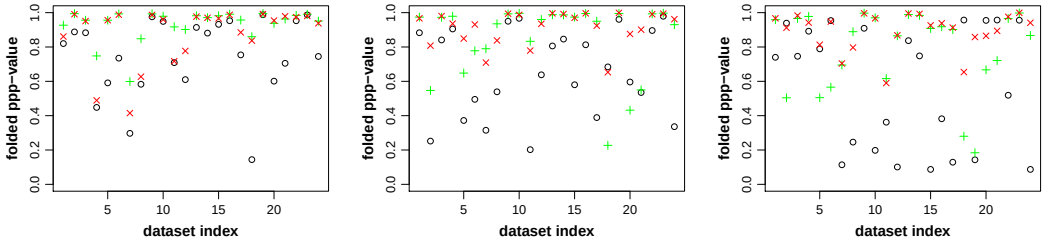


Figure 2.10: Folded ppp-value from the distance method (d_{L_2} distance function), based on unmatched replications, against dataset index using no shifting (black circles), theoretical shifting (green pluses) and distance shifting (d_{L_2}) (red crosses), under clear misspecification, for round 2 ($N = 200$) of simulation study A. Data are generated from the HPP. Left, middle and right plots correspond to the Exp-HM, the Gamma-HM and the Constant-HM models, respectively.

unlike d_{L_1} and d_{L_2} that are defined for unmatched as well.

Results showed that the performance of the distance functions was more or less similar in all scenarios and most rounds, under all cases of (mis)specification. More specifically, under clear misspecification (see figure 2.13) d_{L_1} and d_{L_2} appear to have slightly lower type II error than d_{l_2}, whereas under misspecification (see figure 2.14) this trend is reversed. In any case, these differences are minor and not decisive. Slightly better performance for d_{L_1} and d_{L_2}, compared to d_{l_2}, was exhibited for $N = 1000$ under correct model specification (see figure 2.12), that is, it appeared

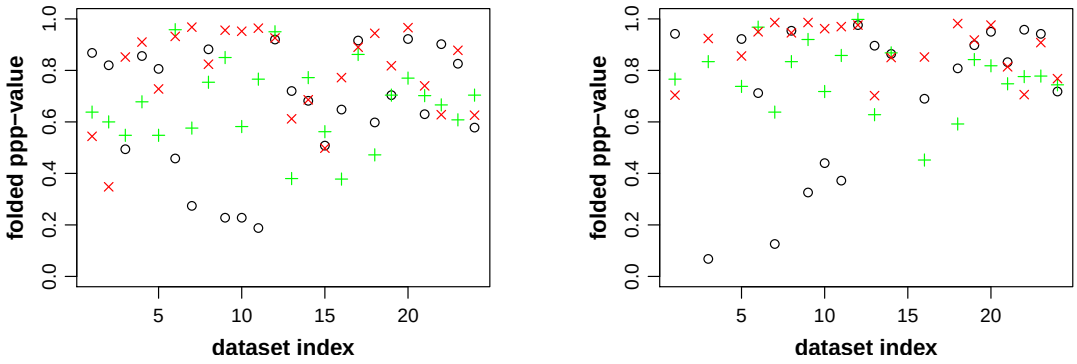


Figure 2.11: Folded ppp-value from the distance method (matched replications, d_{L_2} distance function) against dataset index using no shifting (black circles), theoretical shifting (green pluses) and distance shifting (d_{L_2}) (red crosses), under (less clear) misspecification, for round 4 ($N = 1000$) of simulation study A. Data are generated from the Exp-HM model. Left plot corresponds to the Gamma-HM model and right plot to the Constant-HM model.

that d_{L_1} and d_{L_2} could result in lower type I error; folded ppp-values are in general closer to the optimal value of 0 using d_{L_1} and d_{L_2} compared to d_{l_2}.

Overall, the performance of d_{L_1} and d_{L_2} was almost identical and perhaps slightly better compared to d_{l_2}. Taking into account the fact that d_{L_1} and d_{L_2} can be used for both matched and unmatched datasets, unlike d_{l_2}, it is preferable to choose d_{L_1} or d_{L_2}. From this point and onwards all results are illustrated using d_{L_2}.

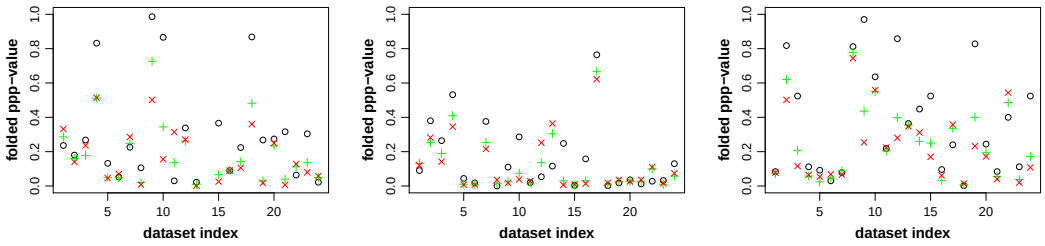


Figure 2.12: Folded ppp-value from the distance method (distance shifting), based on matched replications, against dataset index using d_{l_2} (black circles), d_{L_1} (green pluses) and d_{L_2} (red crosses) distance function, under correct specification, for round 4 ($N = 1000$) of simulation study A. Data are generated from the fitted model. Left, middle and right plots correspond to the Exp-HM, the Gamma-HM and the Constant-HM models, respectively.

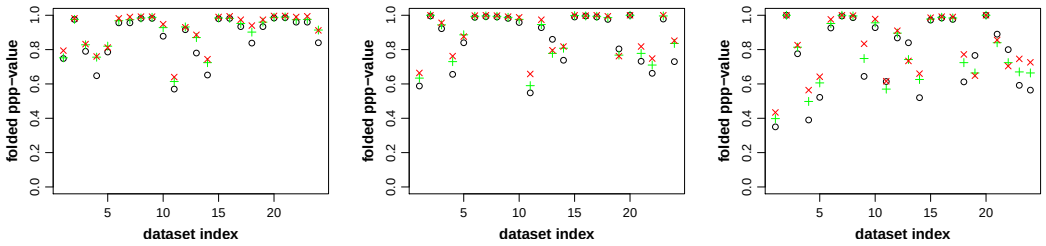


Figure 2.13: Folded ppp-value from the distance method (distance shifting), based on matched replications, against dataset index using d_{l_2} (black circles), d_{L_1} (green pluses) and d_{L_2} (red crosses) distance function, under clear misspecification, for round 1 ($N = 100$) of simulation study A. Data are generated from the HPP. Left, middle and right plots correspond to the Exp-HM, the Gamma-HM and the Constant-HM models, respectively.

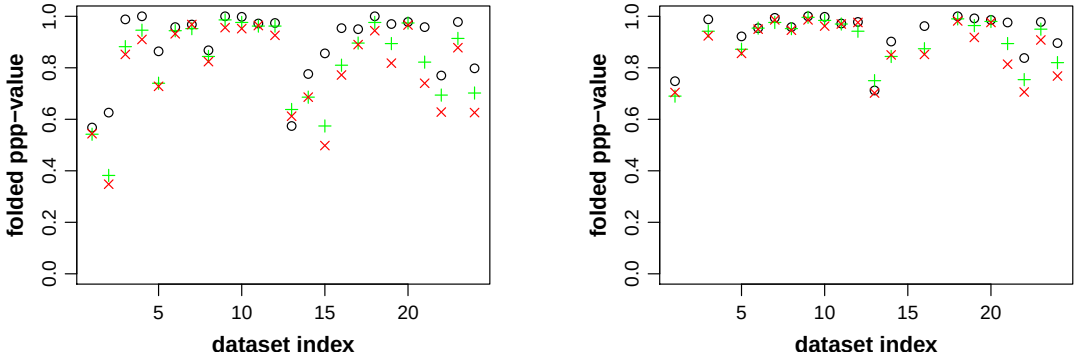


Figure 2.14: Folded ppp-value from the distance method (distance shifting), based on matched replications, against dataset index using d_{l_2} (black circles), d_{L_1} (green pluses) and d_{L_2} (red crosses) distance function,under (less clear) misspecification, for round 4 ($N = 1000$) of simulation study A. Data are generated from the Exp-HM model. Left plot corresponds to the Gamma-HM model and right plot to the Constant-HM model.

Performance of the methods To assess the performance of the distance and the position-time methods, folded ppp-values and $\sqrt{\text{MSE}}$ values are examined for each model, under all cases of mis(specification); recall that the interpretation of the folded ppp-value is given in section 2.5.2 while the $\sqrt{\text{MSE}}$ was defined in section 2.6.4, as a way of summarizing the output of the position-time method for the purposes of simulation studies. The desirable outcome is to have values close to the truth, i.e. in general lower values under correct specification and higher under misspecification.

Particular interest is given to the effect of N and whether it is sensible. More specifically, under any case of misspecification, the sensible behaviour would be for values to increase as N increases, i.e. more data should allow more power to detect lack of fit. Under correct specification, it is desired for values to be either independent of N (e.g. the p-value in the classical setting has a uniform sampling distribution, when the model is true, independently of the dimension of the data) or get smaller as N increases; in any case they should not move to the opposite direction of the truth.

For each model, results are summarized in tables that give the median (95% quantile interval) folded ppp-value and $\sqrt{\text{MSE}}$, for each value of N, under all (mis)specification cases (tables 2.6 to 2.17); these tables allow for the investigation of the effect of N (by choosing a row and looking across columns). For completeness, all (for each generated dataset) folded ppp-values and $\sqrt{\text{MSE}}$ values, are plotted in the Appendix, under each (mis)specification case (figures A.5 to A.10); these plots are particularly useful in providing a visual appreciation on the effect of N (by choosing a column and looking across rows) and on the comparison between matched and unmatched assessment.

Correct specification: distance method Focusing on matched replications assessment, it can be seen from tables 2.6, 2.8 and 2.10 that the folded ppp-values appear sensible for all three models; generally the ppp-values are closer to 0 than 1 and indicate goodness of fit for the models. For Exp-HM (see table 2.6) and Gamma-HM (see table 2.8), as N increases, it appears that the folded ppp-values move closer to the optimal value of 0 whereas for Constant-HM (see table 2.10) the values appear more or less independent of N; in any case the effect of N is sensible.

For the unmatched results, the pattern is very similar with the only difference that the folded ppp-values are in general lower as can be seen from tables 2.7, 2.9 and 2.11. Once again this is sensible as it reflects the facts that (as discussed in section 2.5.5)

the posterior predictive distribution is more uncertain and (as seen earlier in section 2.7.1.4) the models adequately capture the final size. For detailed results on each dataset, under correct specification, see figure A.5 in the Appendix.

Correct specification: position-time method For the matched case, just like the folded ppp-values, the $\sqrt{\text{MSE}}$ values appear sensible (see tables 2.12, 2.14 and 2.16); the median (95% quantile interval) of the $\sqrt{\text{MSE}}$ values (pooling over N as trend was similar between rounds) was 0.21 (0.13, 0.35), 0.19 (0.13, 0.32) and 0.21 (0.13, 0.34) for the Exp-HM, the Gamma-HM and the Constant-HM model, respectively, which is well below the least favourable value of 0.5. Unlike the distance method, it appears that N does not have an effect on the results as the sampling distribution of the $\sqrt{\text{MSE}}$ values is very similar for all values of N.

For the unmatched case the same conclusions as for the distance method apply; values are generally lower due to the incorporation of the final size in the assessment (see tables 2.13, 2.15 and 2.17). Figure A.6 in the Appendix illustrates all $\sqrt{\text{MSE}}$ values for each dataset, under correct specification.

Clear misspecification: distance method Looking at tables 2.6, 2.8 and 2.10 it is obvious that, under clear misspecification (scenario 4), matched folded ppp-values are as derised very close to 1, for all three models, revealing the clear lack of fit. The effect of N on the results is not easily concluded for the matched case as folded ppp-values are mostly available for the rounds of $N = 100$ and $N = 200$ and not for larger N where matching could not be achieved due to the inability of the models to capture the final size (see discussion above in section 2.7.1.4 and tables A.1 to A.3 in the Appendix).

For the unmatched results, folded ppp-values are available for all values of N thus making it possible to investigate the effect of the dimension of the data. It is clear

from tables 2.7, 2.9 and 2.11 that as N increases the folded ppp-values move closer to 1; in fact this behaviour is obvious from $N = 200$. For the rounds of $N = 100$ and $N = 200$, where matched folded ppp-values are also available and comparisons can be made, it is observed that the unmatched values are lower, reflecting the fact that when N is smaller the final size is sufficiently captured by the models; the inability of the models to capture final size becomes much more obvious for larger values of N. Figure A.7 in the Appendix plots all matched and unmatched folded ppp-values, under clear misspecification.

Clear misspecification: position-time method As can be seen in tables 2.12 to 2.17, results for the position-time method are very similar with the distance method; values corresponding to matched replications are very close to the desired value of 0.5, even for $N = 100$ and $N = 200$, and unmatched values behave as desired as N increases. Figure A.8 in the Appendix illustrates the results of the position-time method on all datasets, under clear misspecification.

Misspecification: distance method For matched replications, as seen in tables 2.6, 2.8 and 2.10, the overall pattern under misspecification is similar as in the clear misspecification case but a bit less evident, as one would expect. For example, for $N = 100$, the median (95% quantile interval) matched folded ppp-value of the Constant-HM model (see table 2.10) is 0.24 (0.01, 0.50), 0.52 (0.01, 0.90) and 0.83 (0.51, 1), under correct specification, misspecification and clear misspecification, respectively. Appropriately, the values increase as the level of misspecification increases. It is noted though that for such small population sizes, the folded ppp-values under misspecification are not at the level that they would systematically raise serious concerns for the fit of the model; as it is the case for the clear misspecification case. Thankfully, and more importantly, as N increases the folded ppp-values, under misspecification, become more extreme and the evidence of lack of fit more apparent, i.e. the effect of N is the desirable one. Specifically, the median (95% quantile interval)

matched folded ppp-value for Constant-HM (see table 2.10), under misspecification, is 0.58 (0.16, 0.92), 0.75 (0.28, 0.98) and 0.92 (0.70, 0.99) for $N = 200$, $N = 500$ and $N = 1000$, respectively. Similar conclusions hold for Exp-HM (see table 2.6) and Gamma-HM (see table 2.8). One minor difference is that values for the Gamma-HM model appear to be slightly lower than the Constant-HM model and in turn values for the Exp-HM model are marginally lower than the Gamma-HM model; for example for $N = 1000$ the median (95% quantile interval) folded ppp-value is 0.84 (0.43, 0.97) and 0.75 (0.43, 0.97) for Gamma-HM and Exp-HM, respectively. This pattern is an implication of the single realization setting and the fact that the more stochastic a model is the harder it is to discard when misspecified (see the discussion and the example in section 2.2.3).

As far as the results based on unmatched replications (see tables 2.7, 2.9 and 2.11), a similar pattern as in all other cases holds; the effect of N is the same as in the matched case but the values are in general lower (see figure A.9 in the Appendix for a visual comparison of matched and unmatched results for each dataset). For instance, the median (95% quantile interval) unmatched folded ppp-value for Constant-HM (see table 2.11) is 0.30 (0.03, 0.58) and 0.57 (0.30, 0.98) for $N = 100$ and $N = 1000$, respectively. In practice, this means that it is harder to clearly detect misspecification of the infectious period distribution when using unmatched replications; one might have indications of lack of fit but evidence will not be as severe and as systematic (between datasets) as when using matched data. Note though that this does not imply that the method underperforms. Given the fact that standard SIR models accurately capture final size, even when the infectious distribution is misspecified, and considering that unmatched approaches are simultaneously assessing disease progression dynamics and the final size, the method performs as expected (see the example and the discussion in sections 2.5.5 and 2.6.3). To make this clearer, consider the unmatched folded ppp-values for a specific model (e.g. the Constant-HM) and a specific large enough value of N (e.g. $N = 500$ so that the final size is not captured

under clear misspecification) across all three levels of (mis)specification. Then the median (95% quantile interval) under correct specification (both progression dynamics and final size captured), misspecification (final size captured but progression dynamics not adequately captured) and clear misspecification (neither progression dynamics or final size captured) is 0.03 (0, 0.26), 0.37 (0.10, 0.85) and 1 (0.79, 1), respectively. All folded ppp-values, under misspecification are collected in figure A.9 in the Appendix.

Misspecification: position-time method Tables 2.12 to 2.17 illustrate the results for the position-time method, under misspecification. Conclusions and comments are identical as in the distance method, for both matched and unmatched data. For a visual overview on the output of the method for each dataset see figure A.10 in the Appendix.

Table 2.6: Median (95% quantile interval) folded ppp-value from the distance method (d_{L_2} distance shifting, d_{L_2} distance function) for the Exp-HM model, based on matched replications, for simulation study A. The number of datasets that achieved matching (and the median and quantile interval is taken over) is given in table A.1. Simulation conditions for each scenario are given in table 2.5.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.28 (0.04, 0.79)	0.23 (0.01, 0.65)	0.11 (0, 0.39)	0.13 (0, 0.51)
Scenario 2	0.21 (0.02, 0.57)	0.38 (0.09, 0.80)	0.49 (0.16, 0.77)	0.64 (0.14, 0.92)
Scenario 3	0.27 (0.04, 0.52)	0.36 (0.06, 0.78)	0.59 (0.25, 0.82)	0.75 (0.43, 0.97)
Scenario 4	0.97 (0.70, 0.99)	1 (0.93, 1)	1 (1, 1)	-

Table 2.7: Median (95% quantile interval) folded ppp-value from the distance method (d_{L_2} distance shifting, d_{L_2} distance function) for the Exp-HM model, based on unmatched replications, for simulation study A. Simulation conditions for each scenario are given in table 2.5.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.14 (0.01, 0.49)	0.07 (0.01, 0.39)	0.04 (0, 0.14)	0.03 (0, 0.20)
Scenario 2	0.12 (0.01, 0.43)	0.23 (0.04, 0.59)	0.25 (0.10, 0.52)	0.39 (0.05, 0.74)
Scenario 3	0.16 (0.03, 0.37)	0.23 (0.03, 0.51)	0.32 (0.09, 0.57)	0.50 (0.20, 0.78)
Scenario 4	0.72 (0.24, 0.94)	0.96 (0.46, 0.99)	1 (0.97, 1)	1 (0.99, 1)

Table 2.8: Median (95% quantile interval) folded ppp-value from the distance method (d_{L_2} distance shifting, d_{L_2} distance function) for the Gamma-HM model, based on matched replications, for simulation study A. The number of datasets that achieved matching (and the median and quantile interval is taken over) is given in table A.2. Simulation conditions for each scenario are given in table 2.5.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.48 (0.03, 0.86)	0.54 (0.15, 0.90)	0.65 (0.10, 0.95)	0.84 (0.43, 0.97)
Scenario 2	0.18 (0.01, 0.69)	0.14 (0.01, 0.66)	0.11 (0.01, 0.37)	0.04 (0.01, 0.47)
Scenario 3	0.16 (0.01, 0.53)	0.15 (0.01, 0.51)	0.11 (0.01, 0.45)	0.19 (0.01, 0.66)
Scenario 4	0.98 (0.66, 1)	1 (0.80, 1)	-	-

Table 2.9: Median (95% quantile interval) folded ppp-value from the distance method (d_{L_2} distance shifting, d_{L_2} distance function) for the Gamma-HM model, based on unmatched replications, for simulation study A. Simulation conditions for each scenario are given in table 2.5.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.22 (0.01, 0.52)	0.21 (0.02, 0.48)	0.24 (0.03, 0.67)	0.40 (0.14, 0.69)
Scenario 2	0.08 (0, 0.32)	0.07 (0, 0.44)	0.03 (0, 0.15)	0.01 (0, 0.23)
Scenario 3	0.05 (0, 0.31)	0.07 (0, 0.28)	0.04 (0, 0.21)	0.05 (0, 0.42)
Scenario 4	0.82 (0.39, 0.94)	0.95 (0.69, 1)	1 (0.85, 1)	1 (0.85, 1)

Table 2.10: Median (95% quantile interval) folded ppp-value from the distance method (d_{L_2} distance shifting, d_{L_2} distance function) for the Constant-HM model, based on matched replications, for simulation study A. The number of datasets that achieved matching (and the median and quantile interval is taken over) is given in table A.3. Simulation conditions for each scenario are given in table 2.5.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.52 (0.01, 0.90)	0.58 (0.16, 0.92)	0.75 (0.28, 0.98)	0.92 (0.70, 0.99)
Scenario 2	0.19 (0.04, 0.64)	0.15 (0.01, 0.60)	0.19 (0.02, 0.63)	0.15 (0.01, 0.60)
Scenario 3	0.24 (0.01, 0.50)	0.20 (0.01, 0.50)	0.14 (0.01, 0.52)	0.17 (0.02, 0.64)
Scenario 4	0.83 (0.51, 1)	0.99 (0.78, 1)	-	-

Table 2.11: Median (95% quantile interval) folded ppp-value from the distance method (d_{L_2} distance shifting, d_{L_2} distance function) for the Constant-HM model, based on unmatched replications, for simulation study A. Simulation conditions for each scenario are given in table 2.5.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.30 (0.03, 0.58)	0.29 (0.03, 0.58)	0.37 (0.10, 0.85)	0.57 (0.30, 0.98)
Scenario 2	0.08 (0.01, 0.32)	0.07 (0, 0.33)	0.05 (0, 0.49)	0.03 (0, 0.24)
Scenario 3	0.07 (0, 0.30)	0.04 (0, 0.22)	0.03 (0, 0.26)	0.05 (0, 0.34)
Scenario 4	0.73 (0.26, 0.94)	0.93 (0.63, 1)	1 (0.79, 1)	1 (0.70, 1)

Table 2.12: Median (95% quantile interval) $\sqrt{\text{MSE}}$ from the position-time method (d_{L_2} distance shifting) for the Exp-HM model, based on matched replications, for simulation study A. The number of datasets that achieved matching (and the median and quantile interval is taken over) is given in table A.1. Simulation conditions for each scenario are given in table 2.5.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.23 (0.14, 0.35)	0.22 (0.12, 0.34)	0.20 (0.13, 0.28)	0.22 (0.13, 0.35)
Scenario 2	0.21 (0.12, 0.32)	0.27 (0.15, 0.37)	0.30 (0.23, 0.39)	0.32 (0.22, 0.43)
Scenario 3	0.22 (0.14, 0.30)	0.26 (0.15, 0.39)	0.33 (0.23, 0.39)	0.37 (0.26, 0.43)
Scenario 4	0.44 (0.34, 0.47)	0.48 (0.43, 0.49)	0.49 (0.49, 0.49)	-

Table 2.13: Median (95% quantile interval) $\sqrt{\text{MSE}}$ from the position-time method (d_{L_2} distance shifting) for the Exp-HM model, based on unmatched replications, for simulation study A. Simulation conditions for each scenario are given in table 2.5.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.19 (0.12, 0.26)	0.16 (0.10, 0.26)	0.14 (0.08, 0.19)	0.14 (0.08, 0.21)
Scenario 2	0.17 (0.09, 0.30)	0.23 (0.13, 0.35)	0.25 (0.18, 0.35)	0.29 (0.17, 0.38)
Scenario 3	0.19 (0.10, 0.28)	0.22 (0.12, 0.36)	0.29 (0.17, 0.35)	0.35 (0.23, 0.40)
Scenario 4	0.34 (0.26, 0.41)	0.41 (0.30, 0.45)	0.45 (0.42, 0.47)	0.47 (0.46, 0.48)

Table 2.14: Median (95% quantile interval) $\sqrt{\text{MSE}}$ from the position-time method (d_{L_2} distance shifting) for the Gamma-HM model, based on matched replications, for simulation study A. The number of datasets that achieved matching (and the median and quantile interval is taken over) is given in table A.2. Simulation conditions for each scenario are given in table 2.5.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.28 (0.15, 0.39)	0.31 (0.21, 0.41)	0.35 (0.19, 0.43)	0.40 (0.29, 0.44)
Scenario 2	0.21 (0.13, 0.33)	0.19 (0.14, 0.32)	0.19 (0.14, 0.28)	0.18 (0.14, 0.31)
Scenario 3	0.19 (0.14, 0.30)	0.22 (0.13, 0.35)	0.21 (0.14, 0.31)	0.25 (0.15, 0.34)
Scenario 4	0.44 (0.32, 0.49)	0.48 (0.34, 0.49)	-	-

Table 2.15: Median (95% quantile interval) $\sqrt{\text{MSE}}$ from the position-time method (d_{L_2} distance shifting) for the Gamma-HM model, based on unmatched replications, for simulation study A. Simulation conditions for each scenario are given in table 2.5.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.20 (0.12, 0.26)	0.20 (0.13, 0.25)	0.22 (0.14, 0.30)	0.25 (0.17, 0.31)
Scenario 2	0.16 (0.10, 0.25)	0.16 (0.10, 0.29)	0.16 (0.10, 0.25)	0.13 (0.10, 0.24)
Scenario 3	0.15 (0.11, 0.24)	0.15 (0.11, 0.29)	0.17 (0.10, 0.26)	0.20 (0.10, 0.32)
Scenario 4	0.34 (0.26, 0.40)	0.41 (0.32, 0.45)	0.45 (0.42, 0.46)	0.47 (0.44, 0.48)

Table 2.16: Median (95% quantile interval) $\sqrt{\text{MSE}}$ from the position-time method (d_{L_2} distance shifting) for the Constant-HM model, based on matched replications, for simulation study A. The number of datasets that achieved matching (and the median and quantile interval is taken over) is given in table A.3. Simulation conditions for each scenario are given in table 2.5.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.29 (0.16, 0.38)	0.31 (0.20, 0.42)	0.35 (0.23, 0.43)	0.42 (0.34, 0.45)
Scenario 2	0.19 (0.13, 0.31)	0.19 (0.13, 0.33)	0.20 (0.15, 0.32)	0.22 (0.15, 0.30)
Scenario 3	0.20 (0.13, 0.29)	0.21 (0.12, 0.35)	0.20 (0.13, 0.31)	0.23 (0.15, 0.35)
Scenario 4	0.38 (0.28, 0.49)	0.46 (0.33, 0.49)	-	-

Table 2.17: Median (95% quantile interval) $\sqrt{\text{MSE}}$ from the position-time method (d_{L_2} distance shifting) for the Constant-HM model, based on unmatched replications, for simulation study A. Simulation conditions for each scenario are given in table 2.5.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.22 (0.13, 0.28)	0.23 (0.13, 0.27)	0.25 (0.18, 0.39)	0.28 (0.23, 0.45)
Scenario 2	0.15 (0.10, 0.23)	0.16 (0.11, 0.26)	0.17 (0.11, 0.33)	0.18 (0.10, 0.23)
Scenario 3	0.16 (0.10, 0.24)	0.15 (0.10, 0.27)	0.15 (0.10, 0.24)	0.18 (0.12, 0.31)
Scenario 4	0.32 (0.22, 0.39)	0.40 (0.30, 0.44)	0.45 (0.38, 0.46)	0.47 (0.40, 0.47)

2.7.1.5 Conclusions

The conclusions from simulation study A, under the cases of correct specification, misspecification (of the infectious period distribution) and clear misspecification (i.e. fitting a standard SIR model to data generated from a HPP), are summarized as follows.

Final size

- Under correct specification and under misspecification, the final size is accurately captured by the models, i.e. in practice the final size can not be used to detect misspecification of the infectious period distribution of the models.
- In the case of clear misspecification, the final size gives evidence of this misspecification and its power increases with N; for the specific choice of parameters in the simulation study, a model would typically appear suspicious from $N = 200$ and more clearly discarded for $N = 500$ and $N = 1000$.

Time shifting

- Under all cases of (mis)specification, the application of time shifting significantly improves performance and appears to be essential.

- Under correct specification, time shifting largely reduces type I error, that would persist if no shifting was applied, even for large values of N (e.g $N = 1000$).

- Time shifting reduces type II error under clear misspecification (more evidently for $N = 100$ and $N = 200$ when there is potential for improvement) and under misspecification (more evidently for $N = 500$ and $N = 1000$).

- Between the two methods of time shifting, the distance shifting performs slightly better than the theoretical shifting. Considering the fact that the distance shifting is readily extendable to more general epidemic models, unlike the theoretical shifting, it appears as the more attractive choice to use in practice.

Distance function Regarding the choice of distance function for the distance method, overall performance of d_{L_1} and d_{L_2} is almost identical and perhaps slightly better compared to d_{l_2}. Taking into account the fact that d_{L_1} and d_{L_2} can be used for both matched and unmatched datasets, unlike d_{l_2}, it is preferable to choose d_{L_1} or d_{L_2} in practice.

Distance and position-time methods

- The distance and the position-time methods behave very similarly, under all cases of (mis)specification and matching status (matched or unmatched data). The only point that they appear to differ is on the effect of N, under correct specification. More precisely, for the position-time method values appear to be independent of N whereas for the distance method values seem to decrease as N increases; this is the case for Exp-HM and Gamma-HM, as for Constant-HM values appear more or less unaffected by N. In any case, the effect of changing N is sensible.

- The methods can easily detect lack of fit under clear misspecification, using both matched and unmatched data, even for small values of N (roughly at $N = 100$ and $N = 200$ for matched and unmatched data, respectively). As desired, this ability increases with N. These results encourage the use of the methods as a sensible

tool for model assessment. In addition, they suggest that, in such cases of extreme misspecification, unmatched replications suffice to discard a model and thus the computational cost induced by producing matched replications can be avoided.

- Under misspecification (of the infectious period) and using matched data, the methods can provide indication for lack of fit when $N = 100$ and $N = 200$. The power increases with N and the methods can more systematically detect lack of fit when $N = 500$ and $N = 1000$. An exception is in the case that one of the Gamma-HM model or the Constant-HM model is fitted to data generated from the other, as their removal curves are very similar and the methods can not distinguish between them. For the unmatched case the pattern is similar as for the matched case, with the difference that the power is lower (as it should be) due the incorporation of the final size in the assessment (which is accurately captured by the models). Just like in the matched case, the power of the assessment increases as N increases. This desirable effect of N is of high importance, since in real-life applications interest is in larger rather than smaller population sizes.

2.8 Application of the distance and the position-time methods for assessing the infection rate form assumption of SIR models

This section is concerned with assessing the infection rate form assumption of SIR models, using the distance and the position-time methods. For the purposes of such assessment it is meaningful to consider models that have the same infectious period distribution and solely differ on the form of the infection rate. More precisely, the standard SIR model and the non-linear infection rate SIR model, with Exponential infectious periods ($T_D \sim \text{Exp}(\gamma)$), are considered, defined in sections 1.3.5.5 and 1.3.5.6, and denoted as Exp-HM and Exp-NL, respectively. Note that, the choice of

the Exponential infectious period, over the Gamma or the constant, is intentional in order to create more challenging conditions for the methods to detect lack of fit. More specifically, since we are assessing the infection mechanism of a model, an aspect directly related with the (unobserved) infection curve, a more uncertain infectious period will introduce more noise to the removal curve, which the assessment is based on (see the remark in section 2.2.4.2). That is, if the methods can detect lack of fit in the case of Exponential infectious periods, then they should also (and more easily) be able to detect lack of fit for less uncertain infectious periods, where the loss of information from the infection curve to the removal curve is less. To examine the performance of the methods in assessing the infection rate form, a similar approach as in section 2.7 is taken, and an extensive simulation study is conducted, referred to as simulation study B.

2.8.1 Simulation study B

2.8.1.1 Purpose

Simulation study B aims to examine the performance of the distance and the position-time methods in assessing the infection rate form assumption for the two considered SIR models, namely the Exp-HM and the Exp-NL models. That is, the purpose is to investigate:

- The results of the methods when applied to matched and unmatched data under different simulation scenarios.
- The comparability of the results when methods are applied to matched and unmatched data. As explained in section 2.7.1.1, to conduct this comparison it is useful to examine the performance of the models in capturing the final size.
- The comparability between the distance and the position-time methods.

The above investigations are conducted both under the case that the infection rate form is correctly specified as well as when it is misspecified. Also, just like in

simulation study A (see section 2.7.1), great interest is given to any effect that the dimension of the data, characterized by the number of initial susceptibles N, has on the performance of the methods.

2.8.1.2 Simulation conditions

To carry out the aim of the simulation study the Exp-HM and the Exp-NL models are fitted to data generated under two simulation scenarios, for which the simulation conditions are summarized in table 2.18. In scenario 1 data are generated from Exp-HM and in scenario 2 from Exp-NL; in scenario 1, the Exp-HM model is a correctly specified model and the Exp-NL model is misspecified (due to the infection rate form), and, in scenario 2, the roles of the two models are reversed. Each scenario consists of four rounds, corresponding to the number of initial susceptibles N being set at 100, 200, 500 and 1000, and for each round 24 datasets are generated to capture sampling variability.

In scenario 1 (data generated from Exp-HM), the simulation conditions are identical as in scenario 1 of simulation study A (see section 2.7.1.2). That is, for all rounds, the basic reproduction number R_0 is set at 2.5 and the mean infectious period $E(T_D)$ is set at 10, specifying $\gamma = 0.1$; having specified R_0 and $E(T_D)$, β is given by $\beta = \frac{R_0}{NE(T_D)}$, using equation (1.29). In fact, to avoid unnecessary computational cost associated with model fitting and creating matched replications, the same datasets as in scenario 1 of simulation study A are used; by doing so computational cost is only induced from the Exp-NL model runs, since for the Exp-HM model, all outputs are readily available to use from simulation study A.

In scenario 2 (data generated from Exp-NL) the key parameter to set is the power parameter $p \in [0, 1]$. What seems reasonable is to choose p to be small enough so that the generated data of scenario 2 are different enough from scenario 1 (recall from the remarks in section 1.3.5.6 that for $p = 1$ the non-linear SIR model reduces to the

standard SIR model) and large enough so that it is demanding for the methods to detect lack of fit (the smaller the value of p gets the clearer the distinction between the two models becomes). For example, Alharthi (2016), in a similar simulation scenario, set $p = 0.3$. Our choice, motivated by encouraging results from provisional simulations, is to set p at the more challenging value of 0.5, for all rounds. The remaining parameters are chosen so that generated datasets are in a sense similar to scenario 1 and fair comparability conditions (between scenarios) are ensured, i.e. the goal is to establish some sort of parameter correspondence between the two scenarios. The mean infectious period $E(T_D)$ is, just like in scenario 1, set at 10, yielding $\gamma = 0.1$. The only parameter left to specify is β. Recall that in simulation study A (see section 2.7.1.3), β was implicitly specified by choosing a common value for the basic reproduction number R_0 among all relevant scenarios. Since a basic reproduction number parameter is not defined for the non-linear infection rate model (see the relevant remark in section 1.3.5.6) a sensible alternative is to specify β so that, for each round, the final size sampling distribution (conditioning on major outbreaks) between the two scenarios peaks at similar values. Owing to the fact that there are only four rounds to correspond, this is done by trial and error.

Table 2.18: Simulation conditions for simulation study B. Each simulation scenario consists of 4 rounds, where the number of initial susceptibles N is set at 100, 200, 500 and 1000, respectively. For each round 24 datasets are generated.

	Data generating process	**Parameter values**
Scenario 1	Exp-HM	$R_0 = 2.5, \gamma = 0.1$
Scenario 2	Exp-NL	$p = 0.5, \gamma = 0.1,$ $\beta N = 0.85, 1.05, 1.35, 1.65$

2.8.1.3 Run conditions

The Exp-HM and the Exp-NL models are fitted to each generated dataset via MCMC methods, following Algorithms 5 and 8, respectively, to obtain a sample of size 10000,

after a burn-in of 1000. For better mixing, in all cases, the infection update step is repeated as many times as the number of infections, in each MCMC iteration (see section 1.3.5.3). The prior distributions are assigned as in sections 1.3.5.5 and 1.3.5.6 with the prior parameters being specified so that the uncertainty for all model parameters (except for the label of the initial infective α, which is assigned a prior distribution as $\alpha \sim \mathrm{U}[1:n]$) is expressed via uninformative $\mathrm{Exp}(10^{-3})$ prior distributions. Afterwards, the distance and the position-time methods are applied to assess the fit of each model, using matched and unmatched removal data, as described in algorithms 12 to 15. Note that, based on the findings from simulation study A (see section 2.7.1.4), the chosen distance function is d_{L_2} and the type of shifting applied is distance shifting (d_{L_2}). Replicated datasets are created by choosing 500 posterior values using thinning (choosing every 20^{th} value). To achieve the required number of matched replications a time limit of 15 hours is allowed, which if exceeded matched assessment is not conducted. Runs are performed, in parallel for each dataset.

When fitting the Exp-NL model the power parameter p is taken to be known and fixed at the value of 0.5 rather than being estimated from the data. Recall that the same approach was followed in simulation study A for the shape parameter ν of the Gamma-HM model and the motivation behind it is very similar in both cases (see relevant discussion in section 2.7.1.3). More precisely, if p was allowed to be estimated from the data then distinction between the two models would be compromised e.g. in scenario 1 (data generated from Exp-HM) the posterior distribution of p would simply support values near 1 in which case the Exp-NL model would reduce to the Exp-HM model (see the relevant remark in section 1.3.5.6). Also, mixing issues arise when p is unknown (see e.g. Alharthi (2016)) which in turn could cloud the results of the model assessment methods and limit the ability to extract conclusions from the simulation study.

2.8.1.4 Results

The results of simulation study B are examined following the same procedure as in simulation study A. That is, posterior predictive checking for the final size is conducted by calculating its (mid) ppp-value $P(T_{fs}^{rep} < T_{fs}^{obs}) + \frac{1}{2}P(T_{fs}^{rep} = T_{fs}^{obs})$, conditioning on major outbreaks, and the performance of the methods by analyzing the folded ppp-values and the $\sqrt{\text{MSE}}$ values, for each model, under all cases of (mis)specification (see relevant parts of section 2.7.1.4 for more details).

As for simulation study A (see relevant parts of section 2.7.1.4), tables of median (95% quantile interval) values are used to summarize the results of the methods (tables 2.19 to 2.26), while complete results are provided in the Appendix (figures A.13 to A.16).

Recall that, under correct specification, all results regarding the Exp-HM model have already been examined in simulation study A (see section 2.7.1.4); as discussed in section 2.8.1.2 the same datasets are used for scenario 1 of simulation study A and simulation study B. Hence, when reporting results under correct specification, the focus is on the Exp-NL model.

Matching and posterior predictive checking for the final size All results of the matching procedure as well as median (95% quantile interval) final size ppp-values for both models are given in tables A.7 and A.8, and tables A.9 and A.10 respectively, in the Appendix.

Correct specification Recall from simulation study A (see section 2.7.1.4), that for Exp-HM, matched replications are essentially achieved in all cases and the ppp-value is typically close to the optimal value of 0.5, with small variance; pooling all 96 datasets from the four rounds (as trend over N was similar), median (95% quantile interval) ppp-value is 0.49 (0.43, 0.60).

For Exp-NL the behaviour is again sensible but somewhat different. More precisely, the median ppp-value is, similar to Exp-HM and, as desired, around 0.5 for all rounds but the variance is higher and it appears to increase with N e.g. for $N = 100$ the median (95% quantile interval) ppp-value is 0.49 (0.37, 0.69) and for $N = 1000$ it is 0.44 (0.08, 0.99) (see table A.10 in the Appendix). As far as the matching procedure, it was successfully completed for 95 out of 96 datasets in total (see table A.8 in the Appendix).

Misspecification When the Exp-HM model is misspecified (i.e. when fitted to data generated from the Exp-NL model) the final size is captured quite accurately for smaller values of N but as the dimension of the data increases it becomes harder for the model to produce outbreaks with the same final size as the observed. For instance, the final size median (95% quantile interval) ppp-value for $N = 200$ is 0.52 (0.46, 0.66) and for $N = 1000$ is 0.81 (0.62, 0.99) (see table A.9 in the Appendix). This means that for $N = 1000$ the observed final size typically lies closer to the right tail, rather than the mode, of the final size posterior predictive distribution (see figure A.11 in the Appendix for an example). This pattern is reflected in the number of cases for which the matching procedure is achieved; up to $N = 500$, matching was completed for 70 out of 72 datasets, whereas for $N = 1000$ for only 7 out of 24 datasets (see table A.7 in the Appendix).

Similar behaviour, as far as the effect of the dimension of the data, is exhibited when the Exp-NL model is misspecified (i.e. when fitted to data generated from the Exp-HM model), only in this case the evidence of lack of fit is more apparent. For example, the final size median (95% quantile interval) ppp-value is 0.45 (0.34, 0.59), 0.09 (0.05, 0.20) and 0 (0, 0.05) for $N = 100, 500$ and 1000, respectively (see table A.10 in the Appendix). This implies that, for $N = 500$ and $N = 1000$ the observed final size lies on (and beyond) the left tail of the final size posterior predictive distribution (see figure A.12 in the Appendix for an example). A result of this is that for $N = 1000$

matched replications were achieved for only 2 out of 24 datasets (see table A.8 in the Appendix).

To summarize, simulations suggest that when the infection rate form is misspecified, the models accurately capture the final size for small values of N but as N increases this ability deteriorates and lack of fit is revealed. The power to detect misspecification appears to be higher for Exp-NL compared to Exp-HM (see tables A.9 and A.10 in the Appendix). These observations, particularly the effect of N, are very interesting considering the fact that the general consensus in the literature is that the ability of epidemic models to capture the final size is quite robust under many types of model misspecification. For example in the work of Alharthi (2016), which includes assessment for the Exp-HM and the Exp-NL models under the same simulation scenarios of misspecification (i.e. when one of these models is fitted to data generated from the other), no evidence of model misspecification was found using the final size. The most likely reason for this was that the simulated datasets where of (the same) relatively small dimension ($N = 200$). This highlights the importance of considering the effect of the dimension of the data in such investigations, especially as in real-life applications interest is in larger rather than smaller population sizes.

Performance of the methods

Correct specification: distance method For assessment based on matched replications, as can be seen in table 2.21, the folded ppp-values are typically closer to 0 than 1 indicating, as desired, goodness of fit for the Exp-NL model. It appears that the dimension of the data does not have any apparent effect on the sampling distribution of the folded ppp-values, which is a sensible pattern under correct specification.

As final size is incorporated into the unmatched assessment, which is typically captured by the Exp-NL model independently of N (see results on final size and table

A.10 in the Appendix), folded ppp-values are on average lower than the matched case and appear independent of N, as seen in table 2.22. Once more these results are desirable as they suggest goodness of fit for the model. All folded ppp-values, under correct specification, are illustrated in figure A.13 in the Appendix.

Correct specification: position-time method Tables 2.25 and 2.26 give the Exp-NL model summary results for the position-time method, under correct specification; complete results can be found in figure A.14 in the Appendix. Comments regarding the results and the effect of N are similar to those for the distance method, for both matched and unmatched data.

Misspecification: distance method First, results for the Exp-HM model are reported. For matched replications, as seen in table 2.19, median (95% quantile interval) folded ppp-value, under misspecification, is 0.23 (0.06, 0.82), 0.62 (0.27, 0.97), 0.94 (0.50, 1) and 1 (0.94, 0.1) for $N = 100$, 200, 500 and 1000, respectively; for $N = 1000$ only 7 datasets achieved matching (see table A.7 in the Appendix) but the consistency between the folded ppp-values on these datasets still allows for meaningful conclusions. It is clear that as N increases the folded ppp-values move in the desired direction; in fact, it appears that as N gets large the sampling distribution of the folded ppp-value reduces to a point mass at the optimal value of 1, allowing for systematic detection of lack of fit.

Regarding results based on unmatched replications, folded ppp-values are generally lower than the matched case, for the smaller values of N, and more similar for $N = 1000$ (see table 2.20). Considering the fact that, when applied to unmatched data, the method simultaneously assesses disease progression dynamics and final size, it performs sensibly (see the example and the discussion in sections 2.5.5 and 2.6.3); recall that the Exp-HM model adequately captures the final size for smaller values of N but struggles to do so as N gets larger (see results on final size and table A.9 in the

Appendix). From a practical point of view, it appears that for large enough N (e.g. around N=1000, with the parameters used in this simulation study) lack of fit for the Exp-HM model, due to misspecified infection rate form, could be systematically detected using unmatched replications (thus avoiding the cost of producing matched replications).

The performance of the method for the Exp-NL model is very similar as for the Exp-HM model, for both matched and unmatched case, in the sense that as N increases power increases too. In fact, folded ppp-values are in general higher for the Exp-NL model, particularly for smaller values of N, implying that lack of fit can be detected even easier. For example, as seen in table 2.21, the median (95% quantile interval) folded ppp-value, based on matched replications, is 0.70 (0.08, 0.86), 0.80 (0.28, 0.96) and 0.97 (0.84, 0.99) for $N = 100$, 200 and 500, respectively; for $N = 1000$ only 2 datasets completed matching (see table A.8 in the Appendix) for which the folded ppp-value was 1. As far as unmatched replications, once again the performance of the model in capturing the final size is effectively incorporated into the assessment. More precisely, as seen in table 2.22, for $N = 100$ and $N = 200$, where the final size is relatively adequately captured by the model (see results on final size and table A.10 in the Appendix), the median (95% quantile interval) folded ppp-value is 0.54 (0.09, 0.80) and 0.54 (0.07, 0.90), while for $N = 500$ and $N = 1000$, where the model fails to capture the final size (see results on final size and table A.10 in the Appendix), the median (95% quantile interval) folded ppp-value is 0.90 (0.66, 0.97) and 0.99 (0.97, 1).

All folded ppp-values, under misspecification, are given in figure A.15 in the Appendix.

Misspecification: position-time method Tables 2.23 and 2.24 and tables 2.25 and 2.26 give median (95% quantile interval) folded ppp-values, for the Exp-HM and the Exp-NL models, respectively, under the case of misspecification. Folded ppp-values for all datasets can be seen in figure A.16 in the Appendix. Conclusions

regarding the performance of the position-time method are essentially the same as for the distance method, for both matched and unmatched cases.

Table 2.19: Median (95% quantile interval) folded ppp-value from the distance method (d_{L_2} distance shifting, d_{L_2} distance function) for the Exp-HM model, based on matched replications, for simulation study B. The number of datasets that achieved matching (and the median and quantile interval is taken over) is given in table A.7. Simulation conditions for each scenario are given in table 2.18.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.28 (0.04, 0.79)	0.23 (0.01, 0.65)	0.11 (0, 0.39)	0.13 (0, 0.51)
Scenario 2	0.23 (0.06, 0.82)	0.62 (0.27, 0.97)	0.94 (0.50, 1)	1 (0.94, 0.1)

Table 2.20: Median (95% quantile interval) folded ppp-value from the distance method (d_{L_2} distance shifting, d_{L_2} distance function) for the Exp-HM model, based on unmatched replications, for simulation study B. Simulation conditions for each scenario are given in table 2.18.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.14 (0.01, 0.49)	0.07 (0.01, 0.39)	0.04 (0, 0.14)	0.03 (0, 0.20)
Scenario 2	0.13 (0.01, 0.42)	0.24 (0.07, 0.71)	0.64 (0.16, 0.98)	0.97 (0.68, 0.1)

Table 2.21: Median (95% quantile interval) folded ppp-value from the distance method (d_{L_2} distance shifting, d_{L_2} distance function) for the Exp-NL model, based on matched replications, for simulation study B. The number of datasets that achieved matching (and the median and quantile interval is taken over) is given in table A.8. Simulation conditions for each scenario are given in table 2.18.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.70 (0.08, 0.86)	0.80 (0.28, 0.96)	0.97 (0.84, 0.99)	1 (1, 1)
Scenario 2	0.31 (0.01, 0.67)	0.21 (0.01, 0.60)	0.28 (0.01, 0.63)	0.22 (0.03, 0.58)

Table 2.22: Median (95% quantile interval) folded ppp-value from the distance method (d_{L_2} distance shifting, d_{L_2} distance function) for the Exp-NL model, based on unmatched replications, for simulation study B. Simulation conditions for each scenario are given in table 2.18.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.54 (0.09, 0.80)	0.54 (0.07, 0.90)	0.90 (0.66, 0.97)	0.99 (0.97, 1)
Scenario 2	0.17 (0.02, 0.45)	0.11 (0.01, 0.38)	0.16 (0.04, 0.63)	0.16 (0.01, 0.84)

Table 2.23: Median (95% quantile interval) $\sqrt{\text{MSE}}$ from the position-time method (d_{L_2} distance shifting) for the Exp-HM model, based on matched replications, for simulation study B. The number of datasets that achieved matching (and the median and quantile interval is taken over) is given in table A.7. Simulation conditions for each scenario are given in table 2.18.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.23 (0.14, 0.35)	0.22 (0.12, 0.34)	0.20 (0.13, 0.28)	0.22 (0.13, 0.35)
Scenario 2	0.24 (0.16, 0.39)	0.32 (0.23, 0.43)	0.42 (0.28, 0.48)	0.47 (0.44, 0.48)

Table 2.24: Median (95% quantile interval) $\sqrt{\text{MSE}}$ from the position-time method (d_{L_2} distance shifting) for the Exp-HM model, based on unmatched replications, for simulation study B. Simulation conditions for each scenario are given in table 2.18.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.19 (0.12, 0.26)	0.16 (0.10, 0.26)	0.14 (0.08, 0.19)	0.14 (0.08, 0.21)
Scenario 2	0.18 (0.13, 0.25)	0.20 (0.15, 0.29)	0.29 (0.18, 0.38)	0.37 (0.31, 0.41)

Table 2.25: Median (95% quantile interval) $\sqrt{\text{MSE}}$ from the position-time method (d_{L_2} distance shifting) for the Exp-NL model, based on matched replications, for simulation study B. The number of datasets that achieved matching (and the median and quantile interval is taken over) is given in table A.8. Simulation conditions for each scenario are given in table 2.18.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.30 (0.16, 0.36)	0.34 (0.21, 0.43)	0.43 (0.34, 0.46)	0.42 (0.41, 0.43)
Scenario 2	0.24 (0.12, 0.33)	0.21 (0.11, 0.29)	0.21 (0.13, 0.30)	0.20 (0.13, 0.29)

Table 2.26: Median (95% quantile interval) $\sqrt{\text{MSE}}$ from the position-time method (d_{L_2} distance shifting) for the Exp-NL model, based on unmatched replications, for simulation study B. Simulation conditions for each scenario are given in table 2.18.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.29 (0.16, 0.34)	0.32 (0.18, 0.42)	0.41 (0.32, 0.46)	0.43 (0.38, 0.46)
Scenario 2	0.19 (0.12, 0.30)	0.17 (0.11, 0.27)	0.21 (0.14, 0.25)	0.20 (0.13, 0.31)

2.8.1.5 Removal curve behaviour

To better explain the results of the simulation study it is useful to gain an appreciation on the removal curve behaviour of the two considered models. Figure 2.15 provides a typical example that highlights the difference in the posterior predictive removal curve behaviour between the models. More specifically, the Exp-NL model tends to produce removal curves for which the majority of the events occur more spread out in time, compared to the removal curves of the Exp-HM model; for the Exp-HM model the curves peak faster. This difference in the behaviour of the removal curves is most likely a key reason behind the ability of the methods to successfully detect lack of fit, whenever one of these two models is misspecified.

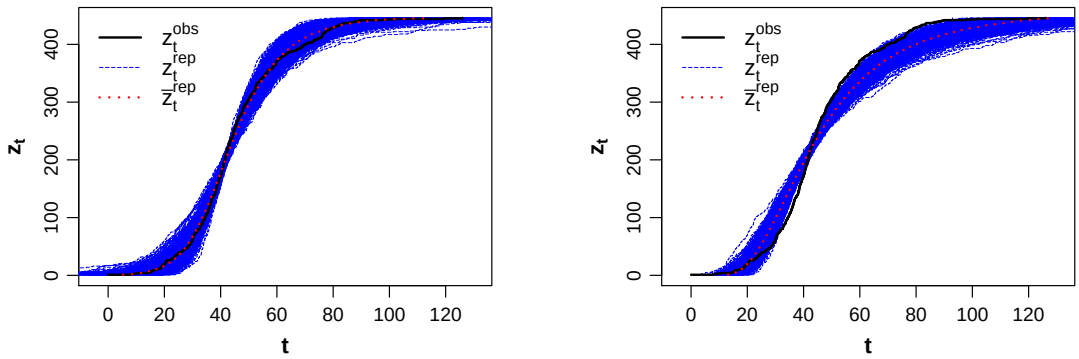


Figure 2.15: Plots of 500 matched replications from the posterior predictive distribution of the removal curve z_t^{rep} with the mean removal curve $\bar{z}_t^{rep}$ (red, dotted line) and the observed removal curve z_t^{obs} (black, solid line) imposed. Observed data is a typical dataset of round 3 ($N = 500$) in scenario 1 (data generated from an Exp-HM) of simulation study B. Left and right plots correspond to the Exp-HM and the Exp-NL models, respectively. For reference the folded ppp-value (d_{L_2} distance shifting, d_{L_2} distance function) and the $\sqrt{\text{MSE}}$ (d_{L_2} distance shifting) are $(0.31, 0.23)$ and $(0.99, 0.43)$ for the Exp-HM and the Exp-NL models, respectively.

2.8.1.6 Conclusions

An outline of the conclusions from simulation study B is given.

Final size

- Under correct specification the final size is, as expected, accurately captured by the models.

- When misspecified the models adequately capture the final size for small values of N but as N increases this ability deteriorates and lack of fit is revealed. The power to detect misspecification appears to be higher for Exp-NL compared to Exp-HM; for the specific choice of parameters in the simulation study serious concerns for the fit of Exp-HM are raised from $N = 1000$, while for Exp-NL from $N = 500$.

Distance and position-time methods

- The distance and the position-time methods behave very similarly, under all cases of (mis)specification and matching status (matched or unmatched data).

- Under correct specification, the methods, as they ought to, suggest goodness of fit. No apparent effect of N is observed, which is a sensible behaviour under correct specification.

- Under misspecification, the methods enjoy increased power as N increases, for both models and for matched and unmatched data. Note that, just like in simulation study A (see section 2.7.1.5), this is the most encouraging conclusion, because in practical examples, outbreaks in larger populations are of far greater interest. Typically, for smaller values of N, unmatched assessment has less power than matched, but for larger values of N power is similar; this is a reflection of the final size performance under misspecification. For example, for the Exp-HM model, systematic lack of fit can be detected around $N = 500$ and $N = 1000$ when using matched and unmatched replications, respectively. The ability to detect misspecification is higher for the Exp-NL model than the Exp-HM model; $N = 200$ and $N = 500$ suffice to raise serious concerns for the fit of the Exp-NL model, using matched and unmatched replications, respectively. The fact that, for large enough N, misspecification can be detected using unmatched replications is

computationally appealing, since the cost of producing matched replications can be avoided.

2.9 Application of the distance and the position-time methods for assessing the population mixing assumption of SIR models

This section is concerned with assessing the population mixing assumption of SIR models using the distance and the position-time methods. For the purposes of such assessment, models that have the same infectious period distribution and different infection processes, distinguished by their population mixing assumption, are considered. More specifically, the standard SIR model and the two-level-mixing model, with constant infectious period ($T_D \equiv c$), are considered, defined in sections 1.3.5.5 and 1.3.5.7, and denoted as Constant-HM and Constant-2L, respectively. The choice of the constant infectious period, rather than Gamma or Exponential, is to create more informative conditions for investigating the performance of the methods. More precisely, since we are assessing the infection mechanism (depending on population mixing assumption) of a model, an aspect directly related with the (unobserved) infection curve, using constant infectious periods prevents any noise from being introduced from the (unobserved) infection curve to the removal curve (assessment is as if it is conducted on the infection curves) and allows for more meaningful conclusions. That is to say, that the methods should first be able to detect lack of fit in the case of constant infectious periods, before being considered for the random infectious periods case, where there is the additional challenge of loss of information from the infection curve to the removal curve (see the remark in section 2.2.4.2). To examine the performance of the methods in assessing the population mixing assumption, the same approach as in sections 2.7 and 2.8 is adopted, and an extensive simulation study is conducted, referred to as simulation study C.

2.9.1 Simulation study C

2.9.1.1 Purpose

The purpose of simulation study C is to examine the performance of the distance and the position-time methods in assessing the population mixing assumption for the two considered SIR models, namely the Constant-HM and the Constant-2L models. At first instance, only matched replications assessment is considered; depending on the results on matched data, final size performance and unmatched assessment might be worth considering at a later point. In short, simulation study C aims to investigate:

- The results of the methods when applied to matched data under different simulation scenarios.
- The comparability between the distance and the position-time methods.

The above investigations are conducted for both the case that the population mixing assumption is correctly specified, as well as when it is misspecified. Similar to simulation studies A and B (see sections 2.7.1 and 2.8.1), it is intended to investigate if and how the dimension of the data, characterized by the number of initial susceptibles N, affects the results. In addition, particular interest is given in examining the performance of the methods on data of varying levels of two-level-mixing effect, that is, on data of varying levels of difference between within and between household infectivity.

2.9.1.2 Simulation conditions

For the purposes of this simulation study, the Constant-HM and the Constant-2L models are fitted to data generated under different simulation scenarios, for which the simulation conditions are summarized in table 2.27. In all scenarios data are generated from the Constant-2L model, i.e. in all instances the Constant-2L model is a correctly specified model, and the Constant-HM model is misspecified due to

the population mixing assumption. To examine the performance of the methods, on data of varying levels of two-level-mixing effect, four simulation scenarios (each corresponding to a different level of two-level-mixing effect) are considered. Each scenario consists of three rounds, corresponding to the population number C being set at 100, 200 and 500, respectively, i.e. the number of initial susceptibles N is set at 99, 199 and 499, respectively. For each round, 24 datasets are generated to capture sampling variability and in all cases the initial infective (and its household) is chosen uniformly at random from the population.

The parameters for the four simulation scenarios are set as follows. For simplicity, all households are taken to have equal size C_H, where C_H is set at 5; this value is large enough to distinguish the two-level-mixing model from the standard SIR model (recall from the relevant remark in section 1.3.5.7 that if $C_H = 1$ the two-level-mixing SIR model reduces to the standard SIR model) and at the same time small enough to be practically relevant. The basic reproduction number R_* is set at 2.5 and the infectious period c at 10 for all scenarios and rounds; the fixed value of R_* allows for an investigation of the effect of the dimension of the data via N (common R_* between rounds) and the effect of the level of the two-level-mixing evidence in the data (common R_* between scenarios). More precisely, the latter examination is achieved as follows. Consider the notation of section 1.3.5.7 and recall from the relevant paragraph of section 1.3.5.7 that $R_* = \mu R_G$ (see equation (1.40)), where μ is the expected number of ever-infected individuals of the within household epidemic, for which only local infections occur, and R_G is the basic reproduction number of the model for which all households are of size 1 and only global infections occur. These three quantities, R_*, μ and R_G, can roughly be thought of as quantifying overall, within household and between household infectivity, respectively. Given R_* (and given $E(T_D)$, N and C_H), R_G and μ are inversely proportional and the two-level-mixing effect in the data becomes more apparent as μ increases (and R_G decreases). To establish an orientation on how to choose μ, for each scenario, we can naively consider the basic reproduction

number of the within household outbreak. More precisely, since the within household outbreak can be seen as an outbreak from a standard SIR model, with one-to-one infection rate β_L and $C_H - 1$ initial susceptibles, its basic reproduction number is given (using equation (1.29)) by $R_0^H = \beta_L(C_H - 1)E(T_D)$; note that this definition is naive, since the number of initial susceptibles $C_H - 1$ is not large (see the relevant paragraph in section 1.3.5.5), but it still suffices to establish a guideline for within household infectivity. By specifying a value for R_0^H, β_L is specified from the previous equation. In turn, given β_L, μ can be specified by solving a system of triangular equations (see Ball (1986)). Finally, given μ and R_*, R_G and β_G are specified using equations (1.40) and (1.41), respectively. To create scenarios of increasing two-level-mixing effect (i.e. increasing within household infectivity), the value of R_0^H is set at 1, 2, 5 and 20, which in turn yields μ =1.65, 3.40, 4.95 and 5, for scenarios 1, 2, 3 and 4, respectively; scenario 1 ($R_0^H = 1$) creates data with a rather mild two-level-mixing effect while scenario 4 ($R_0^H = 20$) represents an extremely apparent case of two-level-mixing effect.

To facilitate a better appreciation of the extent of the two-level-mixing effect in each scenario one can calculate, using simulations, the mean proportion of local infections (from total infections) $\bar{p_L}$, under the sampling distribution of the model with parameters as specified by the simulation scenario. As mentioned in section 1.3.5.7, local infections refer to infections occurring from the action of the local infection process (modelling only withing household contacts) and therefore the higher the number of $\bar{p_L}$ the higher the two-level-mixing effect (i.e. within household infectivity) in the data. The calculated values of $\bar{p_L}$ are 0.32, 0.60, 0.75 and 0.79 for scenarios 1, 2, 3 and 4, respectively, and they are included in table 2.27 for reference.

Note that a simulation scenario where data are generated from the Constant-HM model is not considered. Under such a scenario, the local one-to-one infection rate β_L would be estimated to be very close to 0, causing the two-level mixing SIR model

to essentially reduce to the standard SIR model (see the relevant remark in section 1.3.5.7 for the conditions under which the two-level mixing model reduces to the standard SIR model) making the two models indistinguishable and the simulation scenario uninteresting. One way to avoid this, would be to fix β_L at a some strictly positive value. Such an approach was taken in simulation studies A and B with the shape parameter ν and the power parameter p for the Gamma-HM and the Exp-NL models, respectively. However, unlike ν and p, fixing β_L at a certain value is not an approach that one would typically take in practice, as β_L is one of the main parameters for which inference is desired. Thus, such a simulation scenario is avoided.

Table 2.27: Simulation conditions for simulation study C. Each simulation scenario consists of 3 rounds, where the number of initial susceptibles N is set at 99, 199 and 499, respectively. For each round 24 datasets are generated. The number of individuals in each household is set as $C_H = 5$, in all instances.

	Data generating process	**Parameter values**	$\bar{p_L}$
Scenario 1	Constant-2L	$R_* = 2.5, c = 10, R_0^H = 1, \mu = 1.65$	0.32
Scenario 2	Constant-2L	$R_* = 2.5, c = 10, R_0^H = 2, \mu = 3.4$	0.60
Scenario 3	Constant-2L	$R_* = 2.5, c = 10, R_0^H = 5, \mu = 4.95$	0.75
Scenario 4	Constant-2L	$R_* = 2.5, c = 10, R_0^H = 20, \mu = 5$	0.79

2.9.1.3 Run conditions

The Constant-HM and the Constant-2L models are fitted to each generated dataset via MCMC methods, using Algorithms 7 and 9, to obtain a sample of size 50000, after a burn-in of 10000. The prior distributions are assigned as in sections 1.3.5.5 and 1.3.5.7 with the prior parameters being specified so that all model parameters have uninformative Exp(10^{-3}) prior distributions. Afterwards, the distance and the position-time methods are applied to assess the fit of each model, using matched replications, as described in algorithms 12 and 14. Based on the findings of simulation study A (see section 2.7.1.4) the chosen distance function is d_{L_2} and the type of

shifting applied is distance shifting (d_{L_2}). Replicated datasets are created by choosing 500 posterior values using thinning (choosing every 100^{th} value). To achieve the required number of matched replications a time limit of 15 hours is allowed, which if exceeded matched assessment is not conducted. Runs are performed, in parallel for each dataset.

2.9.1.4 Results

The performance of the methods is examined as in simulation studies A and B (see relevant parts of sections 2.7.1.4 and 2.8.1.4). That is, by examining the folded ppp-values and the $\sqrt{\text{MSE}}$ values, for each model, at each simulation scenario and round. Recall that for the effect of the dimension of the data, quantified by N, to be desirable, it is required that, under misspecification, more data should allow more power to reveal lack of fit (see discussion in relevant parts of section 2.7.1.4 for more details). Similarly, the methods should be expected to have more power in exposing the fit of the Constant-HM model, when the two-level-mixing effect in the data (i.e. the larger the value of R_0^H) becomes more apparent.

Similar to simulation studies A and B (see relevant parts of sections 2.7.1.4 and 2.8.1.4), tables of median (95% quantile interval) values are used to summarize the results of the methods (see tables 2.28 and 2.29), while detailed results for each dataset are provided in the Appendix (see figures A.17 to A.20). In all tables and figures, rows and columns conveniently correspond to the different scenarios (values of R_0^H) and rounds (values of N), respectively; this facilitates the appreciation of any effect the two-level-mixing level (by choosing a column and looking across rows) or the dimension of the data (by choosing a row and looking across columns) have on the results of the methods.

Performance of the methods

Correct specification (Constant-2L model) Table 2.28 illustrates median (95% quantile interval) folded ppp-values from the distance method, for all scenarios and rounds, for the Constant-2L model (correctly specified model). As expected ppp-values are typically closer to 0 than 1 suggesting goodness of fit for the model. Regarding the effect of the dimension of the data, there does not seem to be any clear trend, which under correct specification is a sensible behaviour. Likewise, the effect of the two-level-mixing evidence in the data is sensible. More specifically, it appears that that there is no definite pattern, besides a decrease in the values in scenario 4, where the two-level-mixing effect in the data is quite extreme ($R_0^H = 20$, $\bar{p_L} = 0.79$). For detailed results on each dataset, see figure A.17 in the Appendix.

Results for the effect of N and the effect of R_0^H are very similar for the position-time method, as can be seen in table 2.29 and in figure A.18 in the Appendix.

Misspecification (Constant-HM model) As seen in table 2.30, the performance of the distance method, under misspecification, is not the desirable one as folded ppp-values are not typically large enough to detect lack of fit. In addition, there does not appear to be any clear effect of N, implying that even for larger datasets the method would not be able to expose the misspecification of the model. As far as the effect of the two-level-mixing evidence in the data, it appears sensible, in the sense that as R_0^H increases the power of the method increases too. However, this increase is very slow and it must be emphasized that even for the extreme scenario of $R_0^H = 20$ ($\bar{p_L} = 0.79$), where the level of the two-level-mixing effect in the data is extremely evident, the method can not consistently detect lack of fit; median (95% quantile interval) folded ppp-value (pooling over N) is 0.17 (0.01, 0.73), 0.29 (0.02, 0.94), 0.43 (0.02, 0.97) and 0.57 (0.09, 0.97) for R_0^H =1, 2, 5 and 20, respectively. For a detailed visual appreciation on the results see figure A.19 in the Appendix.

Conclusions are very similar for the position time method as illustrated in table 2.31 and in figure A.20 in the Appendix.

Table 2.28: Median (95% quantile interval) folded ppp-value from the distance method (d_{L_2} distance shifting, d_{L_2} distance function) for the Constant-2L model, based on matched replications, for simulation study C. The number of datasets that achieved matching (and the median and quantile interval is taken over) is given in table A.11. Simulation conditions for each scenario are given in table 2.27.

	$N = 99$	$N = 199$	$N = 499$
Scenario 1	0.24 (0, 0.58)	0.36 (0.03, 0.68)	0.27 (0.04, 0.82)
Scenario 2	0.37 (0.02, 0.79)	0.28 (0.02, 0.89)	0.42 (0.05, 0.89)
Scenario 3	0.38 (0.03, 0.84)	0.37 (0.04, 0.89)	0.44 (0.03, 0.81)
Scenario 4	0.11 (0.01, 0.72)	0.15 (0.01, 0.48)	0.16 (0.01, 0.60)

Table 2.29: Median (95% quantile interval) $\sqrt{\text{MSE}}$ from the position-time method (d_{L_2} distance shifting) for the Constant-2L model, based on matched replications, for simulation study C. The number of datasets that achieved matching (and the median and quantile interval is taken over) is given in table A.11. Simulation conditions for each scenario are given in table 2.27.

	$N = 99$	$N = 199$	$N = 499$
Scenario 1	0.19 (0.13, 0.30)	0.24 (0.14, 0.32)	0.23 (0.14, 0.37)
Scenario 2	0.25 (0.15, 0.39)	0.25 (0.13, 0.39)	0.26 (0.14, 0.40)
Scenario 3	0.23 (0.14, 0.38)	0.25 (0.11, 0.40)	0.25 (0.14, 0.33)
Scenario 4	0.19 (0.12, 0.32)	0.17 (0.10, 0.27)	0.20 (0.11, 0.30)

Table 2.30: Median (95% quantile interval) folded ppp-value from the distance method (d_{L_2} distance shifting, d_{L_2} distance function) for the Constant-HM model, based on matched replications, for simulation study C. The number of datasets that achieved matching (and the median and quantile interval is taken over) is given in table A.12. Simulation conditions for each scenario are given in table 2.27.

	$N = 99$	$N = 199$	$N = 499$
Scenario 1	0.16 (0, 0.51)	0.14 (0.02, 0.69)	0.18 (0.02, 0.81)
Scenario 2	0.29 (0.02, 0.88)	0.41 (0.05, 0.90)	0.20 (0.05, 0.95)
Scenario 3	0.39 (0.03, 0.94)	0.42 (0.07, 0.85)	0.45 (0.08, 0.98)
Scenario 4	0.60 (0.22, 0.95)	0.68 (0.08, 0.93)	0.55 (0.09, 0.99)

Table 2.31: Median (95% quantile interval) $\sqrt{\text{MSE}}$ from the position-time method (d_{L_2} distance shifting) for the Constant-HM model, based on matched replications, for simulation study C. The number of datasets that achieved matching (and the median and quantile interval is taken over) is given in table A.12. Simulation conditions for each scenario are given in table 2.27.

	$N = 99$	$N = 199$	$N = 499$
Scenario 1	0.18 (0.13, 0.26)	0.20 (0.13, 0.34)	0.20 (0.13, 0.37)
Scenario 2	0.22 (0.13, 0.40)	0.25 (0.14, 0.38)	0.20 (0.15, 0.42)
Scenario 3	0.24 (0.13, 0.42)	0.25 (0.15, 0.35)	0.27 (0.15, 0.43)
Scenario 4	0.28 (0.19, 0.40)	0.29 (0.17, 0.39)	0.29 (0.18, 0.42)

2.9.1.5 Removal curve behaviour

In order to gain a better understanding about the results of the simulation study (i.e. the inadequate power of the methods to detect lack of fit, the effect of N and the effect of R_0^H) an appreciation of the removal curve behaviour is in order. Four typical datasets (generated from the Constant-2L model) from different scenarios

and rounds (so that the effect of R_0^H and N can be highlighted) are picked and the posterior predictive removal curves from the Constant-HM model (misspecified model) are produced (see figure 2.16). Choosing a column (value of N) and looking across rows (values of R_0^H), in figure 2.16, it is evident that higher within household infectivity leads to a jump effect in the observed removal curve. To understand this, consider an extreme case of within household infectivity (such as $R_0^H = 20$) so that an infection of an individual within a household is followed by an almost immediate infection of all susceptibles in the household, i.e. all jumps of the removal curve are roughly of size C_H. This is exactly the feature that the methods attempt to make use of, in order to expose the fit of the Constant-HM model when fitted to such data, as removal curves produced from the Constant-HM model can not reproduce this jump effect. To a small extent, the methods succeed as (for a given N) higher values of R_0^H are associated with increased power. However the (non) effect of N is what deprives the methods from systematically detecting lack of fit. More precisely, by choosing a row (value of R_0^H) and looking across columns (values of N), in figure 2.16, one notices that the jump effect in the observed removal curve becomes less evident as the dimension of the data increases; this is because the effect is actually relative to the dimension of the observed data (i.e. the effect is quantified by C_H/N) and since the number of individuals in a household C_H is fixed, while N increases, the effect deteriorates. At the same time, the uncertainty of the posterior predictive distribution of the removal curves is naturally less for larger datasets (see figure 2.16) meaning that there is potentially more power to expose model fit as N gets larger. These two opposing trends of N (the jump effect in the observed removal curve and the uncertainty of the posterior predictive distribution of the removal curves) appear to cancel each other out and yield no apparent overall effect of N on the results; for smaller datasets, where the jump effect is evident, the pack of replicated removal curves is too wide to systematically expose model fit, and for larger datasets, where the pack of replicated removal curves is narrower, the jump effect is not that evident to be regularly detected.

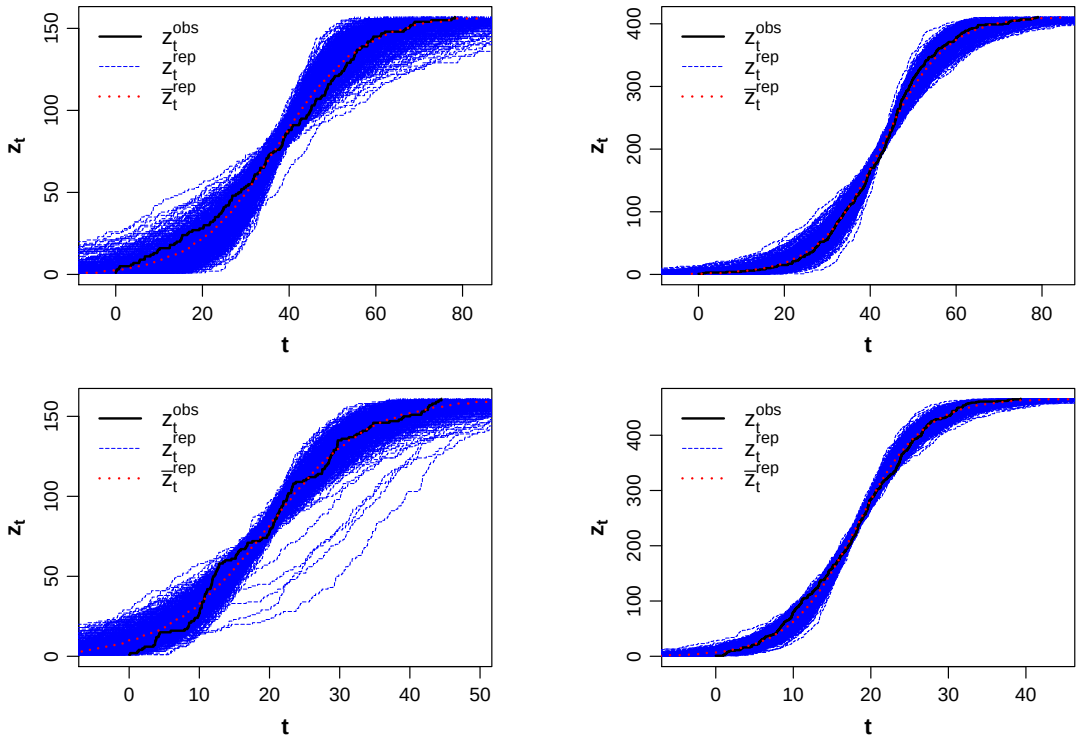


Figure 2.16: Plots of 500 matched replications from the posterior predictive distribution of the removal curve z_t^{rep} with the mean removal curve $\bar{z}_t^{rep}$ (red, dotted line) and the observed removal curve z_t^{obs} (black, solid line) imposed. Fitted model is the Constant-HM model. Observed data are four typical datasets from scenarios (R_0^H) and rounds (N) of simulation study C (data generated from the Constant-2L model). Rows (top to bottom) correspond to R_0^H values of 2 and 20, respectively. Columns (left to right) correspond to N values of 199 and 499, respectively. For reference the folded ppp-value (d_{L_2} distance shifting, d_{L_2} distance function) and the $\sqrt{\text{MSE}}$ (d_{L_2} distance shifting) are (0.33, 0.24), (0.29, 0.20), (0.50, 0.26) and (0.64, 0.28) for R_0^H and N values of (2, 200), (2, 500), (20, 200) and (20, 500), respectively.

2.9.1.6 Conclusions

The conclusions from simulation study C are summarized as follows.

Distance and position-time methods

- The distance and the position-time methods behave very similarly, under all simulation scenarios.

- Under correct specification (Constant-2L model), the methods indicate goodness of fit as expected. No clear effect of N or R_0^H is evident, which is sensible behaviour under correct specification.

- Under misspecification (Constant-HM model), the methods do not have enough power to systematically detect misspecification of the population mixing assumption. Undesirably, power does not increase as N increases (N does not appear to have any effect on the performance of the methods). In practice, this pattern is unwanted as it suggests that even if the dimension of the data is large, the methods would not be able to reliably detect a misspecified population mixing assumption. Methods do perform slightly better, as the evidence of the two-level-mixing effect in the data becomes more apparent, but even in cases of extremely evident effect their ability to detect lack of fit is not consistent.

2.9.1.7 Remarks

Following up on the results of the simulation study, final size assessment (as described in section 2.7.1.4) was conducted for the Constant-HM model, in order to infer whether the methods would perform better on unmatched replications. However, the final size was accurately captured by the model in all instances (see table A.13 in the Appendix), implying that assessment based on unmatched data would have even less power to detect misspecified population mixing assumption.

As discussed in the beginning of section 2.9, using random, rather than constant, infectious periods introduces a loss of information and makes the task of the methods developed more challenging. This was verified with additional simulations, where Exponential infectious periods were used, instead of constant, yielding again similar conclusions; the methods could not systematically detect lack of fit.

To further investigate the (non) effect of N on the methods, and in particular the

relative (to the dimension of the data) jump effect of the removal curve (see section 2.9.1.5), an alternative set-up was considered, where the number of individuals in a household C_H grows proportionally with N. This type of set-up ensures that the relative jump effect C_H/N remains the same as N grows (as opposed to our original set-up where it deteriorates) and allows more power (since the posterior predictive removal curve becomes less uncertain) to detect lack of fit. Provisional simulations indicated that, under this set-up, misspecified models can systematically be detected as N gets larger. However, more extensive simulation studies were not conducted as such a set-up is not practically interesting. More specifically, in practice the more interesting case is, as in our original set-up, where the population size gets large as the number of households gets large (and number of individuals in a household is fixed); as opposed to the alternative set-up where the population size gets large as the number of individuals in a household gets large (and the number of households is fixed).

An alternative, and in a sense natural, way to assess the population mixing assumption is by assessing the posterior predictive distribution of the number (or proportion) of local infections n_L (see e.g. Alharthi (2016)). The simple idea of this approach is that the higher the number (or proportion) of local infections, that the model predicts, the more the evidence of two-level effect in the observed data; so one can deduce the extent of the two-level mixing effect in the observed data by looking at a histogram of n_L and/or by making suitable probability calculations for n_L (such as $P(n_L < c)$ for some $c > 0$), under its posterior predictive distribution (see Alharthi (2016) for more details).

2.10 Discussion

2.10.1 Addressing chapter aims

The work of this chapter described two novel posterior predictive checking methods, based on removal curves. The first method, referred to as the distance method (see section 2.5), is based on the natural idea of measuring the similarity between removal curves by calculating their distance, i.e. a distance function is used as a test statistic. The quantitative end point of the method is the folded ppp-value, a measure of fit that is deterministically connected with the usual ppp-value, under the assumption of symmetric posterior predictive distribution, and thus can easily be interpreted. The second method, referred to as the position-time method (see section 2.6), assesses the fit of the observed curve pointwise, by checking its plausibility under its posterior predictive distribution at the time points that the observed curve is not identically constant; this method is naturally suited for the stochastic process setting. One attractive feature of the method is that it allows the calculation of summaries (over time) of general events (with respect to the posterior predictive distribution) by integrating (over time) the indicator function of the desired event in question. For example, one can calculate the proportion of time that the observed removal curve spends in any (inverse) quantile interval of its posterior predictive distribution.

Both of the methods enjoy increased performance by making use of a time shifting intervention (see section 2.4). This shifting removes the undesired noise that exists in the initial stage of the epidemic (where epidemic processes typically behave like branching processes) and creates more informative conditions for assessing the similarity between observed and replicated data. In addition, both the distance and the position-time methods offer the possibility to be applied to unmatched (major outbreak) replicated datasets; a procedure that is computationally much cheaper than using only matched replicated datasets. This is made possible by a method (see

section 2.3) that classifies each replication from the posterior predictive distribution as a minor or a major outbreak and it is associated with the fact that epidemic models typically demonstrate threshold behaviour.

Extensive simulation studies showed that both the distance and the position-time methods perform very well as tools of assessing the infectious period and the infection rate form assumption of SIR models (see sections 2.7.1 and 2.8.1, respectively). Particularly appealing, is the fact that the methods enjoy increased performance as the dimension of the observed data gets larger; this is very useful from a practical standpoint as in real-life applications interest is in larger scale rather than smaller scale outbreaks. In addition, simulations suggested that when the lack of fit is more apparent (e.g. due to more clear model misspecification cases or due to less clear model misspecification cases but with the dimension of the observed data being large enough), misspecification can successfully be detected using unmatched, rather than matched, replications; once again, this is of high practical importance because avoiding the need to produce matched replications can substantially reduce the computational cost associated with the implementation of the methods.

2.10.2 Limitations

A drawback of the methods is that they failed to detect misspecification of the infectious period distribution when one of the Gamma-HM or the Constant-HM models was fitted to data generated from the other (see section 2.7.1.4). What has to be noted though, is that this should be expected to happen for large values of the shape parameter ν, since as $\nu \to \infty$ the Gamma-HM model reduces to the Constant-HM model (see section 2.7.1.4).

Another drawback of the distance and the position-time methods, is that as seen in section 2.9, they do not have enough power to detect lack of fit in the instances

when the population mixing assumption is misspecified. It must be noted that the methods can not systematically detect lack of fit, even in cases of extreme evidence of two-level-mixing and in addition the performance of the methods does not improve (it remains more or less the same) as the dimension of the observed data gets larger. These observations imply that alternative methods are required to successfully assess the population mixing assumption. This is the topic of chapter 3.

2.10.3 General remarks

Although the performance of the methods was examined via extensive simulation studies, this is never the end of an investigation; there are always more simulation conditions to be considered. Moreover, during the derivation process of the methods, numerous decisions were involved, such as the specific choice of distance function (and the regions of integration) or the choice of interval over which the mean removal curve is not identically constant (in the unmatched case), among others. The decisive factors behind making these decisions were intuition and rational thinking. Ideally, one would be able to investigate the performance of the methods theoretically as well, and base all decisions of the derivation process on mathematical rigor. However, the complexity of the epidemic setting, and the fact that the closed forms of the posterior and the posterior predictive distributions are unknown (if analytically tractable) make this task very challenging, if not infeasible.

The methods developed in this chapter are to be used as tools of model assessment and not model comparison. That is to say, that our interest is in assessing the fit of a considered model without the need of comparing it with other models; if a practitioner is satisfied with the assessment, they can proceed with the analysis. For example, it is not of interest to compare the folded ppp-values from two models that have been fitted to the same data; in fact, such a comparison is not advisable in the context of the single realization setting (see discussion and example in section 2.2.3) as the

more stochastic a model is, the harder it is to yield a large folded ppp-value when misspecified e.g. see the results for the Exp-HM, Gamma-HM and Constant-HM models, under misspecification, in simulation study A (see section 2.7.1.4). Instead, it is much more interesting to compare the folded ppp-values of a given model when fitted to two different datasets; for one dataset the model being correctly specified and for the other being misspecified. This is the reason that the results of the simulation studies (see sections 2.7.1.4, 2.8.1.4 and 2.9.1.4) were presented for a given model under different scenarios rather than for different models on a given scenario.

2.10.4 Further work

Interestingly, some of the methods of this chapter can also be applied in the context of approximate Bayesian computation (ABC) inference. Similarly to MCMC methods, the object of interest of ABC methods is the posterior distribution, but the distinguishing difference is that the latter are simulation-based methods that avoid likelihood calculation (see e.g. Kypraios et al. (2017); McKinley et al. (2018)). The basic idea of ABC is the following. Suppose that some data $\boldsymbol{y}$ have been observed. At first, a candidate parameter $\boldsymbol{\theta}$ is proposed from some prior density $\pi(\boldsymbol{\theta})$. Then, the model is simulated using $\boldsymbol{\theta}$ to produce a dataset $\boldsymbol{y}^*$. If $\boldsymbol{y}$ and $\boldsymbol{y}^*$ are sufficiently close, say $d(\boldsymbol{y}, \boldsymbol{y}^*) < \epsilon$, for some distance function d and some $\epsilon > 0$, then $\boldsymbol{\theta}$ is accepted; or else it is rejected. The procedure is repeated until a desired number of accepted values is achieved and thus its output is a sample of model parameters drawn from the density $\pi(\boldsymbol{\theta} \mid d(\boldsymbol{y}, \boldsymbol{y}^*) < \epsilon)$. In the case that d is suitably chosen and ϵ is sufficiently small, $\pi(\boldsymbol{\theta} \mid d(\boldsymbol{y}, \boldsymbol{y}^*) < \epsilon)$ is a good approximation to the posterior density of interest $\pi(\boldsymbol{\theta} \mid \boldsymbol{y})$. In practice, as suggested in O'Neill (2010), the choice of d is a non-trivial matter and the quality of the above approximation largely depends on d. In the context of stochastic epidemic modelling, the most commonly used distance function is the Euclidean distance between aggregated (by days or weeks) removal vectors (see e.g. Kypraios et al. (2017); McKinley et al. (2018)). Having, as part of the work

in this chapter, defined distance functions to calculate the distance between removal curves (2.5.3) it seems natural to assess whether their use improves the performance of current ABC algorithms. In addition, it is also interesting to investigate whether the application of time shifting (see section 2.4) can improve the efficiency of such algorithms; intuition suggests that by applying time shifting one would be able to achieve the desired number of accepted proposed values more quickly, since if no shifting is applied a lot of proposed values are rejected simply because of the high stochasticity of the simulated removal curves (see 2.2.6) which in turn might yield inappropriately large distances between observed and simulated data. Finally, it is also worth investigating if one should produce simulated data under the proposed values unconditionally, or by imposing some condition on the final size, such as it being the same as the observed or within some proximity to the observed. All these investigations are the topic of research in progress, for which the initial indications are encouraging.

Chapter 3

A Classical Hypothesis Test for Assessing the Population Mixing Assumption of SIR Models

3.1 Introduction

3.1.1 Chapter motivation and aims

As seen in the previous chapter (section 2.9), the distance and position-time methods cannot reliably be used as tools for assessing the population mixing assumption of SIR models; these methods fail to systematically expose the lack of fit of the standard SIR model when fitted to data with a two-level-mixing effect, even in cases where the dimension of the data is large, and the evidence of two-level-mixing is extremely apparent. Therefore, alternative ways to assess the population mixing assumption are required. A key observation behind the methodology of this chapter is that in the cases that the population structure is known (i.e. the household which each individual belongs to is known), and individual event times are available (event times could be either infection times or removal times), the observed data consist not only of event times, but of the household labels which the event times correspond to as

well, i.e. each individual event time is associated with the label of the household that the individual belongs to. Recall that the distance and the position-time methods do not make use of such household label data as they are based solely on removal curves (see sections 2.5 and 2.6). Our speculation is that household label data can be very informative in assessing the population mixing assumption; e.g. in cases of data with two-level-mixing effect (that is, higher within household than between household infectivity), one should expect that events of individuals belonging to the same household would occur closer in time rather than further apart. The aim of this chapter is to develop a method that assesses the population mixing assumption by effectively utilizing the information in the household label data.

All runs and plots in this chapter are produced using the statistical programming language R Core Team (2019).

3.1.2 Chapter layout

The remainder of this chapter is structured as follows. Section 3.2 defines the test, describes its implementation procedure and explaines how is interpreted.

Section 3.3 examines the performance of the test in assessing the population mixing assumptions of SIR models, via the use of an extensive simulation study.

In section 3.4 the test is applied to a widely used real dataset example and its results are compared with what is supported by the literature.

Finally, section 3.5 highlights the main accomplishments of this chapter, gives the limitations and discusses general remarks and further work.

3.2 A classical hypothesis test based on household label data

3.2.1 Setting, notation and rationale

Consider the setting and notation of section 1.3.5.7, where the $C = N + 1$ individuals in the population, of which initially N are susceptible and 1 is infective, are partitioned into l households, labelled as $1, 2, \ldots, l$, with each household m consisting of C_m individuals, $m = 1, 2, \ldots, l$, so that $C = \sum_{m=1}^{l} C_m$. In addition, assume that in each household m there is more than one individual, i.e. assume that $C_m \geq 2$ for all $m = 1, 2, \ldots, l$ (see the relevant remark in section 3.5.3 on how this assumption can be relaxed to allow for the case that $C_m = 1$). Within this setting, consider an epidemic outbreak of n events. Let $\boldsymbol{e} = (e_1, e_2, \ldots, e_n)$ denote the time-ordered event times and $\boldsymbol{g^e} = (g_1^{\boldsymbol{e}}, g_2^{\boldsymbol{e}}, \ldots, g_n^{\boldsymbol{e}})$ their corresponding (time-ordered according to the events) household labels, such that individual k, with event time e_k, belongs to household $g_k^{\boldsymbol{e}} \in \{1, 2, \ldots, l\}$, $k = 1, 2, \ldots, n$. The events times could be either infection times (in which case g_k is the household of individual k, infected at time e_k, $k = 1, 2, \ldots, n$), or removal times (in which case g_k is the household of individual k, removed at time e_k, $k = 1, 2, \ldots, n$) and where relevant a distinction is made. In the instances that the time-ordered event times and their corresponding household labels are random vectors, under some specified sampling distribution, they are denoted as $\boldsymbol{e}^{sam} = (e_1^{sam}, e_2^{sam}, \ldots, e_n^{sam})$ and $\boldsymbol{g}^{\boldsymbol{e}^{sam}} = (g_1^{\boldsymbol{e}^{sam}}, g_2^{\boldsymbol{e}^{sam}}, \ldots, g_n^{\boldsymbol{e}^{sam}})$ respectively, whereas in the case they represent the observed data, they are denoted as $\boldsymbol{e}^{obs} = (e_1^{obs}, e_2^{obs}, \ldots, e_n^{obs})$ and $\boldsymbol{g}^{\boldsymbol{e}^{obs}} = (g_1^{\boldsymbol{e}^{obs}}, g_2^{\boldsymbol{e}^{obs}}, \ldots, g_n^{\boldsymbol{e}^{obs}})$.

The main idea behind the methodology developed, is that if there is a two-level-mixing effect in the data (in which case an infective individual is more likely to infect susceptible individuals within his household rather than outside of it), then when looking at the household labels corresponding to time-ordered event times, one should

expect that same household labels will appear closer together (i.e. clustered) rather that further apart. Conversely, if there is no two-level-mixing effect in the data (and the population is homogeneously mixing) the labels of each household will typically appear completely unpatterned. To make this idea clearer assume that the number of events is $n = 9$, the population structure is given by $l = 4$ and $C_m = 3$, $m = 1, 2, 3, 4$, and that the event times correspond to infection times. In the instance of extreme two-level-mixing effect (where an infection of an individual in a household is followed by an almost immediate infection of all susceptibles within the household) a typically realized household label dataset would be $\boldsymbol{g}^{\boldsymbol{e}} = (3, 3, 3, 1, 1, 1, 4, 4, 4)$, whereas in the instance of homogeneous mixing a realized household label dataset would typically look like $\boldsymbol{g}^{\boldsymbol{e}} = (3, 1, 4, 2, 4, 4, 1, 3, 2)$. This idea of household label clustering is made more precise via the construction of a classical hypothesis test based on household labels, which is now described.

3.2.2 Procedure, null hypothesis and test statistic

In general context, one sufficient (but certainly not necessary) condition to facilitate a classical hypothesis test for a null hypothesis H_0 is via the use of a test statistic T (a scalar function of the data; see section 1.3.3) whose sampling distribution, under H_0, is known and independent of any unknown parameters; the plausibility of H_0 is then tested by examining the consistency of the observed value of T with respect to its sampling distribution under H_0. The test procedure is similar to a posterior predictive check (see section 1.3.3), but the difference is that the plausibility of the observed data is assessed with respect to the sampling distribution under H_0, rather than the posterior predictive distribution. More explicitly, and in the present context, if, as denoted above in section 3.2.1, $\boldsymbol{g}^{\boldsymbol{e}^{obs}}$ are the observed household labels and $\boldsymbol{g}^{\boldsymbol{e}^{sam}}$ is a random vector of household labels having the sampling distribution under H_0 (denoted as $\boldsymbol{g}^{\boldsymbol{e}^{sam}} \sim H_0$), then $T^{obs} := T(\boldsymbol{g}^{\boldsymbol{e}^{obs}})$ is the observed value of T and $T^{sam} := T(\boldsymbol{g}^{\boldsymbol{e}^{sam}})$ is a random variable having the sampling distribution of T under H_0

(denoted as $T^{sam} \sim H_0$). The plausibility of the null hypothesis H_0 is tested visually, by imposing T^{obs} on a histogram of sampled values from T^{sam}, and quantitatively, by calculating the p-value (the probability that $T^{sam} \sim H_0$ is more extreme than the observed value T^{obs}), given by

$$\begin{aligned} \text{p-value} &= P(T^{sam} \leq T^{obs} \mid H_0) \\ &= \mathrm{E}(\mathbb{1}_{\{T^{sam} \leq T^{obs}\}} \mid H_0) = \int \mathbb{1}_{\{T^{sam} \leq T^{obs}\}} \pi(\boldsymbol{g}^{\boldsymbol{e}^{sam}} \mid H_0) d\boldsymbol{g}^{\boldsymbol{e}^{sam}}. \end{aligned} \tag{3.1}$$

The components of the test are constructed as follows. The null hypothesis H_0 is set to be the assumption of a homogeneously mixing population, that is, as H_0: the population is homogeneously mixing. The key is in recognizing that (given a number of events n and a population structure l, C_m, $m = 1, 2, \ldots, l$), under the assumption of homogeneous mixing H_0, the sampling distribution of the discrete random vector $\boldsymbol{g}^{\boldsymbol{e}^{sam}}$ is known and independent of any model parameters (only depends on n, l, C_m, $m = 1, 2, \ldots, l$). To see this, in the case where event times are infection times, note that, under the assumption of homogeneous mixing, the probability that an infective individual contacts (and infects) a susceptible one does not depend on which pair is considered. In the case that event times are removal times, note additionally that the time that an individual remains infective does not depend on the individual; a common assumption of all models in this thesis, as well as most epidemic models, is that infectious periods are i.i.d. according to a random variable T_D (see section 1.3.5.1). Therefore, irrespective of whether the event times are infection or removal times, a realization from the sampling distribution of $\boldsymbol{g}^{\boldsymbol{e}^{sam}} \sim H_0$ is achieved, by choosing uniformly at random a permutation of n out of the total C individuals (i.e. choosing, without replacement and uniformly at random, a sequence of n out of the total C individuals) and recording their corresponding household labels (see section B.2 in Appendix for the analytic expression of the joint p.m.f. of the random vector

$\boldsymbol{g}^{e^{sam}} \sim H_0$).

The challenge then lies in defining an appropriate test statistic T on the space of the n-dimensional household label vectors $\mathbb{Z}_{>0}^n$ (i.e. $T : \boldsymbol{g}^{\boldsymbol{e}} \in \mathbb{Z}_{>0}^n \mapsto T(\boldsymbol{g}^{\boldsymbol{e}}) \in \mathbb{R}$) so that the null hypothesis of homogeneous mixing H_0 is effectively tested. To this end, T is constructed to have an ordinal nature, where the higher (lower) the two-level-mixing effect the lower (higher) the value of T. More specifically, given a realization (observed or simulated) of household label data $\boldsymbol{g}^{\boldsymbol{e}} = (g_1^{\boldsymbol{e}}, g_2^{\boldsymbol{e}}, \ldots, g_n^{\boldsymbol{e}})$, each household m, $m = 1, 2, \ldots, l$, is assigned a value $s_{\boldsymbol{g}^{\boldsymbol{e}}}^{(m)}$, quantifying the two-level-mixing effect associated with the labels of household m, so that the higher (lower) the effect the lower (higher) the value of $s_{\boldsymbol{g}^{\boldsymbol{e}}}^{(m)}$ (details on the specification of $s_{\boldsymbol{g}^{\boldsymbol{e}}}^{(m)}$ to follow right below). Due to the ordinal nature of $s_{\boldsymbol{g}^{\boldsymbol{e}}}^{(m)}$, the total two-level-mixing effect in $\boldsymbol{g}^{\boldsymbol{e}}$ is quantified by summing the $s_{\boldsymbol{g}^{\boldsymbol{e}}}^{(m)}$ of all households. That is to say, that T, defined as $T(\boldsymbol{g}^{\boldsymbol{e}}) = \sum_{m=1}^{l} s_{\boldsymbol{g}^{\boldsymbol{e}}}^{(m)}$, has the desired ordinal nature.

What remains to be specified is $s_{\boldsymbol{g}^{\boldsymbol{e}}}^{(m)}$, $m = 1, 2, \ldots, l$. This is done as follows. Let $\nu_{\boldsymbol{g}^{\boldsymbol{e}}}^{(m)}$ denote the number of times that the label of household m appears in $\boldsymbol{g}^{\boldsymbol{e}}$ and, assuming that $\nu_{\boldsymbol{g}^{\boldsymbol{e}}}^{(m)} \geq 1$, let $\boldsymbol{f}_{\boldsymbol{g}^{\boldsymbol{e}}}^{(m)} = (f_1^{(m)}, f_2^{(m)}, \ldots, f_{\nu_{\boldsymbol{g}^{\boldsymbol{e}}}^{(m)}}^{(m)})$ denote the vector of indices of $\boldsymbol{g}^{\boldsymbol{e}}$ at which the labels of household m appear; note that for $\nu_{\boldsymbol{g}^{\boldsymbol{e}}}^{(m)} = 1$, $\boldsymbol{f}_{\boldsymbol{g}^{\boldsymbol{e}}}^{(m)}$ reduces to a scalar, i.e. $\boldsymbol{f}_{\boldsymbol{g}^{\boldsymbol{e}}}^{(m)} = f_1^{(m)}$ where $f_1^{(m)}$ is the index of $\boldsymbol{g}^{\boldsymbol{e}}$ where the first (and only) appearance of the label of household m occurs. For example, if $\boldsymbol{g}^{\boldsymbol{e}} = (3, 1, 4, 2, 4, 4, 1, 3)$, then $\nu_{\boldsymbol{g}^{\boldsymbol{e}}}^{(3)} = 2$ with $\boldsymbol{f}_{\boldsymbol{g}^{\boldsymbol{e}}}^{(3)} = (1, 8)$ and $\nu_{\boldsymbol{g}^{\boldsymbol{e}}}^{(2)} = 1$ with $\boldsymbol{f}_{\boldsymbol{g}^{\boldsymbol{e}}}^{(2)} = 4$. Provided that the label of household m appears twice or more in $\boldsymbol{g}^{\boldsymbol{e}}$ (i.e. $\nu_{\boldsymbol{g}^{\boldsymbol{e}}}^{(m)} \geq 2$), so that measuring spread is possible, the idea of household label clustering, described in section 3.2.1 (i.e. the idea that the higher (lower) the two-level-mixing effect in the data the closer together (further apart) same household labels appear) is quantified by defining $s_{\boldsymbol{g}^{\boldsymbol{e}}}^{(m)}$ as a measure of spread for the labels of household m that appear in $\boldsymbol{g}^{\boldsymbol{e}}$. More precisely, when $\nu_{\boldsymbol{g}^{\boldsymbol{e}}}^{(m)} \geq 2$, $s_{\boldsymbol{g}^{\boldsymbol{e}}}^{(m)}$ measures the spread of the labels of household

m that appear in $\boldsymbol{g^e}$ by being set as $s^{(m)}_{\boldsymbol{g^e}} = f^{(m)}_{\nu^{(m)}_{\boldsymbol{g^e}}} - f^{(m)}_1 - (\nu^{(m)}_{\boldsymbol{g^e}} - 1)$. For example, if $\boldsymbol{g^e} = (3, 1, 4, 2, 4, 4, 1, 3, 2, 5, 5, 5)$ then $s^{(1)}_{\boldsymbol{g^e}} = 7-2-(2-1) = 4$, $s^{(2)}_{\boldsymbol{g^e}} = 9-4-(2-1) = 4$, $s^{(3)}_{\boldsymbol{g^e}} = 8-1-(2-1) = 6$, $s^{(4)}_{\boldsymbol{g^e}} = 6-3-(3-1) = 1$ and $s^{(5)}_{\boldsymbol{g^e}} = 12-10-(3-1) = 0$. Notice that, as just specified, $s^{(m)}_{\boldsymbol{g^e}}$ can be calculated by counting the number of non-household m labels intervening between the first and last household m label of $\boldsymbol{g^e}$. From this standpoint, $s^{(m)}_{\boldsymbol{g^e}}$ can be thought of as 'penalizing' household m according to the extent that it deviates from the most obvious realization of two-level-mixing effect, where its labels appear in consecutive order (such as household 5 in the example right above).

In the remaining cases that $\nu^{(m)}_{\boldsymbol{g^e}} = 0$ and $\nu^{(m)}_{\boldsymbol{g^e}} = 1$, since measuring the spread of the labels of household m is not possible, $s^{(m)}_{\boldsymbol{g^e}}$ is defined differently; although, as mentioned above, the intention is still for $s^{(m)}_{\boldsymbol{g^e}}$ to quantify the two-level-mixing effect associated with the labels of household m in an ordinal nature (so that T has an ordinal nature). For $\nu^{(m)}_{\boldsymbol{g^e}} = 0$, considering that (given a realization $\boldsymbol{g^e}$ of n events) when there is a two-level-mixing effect in the data (in which case the epidemic typically spreads within rather than between households), the label of less rather than more households should appear in $\boldsymbol{g^e}$, it seems sensible to set $s^{(m)}_{\boldsymbol{g^e}} = 0$. In fact, if one thinks a bit more carefully, setting $s^{(m)}_{\boldsymbol{g^e}} = 0$ (in the instance that $\nu^{(m)}_{\boldsymbol{g^e}} = 0$) appears to be the only sensible assignment. To see this, consider the case of the most obvious two-level-mixing effect such as the example realization in section 3.2.1 with $n = 9$, $l = 4$, $C_m = 3$, $m = 1, 2, 3, 4$, and $\boldsymbol{g^e} = (3, 3, 3, 1, 1, 1, 4, 4, 4)$. For T to have the desired ordinal nature (the higher (lower) the two-level-mixing effect the lower (higher) the value of T) such realizations must yield the minimum value of T (which is 0) and for that to happen, households m for which $\nu^{(m)}_{\boldsymbol{g^e}} = 0$ can only be such that $s^{(m)}_{\boldsymbol{g^e}} = 0$ e.g. in the example above, $\nu^{(1)}_{\boldsymbol{g^e}} = \nu^{(3)}_{\boldsymbol{g^e}} = \nu^{(4)}_{\boldsymbol{g^e}} = 3$, so (from the definition of $s^{(m)}_{\boldsymbol{g^e}}$ for $\nu^{(m)}_{\boldsymbol{g^e}} \geq 2$) $s^{(1)}_{\boldsymbol{g^e}} = s^{(3)}_{\boldsymbol{g^e}} = s^{(4)}_{\boldsymbol{g^e}} = 0$ and thus $T(\boldsymbol{g^e}) = \sum_{m=1}^{4} s^{(4)}_{\boldsymbol{g^e}}$ can only be 0 if household 2 for which $\nu^{(2)}_{\boldsymbol{g^e}} = 0$ is such that $s^{(2)}_{\boldsymbol{g^e}} = 0$.

Turning now to the specification of $s^{(m)}_{\boldsymbol{g^e}}$, when $\nu^{(m)}_{\boldsymbol{g^e}} = 1$, consider the extreme case that $\nu^{(m)}_{\boldsymbol{g^e}} = 1$ for any household m which its label appears in $\boldsymbol{g^e}$; for example, if $n = 5$, $l = 6$, $C_m = 3$ and $m = 1, 2, \ldots, 6$, one such realization might be $\boldsymbol{g^e} = (1, 2, 3, 4, 5)$. This case, referred to as the most extreme case of negative two-level-mixing effect, implies that the outbreak progresses only between and not within households, a pattern that is exactly the opposite of what happens in the case of two-level-mixing effect (where the epidemic is more likely to spread within rather than between households). In order for T to have the required ordinal nature, realizations of this type must produce the maximum value of T. To this end, considering the fact that $n \geq f^{(m)}_{\nu^{(m)}_{\boldsymbol{g^e}}}$ in all instances, $s^{(m)}_{\boldsymbol{g^e}}$ is specified as in the case of $\nu^{(m)}_{\boldsymbol{g^e}} \geq 2$ (see above) with the difference being that $f^{(m)}_{\nu^{(m)}_{\boldsymbol{g^e}}}$ is replaced by n, i.e. as $s^{(m)}_{\boldsymbol{g^e}} = n - f^{(m)}_1 - (\nu^{(m)}_{\boldsymbol{g^e}} - 1) = n - f^{(m)}_1$. For example for $\boldsymbol{g^e} = (2, 3, 3, 5, 3, 4, 4)$, $s^{(2)}_{\boldsymbol{g^e}} = 7 - 1 = 6$ and $s^{(5)}_{\boldsymbol{g^e}} = 7 - 4 = 3$. Notice that, similar to the case of $\nu^{(m)}_{\boldsymbol{g^e}} \geq 2$ (see above), $s^{(m)}_{\boldsymbol{g^e}}$, for $\nu^{(m)}_{\boldsymbol{g^e}} = 1$, can be calculated by counting the number of non-household m labels. The difference is that instead of counting the number of non-household m labels from the first until the last household m label of $\boldsymbol{g^e}$ ($\nu^{(m)}_{\boldsymbol{g^e}} \geq 2$ case), one counts the number of non-household m labels from the first (and only) household m label until the last index of $\boldsymbol{g^e}$ ($\nu^{(m)}_{\boldsymbol{g^e}} = 1$ case). This counting representation of $s^{(m)}_{\boldsymbol{g^e}}$ highlights how a household m, whose label first appears in $\boldsymbol{g^e}$ at index $f^{(m)}_1$, receives the maximum value of $s^{(m)}_{\boldsymbol{g^e}}$ in the instance that its label does not appear again (i.e. in the instance that $\nu^{(m)}_{\boldsymbol{g^e}} = 1$).

Note that from a practical point of view the notion of negative two-level-mixing effect is less useful since it is not usually plausible for real-life epidemic outbreaks to have higher infectivity between rather than within households. Nonetheless, ensuring that T behaves sensibly under such cases is essential in providing it with the required ordinal nature which, as will be shown below, is fundamental in interpretating the test.

For clarity, the definition of T, along with the specification form of $s^{(m)}_{\boldsymbol{g}^{\boldsymbol{e}}}$, for each value of $\nu^{(m)}_{\boldsymbol{g}^{\boldsymbol{e}}}$, are collected in equation (3.2) below.

$$T(\boldsymbol{g}^{\boldsymbol{e}}) = \sum_{m=1}^{l} s^{(m)}_{\boldsymbol{g}^{\boldsymbol{e}}}, \text{ where } s^{(m)}_{\boldsymbol{g}^{\boldsymbol{e}}} = \begin{cases} 0, & \text{if } \nu^{(m)}_{\boldsymbol{g}^{\boldsymbol{e}}} = 0 \\ n - f^{(m)}_1, & \text{if } \nu^{(m)}_{\boldsymbol{g}^{\boldsymbol{e}}} = 1 \\ f^{(m)}_{\nu^{(m)}_{\boldsymbol{g}^{\boldsymbol{e}}}} - f^{(m)}_1 - (\nu^{(m)}_{\boldsymbol{g}^{\boldsymbol{e}}} - 1), & \text{if } \nu^{(m)}_{\boldsymbol{g}^{\boldsymbol{e}}} \geq 2. \end{cases} \tag{3.2}$$

3.2.3 Implementation and interpretation

Since T is a deterministic function of $\boldsymbol{g}^{\boldsymbol{e}}$ (see equation (3.2)), and the distribution of $\boldsymbol{g}^{\boldsymbol{e}^{sam}} \sim H_0$ is known and independent of any model parameters (see section 3.2.2 above), the distribution of $T^{sam} \sim H_0$ is also independent of any model parameters and independent sampling from T^{sam} can easily be achieved by first drawing an independent sample $\{\boldsymbol{g}^{\boldsymbol{e}^{sam^{(1)}}}, \boldsymbol{g}^{\boldsymbol{e}^{sam^{(2)}}}, \ldots, \boldsymbol{g}^{\boldsymbol{e}^{sam^{(S)}}}\}$ from $\boldsymbol{g}^{\boldsymbol{e}^{sam}} \sim H_0$ (following the procedure described in the second paragraph of section 3.2.2) and then evaluating T at each realization $\boldsymbol{g}^{\boldsymbol{e}^{sam^{(s)}}}$, $s = 1, 2, \ldots, S$, using equation (3.2). That is, $\{T^{sam^{(1)}}, T^{sam^{(2)}}, \ldots, T^{sam^{(S)}}\}$, where $T^{sam^{(s)}} := T(\boldsymbol{g}^{\boldsymbol{e}^{sam^{(s)}}})$, $s = 1, 2, \ldots, S$, is an independent sample from the sampling distribution of $T^{sam} \sim H_0$. The procedure of the test, as described in section 3.2.2, can then be implemented by imposing T^{obs} on the histogram of the sampled values $\{T^{sam^{(1)}}, T^{sam^{(2)}}, \ldots, T^{sam^{(S)}}\}$, and by calculating the p-value via Monte Carlo (MC) approximation as

$$\text{p-value} = \int \mathbb{1}_{\{T^{sam} \leq T^{obs}\}} \pi(\boldsymbol{g}^{\boldsymbol{e}^{sam}} \mid H_0) d\boldsymbol{g}^{\boldsymbol{e}^{sam}} \approx \frac{1}{S} \sum_{s=1}^{S} \mathbb{1}_{\{T^{rep^{(s)}} \leq T^{obs}\}}. \tag{3.3}$$

For convenience, all steps required to implement the test are listed in Algorithm 16.

To describe how the test is interpreted, its output from an example dataset, shown in figure 3.1, is used as a visual medium. Given a histogram of sampled values from

Algorithm 16 Scheme for applying the household label data test

1. **Calculate $\boldsymbol{T^{obs}}$:** Given observed time-ordered event times $\boldsymbol{e}^{obs} = (e_1^{obs}, e_2^{obs}, \ldots, e_n^{obs})$ with corresponding (time-ordered according to the events) household labels $\boldsymbol{g}^{\boldsymbol{e}^{obs}} = (g_1^{\boldsymbol{e}^{obs}}, g_2^{\boldsymbol{e}^{obs}}, \ldots, g_n^{\boldsymbol{e}^{obs}})$ calculate $T^{obs} := T(\boldsymbol{g}^{\boldsymbol{e}^{obs}})$, using equation (3.2).

2. **Sample from $\boldsymbol{T^{sam} \sim H_0}$:** For each s, $s = 1, 2, \ldots, S$, generate a realization $\boldsymbol{g}^{\boldsymbol{e}^{sam^{(s)}}} = (e_1^{sam^{(s)}}, e_2^{sam^{(s)}}, \ldots, e_n^{sam^{(s)}})$ from the sampling distribution of $\boldsymbol{g}^{\boldsymbol{e}^{sam}} \sim H_0$, by choosing uniformly at random a permutation of n out of the total C individuals and recording their corresponding household labels. Then $\{\boldsymbol{g}^{\boldsymbol{e}^{sam^{(1)}}}, \boldsymbol{g}^{\boldsymbol{e}^{sam^{(2)}}}, \ldots, \boldsymbol{g}^{\boldsymbol{e}^{sam^{(S)}}}\}$ is an independent sample from $\boldsymbol{g}^{\boldsymbol{e}^{sam}} \sim H_0$. Calculate $T^{sam^{(s)}} := T(\boldsymbol{g}^{\boldsymbol{e}^{sam^{(s)}}})$, for each s, $s = 1, 2, \ldots, S$, using equation (3.2). Then $\{T^{sam^{(1)}}, T^{sam^{(2)}}, \ldots, T^{sam^{(S)}}\}$ is an independent sample from $T^{sam} \sim H_0$.

3. **Apply the test:** Calculate the p-value using equation (3.3) and impose T^{obs} on the histogram of the sampled values $\{T^{sam^{(1)}}, T^{sam^{(2)}}, \ldots, T^{sam^{(S)}}\}$ from $T^{sam} \sim H_0$.

$T^{sam} \sim H_0$, along with the minimum and maximum values of T (as in figure 3.1), and considering the ordinal nature of T (see section 3.2.2 above), observed values of T (along with their corresponding p-values; see equation (3.3)), are interpreted as follows. Values that fall well within the support of $T^{sam} \sim H_0$ (i.e. closer to the mode rather than the tails of the histogram of $T^{sam} \sim H_0$) are consistent with H_0 and provide no evidence against it; in such cases the associated p-value is not too close to the extreme values of 0 or 1. As values move to the left tail (and beyond) of the histogram of $T^{sam} \sim H_0$ and towards the minimum value of T (that represents the most obvious case of two-level-mixing effect) they become inconsistent with H_0, and provide increasing evidence against it and in favour of the hypothesis H_L, that there is two-level-mixing effect in the data; the corresponding p-value is close to or equal to 0. Similarly, as values move to the right tail (and beyond) of the histogram of $T^{sam} \sim H_0$, and towards the maximum value of T (that represents the most obvious case of negative two-level-mixing effect), they become inconsistent with H_0 and provide increasing evidence against it and in favour of the hypothesis H_R, that

there is negative two-level-mixing effect in the data; the associated p-value is close to or equal to 1.

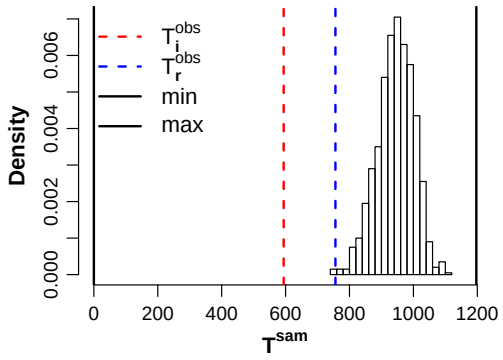


Figure 3.1: Example of assessing the population mixing assumption using the classical hypothesis test for household label data. Observed data are generated from an Exp-2L model ($N = 199$, $C_H = 5$, $R_* = 2.5$, $R_0^H = 1.5$ and $\gamma = 0.1$). The plot is the histogram of 1000 realizations from the sampling distribution of $T^{sam} \sim H_0$ with the observed value (based on infections) $T_{\boldsymbol{i}}^{obs} = 594$ (red, dashed line), the observed value (based on removals) $T_{\boldsymbol{r}}^{obs} = 756$ (blue, dashed line), the minimum value of $T = 0$ (black, solid line) and the maximum value of $T = 1197$ (black, solid line) imposed. The p-values are (based on infections) p-value$_{\boldsymbol{i}} = 0$ and (based on removals) p-value$_{\boldsymbol{r}} = 0.002$.

Note that, as already mentioned in section 3.2.2, cases of negative two-level mixing effect are unlikely to ever be encountered in practice, but they are still useful for providing an ordinal interpretation to the test. Notice also, that plotting a histogram of $T^{sam} \sim H_0$, with the minimum and maximum values of T imposed (as in figure 3.1), is more informative than merely calculating the p-value. For example, a p-value of 0 might correspond to the observed value of T being very near to the left tail of the histogram of $T^{sam} \sim H_0$ (and far from its minimum value) or very near to the minimum value of T (and far from the left tail of the histogram of $T^{sam} \sim H_0$); although the p-value is the same in these two cases, the amount of evidence against H_0 and in favour of H_L is quite different.

An important property of T is that, in addition to the fact that its sampling

distribution $T^{sam} \sim H_0$ is independent of any model parameters, its values are also independent of any model parameters (because T is a statistic, i.e. a function of data only); such statistics (whose sampling distribution does not depend on model parameters) are called ancillary statistics. What this implies, is that the test of H_0 is not just a test for the plausibility of a specific set of parameters of a considered model (that has H_0 as an assumption), but rather a more generic test for the plausibility of the family of models that share H_0 as an assumption, under any set of parameters. Another implication, of the above property of T, is that the test implementation involves no parameter estimation or simulations from the model; recall that the distance and the position-time methods require model fitting via Markov chain Monte Carlo (MCMC) methods and the simulation of replicated outbreaks from the model (see sections 2.5 and 2.6). This is particularly appealing from a computational standpoint, as avoiding the cost of parameter estimation or model simulation makes the test very cheap to perform. What both of these implications suggest is that in practice, it would be most meaningful to conduct the test before any model fitting is done, using it as guide in choosing a model, and proceed after to parameter estimation for the chosen model. For example, if the observed data are consistent with H_0 one may go on and analyze the data choosing a model that assumes homogeneous mixing, such as the standard SIR model or the non-linear SIR model. In the instance that the observed data provide substantial evidence against H_0, and in favour of H_L, one should use this information accordingly and fit a two-level-mixing model to the data in question.

Note that, the generic nature of the test, and in particular the fact that it does not have an alternative hypothesis associated with a set of parameter values of a specific model, does not exclude the possibility that a small value of T (smaller than expected under H_0) might occur by observing clustered label data due to different alternative hypotheses and not necessarily due to H_L. That is to say, that the interpretation of

the test relies on the assumption that the test statistic assumes values according to the ordinal interpretation provided above.

3.2.4 Infection based and removal based assessment

Notice that all steps of the test, from construction to implementation, were conducted without needing to specify whether the event times referred to infection or removal times. For example, as explained in section 3.2.2, the sampling distribution of $\boldsymbol{g}^{e^{sam}} \sim H_0$, and thus also of $T^{sam} \sim H_0$ (see section 3.2.3) is the same, irrespective of whether the event times are infection or removal times. However, it must be emphasized that the observed value of T, and thus the p-value of the test, will generally be different, depending on whether it has been calculated based on observing removals, or infections. This is because, as remarked in section 2.2.4.2, removal times are an i.i.d shift of the infection times, and therefore the household label vector corresponding to time-ordered infection times will generally be different than the household label vector corresponding to time-ordered removal times. To distinguish between infection based and removal based assessment, the observed household label vector, the observed value of T and the associated p-value are denoted as $\boldsymbol{g}^{\boldsymbol{i}^{obs}}$, $T_{\boldsymbol{i}}^{obs}$ and p-value$_{\boldsymbol{i}}$ when calculated based on observing infection times and $\boldsymbol{g}^{\boldsymbol{r}^{obs}}$, $T_{\boldsymbol{r}}^{obs}$ and p-value$_{\boldsymbol{r}}$ when calcuated based on observing removal times. Notice that, in the instance that the shift from infection to removals is constant (i.e. the infectious period is constant) the ordering of individuals remains unchanged, so $\boldsymbol{g}^{\boldsymbol{i}^{obs}} = \boldsymbol{g}^{\boldsymbol{r}^{obs}}$ and in turn $T_{\boldsymbol{i}}^{obs} = T_{\boldsymbol{r}}^{obs}$ and p-value$_{\boldsymbol{i}}$ = p-value$_{\boldsymbol{r}}$. In the instance that the shift is random (i.e. the infectious period is random), any two-level-mixing effect in the data will typically be less apparent under removal based assessment. That is, any clustering effect of same household labels is likely to be, to an extent, clouded from the amount of noise that is introduced in the shift. Hence, in the instances that there is a two-level-mixing effect in the data, one should expect that $T_{\boldsymbol{i}}^{obs}$ will in general be smaller than $T_{\boldsymbol{r}}^{obs}$ and in turn p-value$_{\boldsymbol{i}}$ smaller than p-value$_{\boldsymbol{r}}$.

Nonetheless, applying the test using removal times is still worthwhile (and the best one can do under our assumed framework, where removal times are considered to be observed whereas infection times are missing), since the variation that is introduced from the shift is random and not systematic, i.e. any distortion caused in the order of the households labels, when ordered according to removal times rather than infection times, is not of systematic nature. An appreciation of the above is provided in the example of figure 3.1, where the data are generated from the Exp-2L model. More specifically, although the value of $T_{\boldsymbol{r}}^{obs}$ is slightly larger than the value of $T_{\boldsymbol{i}}^{obs}$, reflecting the fact that the two-level-mixing effect in the data becomes a little less detectable under removal based assessment, it still is small enough to provide substantial evidence against H_0 and in favour of H_L.

The performance of the test, based on both infection and removal data, is thoroughly examined via a simulation study, referred to as simulation study D, which is described in the following section.

3.3 Simulation study D

3.3.1 Purpose

Simulation study D aims to investigate the performance of the household label test in assessing the population mixing assumption. The main interest is in removal based assessment, since under our assumed framework removal times are observed while infection times are missing (see section 1.3.4.2). Nonetheless, in order to quantify the amount of distortion that is introduced in the household label data when conducting removal rather than infection based assessment (see discussion in section 3.2.4), both assessments are conducted and compared. Also of interest is comparing the

performance of the household label test with that of the distance and the position-time methods; recall from section 2.9 and simulation study C (see section 2.9.1), that the distance and the position-time methods were also employed as tools of assessing the population mixing assumption and their performance was not as desired. Lastly, similar to simulation study C (see section 2.9.1), it is important to investigate how the test behaves as the dimension of the data or the level of two-level-mixing effect in the data changes.

3.3.2 Simulation conditions

The simulation conditions of simulation study D (see table 3.1) are very similar to simulation study C (see section 2.9.1.2 for more details), in order to allow for direct comparison between the performance of the household label test and the distance and position-time methods, but there are also differences. More specifically, just like simulation study C, in all instances, data are generated from the two-level-mixing model and all households are taken to have equal size C_H, where C_H is set at 5. The difference is that the choice of infectious period is Exponential ($T_D \sim \text{Exp}(\gamma)$) rather than constant ($T_D \equiv c$), i.e. data are generated from the Exp-2L model rather than the Constant-2L model. This choice allows for the comparison between removal based and infection based assessment to be carried out, since if infectious periods are constant (as in simulation study C), then $\boldsymbol{g}^{\boldsymbol{i}^{obs}} = \boldsymbol{g}^{\boldsymbol{r}^{obs}}$, $T_{\boldsymbol{i}}^{obs} = T_{\boldsymbol{r}}^{obs}$, p-value$_{\boldsymbol{i}}$ = p-value$_{\boldsymbol{r}}$, and the two assessments coincide (see section 3.2.4 above). Notice that, direct comparability conditions with simulation study C are still maintained precisely because p-value$_{\boldsymbol{i}}$ = p-value$_{\boldsymbol{r}}$ for constant infectious periods, and therefore, applying the test on infection data can be seen as applying the test on removal data which were generated from a model with constant infectious period (as in simulation study C). Notice also, that by choosing the more uncertain Exponential infectious period, over its Gamma counterpart, more challenging conditions are created for the removal based assessment to raise evidence against H_0 (see discussion in section 3.2.4 above).

Following the same procedure as in simulation study C, the behaviour of the test, as the dimension of the data or the level of two-level-mixing effect in the data change, is examined by considering simulation scenarios of varying levels of two-level-mixing effect, quantified by R_0^H, and by including different rounds in each scenario, corresponding to different values of initial susceptibles N. More precisely, the scenarios of R_0^H=1, 2, 5, 20 and the rounds of N=99, 199, 499, of simulation study C, are again performed. In addition, utilizing the fact that the test is computationally cheap to implement (see section 3.2.3), the scenario of $R_0^H = 0.5$ and the round of $N = 999$ are considered; the scenario of $R_0^H = 0.5$ is a case of very mild two-level-mixing effect and thus creates very challenging conditions for the test to raise evidence against H_0, while the round of $N = 999$ allows for more informative conclusions on how the test behaves as N gets larger. For the same reason (test being computationally cheap to implement), the sampling variability for each round is thoroughly captured by generating 500 datasets, as opposed to the 24 generated datasets of simulation study C. Note that, for all simulated datasets, the initial infective (and its household) is chosen uniformly at random from the population. Also note, that the generated datasets are conditioned on being major outbreaks, using the approach described in section 2.3 for separating between minor and major outbreaks; the only difference being that the approach is applied to the sampling distribution of the final size instead of its posterior predictive distribution.

As in simulation study C (see section 3.3.2), to establish a better appreciation of the extent of the two-level-mixing effect in each scenario, the mean proportion of local infections (from total infections) $\bar{p_L}$, under the sampling distribution of the model, is calculated. The calculated values of $\bar{p_L}$ are 0.09, 0.27, 0.51, 0.70 and 0.78 for scenarios 1, 2, 3, 4 and 5, respectively, and they are given in table 3.1 for reference.

Note that a simulation scenario for which the data are generated from a homogeneously mixing model (i.e. under H_0) is not considered. This is because it is a well known fact (probability integral transform) that, under the null hypothesis H_0, the sampling distribution of the p-value is uniform.

Table 3.1: Simulation conditions for simulation study D. Each simulation scenario consists of 4 rounds, where the number of initial susceptibles N is set at 99, 199, 499 and 999, respectively. For each round 500 datasets are generated. The number of individuals in each household is set as $C_H = 5$, in all instances.

	Data generating process	**Parameter values**	$\bar{p}_L$
Scenario 1	Exp-2L	$R_* = 2.5, \gamma = 0.1, R_0^H = 0.5, \mu = 0.61$	0.09
Scenario 2	Exp-2L	$R_* = 2.5, \gamma = 0.1, R_0^H = 1, \mu = 1.32$	0.27
Scenario 3	Exp-2L	$R_* = 2.5, \gamma = 0.1, R_0^H = 2, \mu = 2.44$	0.51
Scenario 4	Exp-2L	$R_* = 2.5, \gamma = 0.1, R_0^H = 5, \mu = 3.88$	0.70
Scenario 5	Exp-2L	$R_* = 2.5, \gamma = 0.1, R_0^H = 20, \mu = 4.75$	0.78

3.3.3 Run conditions

Following the procedure described in section 3.2.3 (see Algorithm 16), the test is applied to each generated dataset twice, once based on observing infection times and once based on observing removal times (see section 3.2.4). More specifically, for each dataset, an independent sample of size 10000 from $T^{sam} \sim H_0$ is achieved, and then the two observed values of T, $T_{\boldsymbol{i}}^{obs}$ and $T_{\boldsymbol{r}}^{obs}$, and their corresponding p-values, p-value$_{\boldsymbol{i}}$ and p-value$_{\boldsymbol{r}}$, are calculated and recorded.

3.3.4 Results

The performance of the test is examined by analyzing the p-values, based on both infection and removal data, from each dataset at each simulation scenario and round. Since data are generated from the two-level-mixing model, the desirable effect of N would be for more data (i.e. bigger N) to provide more evidence against H_0 and in

favour of H_L. Similarly, the evidence against H_0 and in favour of H_L should also increase as the two-level-mixing effect in the data increases (i.e. as R_0^H increases).

Tables of median (95% quantile interval) p-values provide summaries from each round of each scenario (tables 3.2 and 3.3) while complete results of all p-values against dataset index are given in the Appendix (figure A.21). Similar to simulation study C (see section 2.9.1.4), to facilitate appreciation of the effect of N and R_0^H, in all tables and figures, rows and columns conveniently correspond to the different scenarios (values of R_0^H) and rounds (values of N), respectively. That is, the effect of R_0^H on the results can be gauged by choosing a column and looking across rows while the effect of N by choosing a row and looking across columns.

3.3.4.1 Infection based assessment

First, infection based results are reported. Recall from section 3.3.2, that applying the test on infection data can be seen as applying the test on removal data which were generated from a model with constant infectious period (as in simulation study C). Hence, the p-value$_{\boldsymbol{i}}$ of simulation study D, can be compared with the ppp-value (distance method) or the $\sqrt{\text{MSE}}$ (position-time method) of simulation study C (see section 2.9.1.4). Table 3.2 below, and figure A.21 in the Appendix, show that infection based results are sensible in all scenarios and rounds. Also, the effect of N and R_0^H is the desirable one, since larger values of N (R_0^H) yield smaller p-value$_{\boldsymbol{i}}$, for a given R_0^H (N). For example, in scenario 1 ($R_0^H = 0.5$) the median (95% quantile interval) p-value$_{\boldsymbol{i}}$ is 0.24 (0, 0.93), 0.17 (0, 0.85), 0.06 (0, 0.72) and 0.02 (0, 0.47) for $N =$ 99, 199, 499 and 999 respectively, implying that even in the case of a very mild two-level-mixing effect (where the mean proportion of local infections is only $\bar{p_L} = 0.09$) the test would still provide adequate evidence against H_0 and in favour of H_L, if the dimension of the data becomes large enough. The power of the test is also exhibited in scenario 2 ($R_0^H = 1$), where the two-level-mixing effect in the data is still relatively mild ($\bar{p_L} = 0.27$), but the sampling distribution of the p-value$_{\boldsymbol{i}}$ is concentrated near 0,

even for the smaller values of N. It is also worth noting that for scenarios 3-5, where the two level-mixing effect becomes increasingly apparent, the sampling distribution of the p-value$_i$ is consistently a point mass at 0, meaning that, as desired, the evidence against H_0 and in favour of H_L would systematically be very strong.

Table 3.2: Median (95% quantile interval) p-value from the household labels test based on observing infection times, p-value$_i$, for simulation study D. The number of datasets that the median and quantile interval is taken over is 500. Simulation conditions for each scenario are given in table 3.1.

	$N = 99$	$N = 199$	$N = 499$	$N = 999$
Scenario 1	0.24 (0, 0.93)	0.17 (0, 0.85)	0.06 (0, 0.72)	0.02 (0, 0.47)
Scenario 2	0.01 (0, 0.43)	0 (0, 0.10)	0 (0, 0)	0 (0, 0)
Scenario 3	0 (0, 0)	0 (0, 0)	0 (0, 0)	0 (0, 0)
Scenario 4	0 (0, 0)	0 (0, 0)	0 (0, 0)	0 (0, 0)
Scenario 5	0 (0, 0)	0 (0, 0)	0 (0, 0)	0 (0, 0)

The performance of the test is in contrast with the performance of the distance and the position-time methods which, as seen in section 2.9.1, did not have enough power to discard the homogeneously mixing model, under similar simulation conditions. Also in contrast is the effect of N and R_0^H on the results, since for the test the power increases as N and R_0^H increase, whereas for the distance and the position-time methods, N and R_0^H had no apparent effect on the results (see section 2.9.1.4); for example, recall that even for the extreme scenario of $R_0^H = 20$, where the level of the two-level-mixing effect in the data is extremely evident, the distance and the position-time methods could not expose the lack of fit of the homogeneously mixing model.

3.3.4.2 Removal based assessment

As can be seen in table 3.3 below and figure A.21 in the Appendix, removal based results are very similar to the infection based results, with the difference being that

p-value$_{\boldsymbol{r}}$ is typically a bit higher than p-value$_{\boldsymbol{i}}$. More precisely, just like in the infection based assessment, results are sensible in all scenarios and rounds and the effect of N and R_0^H is the desirable one, since the larger the values of N (R_0^H) the smaller the p-value$_{\boldsymbol{r}}$ for a given R_0^H (N). As explained in section 3.2.4, the fact that p-value$_{\boldsymbol{r}}$ is typically a bit higher than p-value$_{\boldsymbol{i}}$ for a given round of a given scenario, is to be expected as the two-level-mixing effect in the data becomes less apparent when the observed household labels correspond to removal times, rather than infection times. Nonetheless, and more importantly, the p-value$_{\boldsymbol{r}}$ are still low enough to provide evidence against H_0 and in favour of H_L, where appropriate. More specifically, it is only in scenario 1 ($R_0^H = 0.5$) that the evidence against H_0 and in favour of H_L is not systematic; median (95% quantile interval) p-value$_{\boldsymbol{r}}$ is 0.44 (0.02, 0.97), 0.41 (0.01, 0.94), 0.32 (0.01, 0.94) and 0.29 (0.01, 0.92) for $N = 99$, 199, 499 and 999, respectively. Considering the fact that scenario 1 represents conditions of very mild two-level-mixing effect (where the mean proportion of local infections is only $\bar{p_L} = 0.09$) and that some loss of information when using removals is inevitable (see section 3.2.4), the results are sensible. As soon as the two-level mixing effect becomes a bit less mild, for example as in scenario 2 ($R_0^H = 1$, $\bar{p_L} = 0.27$), it is successfully detected by the removal based test; in scenario 2, median (95% quantile interval) p-value$_{\boldsymbol{r}}$ is 0.09 (0, 0.76), 0.02 (0, 0.59), 0 (0, 0.17) and 0 (0, 0) for $N = 99$, 199, 499 and 999, respectively. Similar to the infection based results, for increasingly apparent two level-mixing effect (scenarios 3-5), the sampling distribution of the p-value$_{\boldsymbol{r}}$ is, as desired, consistently a point mass at 0.

It is worth pointing out that, even with the added challenge of using Exponential, rather than constant, infectious periods (see section 3.3.2), the removal based household label test greatly outperforms the distance and the position-time methods, in assessing the population mixing assumption, under similar simulation scenarios (see section 2.9.1); for example for $R_0^H = 2$ the median (95% quantile interval) folded ppp-value from the distance method (with desired optimal value being 1) was 0.29

Table 3.3: Median (95% quantile interval) p-value from the household labels test based on observing removal times, p-value$_r$, for simulation study D. The number of datasets that the median and quantile interval is taken over is 500. Simulation conditions for each scenario are given in table 3.1.

	$N = 99$	$N = 199$	$N = 499$	$N = 999$
Scenario 1	0.44 (0.02, 0.97)	0.41 (0.01, 0.94)	0.32 (0.01, 0.94)	0.29 (0.01, 0.92)
Scenario 2	0.09 (0, 0.76)	0.02 (0, 0.59)	0 (0, 0.17)	0 (0, 0)
Scenario 3	0 (0, 0.11)	0 (0, 0)	0 (0, 0)	0 (0, 0)
Scenario 4	0 (0, 0)	0 (0, 0)	0 (0, 0)	0 (0, 0)
Scenario 5	0 (0, 0)	0 (0, 0)	0 (0, 0)	0 (0, 0)

(0.02, 0.88), 0.41 (0.05, 0.90) and 0.20 (0.05, 0.95) (see table 2.30), for $N = 99$, 199 and 499, while the corresponding median (95% quantile interval) p-value$_r$ (with desired optimal value being 0) is 0 (0, 0.11), 0 (0, 0) and 0 (0, 0) (see table 3.3).

3.3.5 Conclusions

The conclusions from simulation study D are summarized as follows.

- The performance of the household label test, as a tool for assessing the population mixing assumption, is excellent in all scenarios and rounds, for both infection and removal based assessment. The test, as desired, enjoys increased power as the dimension of the data (quantified by N) or the two-level-mixing effect (quantified by R_0^H) increases.

- Removal based assessment is slightly less powerful than infection based assessment, reflecting the fact that some information is lost when ordering the household labels according to removals times rather than infection times, but still very powerful to consistently provide evidence against H_0 in favour of H_L whenever appropriate. In the context of our assumed framework this is highly important, as only removal times are observed while infection times are missing.

- In assessing the population mixing assumption, the test clearly outperforms the distance and the position-time methods which could not systematically discard a homogeneously mixing model under similar simulation scenarios.

3.3.6 Remarks

In the context of a simulation study, where multiple datasets are considered, it is infeasible to plot the histogram for each one of the datasets for which the test is applied to. Therefore, for the purposes of simulation study D, the test was conducted by considering only the p-value and not the histogram. In practice though, where interest is in analyzing one dataset, the histogram should be plotted as it can provide additional information to that of the p-value (see discussion in section 3.2.3).

All datasets of simulation study D were simulated so that the number of individuals in each household was equal (see table 3.1). It must be noted though, that this condition was not imposed by the test and was only used to replicate the simulation conditions of simulation study C (see table 3.1) so that to allow direct comparisons between simulation studies C and D. That is to say, the household label test can be implemented for populations where the number of individuals in each household is different (see section 3.4 below for an example).

3.4 Application of the test to the Abakaliki smallpox data

3.4.1 Purpose

This section applies the newly derived household label test to a real dataset obtained from a smallpox outbreak in Abakaliki, Nigeria, in 1967 (Bailey, 1975, page 125). It is emphasized, that the intention here is not to conduct an extensive analysis

that provides new insights into the outbreak, but rather to illustrate the use of the test on a real-life example and assess its performance. The Abakaliki dataset provides a good platform to do so for two reasons. First, it contains information regarding the population structure which is necessary to apply the test. Second, it is a widely studied dataset, either analyzed to understand the outbreak (see e.g. Eichner and Dietz (2003); Stockdale et al. (2017)) or used to illustrate new data analysis methodology (see e.g. O'Neill and Roberts (1999); Boys and Giles (2007); Clancy and O'Neill (2008); Kypraios et al. (2017)), and can thus serve as a benchmark for assessing the practical utility of the test.

3.4.2 Data description

The outbreak and the data are described in detail in Thompson and Foege (1968) and Eichner and Dietz (2003). The total population of Abakaliki at the time of the outbreak was 31200 individuals and the total number of smallpox cases was 32. The collected data contained information (to a lesser or greater extent) on age, sex, vaccination status and membership status to a religious organization, for all individuals. In addition, for 251 individuals of the population, information was available on the compound which they belonged to; compound refers to one-storey dwellings built around a central courtyard, capable of housing several families, and indicate the group which individuals belong to (just like households did in the previous sections of this chapter). All 32 ever-infected individuals belonged to a compound and for each one of them the date of onset of rash (case detection) was recorded.

For the purposes of this analysis, a choice is made to ignore age, sex, vaccination and membership status information, as such information are not utilized by the test, and to restrict to the population of individuals that belong to a compound, so that the considered population is partitioned into compounds and the test can be implemented. Specifically, following the notation of section 3.2.1, a

population of size $C = 251$ is considered, where the individuals are partitioned into $l = 9$ compounds, labelled as $1, 2, \ldots, 9$, with each compound m consisting of C_m individuals, $m = 1, 2, \ldots, 9$, where $(C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8, C_9) = (33, 15, 10, 33, 22, 43, 20, 42, 33)$, so that $C = \sum_{m=1}^{l} C_m$. Observed event times are taken to be the $n = 32$ onset of rash (case detection) times and the corresponding (time-ordered according to the events) observed compound label vector is given by $\boldsymbol{g}^{\boldsymbol{e}^{obs}} = (1, 1, 1, 1, 1, 1, 1, 2, 2, 1, 4, 5, 1, 1, 1, 1, 5, 2, 1, 2, 6, 5, 2, 7, 4, 2, 2, 8, 3, 9, 5, 2)$ (see Thompson and Foege (1968, table 1)).

3.4.3 Run conditions

Following the procedure described in section 3.2.3 (see Algorithm 16), the test is applied to the Abakaliki dataset by drawing an independent sample of size 10000 from $T^{sam} \sim H_0$ and calculating the observed value of the test statistic, T^{obs}, and the corresponding p-value. In addition, the plausibility of the null hypothesis H_0 is tested visually, by imposing T^{obs} on the histogram of the sampled values from T^{sam}.

3.4.4 Results and conclusions

Figure 3.2 gives the output of the test when applied to the Abakaliki outbreak data. Having in mind how the test is interpreted (see section 3.2.3) the histogram and the extreme (small) p-value = 0.004 suggest that the data are inconsistent with H_0 (the hypothesis of homogeneous mixing) and provide strong evidence against it and in favour of H_L (the hypothesis of two-level mixing). This conclusion is desirable in the sense that it is in agreement with what is supported by the literature. For example, in both Stockdale et al. (2017) and Eichner and Dietz (2003), where the outbreak was thoroughly analyzed, taking into account all the information in the data, the reported estimates of the basic model parameters suggest a compound effect in the spread of the outbreak; with most infections occurring within rather than between compounds. This is also supported by the epidemiological investigation reported in

Thompson and Foege (1968), which concluded that the spread within compounds, and within families in particular, appeared to drive the epidemic.

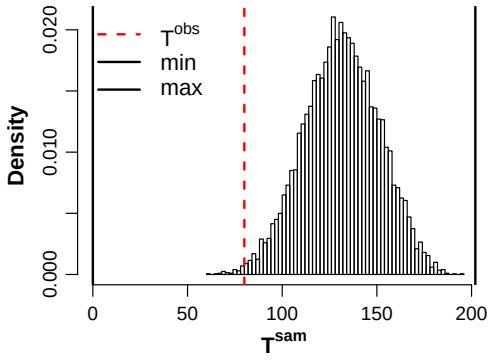


Figure 3.2: Application of the classical hypothesis test for compound label data on the Abakaliki outbreak data. The plot is the histogram of 10000 realizations from the sampling distribution of $T^{sam} \sim H_0$ with the observed value $T^{obs} = 80$ (red, dashed line), the minimum value of $T = 0$ (black, solid line) and the maximum value of $T = 202$ (black, solid line) imposed. The test p-value is p-value $= 0.004$.

3.5 Discussion

3.5.1 Addressing chapter aims

This chapter derived a novel hypothesis test based on household labels. The test is based on the idea that events of individuals belonging to the same household should occur closer in time rather than further apart, in the presence of a two-level-mixing effect. The key in constructing and implementing the test relies on the fact that, under the assumption of homogeneous mixing, the sampling distribution of the discrete random vector of household labels is known and independent of any model parameters. An attractive feature of the test is its ordinal interpretation; the lower the observed value of the test statistic and its corresponding p-value are, the more the evidence against the hypothesis of homogeneous mixing and in favour of the hypothesis of two-level-mixing.

A thorough simulation study demonstrated that the test performs desirably as a tool for assessing the population mixing assumption, for both infection and removal based assessment. The test, as desired, exhibits increased power as the dimension of the data or the two-level-mixing effect increases. Considering that the distance and the position-time methods cannot reliably be used as tools for assessing the population mixing assumption, the household label test is an especially welcome addition to the model assessment toolkit. The apparent edge of the household label test over the distance and the position-time methods is most likely accounted for by the fact that the latter methods do not make use of household label data which are evidently very informative in assessing the population mixing assumption.

When applied to a real dataset, the test again performed well, reaching a conclusion that was in agreement with previous analyses in the literature. Although the test can not (and is not designed to) provide extensive information on all aspects of an outbreak, the fact that it appears to effectively assess the population mixing assumption, by taking minimal computational time and by being straightforward to implement, is particularly appealing from a practical point of view.

3.5.2 Limitations

As discussed in section 2.10 of the previous chapter, although the examination of the performance of a method via simulation studies might be extensive it can never be exhaustive; there are always more simulation scenarios to be considered. For example, it would be interesting to investigate how noisy the infectious period distribution has to be, before all the two-level-mixing information is lost in the transition from infection to removal based assessment (see section 3.2.4); nonetheless, the Exp(γ) infectious period with $\gamma = 0.1$, used in simulation study D, has variance 100, which is larger than what would one typically encounter in practice. Also, it would be interesting

to extensively assess the performance of the test on data for which the number of individuals in each household was not equal; although in principle, and as seen in the Abakaliki data example, there do not appear to be any reasons to believe that the performance of the test would be affected if applied to such data.

3.5.3 General remarks

Although one can sample directly from $T^{sam} \sim H_0$ (see section 3.2.3), deriving an analytic expression for its distribution is rather challenging. More precisely, since T is defined as the sum of the household contributions $s^{(m)}_{\boldsymbol{g}^{\boldsymbol{e}}}$ (see section 3.2.2 and equation (3.2)), $m = 1, 2, \ldots, l$, where l the number of households, knowledge of the distribution of $T^{sam} \sim H_0$ prerequisites knowledge of the joint sampling distribution of the random vector $(s^{(1)}_{\boldsymbol{g}^{\boldsymbol{e}^{sam}}}, s^{(2)}_{\boldsymbol{g}^{\boldsymbol{e}^{sam}}}, \ldots, s^{(l)}_{\boldsymbol{g}^{\boldsymbol{e}^{sam}}}) \sim H_0$, which is complicated by the correlated nature of its components. The above complication could potentially be avoided by opting to pursue an asymptotically (i.e. for large data dimension) approximate, rather than an exact, result for the distribution of $T^{sam} \sim H_0$. For example, if the assumption of equal number of individuals in each household is made, the random vector $(s^{(1)}_{\boldsymbol{g}^{\boldsymbol{e}^{sam}}}, s^{(2)}_{\boldsymbol{g}^{\boldsymbol{e}^{sam}}}, \ldots, s^{(l)}_{\boldsymbol{g}^{\boldsymbol{e}^{sam}}}) \sim H_0$ becomes interchangeable (meaning that the joint distribution of the vector is invariant to any permutation of its components) and various central-limit-type of theorems, for the sum of interchangeable random variables (or processes), exist in the literature, such as Blum et al. (1958); Chernoff and Teicher (1958); Weber (1980). However, the idea of pursuing an asymptotic result was not entertained further as, besides satisfying mathematical curiosity, it would only incur additional challenges (e.g. none of the results in the aforementioned references are directly applicable to our setting and thus, before any such result could be used, one would probably need to modify and then satisfy any theorem conditions in order to accommodate our setting) whilst not having a practical contribution to the test implementation. More specifically, since direct sampling from $T^{sam} \sim H_0$ is possible (see section 3.2.3) and cheap to

perform (e.g. a sample size of 100000 takes less than a second of computer time to achieve) the (exact) distribution of $T^{sam} \sim H_0$, and thus the p-value of the test, can be approximated (via Monte Carlo) arbitrarily accurately (independently of the dimension of the data), by simply increasing the sample size. On the contrary, if the test was to be implemented using an asymptotically approximate distribution for $T^{sam} \sim H_0$, although the p-value calculation would be analytic and not numeric, the approximation, and the test p-value, would only be as accurate as the dimension of the data was large. Based on the above, pursuing an asymptotic approximation result would only be worthwhile in the case that direct sampling from $T^{sam} \sim H_0$ was not possible or too costly to perform; and neither of these apply in this case.

Notice that, in the test formulation (see section 3.2.3), the precise specification of an alternative hypothesis was avoided. The reason for not conforming to the traditional two-outcome hypothesis testing setting, was to prevent unnecessary loss of information in the interpretation of the p-value. More precisely, if a traditional null vs alternative hypothesis test is conducted, with the assumption of homogeneous mixing H_0 being the null hypothesis, and the assumption of two-level-mixing H_L being the alternative hypothesis, the ordinal nature of the test statistic T implies that the test is one-sided with the evidence against H_0 and in favour of H_L being considered substantial, in the instances that the observed value of T is smaller than what would be produced if H_0 was true, that is, when the p-value is small (close to 0). Notice though, that under such formulation, a large p-value (close to 1) is interpreted as merely not providing enough evidence against H_0 whilst it fails to convey any information as far as the plausibility of H_L. This interpretation is less informative than the one resulting when no single alternative hypothesis is specified, where a large p-value (close to 1) is providing evidence against both H_0 and H_L and in favour of the hypothesis H_R, that there is negative two-level-mixing effect in the data, which is the opposite direction hypothesis of H_L (see section 3.2.3). Similar points, as the aforementioned, were also made in Snijders (2001), where an example was used to

highlight the limitations of the traditional two-outcome decision test formulation in the case that a three-outcome decision is more natural.

Recall that throughout this chapter (see section 3.2.1) it was assumed that $C_m \geq 2$ for all $m = 1, 2, \ldots, l$. This is because the alternative case, that $C_m = 1$ for some household m, is far less interesting due to the fact that when $C_m = 1$, the label of household m can only appear zero or one times (i.e. $\nu_{\boldsymbol{g^e}}^{(m)} = 0$ or $\nu_{\boldsymbol{g^e}}^{(m)} = 1$), and measuring spread in such cases is not possible. Nonetheless, the definition of $s_{\boldsymbol{g^e}}^{(m)}$ (and therefore of T) can be extended for the case that $C_m = 1$ by making the only sensible assignment for $s_{\boldsymbol{g^e}}^{(m)}$, which is $s_{\boldsymbol{g^e}}^{(m)} = 0$.

Recall from the previous chapter (see section 2.9.1.7) that a possible way to assess the population mixing assumption is by looking at the posterior predictive distribution of the number (or proportion) of local infections. Although, as already mentioned, this is a rather natural way of conducing the assessment, it requires model fitting via MCMC methods (see Alharthi (2016)) and thus it is much more computationally expensive to implement, compared to the household label test.

It must be noted that classical tests, for assessing the population mixing assumption of epidemic models, already exist in the literature. Particular reference should be made to the tests in Britton (1997a); Britton (1997b); Britton (1997c); Britton (1997d). The general features of the tests in the aforementioned references are the following. First, the setting is the same as the one in this thesis, where the population is partitioned into groups (say households) and information regarding the household of each individual is available. Second, a certain epidemic model is considered that assumes global infection contacts, described by a parameter $\lambda > 0$, and, in addition, local (within household) infection contacts, described by a parameter $\delta > 0$. The model is formulated in such a way so that when $\delta = 0$, only global contacts occur and the model reduces to a homogeneously mixing model; for example, a model that

highly features in these references is the Exp-2L model (see section 1.3.5.7) in which case λ is the one-to-one global infection rate (denoted as β_G throughout this thesis) and δ is the one-to-one local infection rate (denoted as β_L throughout this thesis). Then a test of whether there is additional within household infectivity is conducted by testing the null hypothesis $H_0 : \delta = 0$ against the alternative hypothesis $H_A : \delta > 0$. A third feature of these tests is that they are likelihood ratio (LR) tests i.e. the test statistic is the LR test statistic $\frac{L(\delta)}{L(0)}$, where L is the model likelihood function. Since the LR depends on δ, testing uniformly over all values of δ is not possible and, instead, the alternative hypothesis is restricted to values of δ close to 0. Then, calculation of the observed value of the LR test statistic and derivation of its sampling distribution under H_0, are based on a Taylor approximation of the likelihood around $\delta = 0$, and, additionally, on the use of asymptotic results for the case that the number of households is large (see e.g. theorems 2.1 and 3.1 in Britton (1997a)). The advantage of these tests compared to the test developed in this chapter is that, arguably, they are theoretically more justifiable, in the sense that the former rely on the model likelihood (the test statistic is derived from the model) whereas the latter is motivated by intuition (the test statistic is motivated by the natural structure of household label data). On the other hand, these LR tests rely on approximations (numeric and asymptotic), whereas the test of this chapter does not.

3.5.4 Further work

The household label test could be modified accordingly to assess other assumptions of epidemic models. For example, it is sometimes of interest to understand the spatial component of an outbreak (see e.g. Jewell et al. (2009)) and so it could be required to assess if there is a spatial effect in the spread of an epidemic outbreak among a population of individuals or farms whose location is known; spatial effect refers to an outbreak being more likely to progress between individuals or farms of smaller rather than larger distances. In such a case, by letting H_0 to be the hypothesis of no spatial

effect, H_L the hypothesis of spatial effect and H_R the hypothesis of negative spatial effect (i.e. an outbreak being more likely to progress between individuals or farms of larger rather than smaller distances), a test for assessing the plausibility of H_0, against H_L or H_R, can be implemented and interpretated similarly to the household label test (see section 3.2.3), as follows. First, consider the time-ordered according to event times (individual or farm) labels $\boldsymbol{g^e} = (g_1^{\boldsymbol{e}}, g_2^{\boldsymbol{e}}, \ldots, g_n^{\boldsymbol{e}})$, where n the total number of (infection or removal) events. Second, define an ordinal statistic T, to quantify the spatial effect (so that the smaller the value of T the higher the spatial effect), such as $T(\boldsymbol{g^e}) = \sum_{k=1}^{n-1} d_{g_k^{\boldsymbol{e}}, g_{k+1}^{\boldsymbol{e}}}$, where $d_{g_k^{\boldsymbol{e}}, g_{k+1}^{\boldsymbol{e}}}$, $k = 1, 2 \ldots, n-1$, the distance between individuals or farms, $g_k^{\boldsymbol{e}}$ and $g_{k+1}^{\boldsymbol{e}}$, which experience the event consecutively. Finally, utilize the fact that, under the assumption of no spatial effect H_0, the sampling distribution of the random vector of labels $\boldsymbol{g^{e^{sam}}} = (g_1^{e^{sam}}, g_2^{e^{sam}}, \ldots, g_n^{e^{sam}}) \sim H_0$ is, as in section 3.2.2, known and independent of model parameters and described by choosing uniformly at random a permutation of n out of the total C individuals or farms in the population and recording their corresponding labels. The closer the p-value is to 0 (to 1) the more the evidence against H_0 and in favour of H_L (H_R), while p-values not too extreme (i.e. not too close to 0 or 1) do not provide enough evidence against H_0.

Chapter 4

Efficient Bayesian Inference for Partially Observed Stochastic Epidemic Models

4.1 Introduction

4.1.1 Chapter motivation and aims

In recent years, there has been a significant progress in the area of Markov chain Monte Carlo (MCMC) inference for stochastic epidemic models, fitted to temporal data (see section 1.4.2 and the references therein). However, as discussed in O'Neill (2010), challenges still remain, especially in high-dimensional settings where the computational burden is increased. More precisely, as explained in sections 1.3.5.2 and 1.3.5.3, in order to conduct MCMC inference for the interesting parameters of an epidemic model (i.e. the parameters that carry epidemiological interpretation such as those controlling the infection and the removal processes), one usually needs to introduce the unobserved infection data as additional unknown variables and target the joint posterior distribution of model parameters and augmented infection variables. The challenge in the implementation of such MCMC schemes

is not in updating the components that consist of model parameters, as such components are typically low-dimensional and possess no real difficulties (see e.g. sections 1.3.5.5 to 1.3.5.7 and the MCMC algorithms therein), but in updating the infection component, which is typically of much higher dimension (especially in cases of large-scale outbreaks) and is far harder to update efficiently; the high dimension of the infection space makes it very challenging for an MCMC sampler to move around it efficiently. In turn, the intrinsic dependence between infections and model parameters, typically encountered in epidemic models, implies that any mixing issues relating to the infection component will most likely be inherited by the model parameters of interest, thus affecting the practical utility of the inference. For example, it is typical that the full conditional distributions of components consisting of the model parameters depend heavily on the infections (see e.g. equations (1.21) and (1.24)) and therefore, if the infection component mixes very slowly the model parameter components will mix very slowly as well. Various MCMC algorithms have been employed to address these issues, making use of and sometimes combining different ideas, such as parameter reduction and non-centered parameterizations (see section 1.4.2 and the references therein). However, one thing that has remained the same among most of these algorithms is the fundamental idea according to which infections are proposed to be updated. More specifically, based on the fact that epidemic models typically assume that the infectious periods of individuals are i.i.d. from a distribution $D(\boldsymbol{\phi})$, with parameter $\boldsymbol{\phi}$ (see section 1.3.5.1), most of the currently existing MCMC algorithms (see e.g. O'Neill and Roberts (1999); O'Neill and Becker (2001); Neal and Roberts (2004, 2005); Kypraios (2007); Xiang and Neal (2014)), directly or indirectly, propose infections according to a model-driven independent proposal distribution that proposes a candidate infection time for individual k, say i_k^*, by proposing an infectious period $r_k - i_k^* \sim D(\boldsymbol{\phi}^{(s+1)})$, where r_k is the removal time of individual k and $\boldsymbol{\phi}^{(s+1)}$ is the current value of $\boldsymbol{\phi}$ in the MCMC algorithm. Although some of these MCMC algorithms have managed to mitigate the effect of the problem, by for example attempting to update more than one infections at a time

in a block update step (see Xiang and Neal (2014)), the mixing issues of the infection component still persist and more efficient algorithms are needed. The aim of this chapter, is to develop more efficient MCMC algorithms by introducing alternative proposal mechanisms for the infection component.

Comparing the performance of different MCMC algorithms is an integral part of this chapter and therefore, before proceeding any further, it is important to clarify how algorithm performance is assessed. As mentioned in the paragraph above, the focus throughout this chapter is placed on the update step of the infection component and thus the performance of an MCMC algorithm is assessed by the level of efficiency with which the infection step is performed, i.e. by the quality of mixing of the infection component. Note that, in general, different MCMC algorithms are likely to be more efficient for different components of a posterior vector (that is, a random vector having the posterior distribution). However, in the present context, an optimally mixing algorithm with respect to the infection component will most likely be optimal for the other components as well due to the intrinsic dependence between infections and model parameters (see the previous paragraph). For this reason, throughout this chapter, measures of mixing and efficiency are only considered for the infection component and not the other components.

It is noted that all MCMC algorithms used in this chapter were checked for evidence of non-stationarity (see the part regarding stationarity in section 1.3.2.3) by visually inspecting MCMC trace plots. Utilizing that the algorithms were applied on simulated data, stationarity was also checked by assessing whether the posterior densities of the algorithms appeared to be around the true values of the parameters. Trace plots and posterior density plots are provided in the simulation studies for illustration. Regarding initial values, again utilizing that the algorithms were applied on simulated data, the approach taken throughout this chapter was to initiate all chains at states that were far from the true values in order to better assess whether convergence

appeared to be achieved. All results reported in this chapter are based on chains that appear to have converged.

This chapter makes use of the notation and terminology introduced in sections 1.3.5.1 to 1.3.5.3. Since the focus is placed on the infection component, it is helpful to recall that it is denoted as $(\alpha, i_\alpha, \boldsymbol{i})$, where α is the label of the initial infective, i_α is its corresponding infection time and $\boldsymbol{i} = (i_1, i_2, \ldots, i_n)\backslash\{i_\alpha\}$ are all the remaining infection times, except i_α. The notation $(\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$, where $\boldsymbol{i}^{(s)} = (i_1^{(s)}, i_2^{(s)}, \ldots, i_n^{(s)})\backslash\{i_\alpha^{(s)}\}$, will be used to denote the current value of the infection component at MCMC iteration s, while the notation $(\alpha^*, i_\alpha^*, \boldsymbol{i}^*)$, where $\boldsymbol{i}^* = (i_1^*, i_2^*, \ldots, i_n^*)\backslash\{i_\alpha^*\}$, will be used to denote candidate values. Note that i_k^*, $k = 1, 2, \ldots, n$, could be different to $i_k^{(s)}$ or the same, depending on whether an infection time for individual k is proposed or not, respectively. Similar notation will be used for terms that depend on the infection component. For example, the current (at iteration s) values of the terms $A = \int_{i_\alpha}^{r_n} X_t Y_t dt$ and $B = \sum_{k=1}^{n} (r_k - i_k)$ (see section 1.3.5.5), will be denoted as $A^{(s)}$ and $B^{(s)}$, and their candidate values as A^* and B^*, respectively.

All runs and plots in this chapter are produced using the statistical programming language R Core Team (2019).

4.1.2 Chapter layout

The remainder of this chapter is divided into three parts, two main parts and the last part containing conclusions.

The first part, section 4.2, considers MCMC algorithms based on updating one infection time at each update step, in what is referred to as a 1-dimensional update step.

The second part, section 4.3, proceeds to consider MCMC algorithms based on updating multiple infection times at each update step, in a so-called block update step.

The layout is similar in both main parts; initially, the limitations of existing algorithms are acknowledged, then, new algorithms are developed, and finally, the performance of the new algorithms is compared against the existing ones via simulation studies.

The last part of the chapter, section 4.4, highlights the main accomplishments of the chapter, gives the limitations and discusses general remarks and further work.

4.2 1-dimensional update steps for the infection component

Ultimately, the intention is to develop MCMC algorithms that update many infection times at a time, in a block update step, rather than one at a time, in a 1-dimensional update step, as a block update step will most likely be more efficient than its 1-dimensional counterpart (see e.g. Xiang and Neal (2014)). However, it is sensible to start by considering the simpler problem of updating the infection component in a 1-dimensional step, both because this is interesting in its own right and because this allows for some of the ideas of this chapter to be introduced and illustrated in a simpler setting.

Consider the setting of section 1.3.5.3, where the infection component is updated according to its full conditional density $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \boldsymbol{\beta}^{(s+1)}, \boldsymbol{\phi}^{(s+1)})$, with $\boldsymbol{\beta}^{(s+1)}$ and $\boldsymbol{\phi}^{(s+1)}$ being the current values of $\boldsymbol{\beta}$ and $\boldsymbol{\phi}$ in the MCMC algorithm. As explained

in section 1.3.5.3 (see the second remark following Algorithm 4), having chosen the individual k whose infection time i_k is to be updated, all the remaining infection times remain fixed at their current values during the update step and therefore the target density in a 1-dimensional update step of an infection time i_k is

$$\pi(\alpha, i_\alpha, i_k \mid \boldsymbol{r}, \boldsymbol{\beta}^{(s+1)}, \boldsymbol{\phi}^{(s+1)}, \boldsymbol{i}^{(s)}_{[-k]}), \tag{4.1}$$

where, if $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ is a vector with n entries, $\boldsymbol{x}_{[-k]}$ denotes the vector containing all entries of $\boldsymbol{x}$ except x_k, $k \in \{1, 2, \ldots, n\}$.

4.2.1 Existing 1-dimensional update steps and their limitations: standard 1-dimensional MCMC algorithms

As part of the process of developing new updating schemes, it is useful to acknowledge the limitations of the existing ones. By doing so, the development of new schemes can be guided accordingly. Recall from section 1.3.5.3, that the standard existing method of updating the infection component, in a 1-dimensional step, is by using a MH step and a model-driven independent proposal distribution, that proposes a candidate infection time for individual k, say i_k^*, by proposing an infectious period $r_k - i_k^* \sim D(\boldsymbol{\phi}^{(s+1)})$, where $D(\boldsymbol{\phi})$ is the infectious period distribution of the assumed model, $\boldsymbol{\phi}$ is the parameter of $D(\boldsymbol{\phi})$, and $\boldsymbol{\phi}^{(s+1)}$ is the current value of $\boldsymbol{\phi}$ in the MCMC algorithm (see Algorithm 4). To simplify wording, this proposal distribution will henceforth be referred to as the *standard 1-dimensional (standard-1d) proposal* and MCMC algorithms that use the standard-1d proposal to update the infection component will be referred to as *standard-1d MCMC algorithms*.

To reveal settings where the above updating scheme might underperform, the Exp-HM model is fitted using the standard-1d MCMC algorithm (Algorithm 5) to three datasets, generated from the Exp-HM model itself. Recall that (see section

1.3.5.5), for the Exp-HM model, $\boldsymbol{\beta} = \beta$, $\boldsymbol{\phi} = \gamma$ and $D(\boldsymbol{\phi}) = \text{Exp}(\gamma)$, and thus the target density, in a 1-dimensional update step of an infection time i_k, is $\pi(\alpha, i_\alpha, i_k \mid \boldsymbol{r}, \beta^{(s+1)}, \gamma^{(s+1)}, \boldsymbol{i}^{(s)}_{[-k]})$ and the standard-1d proposal distribution is $D(\boldsymbol{\phi}^{(s+1)}) = \text{Exp}(\gamma^{(s+1)})$. The three datasets are generated by setting the number of initial susceptibles, N, to be 200 and the basic reproduction number parameter, R_0, to be 1.5, 2.5 and 5, and the resulting number of total infections, n, is 131, 179 and 200, respectively. The prior distribution assignment is done as in section 1.3.5.5 and the prior parameters are specified so that the uncertainty for all model parameters (except for the label of the initial infective α which is assigned a prior distribution as $\alpha \sim \text{U}[1 : n]$) is expressed via uninformative $\text{Exp}(10^{-3})$ prior distributions. The algorithm is run for 20000 iterations, after a burn-in of 5000, by repeating the infection component update step as many times as the number of infections so that, in each MCMC iteration, all infection times are attempted to be updated (see last paragraph of section 1.3.5.3).

Figure 4.1 shows the acceptance proportion for the update step of the infection time i_k, for each ever-infected individual k, $k = 1, 2, \ldots, n$. Note that, as in section 1.3.5.2, the n ever-infected individuals are labelled according to the time-ordered removal times $r_1 < r_2 < \cdots < r_n$, so that individual with label 1 is removed first, individual with label 2 is removed second and so on, i.e. the acceptance proportions are plotted against individual labels corresponding to time-ordered removal times. Also notice, that imposed on the plots is the effective sample size (see the part about mixing in section 1.3.2.3) over the actual sample size and the proportion of times that proposed values were inadmissible (meaning that the proposed configuration of infection times would cause the epidemic to cease before r_n; such values are automatically rejected), relating to the update step of each i_k, $k = 1, 2, \ldots, n$. Looking at figure 4.1 one can notice a 'curvature effect' on the acceptance proportions, with individuals being removed closer to first or last, generally having lower acceptance proportions than individuals having order of removal closer to the median order. The effective sample

size proportions follow along this pattern, which is in line with what is known for independent proposals, namely that the higher the acceptance proportion the more efficient an independent proposal will be (see the part regarding independent proposals in section 1.3.2.4). It is also worth noticing that, the inadmissibility proportions are more or less similar across individuals, revealing that the reason behind the curvature effect of the acceptance proportions is not because proposed values for individuals being removed closer to first or last are more often inadmissible.

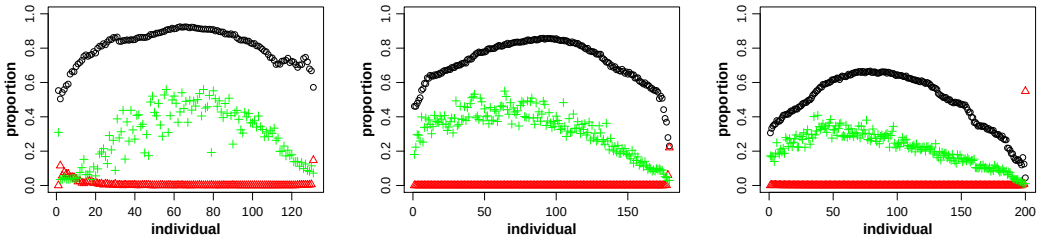


Figure 4.1: Acceptance proportion (black circles), effective sample size over actual sample size (green pluses) and inadmissibility proportion (red triangles) for the 1-dimensional update step of the infection time i_k, of the Exp-HM model, using the standard-1d proposal, against individual label k, $k = 1, 2, \ldots, n$. Columns correspond to three different datasets, generated from an Exp-HM model ($N = 200$, $\gamma = 0.1$), where R_0 (left to right) is set at 1.5, 2.5 and 5, respectively.

To further investigate this curvature effect, the target density of the update step of i_k, as a function of i_k, is plotted for $k = 1$, $k = \lfloor n/2 \rfloor$ and $k = n$, with all remaining variables being fixed at their true (rather than their current) values, and the standard-1d proposal density (using as parameter the true value of γ rather than its current value) is imposed, for all three datasets (figure 4.2). Guidance on how to interpret the plots of figure 4.2 can be gauged by recalling that for an independent proposal density to work well it should resemble the target density (see the part regarding independent proposals in section 1.3.2.4). Looking across columns (values of k), in figure 4.2, one can see that the proposal density is much more similar to the target density for $k = \lfloor n/2 \rfloor$, compared to $k = 1$ or $k = n$, which helps explain the curvature effect on the acceptance (and effective sample size) proportions, exhibited

in figure 4.1. More precisely, it appears that, for all three datasets, the proposal density overestimates the length of the infectious period of individual with label 1 and underestimates the length of the infectious period of individual with label n, both with reference to the target density. Notice also that (by looking across rows in figure 4.2), this pattern becomes more evident as the value of R_0 becomes larger. To appreciate this, consider an extreme case where R_0 is very large so that all infections occur almost instantaneously immediately after the first infection and before the first removal $r_1 = 0$, around a time point, say t_I. In such a case, the target density of the update step of i_k, for all individuals $k = 1, 2, \ldots, n$, irrespectively of their removal time r_k, would put most of its mass around t_I; this is because the target density of the update step of i_k is conditioned on the values of the remaining infection times, which would all occur around t_I (i.e. the target density would 'know' that all remaining infection times occur around t_I), and would therefore suggest that i_k should occur around t_I too. Note that, precisely because individuals are labelled according to their time-ordered removal times $r_1 < r_2 < \cdots < r_n$, the smaller (larger) the label of individual k, $k = 1, 2, \ldots, n$, the closer (further) from t_I, its removal time r_k would be. That is to say that, according to the target density of i_k, the smaller (larger) the label of individual k, $k = 1, 2, \ldots, n$, the shorter (longer) its infectious period $r_k - i_k$ would generally be. However, the standard-1d proposal scheme generates candidate infectious periods $r_k - i_k^*$, from the same distribution, for all individuals $k = 1, 2, \ldots, n$ (in this case $r_k - i_k^* \overset{\text{i.i.d.}}{\sim} D(\boldsymbol{\phi}^{(s+1)}) = \text{Exp}(\gamma^{(s+1)})$, $k = 1, 2, \ldots, n$), so in this case it would end up overestimating the length of the infectious period of individuals with labels close to 1, and underestimating the length of the infectious period of individuals with labels close to n.

A similar investigation as the above is also conducted using the Gamma-HM model, leading to very similar conclusions. More specifically, the Gamma-HM model is fitted (under the same run conditions as the Exp-HM model; using the same prior assignment and running the chain for 20000 iterations, after a burn-in of 5000, by

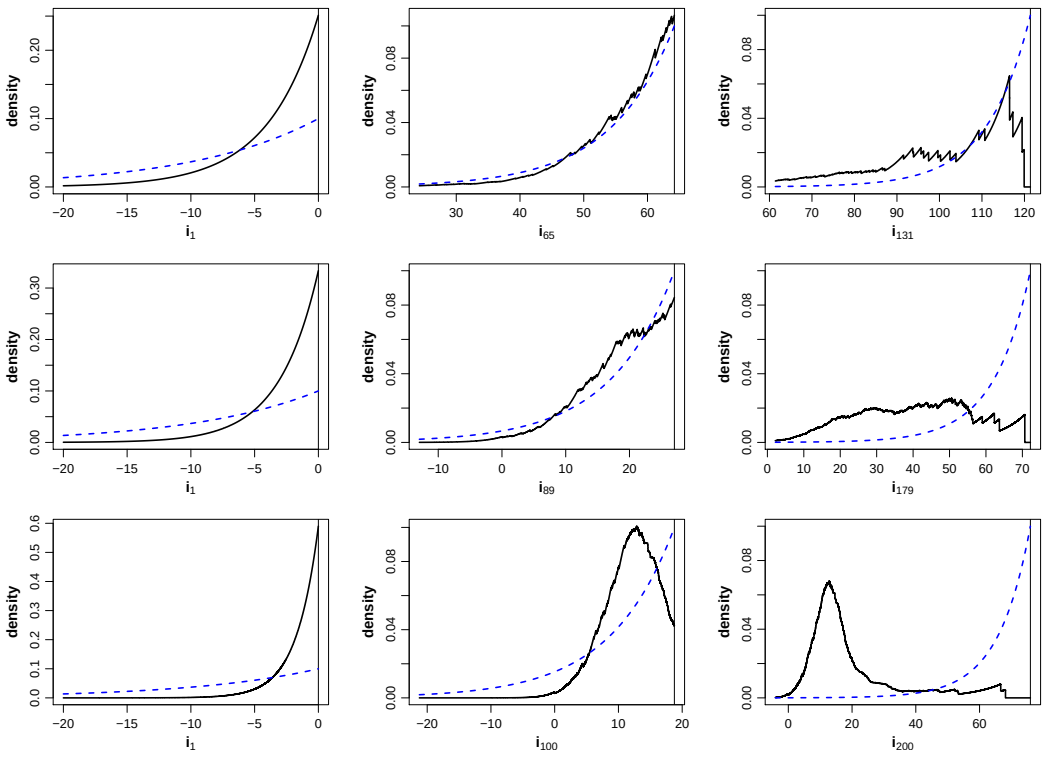


Figure 4.2: Target density (black, solid line) and standard-1d proposal density (blue, dashed line) for the 1-dimensional update step of the infection time i_k, of individual label k, of the Exp-HM model. Imposed (vertical, black, solid line) is the observed removal time r_k (maximum value of i_k). Rows correspond to three different datasets, generated from an Exp-HM model ($N = 200$, $\gamma = 0.1$), where R_0 (top to bottom) is set at 1.5, 2.5 and 5, respectively. Columns (left to right) correspond to k values of 1, $\lfloor n/2 \rfloor$ and n, respectively.

repeating the infection component update step as many times as the number of infections) using the standard-1d MCMC algorithm (Algorithm 6) to two datasets generated from the model itself, by setting $N = 200$ and R_0 to be 1.5 and 2.5, respectively. Recall that (see section 1.3.5.5), for the Gamma-HM model, $\boldsymbol{\beta} = \beta$, $\boldsymbol{\phi} = (\nu, \lambda)$ (but reduces to $\boldsymbol{\phi} = \lambda$ since ν is assumed known) and $D(\boldsymbol{\phi}) = \text{Gamma}(\nu, \lambda)$, and thus the target density, in a 1-dimensional update step of an infection time i_k, is $\pi(\alpha, i_\alpha, i_k \mid \boldsymbol{r}, \nu, \beta^{(s+1)}, \lambda^{(s+1)}, \boldsymbol{i}^{(s)}_{[-k]})$ and the standard-1d proposal distribution is $D(\boldsymbol{\phi}^{(s+1)}) = \text{Gamma}(\nu, \lambda^{(s+1)})$. Figure 4.3 shows the acceptance, effective sample size and inadmissibility proportions for the update step of the infection time i_k, for each

ever-infected individual k, $k = 1, 2, \ldots, n$. Just like for the Exp-HM model (see figure 4.1), one can notice a curvature effect on the acceptance (and effective sample size) proportions. Figure 4.4 plots the target density of the update step of i_k, as a function of i_k, for individuals $k = 1$, $k = \lfloor n/2 \rfloor$ and $k = n$, with all remaining variables being fixed at their true (rather than their current) values, and imposes the standard-1d proposal density (using as parameter the true value of λ rather than its current). Again, similar to the Exp-HM model case (see figure 4.2), one can see, especially as R_0 gets larger, that the proposal density proposes quite accurately the infectious periods of individuals with labels close to $\lfloor n/2 \rfloor$, but tends to overestimate the length of the infectious period of individuals with labels close to 1 and to underestimate the length of the infectious period of individuals with labels close to n; a pattern that helps explain the curvature effect exhibited in figure 4.3.

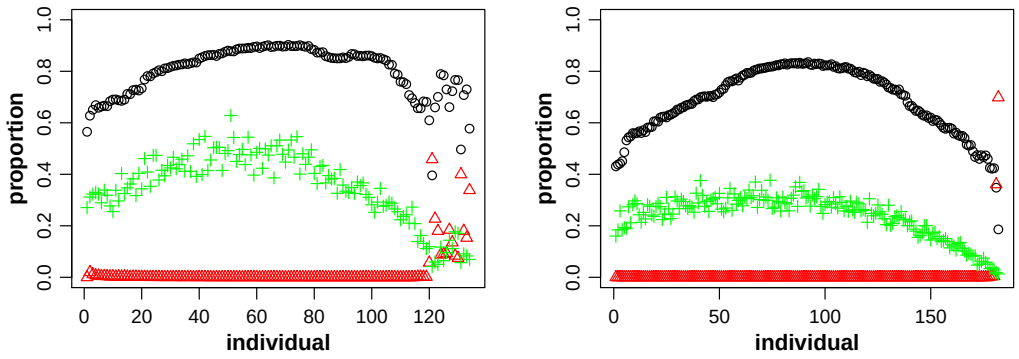


Figure 4.3: Acceptance proportion (black circles), effective sample size over actual sample size (green pluses) and inadmissibility proportion (red triangles) for the 1-dimensional update step of the infection time i_k, of the Gamma-HM model, using the standard-1d proposal, against individual label k, $k = 1, 2, \ldots, n$. Columns correspond to two different datasets, generated from a Gamma-HM model ($N = 200$, $\nu = 2$, $\lambda = 0.2$), where R_0 (left to right) is set at 1.5 and 2.5, respectively.

4.2.2 Individual-specific 1-dimensional MCMC algorithms

The above investigations, illustrate that (the performance of) the standard-1d proposal for updating an infection time i_k is not homogeneous across all individuals,

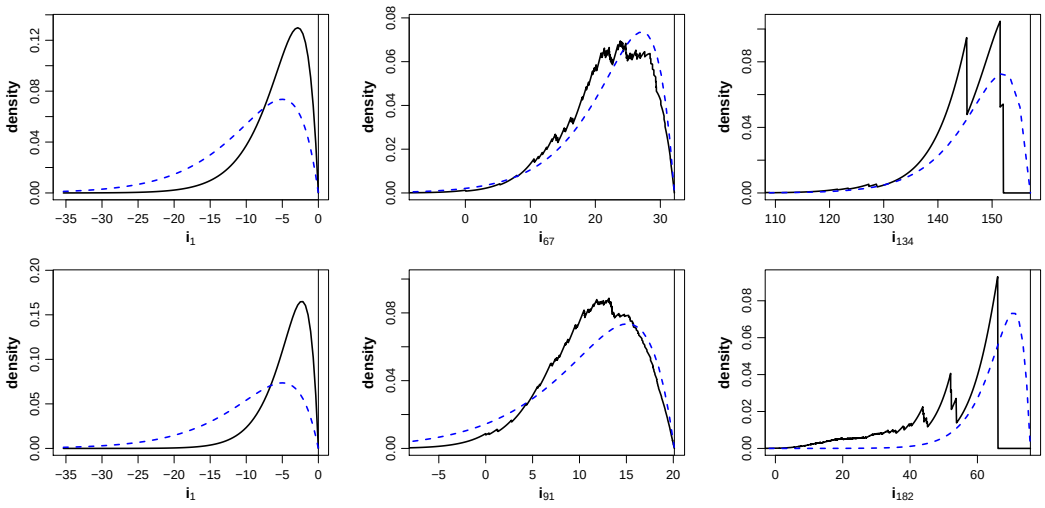


Figure 4.4: Target density (black, solid line) and standard-1d proposal density (blue, dashed line) for the 1-dimensional update step of the infection time i_k, of individual label k, of the Gamma-HM model. Imposed (vertical, black, solid line) is the observed removal time r_k (maximum value of i_k). Rows correspond to two different datasets, generated from a Gamma-HM model ($N = 200$, $\nu = 2$, $\lambda = 0.2$), where R_0 (top to bottom) is set at 1.5 and 2.5, respectively. Columns (left to right) correspond to k values of 1, $\lfloor n/2 \rfloor$ and n, respectively.

$k = 1, 2, \ldots, n$, and there is a pattern according to the order which individuals are removed. This pattern, suggests that, instead of using a proposal distribution that is common for all individuals $k = 1, 2, \ldots, n$ (as is the standard-1d proposal), it might be preferable to consider proposal distributions that are specific to each individual k, $k = 1, 2, \ldots, n$. More precisely, the following alternative 1-dimensional update scheme, for the infection component $(\alpha, i_\alpha, \boldsymbol{i})$, is considered. First, choose one of the n ever-infected individuals, say k, according to a discrete uniform distribution on $\{1, 2, \ldots, n\}$, i.e. as $k \sim \mathrm{U}[1 : n]$. Then, propose a candidate infection time for individual k, say i_k^*, using a MH step and an independent proposal distribution, as $r_k - i_k^* \sim D(\boldsymbol{\phi}_k)$, where $D(\boldsymbol{\phi})$ is the infectious period distribution of the assumed model, $\boldsymbol{\phi}$ is the parameter of $D(\boldsymbol{\phi})$, and $\boldsymbol{\phi}_k$ is a suitably specified parameter, specific to individual k (the specification of $\boldsymbol{\phi}_k$ is discussed below). Owing to their nature, these proposal distributions are referred to as *individual-specific 1-dimensional (IS-1d) proposals* and MCMC algorithms that use the IS-1d proposals to update the

infection component will be referred to as *IS-1d MCMC algorithms*.

Note that, the above defined IS-1d proposal scheme is the same as the standard-1d proposal scheme (mentioned in the first paragraph of section 4.2.1 and fully described in section 1.3.5.3 and Algorithm 4) but with one fundamental difference. Specifically, although the family of the proposal distribution of i_k is the same for both schemes, and is $D(\boldsymbol{\phi})$, the parameter of the proposal distribution is different, as for the standard-1d proposal scheme it is the current value of $\boldsymbol{\phi}$, say $\boldsymbol{\phi}^{(s+1)}$, common for all individuals $k = 1, 2, \ldots, n$, whereas for the IS-1d proposal scheme it is the individual-specific parameter $\boldsymbol{\phi}_k$, generally different for each individual k, $k = 1, 2, \ldots, n$.

To specify the individual-specific parameters $\boldsymbol{\phi}_k$, $k = 1, 2, \ldots, n$, a practical approach is taken, where the $\boldsymbol{\phi}_k$'s are treated as tuning parameters (see the part regarding dependent and independent proposals in section 1.3.2.4) and are specified using the burn-in iterations. The intention for specifying the $\boldsymbol{\phi}_k$'s in such a way is to allow for any information about the target posterior distribution, provided by the initial burn-in iterations, to be incorporated into the proposal distributions. In this way, the independent IS-1d proposal distributions for i_k can be made more similar to their associated target distributions of i_k, which is what is desired from a good performing independent proposal distribution (see the part regarding independent proposals in section 1.3.2.4). Specifically, the $\boldsymbol{\phi}_k$'s are specified as follows. The first, say S_B, chain iterations, corresponding to the burn-in period, are run using an already existing proposal scheme, such as the standard-1d scheme. Having obtained a sample of size S_B for each i_k, say $\{i_k^{(1_B)}, i_k^{(2_B)}, \ldots, i_k^{(S_B)}\}$, the IS-1d proposal distribution, $D(\boldsymbol{\phi}_k)$, is fitted to the sampled infectious periods of individual k, $\{r_k - i_k^{(1_B)}, r_k - i_k^{(2_B)}, \ldots, r_k - i_k^{(S_B)}\}$, and a method of moments (MOM) estimation is used to specify the parameter $\boldsymbol{\phi}_k$, $k = 1, 2, \ldots, n$. In the case that the assumed infectious period distribution is Exp(γ), the IS-1d proposal for individual k is $D(\boldsymbol{\phi}_k) = \text{Exp}(\gamma_k)$ and the MOM estimation specifies γ_k as $\gamma_k = 1/\bar{x}_k$, where $\bar{x}_k = \frac{1}{S_B}\sum_{s=1}^{S_B}(r_k - i_k^{(s_B)})$, $k = 1, 2, \ldots, n$.

In the case that the assumed infectious period distribution is Gamma(ν, λ), the IS-1d proposal for individual k is $D(\boldsymbol{\phi}_k) = \text{Gamma}(\nu_k, \lambda_k)$. In this latter case, one may conduct MOM estimation for both ν_k and λ_k, yielding $\nu_k = \frac{\bar{x}_k^2}{s_k^2}$ and $\lambda_k = \frac{\bar{x}_k}{s_k^2}$, where $s_k^2 = \frac{1}{S_B - 1} \sum_{s=1}^{S_B} (r_k - i_k^{(s_B)} - \bar{x}_k)$, or alternatively, since ν is assumed to be known for inferences purposes (see section 1.3.5.4), set $\nu_k = \nu$ for all $k = 1, 2, \ldots, n$ and conduct MOM estimation only for λ_k, yielding $\lambda_k = \frac{\nu}{\bar{x}_k}$, $k = 1, 2, \ldots, n$.

Recall that (see the part regarding proposal distributions in section 1.3.2.3) a proposal distribution can be specified arbitrarily, as long as the required ergodic properties are satisfied; in the present context this is ensured, since the $\boldsymbol{\phi}_k$'s are specified at the end of the burn-in iterations and remain fixed henceforth, thus not altering the ergodic properties of the chain. Nonetheless, whether or not a proposal specification is a good one is ultimately determined by algorithm efficiency. To develop a visual appreciation of how the IS-1d proposal scheme might perform, the simulated datasets of section 4.2.1 are again considered. First, the three datasets generated from the Exp-HM model are considered. Using the $S_B = 5000$ burn-in iterations from the standard-1d MCMC run (i.e. the MCMC run using Algorithm 5) each parameter γ_k, of the IS-1d proposal distribution Exp(γ_k), $k = 1, 2, \ldots, n$, is specified as described in the paragraph above. Figure 4.5 is a repetition of figure 4.2 (discussed earlier in section 4.2.1), with the addition that, besides the standard-1d proposal density, the IS-1d proposal density is also imposed on the target density of i_k. The plots show that the IS-1d proposal density, more or less, corrects for the overestimation (underestimation) of the length of the infectious period, exhibited by the standard-1d proposal density, for individuals with labels close to 1 (close to n). This happens precisely because the IS-1d proposals are allowed to have a different parameter for each individual k, and because these parameters are specified according to the information about the target posterior posterior distribution, provided by the initial burn-in iterations. Therefore, they have the ability to capture the pattern sometimes (see the above paragraph) exhibited in the target distribution, where the smaller (larger) the label of individual

k, $k = 1, 2, \ldots, n$, the shorter (longer) its infectious period $r_k - i_k$ typically is. A further appreciation of this ability, is given by figure 4.6, where the mean infectious period according to the IS-1d proposal distribution, given by $1/\gamma_k$, is plotted for each individual k, $k = 1, 2, \ldots, n$. Finally, a thing that is worth noticing in figure 4.5 is that for individuals with label 1, the IS-1d proposal density appears to be optimal, since it is almost the same as the target density, but for individuals with label n, although it still improves upon the standard-1d density, it is not that similar to the target density, particularly as R_0 gets larger.

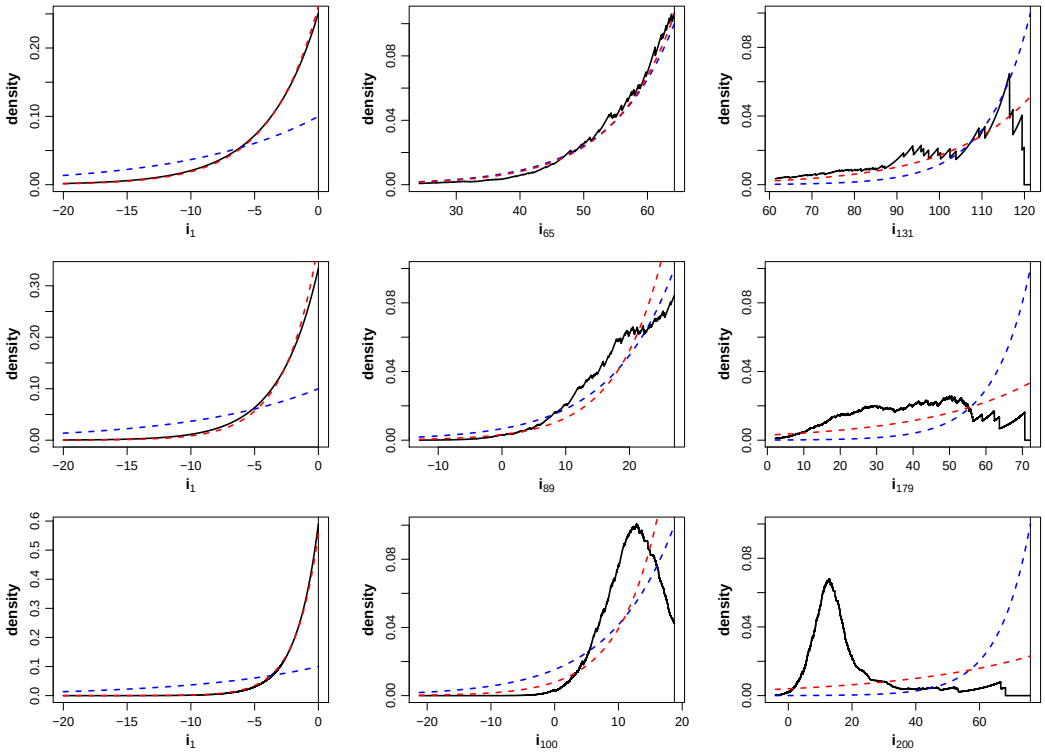


Figure 4.5: Target density (black, solid line), standard-1d proposal density (blue, dashed line) and IS-1d proposal density (red, dashed line) for the 1-dimensional update step of the infection time i_k, of individual label k, of the Exp-HM model. Imposed (vertical, black, solid line) is the observed removal time r_k (maximum value of i_k). Rows correspond to three different datasets, generated from an Exp-HM model ($N = 200$, $\gamma = 0.1$), where R_0 (top to bottom) is set at 1.5, 2.5 and 5, respectively. Columns (left to right) correspond to k values of 1, $\lfloor n/2 \rfloor$ and n, respectively.

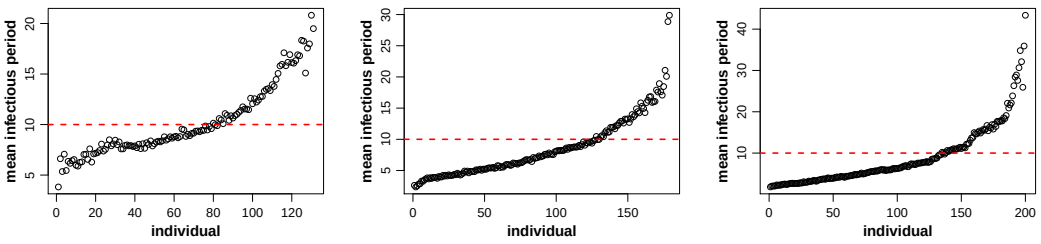


Figure 4.6: Mean infectious period $1/\gamma_k$, according to the IS-1d proposal distribution $\text{Exp}(\gamma_k)$, against individual label k, $k = 1, 2, \ldots, n$. Imposed (horizontal, red, dashed line) is the true infectious period. Columns correspond to three different datasets, generated from an Exp-HM model ($N = 200$, $\gamma = 0.1$), where R_0 (left to right) is set at 1.5, 2.5 and 5, respectively.

As can be seen from figures 4.7 and 4.8, similar observations can be made for the two datasets generated from the Gamma-HM model. It is noted that these plots are produced by estimating both of the parameters, ν_k and λ_k, of the IS-1d proposal distribution $\text{Gamma}(\nu_k, \lambda_k)$, $k = 1, 2, \ldots, n$, and not only λ_k (see earlier in this section on how ν_k and λ_k are estimated). Note also, that the $S_B = 5000$ burn-in iterations, according to which the parameters are calculated, are taken from the standard-1d MCMC run (i.e. the MCMC run using Algorithm 6).

These observations suggest that it might be more efficient to conduct the 1-dimensional update step of the infection component using the IS-1d proposals instead of the standard-1d proposals i.e that IS-1d MCMC algorithms might be more efficient than standard-1d MCMC algorithms. To further examine the validity of this speculation, simulation studies are conducted. Since both proposal schemes are tightly related to the infectious period distribution of the model in question (in the sense that the family of the proposal distributions is the same as the infectious period distribution assumed by the model; see section 4.2.1 and the earlier parts of this section), two simulation studies are conducted, referred to as simulation study E and simulation study F, one for the case that the infectious period is assumed to be $\text{Exp}(\gamma)$, and one for the case that it is assumed to be $\text{Gamma}(\nu, \lambda)$, respectively. Recall that

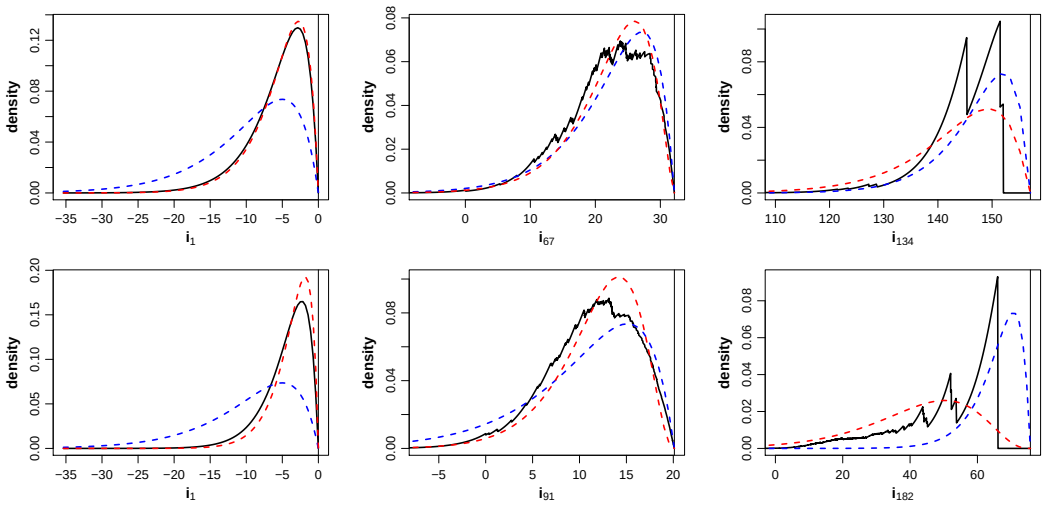


Figure 4.7: Target density (black, solid line), standard-1d proposal density (blue, dashed line) and IS-1d proposal density (red, dashed line) for the 1-dimensional update step of the infection time i_k, of individual label k, of the Gamma-HM model. Imposed (vertical, black, solid line) is the observed removal time r_k (maximum value of i_k). Rows correspond to two different datasets, generated from a Gamma-HM model ($N = 200$, $\nu = 2$, $\lambda = 0.2$), where R_0 (top to bottom) is set at 1.5 and 2.5, respectively. Columns (left to right) correspond to k values of 1, $\lfloor n/2 \rfloor$ and n, respectively.

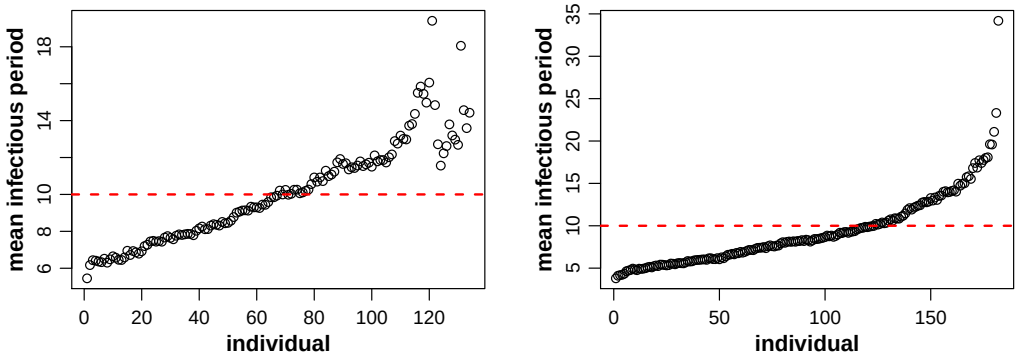


Figure 4.8: Mean infectious period ν_k/λ_k, according to the IS-1d proposal distribution Gamma(ν_k, λ_k), against individual label k, $k = 1, 2, \ldots, n$. Imposed (horizontal, red, dashed line) is the true infectious period. Columns correspond to two different datasets, generated from a Gamma-HM model ($N = 200$, $\nu = 2$, $\lambda = 0.2$), where R_0 (left to right) is set at 1.5 and 2.5, respectively.

for the other choice of infectious period distribution considered in this thesis, namely the constant distribution, no infection step is needed since the infection times are

automatically updated at the update step of the constant value of the infectious period (see the part about Bayesian inference and MCMC algorithm for the Constant-HM model in section 1.3.5.5).

4.2.3 Simulation study E

4.2.3.1 Purpose

The purpose of simulation study E is to compare the performance of the standard-1d and the IS-1d MCMC algorithms, for the case that the infectious period distribution is assumed to be Exp(γ). To perform this comparison, the relevant version of the widely used standard SIR model (see section 1.3.5.5) is considered, namely the Exp-HM model. Both from a methodological (see section 4.1.1) and a practical standpoint, interest is mostly focused on large-scale outbreaks and so it is important to investigate how the comparative performance of the two algorithms changes as the dimension of the data increases. The dimension of the data is quantified by N. Also of interest is to investigate the effect that R_0 might have on the comparison, since as seen in sections 4.2.1 and 4.2.2 above, the value of R_0 may affect algorithm efficiency.

4.2.3.2 Simulation conditions

To address the tasks of the simulation study, datasets are simulated from the Exp-HM model under all combinations of selected values of the parameters R_0 and N. The values for R_0 are 1.5, 2.5 and 5 and for N are 200 and 1000. This yields six simulation scenarios, one for each different pair of selected values of (R_0, N). In all instances, the mean infectious period $\mathrm{E}(T_D)$ is set to 10, specifying γ to be $\gamma = 0.1$. Note that, unlike previous simulations studies of this thesis (see for example simulation study A in section 2.7.1.2), it is not currently of interest to capture sampling variability (and investigate the sampling properties of some model assessment measure) and therefore it is sufficient to consider one dataset for each simulation scenario. To ensure that datasets are, in a sense, representative of their corresponding simulation scenario,

they are simulated so that they have final size equal to the (major outbreak) mode of the final size with respect to the sampling distribution.

4.2.3.3 Run conditions

The Exp-HM model is fitted to each generated dataset, using both of the MCMC algorithms under comparison, namely the standard-1d MCMC algorithm and the IS-1d MCMC algorithm. Recall that the standard-1d MCMC algorithm for the Exp-HM model (Algorithm 5) was given in section 1.3.5.5. The IS-1d MCMC algorithm differs from the standard-1d MCMC algorithm only in the update step of the infection component, which is conducted as described in section 4.2.2 above. For clarity, all steps of the IS-1d MCMC algorithm are listed in Algorithm 17 below. The run conditions of the two algorithms are set to be identical, in order to ensure that the runtime of the two algorithms is roughly equal so that performance can be compared by looking only at the mixing and not the runtime. Note that, the computations and random number generations involved, are the same for both algorithms with the exception of the calculation of the individual-specific proposal parameters for the IS-1d algorithm, which takes minimal computational time and is done only once in the algorithm, after the burn-in period.

Specifically, both algorithms are run for 20000 iterations, after a burn-in of 5000, by repeating the infection component update step as many times as the number of infections, so that in each MCMC iteration all infection times are attempted to be updated (see last paragraph of section 1.3.5.3), and the target space is sufficiently explored. The burn-in iterations of the IS-1d algorithm, according to which the proposal parameters are calculated (see section 4.2.2 for details), are run using the standard-1d algorithm. The prior distribution assignment, for both algorithms, is as in section 1.3.5.5, with the prior parameters being specified so that the uncertainty for all model parameters (except for the label of the initial infective α which is assigned

a prior distribution as $\alpha \sim \text{U}[1:n]$) is expressed via uninformative $\text{Exp}(10^{-3})$ prior distributions.

Algorithm 17 IS-1d MCMC algorithm for the Exp-HM model

1. Suppose the current state is $(\beta^{(s)}, \gamma^{(s)}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$
2. Sample $\beta^{(s+1)} \sim \pi(\beta \mid \boldsymbol{r}, \gamma^{(s)}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)}) \equiv \text{Gamma}\left(n - 1 + \nu_\beta, A^{(s)} + \lambda_\beta\right)$ using a Gibbs step
3. Sample $\gamma^{(s+1)} \sim \pi(\gamma \mid \boldsymbol{r}, \beta^{(s+1)}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)}) \equiv \text{Gamma}\left(n + \nu_\gamma, B^{(s)} + \lambda_\gamma\right)$ using a Gibbs step
4. Generate $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)})$ according to $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \beta^{(s+1)}, \gamma^{(s+1)})$ using a MH step as follows
   (a) Choose one of the n ever-infected individuals, say k, as $k \sim \text{U}[1:n]$
   (b) Propose a candidate infection time for individual k, say i_k^*, as $r_k - i_k^* \sim \text{Exp}(\gamma_k)$, where γ_k is specified using the burn-in iterations, as described in section 4.2.2
   (c) Calculate the acceptance ratio $r = \frac{\pi(\alpha^*, i_\alpha^*, \boldsymbol{i}^* \mid \boldsymbol{r}, \beta^{(s+1)}, \gamma^{(s+1)})}{\pi(\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)} \mid \boldsymbol{r}, \beta^{(s+1)}, \gamma^{(s+1)})} \times \frac{q_k(r_k - i_k^{(s)})}{q_k(r_k - i_k^*)}$, where $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \beta, \gamma)$ is given by expression (1.22) and $q_k(x)$ is the p.d.f. of a random variable $X_k \sim \text{Exp}(\gamma_k)$
   (d) Set $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}) = (\alpha^*, i_\alpha^*, \boldsymbol{i}^*)$ with probability $1 \wedge r$; otherwise set $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}) = (\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$
5. Set the next state as $(\beta^{(s+1)}, \gamma^{(s+1)}, \alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)})$.

4.2.3.4 Results

Before looking at the results and comparing algorithm performance, both algorithms were checked for evidence of non-stationarity (see the part regarding stationarity in section 1.3.2.3) by visually inspecting MCMC trace plots and by assessing whether the posterior densities of the two algorithms appeared to be the same. For all datasets, both algorithms appeared to have converged to the (same) desired posterior distribution. As an illustration, the relevant plots are provided for one of the datasets,

in figure A.22 in the Appendix.

As explained in the last paragraph of section 4.1.1, algorithm performance is assessed by the level of efficiency with which the infection step is performed, i.e. by the quality of mixing of the infection component. Following Xiang and Neal (2014), the sum of the infectious periods, $B = \sum_{k=1}^{n}(r_k - i_k)$, is chosen as the key summary statistic of the infection component and mixing and efficiency are assessed with respect to B. As described in section 1.3.2.3 (see the part about chain mixing), this is done by measuring the amount of autocorrelation in the sampled values of B; the higher the autocorrelation, the slower the mixing and the less the efficiency. Again, following the aforementioned part of section 1.3.2.3, the amount of autocorrelation in the sampled values of B is quantified by calculating the sample autocorrelations at lag-k, denoted as $\hat{\rho}_k$, and by producing an autocorrelation function (ACF) plot, of $\hat{\rho}_k$ against k, and additionally by calculating the effective sample size associated with the MCMC sample of B, interpreted as the number of i.i.d. sampled values required to give the same precision as the MCMC sample in question.

Table 4.1 and figure 4.9, respectively, give the effective sample sizes and the ACF plots, for the two compared algorithms, for each of the six datasets of the simulation study. As observed, from both table 4.1 and figure 4.9, the IS-1d algorithm, to a lesser or a greater extent, exhibits improved mixing compared to the standard-1d algorithm, for all considered datasets. The magnitude of this improvement for the different simulation scenarios, as quantified by the effective sample size ratio of the IS-1d algorithm over the standard-1d algorithm, ranges from 1.08 to 2.51.

As far as the effect of the dimension of the data on the comparison, quantified by N, there is no evidence to suggest that the comparative performance of the two algorithms changes as N increases. More precisely, by focusing on pairs of datasets for which R_0 is fixed and N changes (i.e. considering datasets 1 and 2, or, 3 and

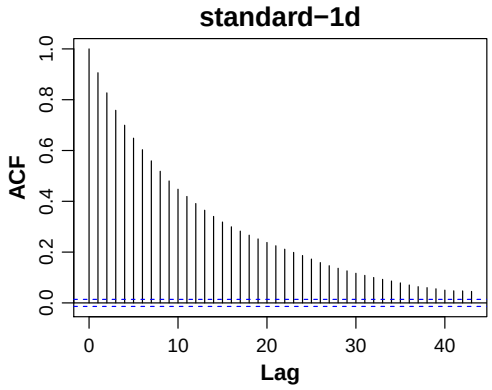
4, or, 5 and 6) it appears that the comparative performance of the algorithms is not affected by N e.g. for dataset 3 ($R_0 = 2.5, N = 200$) the effective sample size ratio of the IS-1d over the standard-1d algorithm is 1122/540=2.08, and for dataset 4 ($R_0 = 2.5, N = 1000$) is 868/361=2.40, suggesting that (when $R_0 = 2.5$) for both $N = 200$ and $N = 1000$, the IS-1d algorithm is more than twice as efficient than the standard-1d algorithm.

Unlike N, R_0 has an evident effect on the comparison of the two algorithms. More specifically, as seen from table 4.1 and figure 4.9, the performance of the standard-1d algorithm becomes worse as R_0 increases, whereas the performance of the IS-1d algorithm improves from $R_0 = 1.5$ to $R_0 = 2.5$, and becomes worse from $R_0 = 2.5$ to $R_0 = 5$. These patterns, and the driving reasons behind them, are made more clear by figure 4.10, where the acceptance (and effective sample size) proportions for the update step of the infection time i_k, are plotted against the individual label k, $k = 1, 2, \ldots, n$, for both algorithms. Notice that, to highlight the effect of R_0, datasets 2, 4 and 6 (i.e. the datasets for which $N = 1000$ and R_0 is 1.5, 2.5 and 5, respectively) of the simulation study are considered. Looking at the left column of figure 4.10, which corresponds to the standard-1d algorithm, one can see the curvature effect (first discussed in section 4.2.1), which becomes more apparent as R_0 increases. As already explained in section 4.2.1, this curvature effect reflects the tendency of the standard-1d proposals to overestimate (underestimate) the length of the infectious period of individuals with labels close to 1 (n), especially as R_0 increases (see figure 4.2). Looking at the right column of figure 4.10, corresponding to the 1S-1d algorithm, one can see that the IS-1d proposals, more or less, improve the acceptance (and effective sample size) proportions of individuals. As discussed in section 4.2.2, this is because the IS-1d proposals are allowed to have a different parameter for each individual k, and can thus accordingly adjust to having shorter or longer infectious periods. It is noted though, that the performance of the IS-1d proposals is clearly better for labels closer to 1, rather than closer to n, and this pattern becomes increasingly apparent

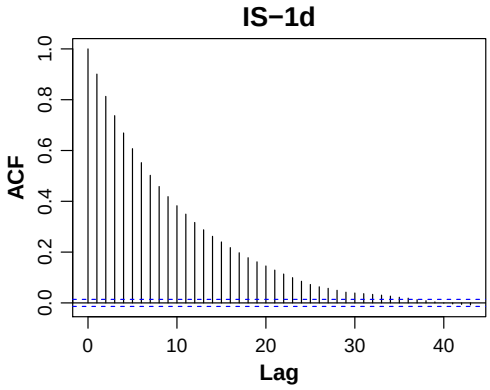
as R_0 increases. Looking at figure 4.5 in section 4.2.2 and recalling the discussion on what happens in the case that R_0 is extremely large in section 4.2.1, this is to be expected, because for individuals with labels close to n, and for larger values of R_0, it becomes increasingly harder for any Exponential proposal distribution (irrespectively of its parameter) to resemble the target distribution. This explains why the IS-1d algorithm is less efficient for $R_0 = 5$ compared to $R_0 = 2.5$.

Table 4.1: Effective sample size for $B = \sum_{k=1}^{n}(r_k - i_k)$, for the two compared MCMC algorithms, standard-1d and IS-1d, for each of the six datasets of simulation study E. The simulation and run conditions are described in sections 4.2.3.2 and 4.2.3.3, respectively.

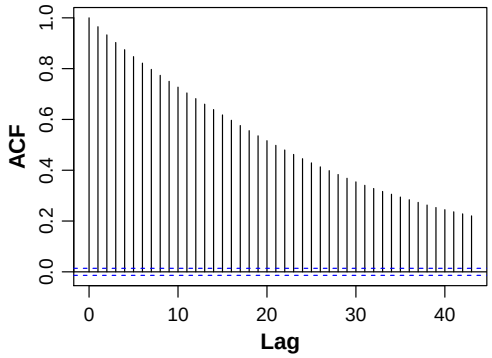
	Algorithm	
	standard-1d	**IS-1d**
Dataset 1 ($R_0 = 1.5, N = 200$)	763	1032
Dataset 2 ($R_0 = 1.5, N = 1000$)	408	438
Dataset 3 ($R_0 = 2.5, N = 200$)	540	1122
Dataset 4 ($R_0 = 2.5, N = 1000$)	361	868
Dataset 5 ($R_0 = 5, N = 200$)	284	715
Dataset 6 ($R_0 = 5, N = 1000$)	257	626



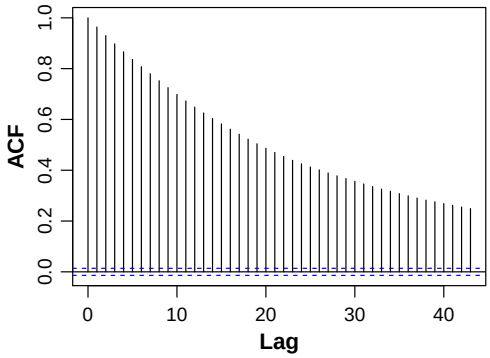
(a) Dataset 1 ($R_0 = 1.5, N = 200$)



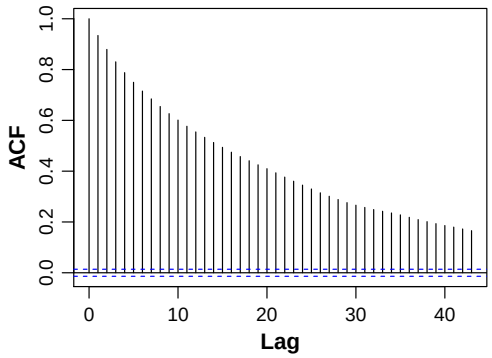
(b) Dataset 1 ($R_0 = 1.5, N = 200$)



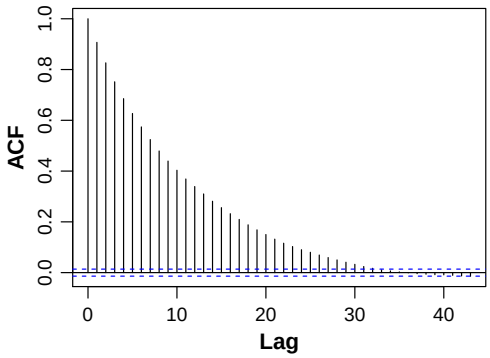
(c) Dataset 2 ($R_0 = 1.5, N = 1000$)



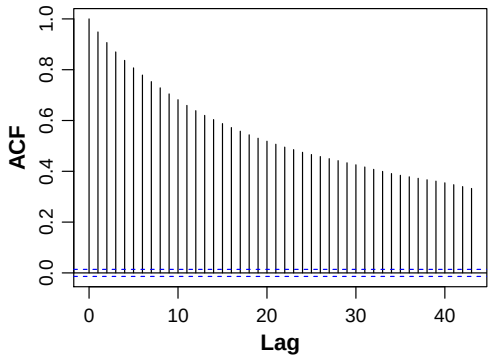
(d) Dataset 2 ($R_0 = 1.5, N = 1000$)



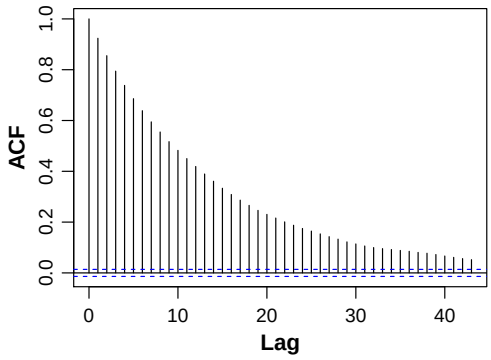
(e) Dataset 3 ($R_0 = 2.5, N = 200$)



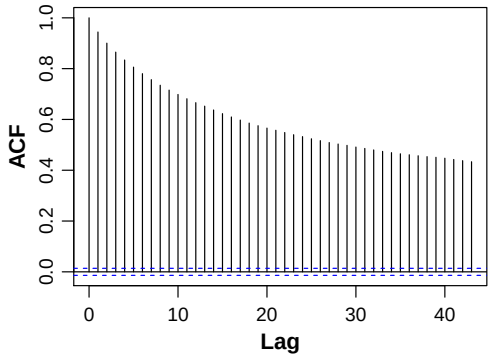
(f) Dataset 3 ($R_0 = 2.5, N = 200$)



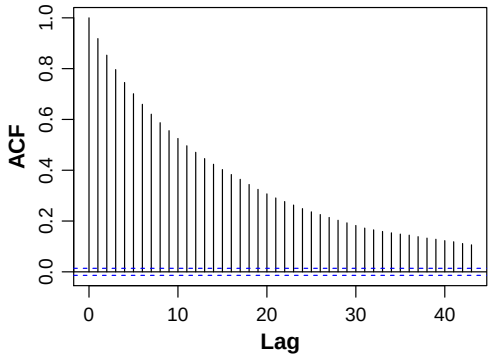
(g) Dataset 4 ($R_0 = 2.5, N = 1000$)



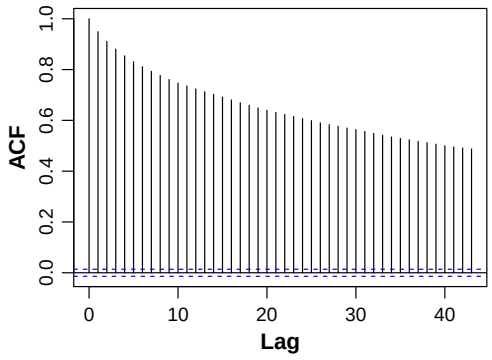
(h) Dataset 4 ($R_0 = 2.5, N = 1000$)



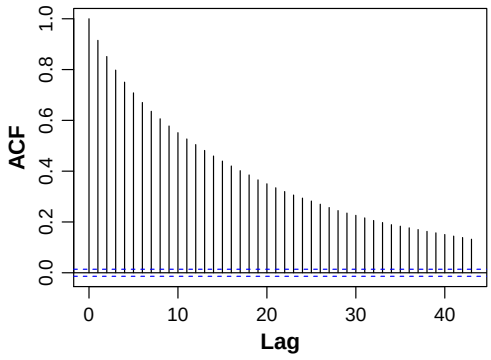
(i) Dataset 5 ($R_0 = 5, N = 200$)



(j) Dataset 5 ($R_0 = 5, N = 200$)

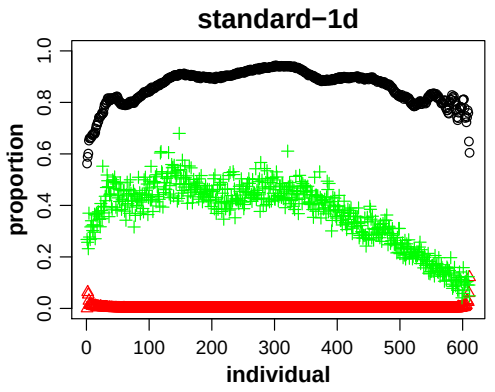


(k) Dataset 6 ($R_0 = 5, N = 1000$)

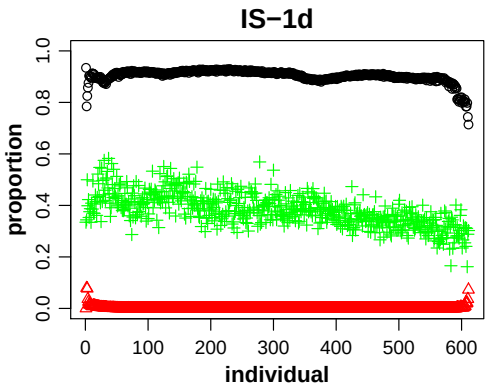


(l) Dataset 6 ($R_0 = 5, N = 1000$)

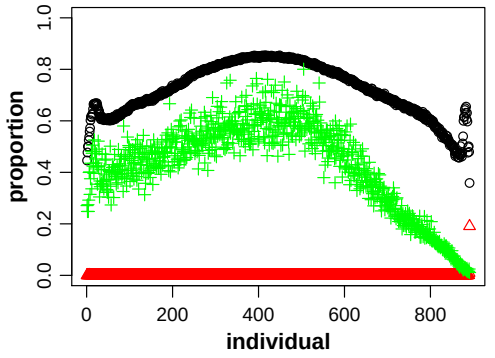
Figure 4.9: ACF plots for $B = \sum_{k=1}^{n}(r_k - i_k)$, for each of the six datasets of simulation study E. The simulation and run conditions are described in sections 4.2.3.2 and 4.2.3.3, respectively. Left column corresponds to the standard-1d MCMC algorithm and right column corresponds to the IS-1d MCMC algorithm.



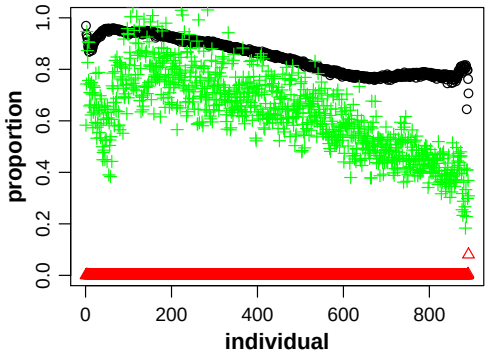
(a) Dataset 2 ($R_0 = 1.5, N = 1000$)



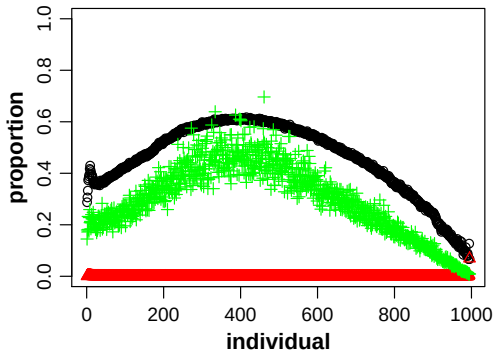
(b) Dataset 2 ($R_0 = 1.5, N = 1000$)



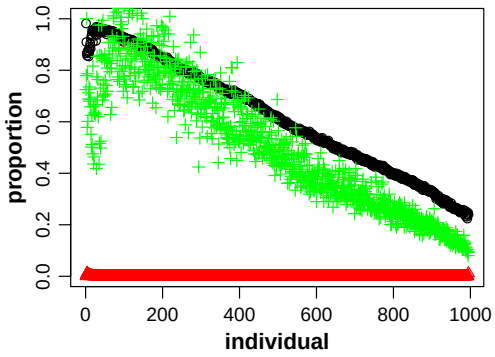
(c) Dataset 4 ($R_0 = 2.5, N = 1000$)



(d) Dataset 4 ($R_0 = 2.5, N = 1000$)



(e) Dataset 6 ($R_0 = 5, N = 1000$)



(f) Dataset 6 ($R_0 = 5, N = 1000$)

Figure 4.10: Acceptance proportion (black circles), effective sample size over actual sample size (green pluses) and inadmissibility proportion (red triangles) for the 1-dimensional update step of the infection time i_k, against individual label k, $k = 1, 2, \ldots, n$, for datasets 2, 4 and 6 of simulation study E. The simulation and run conditions are described in sections 4.2.3.2 and 4.2.3.3, respectively. Left column corresponds to the standard-1d MCMC algorithm and right column corresponds to the IS-1d MCMC algorithm.

4.2.3.5 Conclusions

The conclusions from simulation study E are summarized as follows.

- For all scenarios of the simulation study, the IS-1d algorithm, to a lesser or a greater extent, exhibits improved mixing compared to the standard-1d algorithm. Specifically, for the considered datasets of the simulation study, the IS-1d algorithm is from 1.08 times up to 2.51 times more efficient than the standard-1d algorithm.
- The dimension of the data, quantified by N, does not appear to have an evident effect on the comparative performance of the two algorithms suggesting that for the practically interesting cases of large-scale outbreaks, the comparative performance of the two algorithms would be similar to small-scale outbreaks.
- The severity of the outbreak, quantified by R_0, appears to have an effect on the performance of the two algorithms suggesting that the IS-1d algorithm can handle the cases that R_0 is large better than the standard-1d algorithm, although the performance of both algorithms will drop if R_0 gets too large (around 5 for the scenarios of this simulation study).

Overall, the IS-1d algorithm offers a not drastic but still welcome improvement in mixing, compared to the standard-1d algorithm. Considering that the additional computational complexity and cost associated with the 1S-1d algorithm is minimal, there does not appear to be any reason not to use the IS-1d algorithm over the standard-1d algorithm in practice.

4.2.3.6 Remarks

As seen from the results of the simulation study, the performance of the IS-1d proposals is better for the infection times of individuals with labels closer to 1 rather than closer to n, and this pattern becomes increasingly apparent as R_0 increases (see figure 4.10). As explained, this is because, for large values of R_0,

the Exponential family of the proposal distributions, can not capture the shape of the target distribution, even if its parameter is specified optimally. A natural way of dealing with this problem would be to consider different, more flexible families of proposal distributions. For example, one choice that was considered was Gamma proposal distributions. However, such a choice appeared to be problematic as the proposal distribution was lighter-tailed than the target distribution creating an issue of the sampler ever visiting or leaving the tails (see the part about independent proposals in section 1.3.2.4).

Another solution would be to consider a Gibbs step for each infection time i_k, $k = 1, 2, \ldots n$. More specifically, the full conditional distribution of an infection time i_k is actually known and is described by a piecewise Exponential distribution in each interevent interval. That is, if all the rest (besides i_k) $2n-1$ infection and removal event times are ordered in time as $t_1 < t_2 < \ldots t_{2n-1}$, the full conditional distribution of i_k in the interval (t_m, t_{m+1}), $m = 0, 1, \ldots, 2n-2$, where $t_0 = -\infty$, is described by an $\text{Exp}(\delta_m)$ distribution and a parameter p_m, where p_m is the probability mass corresponding to the interval (with $\sum_{m=0}^{2n-2} p_m = 1$ so that the full conditional distribution of i_k is a probability distribution). Although such a scheme would be in a sense optimal for a 1-dimensional independent MH update step, as the acceptance proportion would move to 1 for any infection time i_k, practical implementation is computationally too costly. More precisely, at each execution of the update step of the infection component, before a candidate infection time could be proposed, one would need to order all event times according to time, and calculate the likelihood of the model $2n-1$ times, one for each interval (t_m, t_{m+1}), $m = 0, 2, \ldots, 2n-2$, so that the parameters δ_m and p_m, could be specified. Assuming that the infection step is repeated n times in each MCMC iteration, a Gibbs step for each infection time would require $(2n-1)n$ calculations of the likelihood in each MCMC iteration. Considering the computational cost associated with these calculations and how it grows with the

number of infections n, it is easy to see why such a Gibbs scheme would be essentially impossible to implement for any practically useful case.

4.2.4 Simulation study F

4.2.4.1 Purpose, simulation and run conditions

The purpose of simulation study F is the same as simulation study E (see section 4.2.3 right above) with the difference that the infectious period distribution is now assumed to be Gamma(ν, λ) instead of Exp(γ). Therefore, the simulation and run conditions for simulation study F are set almost identically as for simulation study E.

More precisely, simulation study F aims to compare the performance of the standard-1d and the IS-1d MCMC algorithms, for the case that the infectious period distribution is assumed to be Gamma(ν, λ). As in simulation study E, it is of interest to investigate the effect that N or R_0 might have on this comparison. In addition, the effect of the shape parameter ν, of the Gamma(ν, λ) infectious period distribution, is also investigated.

For each combination of selected values of the parameters R_0, ν and N, one dataset is simulated from the Gamma-HM model conditioning on the final size being equal to the (major outbreak) mode of the final size with respect to the sampling distribution. The values for R_0 are 1.5 and 2.5, for ν are 2 and 5 and for N are 200 and 1000, yielding eight simulation scenarios in total, one for each different trio of selected values of (R_0, ν, N). Note that, the selected values of ν serve to investigate how the algorithms compare when the Gamma distribution becomes increasingly different than the Exponential (recall from section 1.3.5.4 that the Gamma(ν, λ) distribution reduces to an Exp(λ) distribution in the cases that its shape parameter $\nu = 1$). The mean infectious period E(T_D) is set at 10 in all cases, specifying λ as $\lambda = 0.2$ and

$\lambda = 0.5$, for the instances that $\nu = 2$ and $\nu = 5$, respectively.

The Gamma-HM model is fitted to each simulated dataset, using both of the MCMC algorithms under comparison, the standard-1d MCMC algorithm and the IS-1d MCMC algorithm. Note that, for both algorithms, the shape parameter ν is assumed to be known. This is done to avoid mixing issues that are induced in the instance that ν is an unknown parameter to be estimated from the data (see e.g. Kypraios (2007); Jewell et al. (2009); Alharthi (2016) where ν was also treated as known). The standard-1d MCMC algorithm for the Gamma-HM model (Algorithm 6) was already given in section 1.3.5.5. The IS-1d MCMC algorithm differs from the standard-1d MCMC algorithm only in the update step of the infection component, which is conducted as described in section 4.2.2. For reference, all steps of the IS-1d MCMC algorithm are given in Algorithm 18 below. In all instances that the IS-1d MCMC algorithm is run, both of the parameters, ν_k and λ_k, of the IS-1d proposal distribution $\text{Gamma}(\nu_k, \lambda_k)$, $k = 1, 2, \ldots, n$, are tuned and not only λ_k (see section 4.2.2 on how this is done). The burn-in iterations of the IS-1d algorithm, according to which the proposal parameters are tuned, are run using the standard-1d algorithm. Just like in simulation study E, the run conditions of the two algorithms are set to be identical, ensuring that the runtime of the two algorithms is roughly equal so that performance can be compared by focusing only at the mixing and not the runtime. Specifically, both algorithms are run for 20000 iterations, after a burn-in of 5000, by repeating the infection component update step as many times as the number of infections, as described in the last paragraph of section 1.3.5.3. The prior distribution assignment, for both algorithms, is done as in section 1.3.5.5, with the prior parameters being set so that the prior uncertainty for all model parameters (except for the label of the initial infective α which is assigned a prior distribution as $\alpha \sim \text{U}[1 : n]$) is expressed via uninformative $\text{Exp}(10^{-3})$ distributions.

Algorithm 18 IS-1d MCMC algorithm for the Gamma-HM model

1. Suppose the current state is $(\beta^{(s)}, \lambda^{(s)}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$
2. Sample $\beta^{(s+1)} \sim \pi(\beta \mid \boldsymbol{r}, \nu, \lambda^{(s)}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)}) \equiv \text{Gamma}\left(n - 1 + \nu_\beta, A^{(s)} + \lambda_\beta\right)$ using a Gibbs step
3. Sample $\lambda^{(s+1)} \sim \pi(\lambda \mid \boldsymbol{r}, \nu, \beta^{(s+1)}, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)}) \equiv \text{Gamma}\Big(\nu n + \nu_\lambda, B^{(s)} + \lambda_\lambda\Big)$ using a Gibbs step
4. Generate $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)})$ according to $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu, \beta^{(s+1)}, \lambda^{(s+1)})$ using a MH step as follows
   (a) Choose one of the n ever-infected individuals, say k, as $k \sim \text{U}[1:n]$
   (b) Propose a candidate infection time for individual k, say i_k^*, as $r_k - i_k^* \sim \text{Gamma}(\nu_k, \lambda_k)$, where ν_k and λ_k are specified using the burn-in iterations, as described in section 4.2.2
   (c) Calculate the acceptance ratio $r = \frac{\pi(\alpha^*, i_\alpha^*, \boldsymbol{i}^* | \boldsymbol{r}, \nu, \beta^{(s+1)}, \lambda^{(s+1)})}{\pi(\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)} | \boldsymbol{r}, \nu, \beta^{(s+1)}, \lambda^{(s+1)})} \times \frac{q_k(r_k - i_k^{(s)})}{q_k(r_k - i_k^*)}$, where $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu, \beta, \lambda)$ is given by expression (1.25) and $q_k(x)$ is the p.d.f. of a random variable $X_k \sim \text{Gamma}(\nu_k, \lambda_k)$
   (d) Set $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}) = (\alpha^*, i_\alpha^*, \boldsymbol{i}^*)$ with probability $1 \wedge r$; otherwise set $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}) = (\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$
5. Set the next state as $(\beta^{(s+1)}, \lambda^{(s+1)}, \alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)})$.

4.2.4.2 Results and conclusions

The comparison of the algorithms is conducted identically to simulation study E (see section 4.2.3.4 for more details). Initially, the stationarity of the algorithms was checked (see the part regarding stationarity in section 1.3.2.3) by visually inspecting MCMC trace plots and by assessing whether the posterior densities of the two algorithms appeared to be the same. In all cases, both algorithms appeared to have converged to the (same) desired posterior distribution. For reference, the relevant plots are given for one of the datasets of the simulation study, in figure A.23 in the Appendix. Then, mixing and efficiency were assessed with respect to $B = \sum_{k=1}^{n}(r_k - i_k)$, specifically by producing ACF plots and calculating the effective

sample size associated with the MCMC sample of B.

Table 4.2 and figure 4.11 respectively give the effective sample sizes and the ACF plots, for the two compared algorithms, for each of the eight datasets of the simulation study. Similar to simulation study E (see section 4.2.3.4), the IS-1d algorithm, overall, has better mixing compared to the standard-1d algorithm, although the extent of the improvement is less in the present simulation study compared to simulation study E. Specifically, the effective sample size ratio of the IS-1d algorithm over the standard-1d algorithm, ranges from 0.91 to 1.99; for three of the considered datasets the mixing of the algorithms is essentially the same and for the remaining five datasets the IS-1d algorithm has, to a lesser or a greater extent, better mixing.

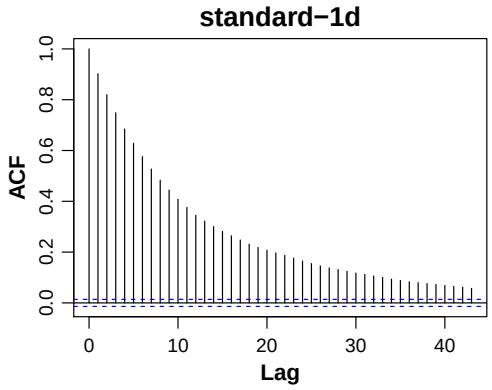
The effect of N and R_0 on the comparison is similar as in simulation study E (see section 4.2.3.4). Specifically, the comparative performance of the two algorithms does not systematically change with N (this can be seen by focusing on pairs of datasets for which R_0 and ν are the same and N changes), whereas regarding R_0, the two algorithms have similar mixing for $R_0 = 1.5$ but the IS-1d algorithm has better mixing for $R_0 = 2.5$. The explanation behind the effect of R_0 on the performance of the algorithms is the same as in simulation study E (see section 4.2.3.4 for more details) and can be gauged by looking at the acceptance (and effective sample size) proportion plots (figure 4.12) and observing how the IS-1d algorithm corrects for the curvature effect exhibited by the standard-1d algorithm, for the different values of R_0. As far as the effect of ν, it is evident (see table 4.2 and figure 4.11) that the quality of mixing of both algorithms becomes worse as ν increases but there is no evidence to suggest that the relative quality of mixing of the two algorithms changes with ν.

The main conclusion from simulation study F is the same as from simulation study E (see section 4.2.3.5). In general, the IS-1d algorithm provides a welcome, although not a radical, improvement in mixing, compared to the standard-1d algorithm. Since

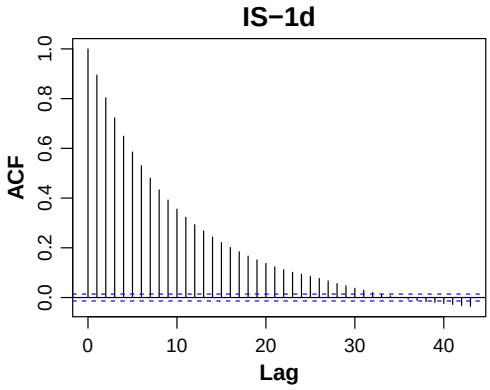
the computational cost and complexity of the two algorithms is essentially the same, the IS-1d algorithm appears to be the better choice, between the two, in practice.

Table 4.2: Effective sample size for $B = \sum_{k=1}^{n}(r_k - i_k)$, for the two compared MCMC algorithms, standard-1d and IS-1d, for each of the eight datasets of simulation study F. The simulation and run conditions are described in section 4.2.4.1.

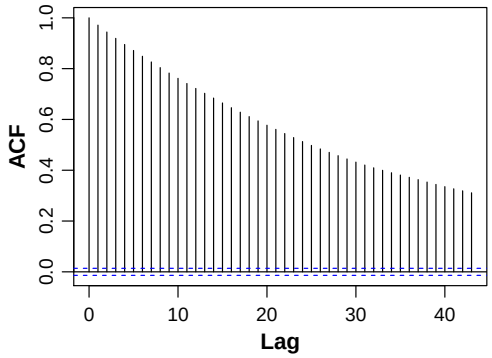
	Algorithm	
	standard-1d	**IS-1d**
Dataset 1 ($R_0 = 1.5, \nu = 2, N = 200$)	852	1192
Dataset 2 ($R_0 = 1.5, \nu = 2, N = 1000$)	346	316
Dataset 3 ($R_0 = 1.5, \nu = 5, N = 200$)	436	410
Dataset 4 ($R_0 = 1.5, \nu = 5, N = 1000$)	203	211
Dataset 5 ($R_0 = 2.5, \nu = 2, N = 200$)	407	810
Dataset 6 ($R_0 = 2.5, \nu = 2, N = 1000$)	359	697
Dataset 7 ($R_0 = 2.5, \nu = 5, N = 200$)	267	436
Dataset 8 ($R_0 = 2.5, \nu = 5, N = 1000$)	287	385



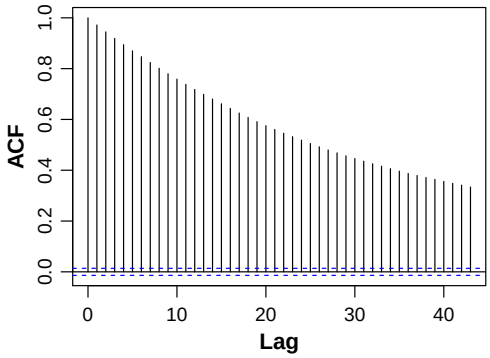
(a) Dataset 1 ($R_0 = 1.5, \nu = 2, N = 200$)



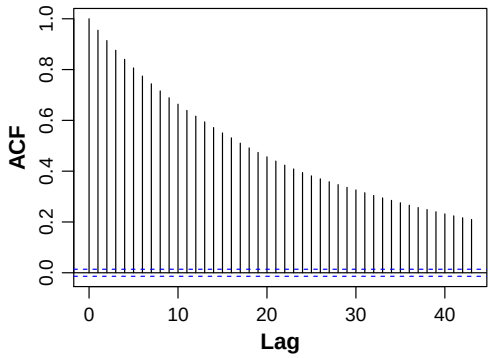
(b) Dataset 1 ($R_0 = 1.5, \nu = 2, N = 200$)



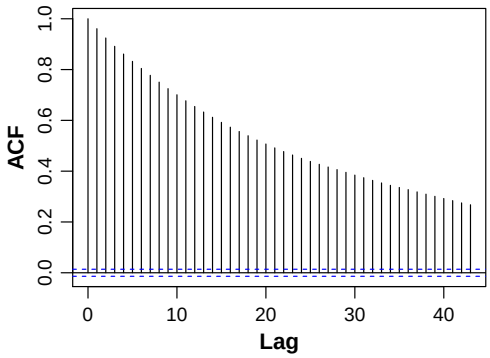
(c) Dataset 2 ($R_0 = 1.5, \nu = 2, N = 1000$)



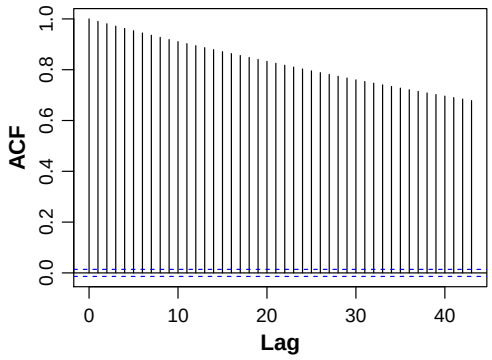
(d) Dataset 2 ($R_0 = 1.5, \nu = 2, N = 1000$)



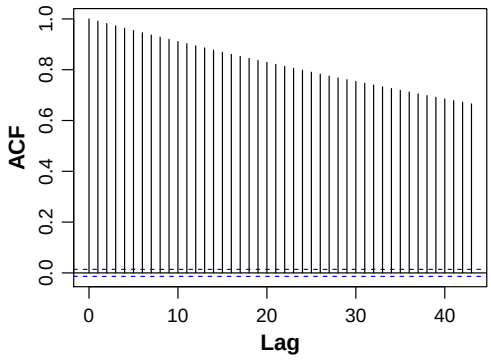
(e) Dataset 3 ($R_0 = 1.5, \nu = 5, N = 200$)



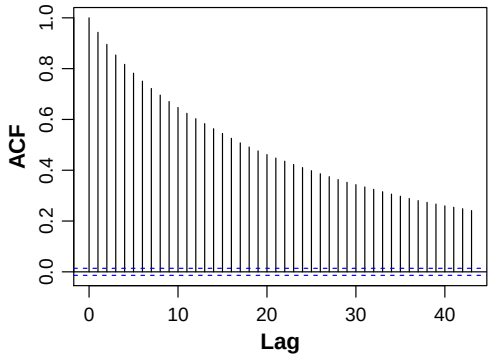
(f) Dataset 3 ($R_0 = 1.5, \nu = 5, N = 200$)



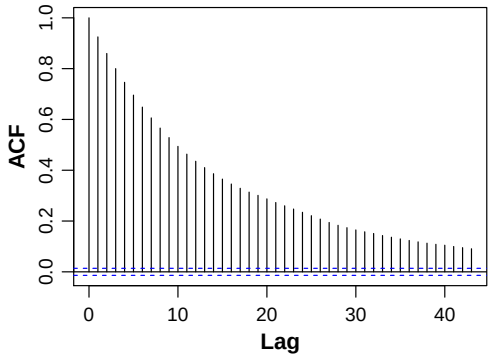
(g) Dataset 4 ($R_0 = 1.5, \nu = 5, N = 1000$)



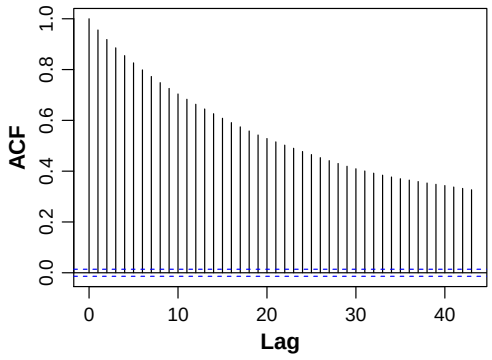
(h) Dataset 4 ($R_0 = 1.5, \nu = 5, N = 1000$)



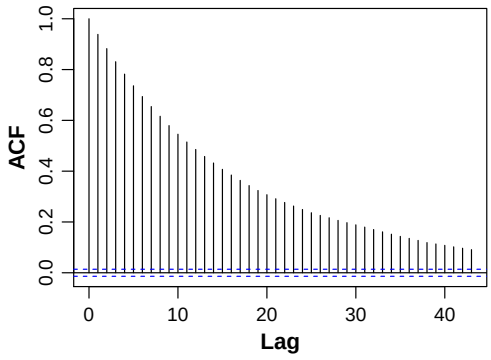
(i) Dataset 5 ($R_0 = 2.5, \nu = 2, N = 200$)



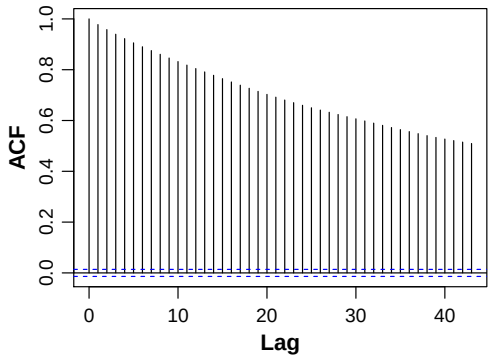
(j) Dataset 5 ($R_0 = 2.5, \nu = 2, N = 200$)



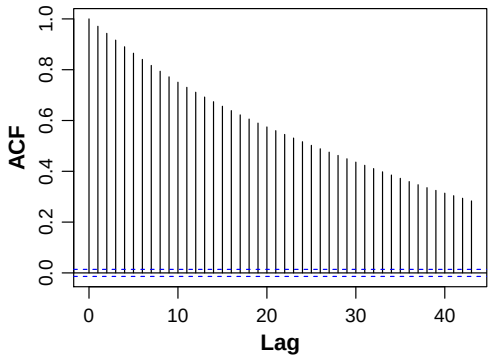
(k) Dataset 6 ($R_0 = 2.5, \nu = 2, N = 1000$)



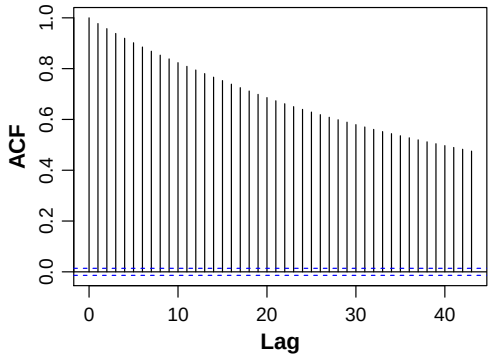
(l) Dataset 6 ($R_0 = 2.5, \nu = 2, N = 1000$)



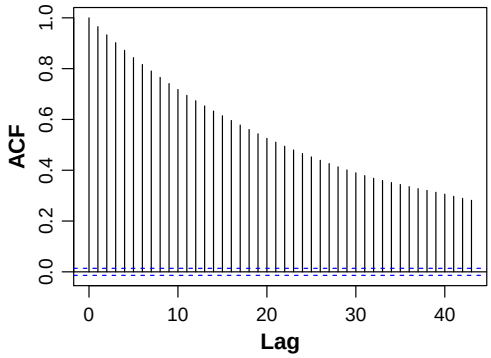
(m) Dataset 7 ($R_0 = 2.5, \nu = 5, N = 200$)



(n) Dataset 7 ($R_0 = 2.5, \nu = 5, N = 200$)

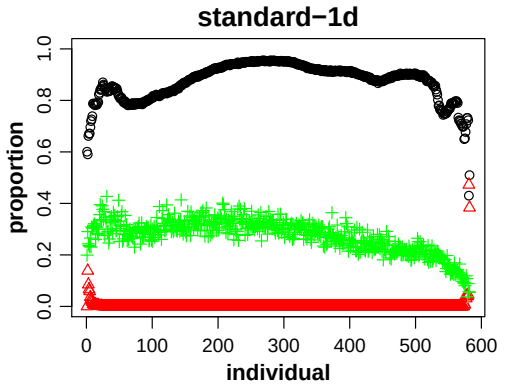


(o) Dataset 8 ($R_0 = 2.5, \nu = 5, N = 1000$)

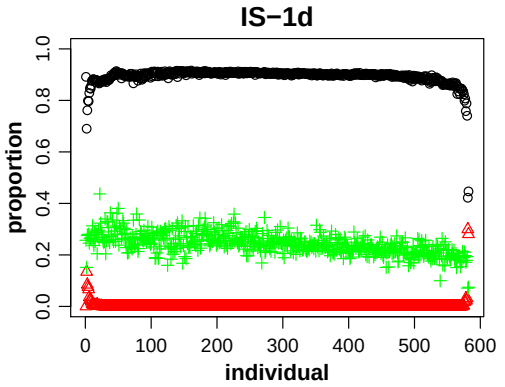


(p) Dataset 8 ($R_0 = 2.5, \nu = 5, N = 1000$)

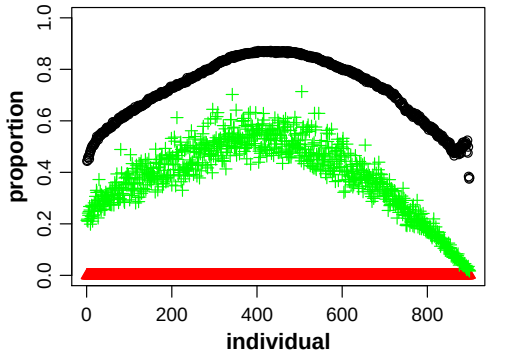
Figure 4.11: ACF plots for $B = \sum_{k=1}^{n}(r_k - i_k)$, for each of the eight datasets of simulation study F. The simulation and run conditions are described in section 4.2.4.1. Left column corresponds to the standard-1d MCMC algorithm and right column corresponds to the IS-1d MCMC algorithm.



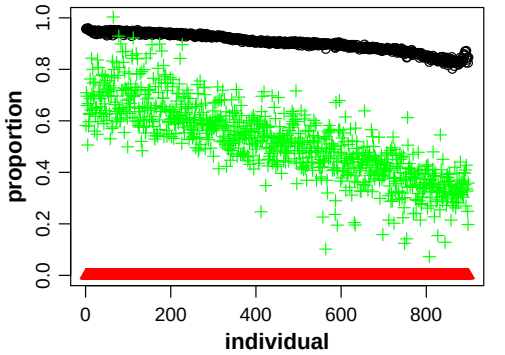
(a) Dataset 2 ($R_0 = 1.5, \nu = 2, N = 1000$)



(b) Dataset 2 ($R_0 = 1.5, \nu = 2, N = 1000$)



(c) Dataset 6 ($R_0 = 2.5, \nu = 2, N = 1000$)



(d) Dataset 6 ($R_0 = 2.5, \nu = 2, N = 1000$)

Figure 4.12: Acceptance proportion (black circles), effective sample size over actual sample size (green pluses) and inadmissibility proportion (red triangles) for the 1-dimensional update step of the infection time i_k, against individual label k, $k = 1, 2, \ldots, n$, for datasets 2 and 6 of simulation study F. The simulation and run conditions are described in section 4.2.4.1. Left column corresponds to the standard-1d MCMC algorithm and right column corresponds to the IS-1d MCMC algorithm.

4.2.4.3 Remarks

As mentioned above, in all instances that the IS-1d MCMC algorithm was run, both of the parameters, ν_k and λ_k, of the IS-1d proposal distribution Gamma(ν_k, λ_k), $k = 1, 2, \ldots, n$, were tuned. The alternative approach (see section 4.2.2 for more details) was to set $\nu_k = \nu$, for all $k = 1, 2, \ldots, n$, where ν the known value of the shape parameter, and tune only the rate parameter λ_k, $k = 1, 2, \ldots, n$. As suggested by later investigations (see the relevant discussion in section 4.3.2.2), in the cases that $R_0 = 2.5$, such individually tuned proposals perform optimally when the parameter

tuning is done using the former approach (as done in the present simulation study), but in the cases that $R_0 = 1.5$, optimal performance is achieved when the tuning is done using the latter approach. That is to say, that the performance of the IS-1d algorithm, in the instances that $R_0 = 1.5$, both in the simulation study above and in general, could be further improved by tuning only the rate parameter, as opposed to tuning both the shape and the rate parameter.

4.3 Block update steps for the infection component

Section 4.3 turns attention from 1-dimensional update steps of the infection component to block update steps. As already mentioned in the beginning of section 4.2, developing MCMC algorithms that update many infection times at a time, in a block update step, is the main intention of this chapter as such algorithms are in general more efficient than their 1-dimensional counterparts (see e.g. Xiang and Neal (2014)).

All MCMC algorithms considered throughout section 4.3 are described and applied to the Gamma-HM model, that is the standard SIR model with one-to-one infection rate parameter β and Gamma(ν, λ) infectious periods (see section 1.3.5.5). There are two reasons for choosing to focus on the Gamma-HM model. First, this is the model also used to describe and apply the block update MCMC algorithm in Xiang and Neal (2014) (see section 4.3.1 to follow for more details), the algorithm that serves as a reference point and a comparator for a new algorithm to be developed in this section; choosing to apply the to-be-developed algorithm on the same model as the existing algorithm of Xiang and Neal (2014), creates more direct conditions for comparison. Second, the Gamma(ν, λ) distribution is the most general choice for the infectious period distribution, among the considered choices of this thesis (see section 1.3.4.1); in fact, as remarked in section 1.3.5.4 the Exponential distribution is

a special case of the Gamma distribution when the shape parameter ν is equal to 1. It is noted that, similar to the 1-dimensional update step case and for reasons already explained in section 4.2.4.1, the shape parameter ν of the Gamma(ν, λ) distribution of the infectious periods, is assumed to be known for both algorithms, for all inference purposes, throughout section 4.3.

4.3.1 Existing block update steps and their limitations: standard block MCMC algorithm

There are many different existing MCMC algorithms that attempt to perform block update steps for the infection component, using and sometimes combining different ideas such as non-centered parameterizations and parameter reduction (see section 1.4.2 and the references therein). Arguably though, the optimally performing existing MCMC algorithm, for block update of the infection component, is the algorithm of Xiang and Neal (2014). This algorithm, plays an important part for the purposes of section 4.3 as it first serves as a foundation, for the to-be-developed block update MCMC algorithm to built upon, and then as a comparator to its performance. Therefore, before proceeding any further, it is necessary to describe its features and implementation procedure. The description of the algorithm is provided for the case that the shape parameter ν is known, as assumed throughout section 4.3, but it is noted that in the paper of Xiang and Neal (2014) the shape parameter ν is considered unknown and is an additional parameter to be estimated from the data (see section 4.4.2 for a further discussion on this topic).

4.3.1.1 Features and implementation procedure

The first feature of the algorithm is parameter reduction. Recall from section 1.3.5.5 (see the part about Bayesian inference and MCMC algorithm for the Gamma-HM model and in particular equation (1.23)) that the 'default' target posterior density for the Gamma-HM model (when ν is assumed to be known) is $\pi(\beta, \lambda, \alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu)$.

In the algorithm of Xiang and Neal (2014), β and λ are analytically integrated out, reducing the target posterior density to $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu)$ (the expression for $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu)$ is given further below). As argued in Xiang and Neal (2014), the motivation behind this is twofold. First, it makes the target space of the algorithm smaller, and therefore easier to explore. Second, it places the focus on the update of the augmented infection component, rather than on model parameter components, and thus allows more freedom in the exploration of the infection space; which, as mentioned in the beginning of this chapter (see section 4.1.1), is by far the most challenging part of the inference. Note that, since the target density is $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu)$, the algorithm produces samples from $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu)$, the posterior density of the infection component. However, given an MCMC sample from $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu)$, samples from the marginal posterior distributions of the parameters of interest, β and λ, can easily be achieved by sampling from their respective known form full conditional distributions (see equation (1.24)), conditioning on the already sampled values of the infection component $(\alpha, i_\alpha, \boldsymbol{i})$.

The second feature of the algorithm is that the infection component is updated using a MH block update step, where many infection times are updated simultaneously. This is done as follows. Suppose that the chain is transitioning from its s^{th} to its $(s+1)^{\text{th}}$ iteration so that the current state of the infection component is $(\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$. Initially, one chooses the number of infection times to be updated, say m, referred to as the *block step size* (the choice of m is discussed in the next paragraph). Given the block step size m, one then chooses, uniformly at random, a set of m out of the total n individual labels, say $\boldsymbol{b} = \{b_1, b_2, \ldots, b_m\}$, for which their corresponding infection times, are to be updated. Note that, $\boldsymbol{b}$ is chosen randomly at each iteration, and thus depends on the iteration, however this dependence is suppressed in the notation for visual clarity. Also note that, having chosen $\boldsymbol{b}$, the infection times of individuals not in $\boldsymbol{b}$, remain fixed at their current values during the update step and therefore, to specify the candidate state $(\alpha^*, i_\alpha^*, \boldsymbol{i}^*)$ of the chain, one proposes infection times only for the

remaining individuals that are in $\boldsymbol{b}$. A candidate infection time for each individual $k \in \boldsymbol{b}$, say i_k^*, is proposed as $r_k - i_k^* \overset{\text{i.i.d.}}{\sim} \text{Gamma}(\nu, \lambda')$, where λ' is drawn from the full conditional distribution of λ (equation (1.24)), conditioned on the current value of the infection component, i.e. $\lambda' \sim \pi(\lambda \mid \boldsymbol{r}, \nu, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$. Finally, one accepts or rejects the proposed move from $(\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$ to $(\alpha^*, i_\alpha^*, \boldsymbol{i}^*)$ after calculating the MH acceptance ratio. Note that, since λ' is drawn from the full conditional distribution of λ, conditioned on the current value of the infection component, the above proposal distribution is essentially the same model-driven proposal distribution used in the standard-1d update MCMC scheme of the model (see Algorithm 6) but with two differences. The first and most obvious difference is that the proposed infection times are now accepted or rejected as a block instead of one at a time. The second difference is that the rate parameter of the proposal distribution is only introduced as an intermediary (since λ has been integrated out it is not a part of the MCMC scheme), to facilitate the proposal of the infection component, and is analytically integrated out in calculating the proposal ratio (more details to follow below). Due to having essentially the same proposal mechanism as the standard-1d scheme, this proposal proposal distribution is henceforth referred to as the *standard block (standard-block) proposal* and the block update algorithm of Xiang and Neal (2014) as the *standard block (standard-block) MCMC algorithm*.

The third and final feature of the algorithm is that the block step size is specified using the burn-in iterations so that the algorithm performs optimally. In general, at each MCMC iteration, the block step size can be chosen according to a discrete random variable, say M, taking values in $\{1, 2, \ldots, n\}$, whose distribution is described by its probability mass function (p.m.f.), $f_M(j) = P(M = j) = p_j$, $j = 1, 2, \ldots n$. The key in optimizing the performance of the scheme is in choosing the probabilities p_j, $j = 1, 2, \ldots n$. Let a_j be the acceptance proportion of the update step, corresponding to block step size j, $j = 1, 2, \ldots n$; e.g. a_5 is the acceptance proportion of the update step when a block of 5 infections is proposed to be updated. In general, the quality

of mixing of the chain, for each block step size j, $j = 1, 2, \dots n$, is quantified by $q_j = ja_j$, where the higher the value of q_j the better the mixing. To see this, notice that the higher the value of j the larger the size of the attempted jump, and, the higher the acceptance proportion a_j the more often the jump is performed. To this end, the idea of Xiang and Neal (2014) (can be modified depending on the application in context e.g. on how big is n), and what is done in this thesis, is to run a batch of burn-in iterations by setting $p_j = 1/n$, for all $j = 1, 2, \dots n$ (i.e. set M to have a discrete uniform distribution on $\{1, 2, \dots, n\}$), and record the value of each a_j, $j = 1, 2, \dots n$. Then, to optimize mixing, in the case that the block step size is desired to be random at each iteration, one runs the subsequent chain iterations, by setting $p_j \propto q_j^d$, $j = 1, 2, \dots n$, for some $d > 0$. Alternatively, the block step size can remain fixed, for all subsequent iterations, by setting $p_m = 1$, for some $m \in \{1, 2, \dots, n\}$, and $p_j = 0$, for all $j = 1, 2, \dots, n$, $j \neq m$. The obvious choice of m that optimizes mixing is the one such that $q_m \geq q_j$ for all $j = 1, 2, \dots, n$. Xiang and Neal (2014) argue that there could be some benefits in allowing the block step size to be random, however for simplicity and to ease the comparison between algorithms, in this thesis, only the case that the block step size is fixed is considered.

Before giving the step-by-step implementation procedure of the algorithm, expressions for the target posterior density and the proposal ratio, both necessary to calculate the MH acceptance ratio, are given. The target posterior density of the scheme, $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu)$, is such that $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu) = \iint \pi(\beta, \lambda, \alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu) d\beta d\lambda \propto \iint \pi(\boldsymbol{r}, \boldsymbol{i} \mid \beta, \lambda, \nu, \alpha, i_\alpha) \pi(\beta, \lambda, \alpha, i_\alpha) d\beta d\lambda$, where $\pi(\boldsymbol{r}, \boldsymbol{i} \mid \beta, \lambda, \nu, \alpha, i_\alpha)$ (as previously seen by equations (1.17) and (1.12)) is given by

$$\begin{aligned}\pi(\boldsymbol{r}, \boldsymbol{i} \mid \beta, \lambda, \nu, \alpha, i_\alpha) = &\left(\prod_{k=1, k\neq\alpha}^{n} \beta Y_{i_k^-} \right) \times \exp(-\beta A) \\ &\times \left(\frac{\lambda^\nu}{\Gamma(\nu)} \right)^n \left(\prod_{k=1}^{n} (r_k - i_k) \right)^{\nu-1} \exp(-\lambda B),\end{aligned}$$

and $\pi(\beta, \lambda, \alpha, i_\alpha)$ (as previously described in section 1.3.5.5) is such that $\pi(\beta, \lambda, \alpha, i_\alpha) = \pi(\beta)\pi(\lambda)\pi(\alpha)\pi(i_\alpha)$ with $\pi(\beta) \equiv \text{Gamma}(\nu_\beta, \lambda_\beta)$, $\pi(\lambda) \equiv \text{Gamma}(\nu_\lambda, \lambda_\lambda)$, $\pi(\alpha) \equiv \text{U}[1:n]$ and $\pi(-i_\alpha) \equiv \text{Exp}(\xi_{i_\alpha})$. This integration is straightforward to perform and yields that

$$\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu) \propto \left(\prod_{k=1, k\neq\alpha}^{n} Y_{i_k^-}\right) \times (A + \lambda_\beta)^{-(n-1+\nu_\beta)} \times \left(\prod_{k=1}^{n}(r_k - i_k)\right)^{\nu-1} \times (B + \lambda_\lambda)^{-(\nu n+\nu_\lambda)} \times \exp(\xi_{i_\alpha} i_\alpha)\mathbb{1}_{\{i_\alpha<0\}}. \tag{4.2}$$

For the proposal ratio, consider first the forward proposal density, denoted as $h(s \to *)$, associated with the move from $(\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$ to $(\alpha^*, i_\alpha^*, \boldsymbol{i}^*)$. Based on the proposal mechanism described above, it should be clear that $h(s \to *)$ is such that $h(s \to *) = \int \prod_{k\in\boldsymbol{b}} q(r_k - i_k^* \mid \lambda)\pi(\lambda \mid \boldsymbol{r}, \nu, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})d\lambda$, where $q(x \mid \lambda)$ is the p.d.f. of a random variable $X \sim \text{Gamma}(\nu, \lambda)$ and $\pi(\lambda \mid \boldsymbol{r}, \nu, \alpha, i_\alpha, \boldsymbol{i})$ is the full conditional density of λ (equation (1.24)). A few lines of algebra (see Xiang and Neal (2014, page 244)) yield that $h(s \to *) = \frac{\Gamma(\nu m+\nu n+\nu_\lambda)}{\Gamma(\nu)^m\Gamma(\nu n+\nu_\lambda)} \times \frac{\left(\prod_{k\in\boldsymbol{b}}(r_k - i_k^*)\right)^{\nu-1}(B^{(s)}+\lambda_\lambda)^{\nu n+\nu_\lambda}}{(B^{(s)}+B^*\lambda_\lambda)^{\nu m+\nu n+\nu_\lambda}}$. The expression for the backward proposal density $h(* \to s)$, associated with the move from $(\alpha^*, i_\alpha^*, \boldsymbol{i}^*)$ to $(\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$, is acquired by simply exchanging the role of current and proposed values in the expression of the forward proposal density and therefore, the proposal ratio is given by

$$\frac{h(* \to s)}{h(s \to *)} = \left(\frac{B^* + \lambda_\lambda}{B^{(s)} + \lambda_\lambda}\right)^{\nu n+\nu_\lambda} \left(\frac{\prod_{k\in\boldsymbol{b}}(r_k - i_k^{(s)})}{\prod_{k\in\boldsymbol{b}}(r_k - i_k^*)}\right)^{\nu-1}. \tag{4.3}$$

All implementation steps of the standard-block MCMC algorithm are collected in Algorithm 19 below.

4.3.1.2 Run conditions

In all instances that the standard-block MCMC algorithm is run in this chapter, the run conditions are the same. Therefore, for ease of reference, these conditions are

Algorithm 19 Standard-block MCMC algorithm for the Gamma-HM model

1. Suppose the current state is $(\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$
2. Generate $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)})$ according to $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu)$ using a MH step as follows
   (a) Choose, uniformly at random, m of the n ever-infected individuals, say $\boldsymbol{b} = \{b_1, b_2, \ldots, b_m\}$, where m is specified using the burn-in iterations, as described in section 4.3.1
   (b) Sample λ' from the full conditional distribution of λ as $\lambda' \sim \pi(\lambda \mid \boldsymbol{r}, \nu, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)}) \equiv \text{Gamma}\Big(\nu n + \nu_\lambda, B^{(s)} + \lambda_\lambda\Big)$
   (c) Propose a candidate infection time for each individual $k \in \boldsymbol{b}$, say i_k^*, as $r_k - i_k^* \overset{\text{i.i.d.}}{\sim} \text{Gamma}(\nu, \lambda')$
   (d) Calculate the acceptance ratio $r = \frac{\pi(\alpha^*, i_\alpha^*, \boldsymbol{i}^* \mid \boldsymbol{r}, \nu)}{\pi(\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)} \mid \boldsymbol{r}, \nu)} \times \frac{h(*\to s)}{h(s\to *)}$, where $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu)$ is given by expression (4.2) and $\frac{h(*\to s)}{h(s\to *)}$ by equation (4.3)
   (e) Set $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}) = (\alpha^*, i_\alpha^*, \boldsymbol{i}^*)$ with probability $1 \wedge r$; otherwise set $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}) = (\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$
3. Set the next state as $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)})$.

collected in the present paragraph. As mentioned above, the standard-block MCMC algorithm is implemented as in Algorithm 19. The prior distribution assignment follows section 1.3.5.5 and the prior parameters are specified so that the uncertainty for all model parameters (except for the label of the initial infective α which is assigned a prior distribution as $\alpha \sim \text{U}[1:n]$) is quantified via uninformative $\text{Exp}(10^{-3})$ prior distributions. The block step size m is specified using a batch of $S_{B_1} = 500n$ burn-in iterations, as described earlier in section 4.3.1 (see the paragraph about the third feature of the algorithm). Notice that, setting $S_{B_1} = 500n$, ensures that, for each block step size j, $j = 1, 2, \ldots n$, more or less, 500 update steps are performed and thus the corresponding acceptance proportions a_j, needed to specify m, are accurately estimated. Besides specifying m, these S_{B_1} iterations are also used to specify L, where L is the value according to which post burn-in iterations are thinned by. The

value of L is specified so that, roughly a proportion of 0.8 of the infection times are changed between consecutive values of the thinned chain; this is done by first calculating the expected number of infection times that are successfully updated, at a given iteration, based on a fixed block step size m, given by $q_m = ma_m$, and then setting $L = 0.8n/q_m$. Using the specified values of m and L, the algorithm is run for $S = 20000L$ post burn-in iterations, keeping only every L^{th} iteration, following a second batch of burn-in iterations, of length $S_{B_2} = 4500L$.

4.3.1.3 Visualizing the movement of the sampler

Similar to the 1-dimensional case (see sections 4.2.1 and 4.2.2), the development of a new block update scheme is guided by acknowledging the limitations of the existing scheme. To this end, the Gamma-HM model is fitted using the standard-block MCMC algorithm (Algorithm 19), under the run conditions of section 4.3.1.2, to an example dataset. The dataset is generated from the Gamma-HM model itself by setting $N = 500$, $R_0 = 2.5$ and $\nu = 5$, and results in $n = 448$ total infections.

Figure 4.13 shows the acceptance and the inadmissibility proportion, for each block step size, associated with the initial $S_{B_1} = 500n$ iterations used to specify the optimal block step size. Looking at figure 4.13, it is clear that the acceptance (inadmissibility) proportion reduces (increases) as the block step size increases. Intuitively, this should be expected, to lesser or greater extent, as the larger the block step size the larger the size of the attempted jump and the higher the chance of rejection. However, it is very interesting to notice how quickly the acceptance proportion drops to 0. More precisely, if one considers a random block step size, distributed according to a p.m.f. $p_j \propto q_j$, $j = 1, 2, \ldots n$, so that the mixing is optimized, where p_j and q_j are as in section 4.3.1.1, then $\sum_{j=1}^{25} p_j = 0.54$ and $\sum_{j=1}^{50} p_j = 0.80$. That is to say, that the majority, specifically 0.54, of the proposed moves favour proposing no more than 25 infection times; the proportion increases to 0.8 if block step sizes up to 50 are considered. Note that, if one considers a fixed block step size (following the procedure

described in section 4.3.1.1), the block step size value that is found to optimize mixing is $m = 12$, i.e. updating 12 infection times achieves optimal performance for the algorithm. Also interesting, is that the low acceptance proportions for all, except the relatively small block step sizes, can not be entirely accounted to the proposed values being inadmissible (and thus being automatically rejected). For example, the acceptance proportion corresponding to block step size 100 is 0 but the associated inadmissibility proportion is 0.16, implying that the remaining 0.84 of the proposed moves are admissible but are still rejected based on the MH acceptance probability. These observations suggest that the standard-block MCMC algorithm struggles to move around the target space, unless the block step size is relatively small, and the reasons for this can not be solely accounted to inadmissibility. Note that, similar observations were also made in Xiang and Neal (2014), when the authors applied the algorithm to a foot and mouth disease dataset of 1021 infections; the optimal performance of the algorithm was found to be achieved for block step sizes around 16 and no proposed moves were accepted for block step sizes larger than 64.

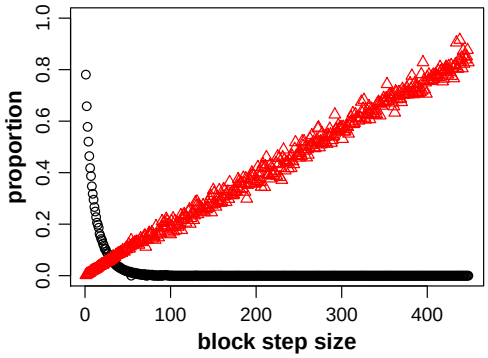


Figure 4.13: Acceptance proportion (black circles) and inadmissibility proportion (red triangles) for the block update step of the standard-block MCMC algorithm, against block step size. The dataset is generated from a Gamma-HM model ($N = 500$, $R_0 = 2.5$, $\nu = 5$) and the number of infections is $n = 448$.

To make the reasons behind the above limitations more precise, an attempt is made

to develop a visual appreciation, on how the sampler moves around the target space. To this end, the focus is placed on the target posterior density of the scheme, $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu)$, given by expression (4.2). Looking carefully at expression (4.2), one notices that, if any contributions from the prior distribution are ignored, $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu)$ (in log scale) depends on the infection component only via four real-valued terms, namely $A = \int_{i_\alpha}^{r_n} X_t Y_t dt$, $B = \sum_{k=1}^{n} (r_k - i_k)$, $C = \sum_{k=1, k\neq\alpha}^{n} \log(Y_{i_k^-})$ and $D = \sum_{k=1}^{n} \log(r_k - i_k)$. That is to say, that the 4-dimensional vector (A, B, C, D) is in this sense sufficient for $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu)$. What this implies, is that one can produce density plots of (A, B, C, D) and gain a visual overview on the movement of the sampler. More specifically, this can be done as follows. First, given a posterior sample of the infection component from the run of the algorithm on some dataset, the bivariate posterior density plots (contour plots) of (A, B, C, D) can be produced. Then, given a current state of the chain and a block step size, one can follow steps 2(b) and (c) of Algorithm 19 and generate values from the proposal distribution, thus producing a proposal sample of the infection component. Note that, the proposed values of the infection component are generated under the condition of being admissible as interest is in understanding why admissible moves are systematically being rejected. Given a proposal sample of the infection component, the movement of the sampler on the target space can be visualized, by imposing on the plotted bivariate posterior density plots the corresponding bivariate proposal density plots.

Figure 4.14 illustrates these plots (with the exception of plots involving the term D in order to reduce the number of plots and maintain visual clarity) for the example dataset, considering three different block step sizes, 15, 100 and 250. The general observation from figure 4.14 is that, as suggested above, the sampler struggles to move around the target space, unless the block step size is relatively small. Specifically, as the block step size increases, the sampler appears to increasingly suffer from a loss of orientation and proposes to move to regions that are beyond the support of the target density. Before more can be said, two useful remarks must be made.

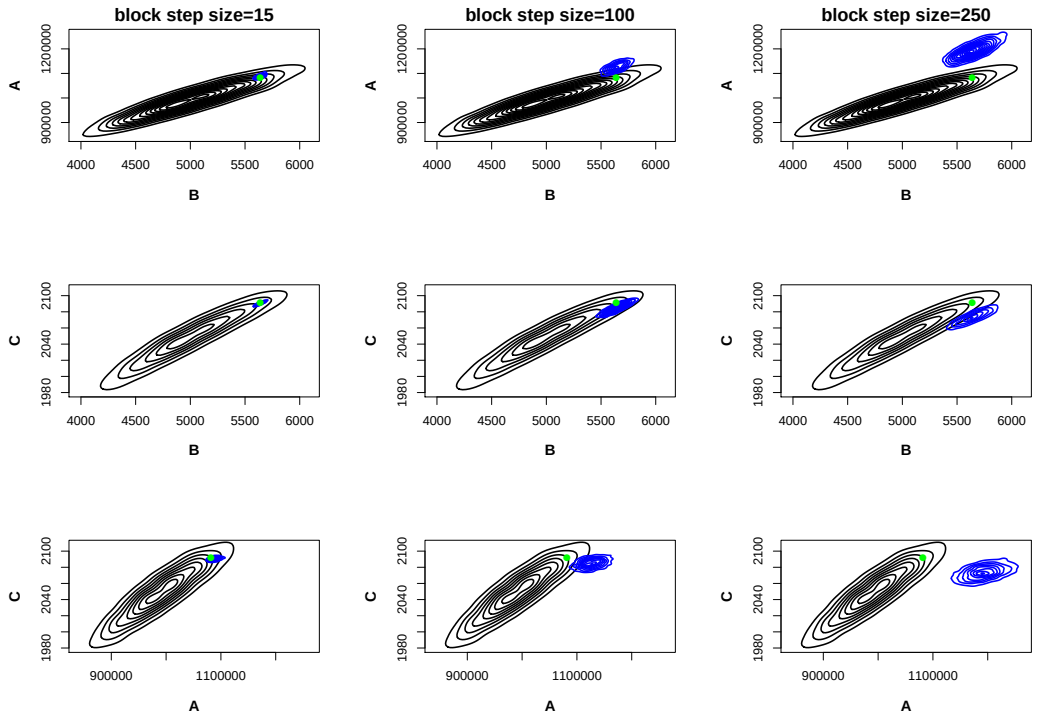


Figure 4.14: Bivariate target posterior densities (black, solid contours) and bivariate standard-block proposal densities (blue, solid contours), for the vector (A, B, C). Imposed (green, circle) is the current state. Columns (left to right) correspond to block step size values of 15, 100 and 250, respectively. The dataset is generated from a Gamma-HM model ($N = 500$, $R_0 = 2.5$, $\nu = 5$) and the number of infections is $n = 448$.

The first remark is related to the value of the block step size and how it affects the properties of the sampler. Recall that (see the first paragraph in section 4.3.1.1) the target posterior density of the scheme is $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu)$. Note however, that when transitioning from iteration s to $s+1$, given a block step size m and having chosen the set of individuals $\boldsymbol{b}$, whose infection times are to be updated, the infection times of individuals not in $\boldsymbol{b}$, remain fixed at their current values during the update step. That is to say, that the target posterior density reduces to

$$\pi(\alpha, i_\alpha, \boldsymbol{i}_{[\boldsymbol{b}]} \mid \boldsymbol{r}, \nu, \boldsymbol{i}^{(s)}_{[-\boldsymbol{b}]}), \tag{4.4}$$

where, if $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ is a vector with n entries, $\boldsymbol{x}_{[A]}$ denotes the vector containing all entries of $\boldsymbol{x}$ whose index is in $A \subseteq \{1, 2, \ldots, n\}$ and $\boldsymbol{x}_{[-A]}$ the vector containing all entries of $\boldsymbol{x}$ whose index is not in $A \subseteq \{1, 2, \ldots, n\}$.

The second remark, is related to the proposal parameter λ' and to how it induces dependency on the current state, with respect to the term B. Suppose that the chain is transitioning from its s^{th} to its $(s+1)^{\text{th}}$ iteration, so that $B^{(s)}$ is the current value of $B = \sum_{k=1}^{n}(r_k - i_k)$, and assume that $\boldsymbol{b} = \{b_1, b_2, \ldots, b_m\}$ is the set of individuals for which infection times are to be proposed. Consider the to-be-proposed value of B, denoted as B^*. This is a random variable, with respect to the proposal distribution, i.e. under the randomness induced by the to-be-proposed infection times i_k^*, $k \in \boldsymbol{b}$, which are such that $r_k - i_k^* \overset{\text{i.i.d.}}{\sim} \text{Gamma}(\nu, \lambda')$, $\lambda' \sim \pi(\lambda \mid \boldsymbol{r}, \nu, \alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)}) \equiv \text{Gamma}(\nu n + \nu_\lambda, B^{(s)} + \lambda_\lambda)$ (see equation (1.24)). Taking into account the fact that infection times are only proposed for individuals $k \in \boldsymbol{b}$ (see the remark in the previous paragraph), and using the linearity of expectation, one can see that the expectation of B^* is given by $\text{E}(B^*) = \sum_{k \notin \boldsymbol{b}}(r_k - i_k^{(s)}) + \sum_{k \in \boldsymbol{b}} \text{E}(r_k - i_k^*)$. Heuristically, based on this expression, if the block step size m is much smaller than n, the value of $\text{E}(B^*)$ is roughly specified by the term $\sum_{k \notin \boldsymbol{b}}(r_k - i_k^{(s)})$, which in that case is roughly equal to $B^{(s)}$, and therefore $E(B^*)$ is close to $B^{(s)}$. Alternatively, if the block step size m is close to n, then the value of $\text{E}(B^*)$ is roughly specified by the term $\sum_{k \in \boldsymbol{b}} \text{E}(r_k - i_k^*)$. Noting that, $\text{E}(r_k - i_k^*) = \nu/\lambda'$, $k \in \boldsymbol{b}$, and $\text{E}(1/\lambda') = (B^{(s)} + \lambda_\lambda)/(\nu n + \nu_\lambda - 1)$, and using the law of total expectation, one can see that $\sum_{k \in \boldsymbol{b}} \text{E}(r_k - i_k^*) = \frac{\nu m (B^{(s)} + \lambda_\lambda)}{\nu n + \nu_\lambda - 1}$, and thus $\text{E}(B^*)$ is again roughly close to $B^{(s)}$. What these suggest, is that the sampler has a somewhat dependent nature in the exploration of the target space, in the sense that it tries to centre itself around the current value of B and propose moves around this value (see the part about dependent proposal distributions in section 1.3.2.4).

Turning attention back to figure 4.14, one can see how this last remark is reflected on the relevant plots (i.e. the bivariate densities involving the term B), as the proposed

moves appear to be, on average, around the current value, with respect to B; e.g. in figure 4.14, $B^{(s)} = 5639$, and $E(B^*)$ is 5640, 5641 and 5644 for block step size 15, 100 and 250, respectively. This feature of the sampler appears to be desirable, in the sense that proposed moves are in most cases within the support of the posterior distribution, with respect to B. However, looking at figure 4.14, the same cannot be said with respect to A and C. That is to say, that the proposal distribution does not match the posterior distribution, in the sense that values of (A, B, C) that are very plausible under the proposal distribution (i.e. typically proposed) are not supported under the posterior distribution. This pattern becomes increasingly apparent as the block step size increases and the first remark made above (see expression (4.4)), can help explain why. When the block step size is small, most infection times remain fixed at their current values. As a result, the area of the proposed region is small and also there is less freedom for the proposal distribution of (A, B, C) to demonstrate any differences that it might have with the corresponding posterior distribution. However, as the block step size increases, fewer infections remain fixed at their current values, the area of the proposed region becomes larger and the potential for the proposal distribution of (A, B, C) to be dissimilar to the corresponding posterior distribution increases.

Note that, the above offer an explanation on why for both the example dataset in question, and the foot and mouth disease dataset in Xiang and Neal (2014), no proposed moves are accepted for block step sizes larger than a certain value.

4.3.2 Dependent individual-specific block MCMC algorithm

The limitations of the standard-block MCMC algorithm, exhibited in section 4.3.1.3 above, guide the development of an alternative proposal scheme, for block updating the infection component, which gives rise to a novel, block update MCMC algorithm. For reasons soon to be made apparent, this new algorithm is referred to as *dependent*

individual-specific block (DIS-block) MCMC algorithm and its associated proposal distribution as *DIS-block proposal*. At first, section 4.3.2.1, describes the features and the implementation procedure of the algorithm. Subsequently, section 4.3.2.2, collects the conditions according to which the algorithm is run. Then, section 4.3.2.3, applies the algorithm to the example dataset of section 4.3.1 and investigates how it moves around the target space, while drawing comparisons with the standard-block MCMC algorithm.

4.3.2.1 Features and implementation procedure

The DIS-block algorithm is designed to address the limitations of the standard-block algorithm while also maintaining its good features. To this end, the first and third feature of the standard-block algorithm, namely parameter reduction and specification of the block step size using burn-in, are maintained in the DIS-block algorithm and performed in an identical manner (see section 4.3.1.1 for all the details). Note that, this means that the target density of the DIS-block algorithm is, as for the standard-block algorithm, $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu)$, and it is given by expression (4.2). Note also that, despite the fact that the block step size can in general be random, a similar approach as for the standard-block algorithm is taken and only the case that the block step size is fixed is considered. As already mentioned in section 4.3.1.1, taking this approach, facilitates better conditions for comparison.

The difference between the two algorithms lies in the proposal mechanism of the infection component and in particular on the parameters of the proposal distribution of the infection times. More precisely, the DIS-block algorithm updates the infection component using a MH step as follows. Suppose that the chain is transitioning from its s^{th} to its $(s+1)^{\text{th}}$ iteration, so that the current state of the infection component is $(\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$, and let m be the block step size. First, choose, uniformly at random, a set of m out of the total n individual labels, say $\boldsymbol{b} = \{b_1, b_2, \ldots, b_m\}$, for which their corresponding infection times, are to be updated. Then, propose a candidate infection

time for each individual $k \in \boldsymbol{b}$, say i_k^*, independently, as $r_k - i_k^* \sim \text{Gamma}(\nu_k, \lambda_k d^{(s)})$, where ν_k and λ_k are individual-specific parameters, specified using burn-in iterations, and $d^{(s)}$ is a dependence inducing parameter, specified so that the expectation of B, with respect to the proposal distribution, is equal to the current value of B (the specification of ν_k, λ_k, $k = 1, 2, \ldots, n$, and $d^{(s)}$ is discussed in detail below). The proposed move from $(\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$ to $(\alpha^*, i_\alpha^*, \boldsymbol{i}^*)$ is accepted or rejected, after calculating the MH acceptance ratio.

As mentioned right above, the difference between the two algorithms is in the parameters of the proposal distribution of the infection times. Specifically, whilst the standard-block proposal associated with the infection time of individual k, $k = 1, 2, \ldots, n$, was $\text{Gamma}(\nu, \lambda')$ (see section 4.3.1.1 and steps 2(b) and (c) of Algorithm 19), the corresponding DIS-block proposal is $\text{Gamma}(\nu_k, \lambda_k d^{(s)})$. To explain the rationale under which ν_k, λ_k, $k = 1, 2, \ldots, n$, and $d^{(s)}$ are specified, note that $r_k - i_k^* \sim \text{Gamma}(\nu_k, \lambda_k d^{(s)})$ is equivalent to $(r_k - i_k^*)/d^{(s)} \sim \text{Gamma}(\nu_k, \lambda_k)$, due to the scaling property of the Gamma distribution. That is to say, that the procedure of proposing infectious periods as $r_k - i_k^* \sim \text{Gamma}(\nu_k, \lambda_k d^{(s)})$, independently for each individual $k \in \boldsymbol{b}$, can equivalently be seen as a two-step procedure, where infectious periods are first drawn from a $\text{Gamma}(\nu_k, \lambda_k)$ distribution, independently for each $k \in \boldsymbol{b}$, and subsequently scaled by a factor of $1/d^{(s)}$. Parameters ν_k and λ_k, $k = 1, 2, \ldots, n$, are associated with the first step and are tasked to address the limitations of the standard-block sampler, described in section 4.3.1.3, by attempting to make the proposal distribution more similar to the posterior distribution so that typically proposed values do not fall beyond the support of the posterior distribution. To this end, ν_k and λ_k, $k = 1, 2, \ldots, n$, are set to have two characteristics. First, they are individual-specific, so that they have the flexibility to capture the pattern sometimes exhibited in the target distribution, where the infectious periods of individuals are not homogeneous (see section 4.2.1). Second, they are specified using burn-in iterations, in order to allow for information about the target posterior distribution, as the above,

to be incorporated into the proposal distribution. Parameter $d^{(s)}$ is associated with the second step and is tasked to induce dependency on the current state, with respect to $B = \sum_{k=1}^{n} (r_k - i_k)$, similar to what the parameter λ' does for the standard-block algorithm (see 4.3.1.3). The following two paragraphs give the details regarding the specification of ν_k, λ_k, $k = 1, 2, \ldots, n$, and $d^{(s)}$.

To specify the individual-specific parameters, ν_k and λ_k, $k = 1, 2, \ldots, n$, a similar approach as for the IS-1d algorithm is followed (see the third paragraph of section 4.2.2), where ν_k and λ_k are specified using a MOM estimation, based on a batch of burn-in iterations. Note that, based on the investigations of section 4.2.2 and the results of simulation studies E (see section 4.2.3.4) and F (see section 4.2.4.2), specifying the proposal parameters in such a way, allows for the Gamma(ν_k, λ_k) IS-1d proposal distributions to become more similar to their associated target posterior distributions. The specification procedure is as follows. First, run a batch of burn-in iterations, using a different MCMC algorithm (such as the standard-1d or the standard-block algorithm), and store the sampled values of the infectious period of each individual k, $k = 1, 2, \ldots, n$, say $\{r_k - i_k^{(1_{B_1})}, r_k - i_k^{(2_{B_1})}, \ldots, r_k - i_k^{(S_{B_1})}\}$. Then, for each k, $k = 1, 2, \ldots, n$, fit a Gamma(ν_k, λ_k) distribution to $\{r_k - i_k^{(1_{B_1})}, r_k - i_k^{(2_{B_1})}, \ldots, r_k - i_k^{(S_{B_1})}\}$ and conduct a MOM estimation to specify ν_k and λ_k. Note that, one can conduct MOM estimation for both ν_k and λ_k, yielding $\nu_k = \frac{\bar{x}_k^2}{s_k^2}$ and $\lambda_k = \frac{\bar{x}_k}{s_k^2}$, where $\bar{x}_k = \frac{1}{S_{B_1}} \sum_{s=1}^{S_{B_1}} (r_k - i_k^{(s_{B_1})})$ and $s_k^2 = \frac{1}{S_{B_1}-1} \sum_{s=1}^{S_{B_1}} (r_k - i_k^{(s_{B_1})} - \bar{x}_k)$, or alternatively, since ν is assumed to be known, one can set $\nu_k = \nu$ for all $k = 1, 2, \ldots, n$ and conduct MOM estimation only for λ_k, yielding $\lambda_k = \frac{\nu}{\bar{x}_k}$, $k = 1, 2, \ldots, n$.

The dependence inducing parameter $d^{(s)}$ is specified as follows. Suppose that the chain is transitioning from its s^{th} to its $(s+1)^{\text{th}}$ iteration, so that $B^{(s)}$ is the current value of $B = \sum_{k=1}^{n} (r_k - i_k)$, and assume that $\boldsymbol{b} = \{b_1, b_2, \ldots, b_m\}$ is the set of individuals for which infection times are to be proposed. Consider the to-be-proposed value of B, denoted as B^*, which is a random variable, with respect to the proposal distribution,

i.e. under the randomness induced by the to-be-proposed infection times i_k^*, $k \in \boldsymbol{b}$, which are such that $r_k - i_k^* \sim \text{Gamma}(\nu_k, \lambda_k d^{(s)})$, independently. The value of $d^{(s)}$ is specified so that $\text{E}(B^*) = B^{(s)}$. Since $\text{E}(r_k - i_k^*) = \nu_k/(\lambda_k d^{(s)})$, for $k \in \boldsymbol{b}$, using the linearity property of the expectation, it is straightforward to see that the value of $d^{(s)}$, such that $\text{E}(B^*) = B^{(s)}$, is $d^{(s)} = \frac{\sum_{k\in\boldsymbol{b}}(\nu_k/\lambda_k)}{\sum_{k\in\boldsymbol{b}}(r_k - i_k^{(s)})}$. As already mentioned, parameter $d^{(s)}$ plays a similar role as parameter λ' does in the standard-block algorithm, in the sense that both induce dependency on the current state, with respect to B. Arguably though, $d^{(s)}$ does this in a more direct and favourable way as it enforces $\text{E}(B^*)$ to be exactly equal to $B^{(s)}$ whereas λ' only does this in an approximate sense (see the second remark in secion 4.3.1.3).

Before giving the step-by-step implementation procedure of the algorithm, an expression for calculating the proposal ratio is provided. Consider initially the forward proposal density, denoted as $g(s \to *)$, associated with the move from $(\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$ to $(\alpha^*, i_\alpha^*, \boldsymbol{i}^*)$. According to the proposal mechanism described above, it should be evident that $g(s \to *) = \prod_{k\in\boldsymbol{b}} q_k(r_k - i_k^* \mid d^{(s)})$, where (as shown above) $d^{(s)} = \frac{\sum_{k\in\boldsymbol{b}}(\nu_k/\lambda_k)}{\sum_{k\in\boldsymbol{b}}(r_k - i_k^{(s)})}$ and $q_k(x \mid d)$ is the p.d.f. of a random variable $X_k \sim \text{Gamma}(\nu_k, \lambda_k d)$, $k \in \boldsymbol{b}$. Exchanging the role of current and proposed values, yields that the backward proposal density $g(* \to s)$, associated with the move from $(\alpha^*, i_\alpha^*, \boldsymbol{i}^*)$ to $(\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$, is $g(* \to s) = \prod_{k\in\boldsymbol{b}} q_k(r_k - i_k^{(s)} \mid d^*)$, where $d^* = \frac{\sum_{k\in\boldsymbol{b}}(\nu_k/\lambda_k)}{\sum_{k\in\boldsymbol{b}}(r_k - i_k^*)}$. That is to say, that the proposal ratio is given by

$$\frac{g(* \to s)}{g(s \to *)} = \frac{\prod_{k\in\boldsymbol{b}} q_k(r_k - i_k^{(s)} \mid d^*)}{\prod_{k\in\boldsymbol{b}} q_k(r_k - i_k^* \mid d^{(s)})}, \tag{4.5}$$

where $d^{(s)}$, d^* and $q_k(x \mid d)$, $k \in \boldsymbol{b}$, are as above. The implementation steps of the DIS-block MCMC algorithm are collected below, in Algorithm 20.

Algorithm 20 DIS-block MCMC algorithm for the Gamma-HM model

1. Suppose the current state is $(\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$
2. Generate $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)})$ according to $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu)$ using a MH step as follows
   (a) Choose, uniformly at random, m of the n ever-infected individuals, say $\boldsymbol{b} = \{b_1, b_2, \ldots, b_m\}$, where m is specified using the burn-in iterations, as described in section 4.3.2.1
   (b) Propose a candidate infection time for each individual $k \in \boldsymbol{b}$, say i_k^*, independently, as $r_k - i_k^* \sim \text{Gamma}(\nu_k, \lambda_k d^{(s)})$, where ν_k and λ_k are specified using the burn-in iterations, and $d^{(s)}$ is specified so that $\text{E}(B^*) = B^{(s)}$, as described in section 4.3.2.1
   (c) Calculate the acceptance ratio $r = \frac{\pi(\alpha^*, i_\alpha^*, \boldsymbol{i}^* \mid \boldsymbol{r}, \nu)}{\pi(\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)} \mid \boldsymbol{r}, \nu)} \times \frac{g(* \to s)}{g(s \to *)}$, where $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu)$ is given by expression (4.2) and $\frac{g(* \to s)}{g(s \to *)}$ by equation (4.5)
   (d) Set $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}) = (\alpha^*, i_\alpha^*, \boldsymbol{i}^*)$ with probability $1 \wedge r$; otherwise set $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)}) = (\alpha^{(s)}, i_\alpha^{(s)}, \boldsymbol{i}^{(s)})$
3. Set the next state as $(\alpha^{(s+1)}, i_\alpha^{(s+1)}, \boldsymbol{i}^{(s+1)})$.

4.3.2.2 Run conditions

The present section collects the conditions according to which the DIS-block algorithm is run in this thesis. The rationale is to set these conditions to be as similar as possible as for the standard-block algorithm (see section 4.3.1.2), in order to ensure that the runtime of the two algorithms is roughly equal and their performance can be compared by focusing on mixing properties and not runtime.

As already mentioned, the standard-block MCMC algorithm is implemented as in Algorithm 20. The prior distribution assignment is identical as for the standard-block algorithm (see section 4.3.1.2). A difference in the run conditions of the two algorithms, is that for the standard-block algorithm, two batches of burn-in iterations are run, whereas for the DIS-block algorithm, an additional batch is required. This

additional batch of burn-in iterations, is run first and it is used to specify the individual specific parameters, ν_k and λ_k, $k = 1, 2, \ldots, n$, required for all subsequent iterations, following a MOM estimation procedure, as described in section 4.3.2.1 (the decision to estimate both ν_k and λ_k, in the procedure, or only λ_k, is discussed in the next paragraph). This batch, is run using the standard-1d MCMC algorithm (Algorithm 5), for a length of $S_{B_1} = 2250$ iterations, by repeating the infection component update step as many times as the number of infections so that, in each MCMC iteration, all infection times are attempted to be updated (see last paragraph of section 1.3.5.3). The second batch of burn-in iterations, of length $S_{B_2} = 500n$, is used to specify the optimal block step size m. As already mentioned in section 4.3.2.1, the procedure to specify m is identical to the standard-block algorithm (see section 4.3.1.1 for details). Note that, as is the case for the standard-block algorithm, these S_{B_2} iterations can also be used to specify the value according to which post burn-in iterations are thinned by, following the procedure described in section 4.3.1.1. However, it is preferable for comparison purposes to conduct this procedure only for the standard-block algorithm and then set the specified thinning value, say L, to be the same for the DIS-block algorithm. Using the specified values of ν_k, λ_k, $k = 1, 2, \ldots, n$, m and L, the algorithm is run for $S = 20000L$ post burn-in iterations, keeping only every L^{th} iteration, following a third batch of burn-in iterations, of length $S_{B_3} = 2250L$.

Based on provisional runs of the algorithm on simulated data, it was indicated that, in cases that R_0 is relatively small (roughly around 1.5 or smaller), optimal algorithm performance was achieved when estimating only λ_k, $k = 1, 2, \ldots, n$, in the MOM estimation procedure (after setting $\nu_k = \nu$, for all $k = 1, 2, \ldots, n$, where ν the known value of the shape parameter), whereas for larger values of R_0 (roughly around 2.5 or larger) it was favourable to allow both ν_k and λ_k, $k = 1, 2, \ldots, n$, to be estimated. A possible explanation for this can be given by considering the effect that R_0 has on the posterior distribution of the infectious periods of individuals, where the larger the

value of R_0 the higher the deviation from homogeneity among individuals (see section 4.2.1 and in particular the discussion regarding what happens when R_0 becomes very large). Therefore, the larger the value of R_0 the larger the need for flexibility in the proposal distribution parameters, to help capture the trend in the posterior distribution. According to the above, for all runs of the algorithm in this thesis, the decision of estimating both ν_k and λ_k, or only λ_k, is based on the value of R_0 under which the dataset is simulated; more specifically for datasets such that $R_0 = 1.5$ only λ_k, $k = 1, 2, \ldots, n$, is estimated and for datasets such that $R_0 = 2.5$ both ν_k and λ_k, $k = 1, 2, \ldots, n$, are estimated. Note that, in practice, where the algorithm is applied on real data and the true value of R_0 is not known, one can accordingly make the decision of estimating both ν_k and λ_k, or only λ_k, based on the sampled values of R_0 from the first batch of burn-in iterations.

Note that, despite requiring an additional batch of burn-in iterations, compared to the standard-block algorithm, the DIS-block algorithm does not require additional runtime. This is because the total burn-in iterations can remain the same (in this thesis the run conditions are such that the total burn-in iterations are $500n + 4500L$ and $2250n + 500n + 2250L$, for the standard-block and the DIS-block algorithm, respectively), only be split in three batches rather than two. Also note that, the standard-block algorithm includes an additional random number generation, at each MCMC iteration, as it needs to draw λ', the parameter needed to facilitate the proposal of the infections (step 2(b) in Algorithm 19). Although the additional time related to this draw is very small, it is likely to make a difference in the case that the number of MCMC iterations is very large. Conversely, the additional calculations associated with the DIS-block algorithm, that is the calculations of ν_k, λ_k, $k = 1, 2, \ldots, n$ and $d^{(s)}$, take minimal computational time as they are all based on analytic expressions (see section 4.3.2.1). All things considered, for all instances that the two algorithms were run in this thesis, runtime was found to be roughly equal and therefore their performance is compared solely by examining their mixing properties.

4.3.2.3 Visualizing the movement of the sampler

As mentioned in section 4.3.2.1, the DIS-block algorithm was developed to address the limitations of the standard-block algorithm. To highlight how these limitations are addressed, the present section provides a visual illustration on how the DIS-block sampler moves around the target space and compares its movement with that of the standard-block sampler. Recall that, a procedure that provided a visual overview on the movement of the standard-block sampler, was described and conducted in section 4.3.1.3. More specifically, after applying the standard-block algorithm on an example dataset, figures 4.13 and 4.14 were used to illustrate the nature and the properties of the sampler. The present section, repeats this procedure for the DIS-block algorithm, i.e. it applies the DIS-block algorithm on the same example dataset and repeats figures 4.13 and 4.14, by including in addition the corresponding information from the run of the DIS-block algorithm.

Figure 4.15 shows the acceptance and inadmissibility proportions, versus block step size, for both algorithms, based on the batch of burn-in iterations used to specify the optimal block step size. Recall from section 4.3.1.3, that for the standard-block algorithm, the main observation was that the acceptance (inadmissibility) proportion reduces (increases) quite rapidly, as the block step size increases. Although, as expected, the acceptance (inadmissibility) proportion reduces (increases), as the block step size increases, for the DIS-block algorithm as well, the rate at which this happens is much lower. More specifically, whereas for the standard-block algorithm, the acceptance proportion becomes essentially 0 (smaller than 0.001) for block step sizes larger than 71, for the DIS-block algorithm, the acceptance proportion does not drop below 0.07, even for moves where all infection times are attempted to be updated, i.e. moves of maximum block step size (equal to the number of infections $n = 448$). Also, as already mentioned in section 4.3.1.3, for the standard-block algorithm, mixing is optimized for block step size values around 12, whereas the corresponding value for

the DIS-block algorithm is 271. These observations are illustrative of the ability of the DIS-block algorithm to perform moves of much larger block step size, compared to the standard-block algorithm. What should be taken into account though, when conducting such comparisons, is that the parameter λ' of the standard-block proposal is itself random (see Algorithm 19), unlike the parameters of the DIS-block proposal which are not random, and therefore it should be expected that the area of the proposed region of the standard-block algorithm will be larger compared to that of the DIS-block algorithm, for a given block step size (see figure 4.16 below for a visual appreciation).

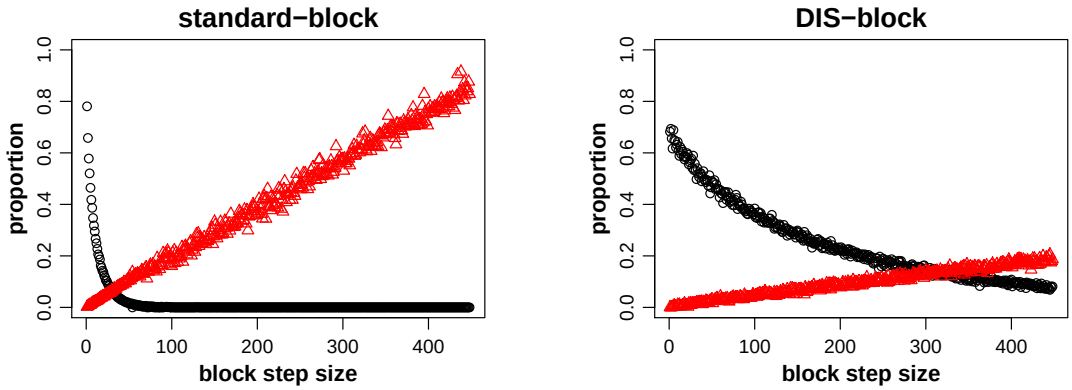


Figure 4.15: Acceptance proportion (black circles) and inadmissibility proportion (red triangles) for the block update step, against block step size. Left column corresponds to the standard-block MCMC algorithm and right column to the DIS-block MCMC algorithm. The dataset is generated from a Gamma-HM model ($N = 500$, $R_0 = 2.5$, $\nu = 5$) and the number of infections is $n = 448$.

Figure 4.16 presents the bivariate posterior density plots of the vector (A, B, C), having imposed the corresponding bivariate proposal density plots, of both samplers, for three different block step sizes, 15, 100 and 250 (the procedure to produce these plots was described in section 4.3.2.1). What is evident in figure 4.16, is that the DIS-block sampler, unlike the standard-block sampler, does not struggle to move around the target space for larger block step sizes. More specifically, reflected in the plots (see the bivariate density plots involving the term B) is the dependent

manner according to which the DIS-block sampler performs the exploration of the target space, that is, by centering itself around the current value of B and proposing moves around it (see the discussions about $d^{(s)}$ in section 4.3.2.1). In this regard, the two samplers are somewhat similar, as the standard-block sampler also tries to centre itself around the current value of B and propose moves around it (see the second remark in section 4.3.1.3 and the relevant bivariate density plots in figure 4.16). This common feature results in both samplers proposing moves that are typically within the support of the posterior distribution, with respect to B, for all block step sizes. However, as the block step sizes increases, it is evident from figure 4.16, that from the two proposal distributions of (A, B, C), that associated with the DIS-block sampler and that associated with the standard-block sampler, only the former is similar to the corresponding posterior distribution; in the sense that, typically, proposed values of (A, B, C), under the DIS-block proposal, are supported under the posterior distribution whereas proposed values of (A, B, C), under the standard-block proposal, are not. What makes the two proposals, behave so differently in this regard, is the effect of the individual-specific parameters, ν_k and λ_k, $k = 1, 2, \ldots, n$, of the DIS-block proposal (see the discussions about ν_k and λ_k in section 4.3.2.1).

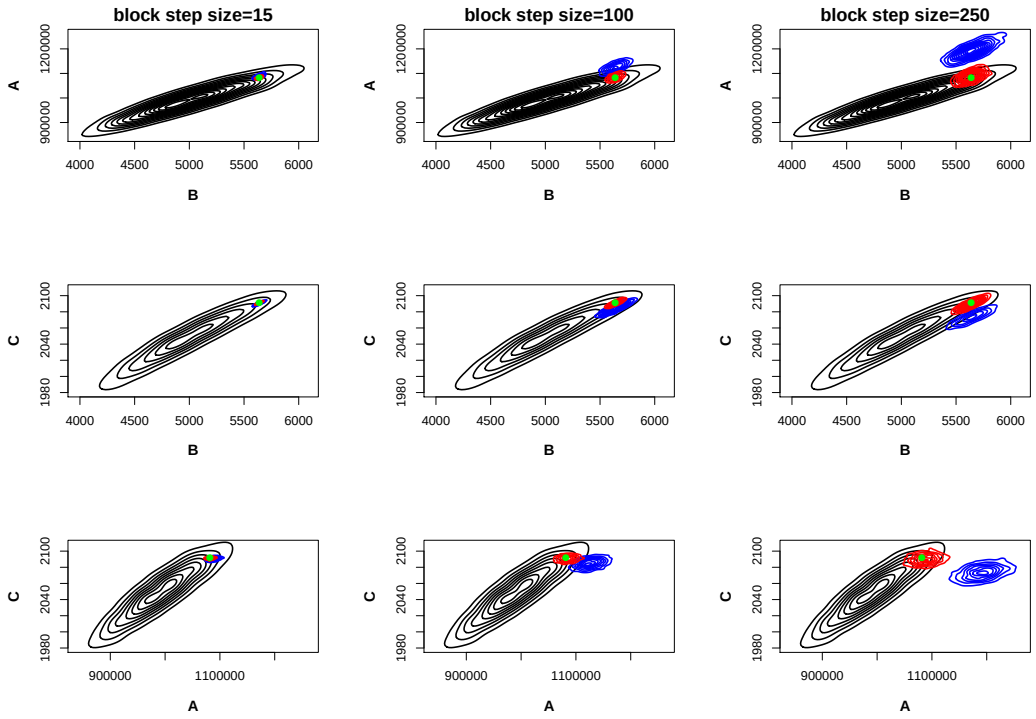


Figure 4.16: Bivariate target posterior densities (black, solid contours), bivariate standard-block proposal densities (blue, solid contours) and bivariate DIS-block proposal densities (red, solid contours), for the vector (A, B, C). Imposed (green, circle) is the current state. Columns (left to right) correspond to block step size values of 15, 100 and 250, respectively. The dataset is generated from a Gamma-HM model ($N = 500$, $R_0 = 2.5$, $\nu = 5$) and the number of infections is $n = 448$.

The above suggest that the DIS-block MCMC algorithm might be more efficient than the standard-block MCMC algorithm. To examine this speculation a simulation study is conducted, referred to as simulation study G, where the performance of the two algorithms is more formally quantified and compared. Before proceeding to simulation study G, section 4.3.2 concludes by making some interesting remarks related to the DIS-block algorithm.

4.3.2.4 Remarks

The feature of parameter reduction carries particular importance in the DIS-block algorithm. Specifically, in the context of the DIS-block algorithm, the biggest benefit from integrating out β and λ, does not come from the fact that the target space becomes smaller (although any such related benefit is still welcome, no matter how small) but from the fact that it naturally suits with the way that the proposal parameters ν_k and λ_k, $k = 1, 2, \ldots, n$, are specified. As described in section 4.3.2.1, ν_k and λ_k, $k = 1, 2, \ldots, n$, are specified by fitting a Gamma(ν_k, λ_k) distribution to a burn-in sample of infectious periods or equivalently of infection times, i.e. specified according to a sample that is roughly (in the sense that it is a burn-in sample) from the posterior density $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu)$. The intention of specifying these parameters in such a way is to make the proposal density similar to its associated target density. The subtle point to note here is that, this specification implicitly benefits from the fact that, when β and λ are integrated out, the associated target density is $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu)$ (see the first paragraph of section 4.3.2.1), which is the same density as the density according to which ν_k, λ_k, $k = 1, 2, \ldots, n$, are specified. That is to say, that the proposal density is targeting the same density as the one it is made to resemble. Notice though, that if β and λ were not integrated out, this would not be the case, as the associated target density would then be $\pi(\alpha, i_\alpha, \boldsymbol{i} \mid \boldsymbol{r}, \nu, \beta^{(s)}, \lambda^{(s)})$, where $\beta^{(s)}$ and $\lambda^{(s)}$, the current values of β and λ in the algorithm, and therefore the specification of ν_k, λ_k, $k = 1, 2, \ldots, n$, would be in this sense suboptimal. This is also supported by investigations based on simulated data.

The DIS-block algorithm relies on the combined effect of the individual-specific parameters ν_k, λ_k, $k = 1, 2, \ldots, n$, and the dependence inducing parameter $d^{(s)}$. For example, if one considers a proposal scheme, say P_1, that makes use only of $d^{(s)}$ and not of ν_k, λ_k, $k = 1, 2, \ldots, n$ (i.e. the scheme that proposes infection times as $r_k - i_k^* \sim$ Gamma$(\nu, d^{(s)})$, $k = 1, 2, \ldots, n$, where $d^{(s)}$ such that $\mathrm{E}(B^*) = B^{(s)}$) the

behaviour of the sampler will be similar to that of the standard-block sampler. To see this, notice that the P_1 proposal is the same as the standard-block Gamma(ν, λ') proposal, only that λ' is replaced by $d^{(s)}$, and consider that the role of $d^{(s)}$ in P_1 is similar to that of λ' in the standard-block scheme (see the second remark in section 4.3.1.3). A visual appreciation of this, using the example dataset of section 4.3.1.3, is given by figure A.2 in the Appendix. Alternatively, if one considers a proposal scheme, say P_2, that makes use only of ν_k, λ_k, $k = 1, 2, \ldots, n$ and not of $d^{(s)}$ (i.e. the scheme proposing infection times as $r_k - i_k^* \sim$ Gamma(ν_k, λ_k), $k = 1, 2, \ldots, n$, where ν_k and λ_k are specified as in section 4.3.2.1), although the effect of ν_k, λ_k, $k = 1, 2, \ldots, n$, might allow the proposal distribution to reproduce the structure of the posterior distribution, the sampler could face issues of ever visiting or leaving the tails of the target distribution, for large block step sizes. To see how this might happen, consider the case that the block step size m is set at its maximum value n. Then, since there is no dependence inducing parameter in the Gamma(ν_k, λ_k) proposal and since all infection times are attempted to be updated (see the first remark in section 4.3.1.3), Gamma(ν_k, λ_k) is strictly an independent proposal and unless it is heavier-tailed than the target distribution it carries the risk of never proposing moves to the tails of the target distribution or, if it does and the sampler moves there, of never leaving (see the discussion about independent proposals in section 1.3.2.4). A visual appreciation of this, using the example dataset of section 4.3.1.3, is provided by figure A.3 in the Appendix.

4.3.3 Simulation study G

4.3.3.1 Purpose

Simulation study G compares the performance of the standard-block and the DIS-block MCMC algorithms on datasets that are generated under different simulation scenarios. The structure of simulation study G is very similar to that of simulation study F, where again MCMC algorithms for the Gamma-HM model were compared;

the difference is that simulation study F compared 1-dimensional update algorithms (see section 4.2.4) while G compares block update algorithms. As in simulation study F, it is of interest to investigate how the comparison of the algorithms might be affected by the scale of the outbreak, quantified by N, the severity of the outbreak, quantified by R_0, and the value of the shape parameter ν, of the assumed Gamma(ν, λ) distribution of the infectious periods.

4.3.3.2 Simulation and run conditions

The simulation conditions are the same as in simulation study F, except that in the present simulation study an additional value of N is considered, namely $N = 500$. Specifically, for each combination of selected values of the parameters R_0, ν and N, one dataset is simulated from the Gamma-HM model under the condition that the final size is equal to the (major outbreak) mode of the final size with respect to the sampling distribution. As previously mentioned, this condition ensures that, in a sense, the datasets are representative of their corresponding simulation scenario. The values for R_0 are 1.5 and 2.5, for ν are 2 and 5 and for N are 200, 500 and 1000, resulting in twelve simulation scenarios in total, one for each distinct trio of selected values of (R_0, ν, N). The mean infectious period $\mathrm{E}(T_D)$ is set to be 10 in all instances, specifying λ to be $\lambda = 0.2$ and $\lambda = 0.5$, for the case that $\nu = 2$ and $\nu = 5$, respectively.

The Gamma-HM model is fitted to each simulated dataset, using both of the MCMC algorithms under comparison, namely the standard-block MCMC algorithm and the DIS-block MCMC algorithm. The run conditions for the two algorithms are described in detail in sections 4.3.1.2 and 4.3.2.2, respectively. It is worth reiterating that (see section 4.3.2.2) the run conditions of the two algorithms are set to be as similar as possible, and therefore, in all instances, runtime is roughly equal and algorithm performance is compared by focusing only at mixing properties and not runtime.

4.3.3.3 Results

The comparison of the algorithms is conducted in the same way as for the previous two simulation studies of this chapter, E and F (see sections 4.2.3.4 and 4.2.4.2). First, before looking at the results, both algorithms were checked for evidence of non-stationarity (see the part regarding stationarity in section 1.3.2.3) by visually inspecting MCMC trace plots and by assessing whether the posterior densities of the two algorithms appeared to be the same. For all datasets, both algorithms appeared to have converged to the (same) desired posterior distribution. For reference, the relevant plots are given for one of the datasets, in figure A.24 in the Appendix. Then, following Xiang and Neal (2014), mixing and efficiency were assessed with respect to $B = \sum_{k=1}^{n}(r_k - i_k)$. This was done by producing ACF plots and by calculating the effective sample size associated with the MCMC sample of B, as described in section 1.3.2.3.

Table 4.3 and figure 4.17, respectively, give the effective sample sizes and the ACF plots, for the two compared algorithms, for each of the twelve datasets of the simulation study. Looking at table 4.1 and figure 4.9, one can see that the DIS-block algorithm, to a lesser or greater extent, has better mixing compared to the standard-block algorithm, for all considered datasets. Specifically, considering all twelve datasets, the effective sample size ratio of the DIS-block algorithm over the standard-block algorithm, ranges from 1.41 to 6.57 and it is equal to 3.35 on average, i.e. the DIS-block algorithm is from 1.41 times up to 6.57 times more efficient than the standard-block algorithm, and 3.35 times on average.

To examine the effect of each one of the factors, N, R_0 and ν, one can compare the effective sample size ratio (of the DIS-block algorithm over the standard-block algorithm) over the different values of the factor. Regarding N, the average (minimum, maximum) effective sample size ratio is 3.02 (1.50, 5.71), 3.40 (1.79, 6)

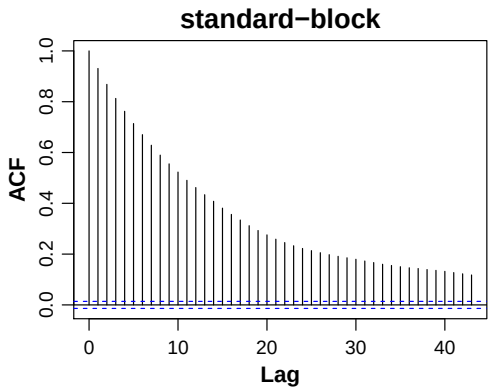
and 3.64 (1.41, 6.57) for datasets for which $N = 200$, $N = 500$ and $N = 1000$, respectively. Although there appears to be a slight increase with N, it is not enough to suggest that the relative quality of mixing of the two algorithms changes with N.

As far as R_0, the average (minimum, maximum) effective sample size ratio is 1.91 (1.41, 2.74) and 4.80 (3.19, 6.57) for datasets such that $R_0 = 1.5$ and $R_0 = 2.5$, respectively. These values indicate that the advantage of the DIS-block algorithm, over the standard-block algorithm, is larger for $R_0 = 2.5$ compared to $R_0 = 1.5$. A possible explanation for this can be given by considering the effect that R_0 has on the posterior distribution of the infectious periods of individuals, where the larger the value of R_0 the higher the deviation from homogeneity among individuals (see section 4.2.1 and in particular the discussion regarding what happens when R_0 becomes very large). Since the DIS-block proposal scheme allows for nonhomogeneity among individuals and the standard-block proposal scheme does not, it seems reasonable that the advantage of the DIS-block algorithm is larger for $R_0 = 2.5$ compared to $R_0 = 1.5$. Recall from simulation studies E and F (see sections 4.2.3.4 and 4.2.4.2) that similar observations were made when comparing the IS-1d algorithm (whose proposal scheme is also based on individual-specific parameters, similar to that of the DIS-block algorithm) to the standard-1d algorithm (whose proposal scheme is essentially the 1-dimensional version of that of the standard-block algorithm and is based on parameters that are the same over individuals).

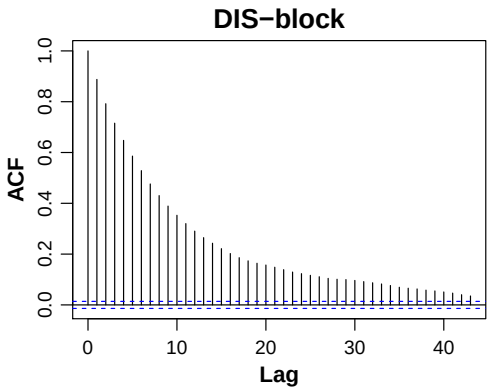
Regarding the effect of ν, the average (minimum, maximum) effective sample size ratio is 2.65 (1.41, 3.83) and 4.05 (1.50, 6.57) for datasets for which $\nu = 2$ and $\nu = 5$, respectively. Based on these values, it appears that the advantage of the DIS-block algorithm increases for larger values of ν, however the evidence for this is not as decisive as it is for R_0.

Table 4.3: Effective sample size for $B = \sum_{k=1}^{n}(r_k - i_k)$, for the two compared MCMC algorithms, standard-block and DIS-block, for each of the twelve datasets of simulation study G. The simulation and run conditions are described in section 4.3.3.2.

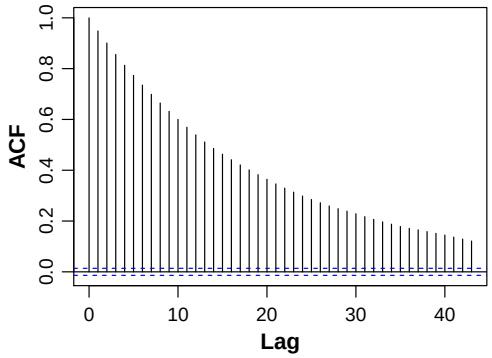
	Algorithm	
	standard-block	**DIS-block**
Dataset 1 ($R_0 = 1.5, \nu = 2, N = 200$)	689	1165
Dataset 2 ($R_0 = 1.5, \nu = 2, N = 500$)	546	1265
Dataset 3 ($R_0 = 1.5, \nu = 2, N = 1000$)	638	900
Dataset 4 ($R_0 = 1.5, \nu = 5, N = 200$)	439	660
Dataset 5 ($R_0 = 1.5, \nu = 5, N = 500$)	290	518
Dataset 6 ($R_0 = 1.5, \nu = 5, N = 1000$)	371	1016
Dataset 7 ($R_0 = 2.5, \nu = 2, N = 200$)	631	2011
Dataset 8 ($R_0 = 2.5, \nu = 2, N = 500$)	780	2707
Dataset 9 ($R_0 = 2.5, \nu = 2, N = 1000$)	848	3248
Dataset 10 ($R_0 = 2.5, \nu = 5, N = 200$)	446	2547
Dataset 11 ($R_0 = 2.5, \nu = 5, N = 500$)	620	3725
Dataset 12 ($R_0 = 2.5, \nu = 5, N = 1000$)	749	4918



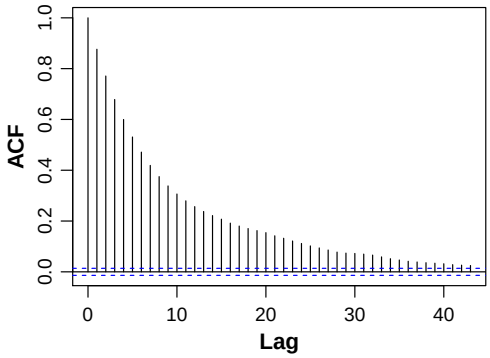
(a) Dataset 1 ($R_0 = 1.5, \nu = 2, N = 200$)



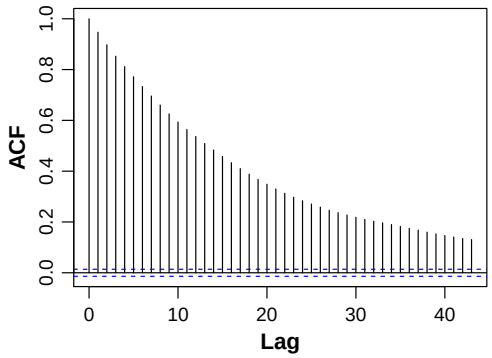
(b) Dataset 1 ($R_0 = 1.5, \nu = 2, N = 200$)



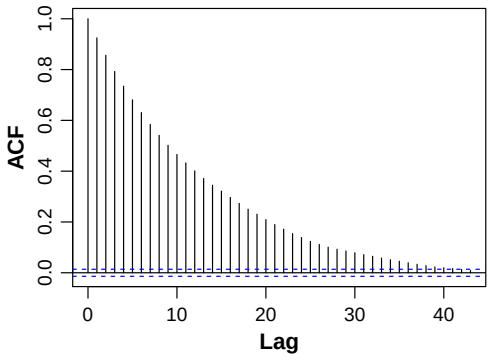
(c) Dataset 2 ($R_0 = 1.5, \nu = 2, N = 500$)



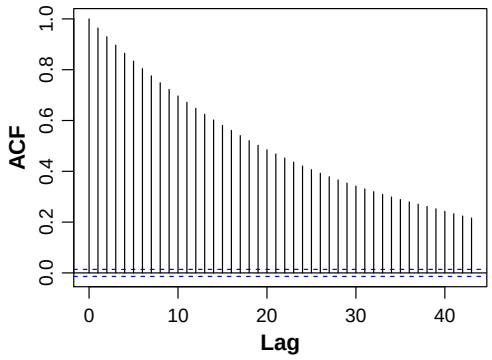
(d) Dataset 2 ($R_0 = 1.5, \nu = 2, N = 500$)



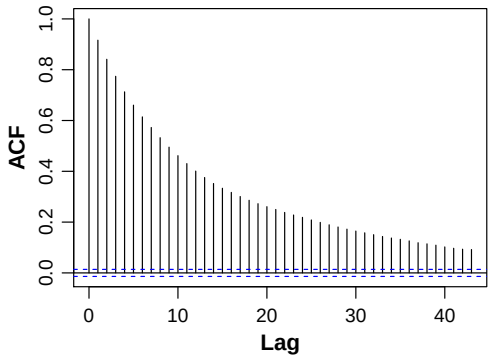
(e) Dataset 3 ($R_0 = 1.5, \nu = 2, N = 1000$)



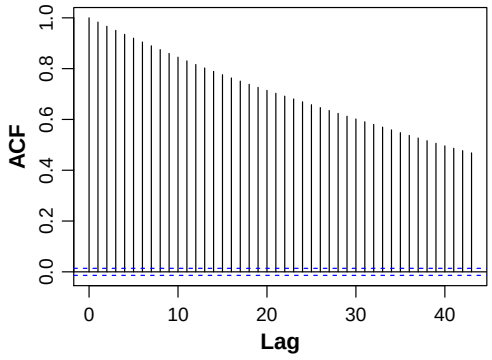
(f) Dataset 3 ($R_0 = 1.5, \nu = 2, N = 1000$)



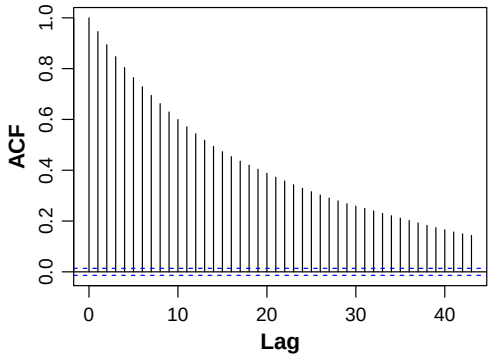
(g) Dataset 4 ($R_0 = 1.5, \nu = 5, N = 200$)



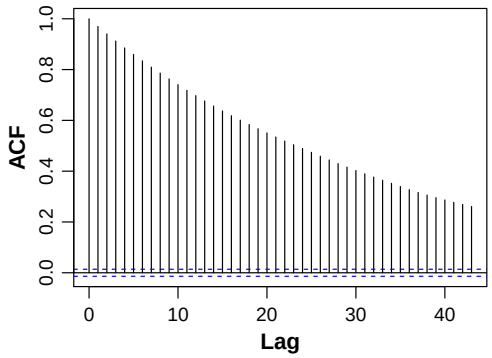
(h) Dataset 4 ($R_0 = 1.5, \nu = 5, N = 200$)



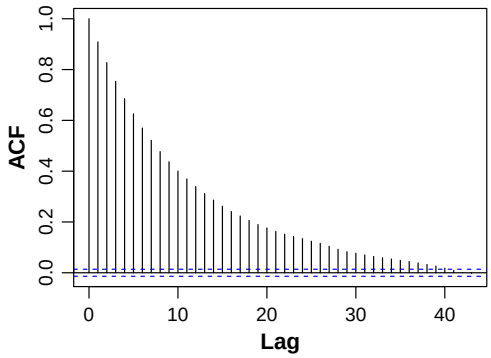
(i) Dataset 5 ($R_0 = 1.5, \nu = 5, N = 500$)



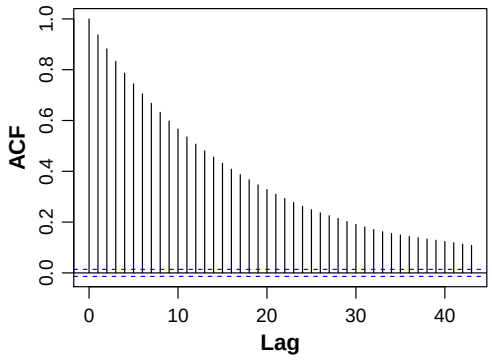
(j) Dataset 5 ($R_0 = 1.5, \nu = 5, N = 500$)



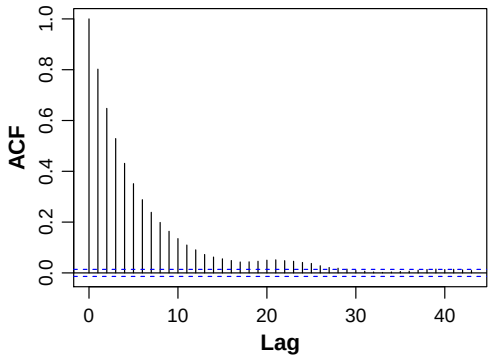
(k) Dataset 6 ($R_0 = 1.5, \nu = 5, N = 1000$)



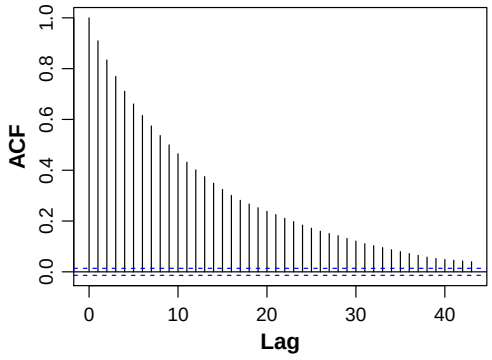
(l) Dataset 6 ($R_0 = 1.5, \nu = 5, N = 1000$)



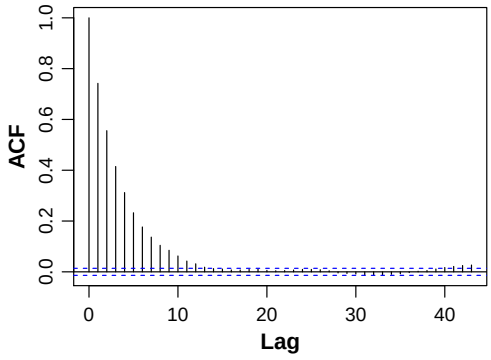
(m) Dataset 7 ($R_0 = 2.5, \nu = 2, N = 200$)



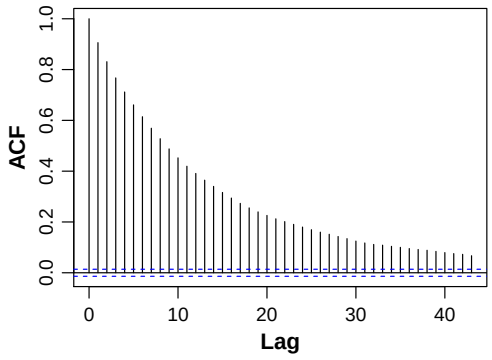
(n) Dataset 7 ($R_0 = 2.5, \nu = 2, N = 200$)



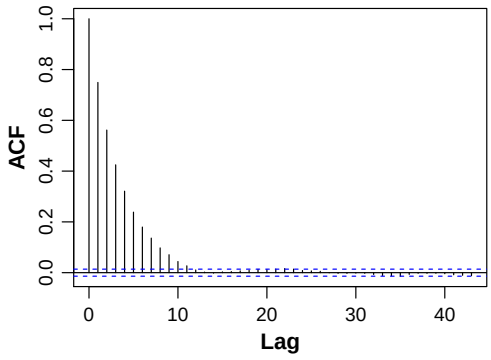
(o) Dataset 8 ($R_0 = 2.5, \nu = 2, N = 500$)



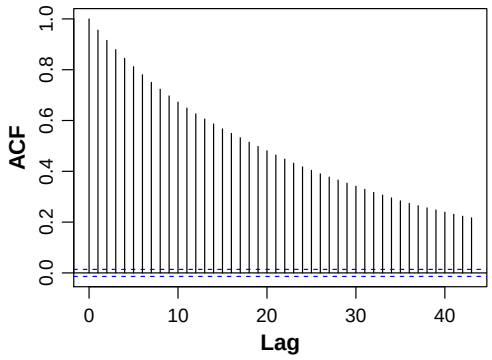
(p) Dataset 8 ($R_0 = 2.5, \nu = 2, N = 500$)



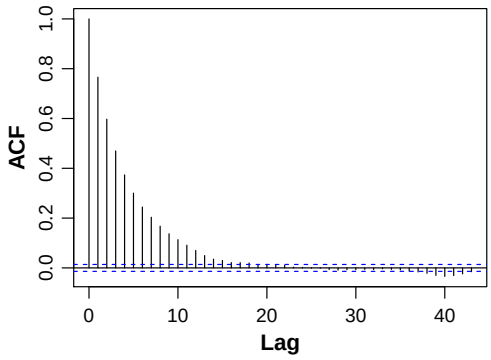
(q) Dataset 9 ($R_0 = 2.5, \nu = 2, N = 1000$)



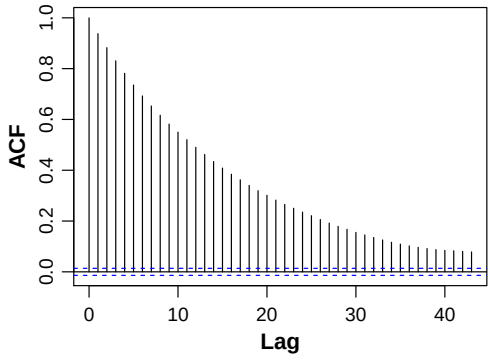
(r) Dataset 9 ($R_0 = 2.5, \nu = 2, N = 1000$)



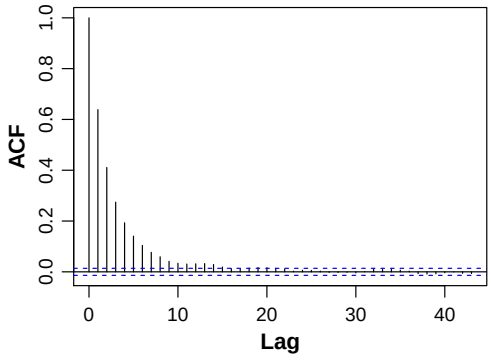
(s) Dataset 10 ($R_0 = 2.5, \nu = 5, N = 200$)



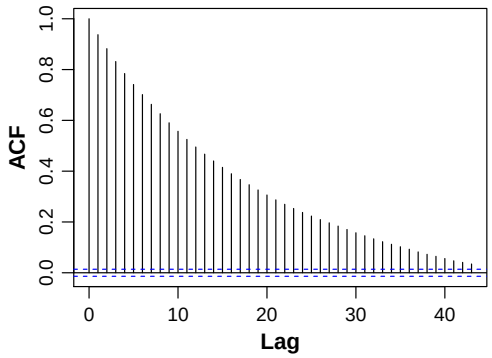
(t) Dataset 10 ($R_0 = 2.5, \nu = 5, N = 200$)



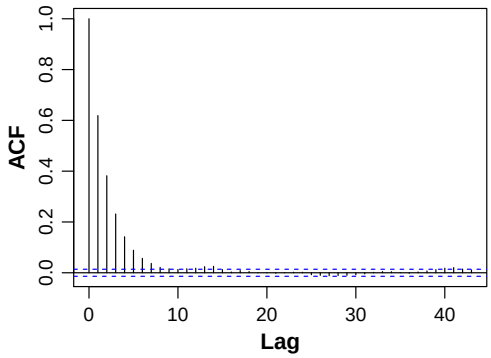
(u) Dataset 11 ($R_0 = 2.5, \nu = 5, N = 500$)



(v) Dataset 11 ($R_0 = 2.5, \nu = 5, N = 500$)



(w) Dataset 12 ($R_0 = 2.5, \nu = 5, N = 1000$)



(x) Dataset 12 ($R_0 = 2.5, \nu = 5, N = 1000$)

Figure 4.17: ACF plots for $B = \sum_{k=1}^{n}(r_k - i_k)$, for each of the twelve datasets of simulation study G. The simulation and run conditions are described in section 4.3.3.2. Left column corresponds to the standard-block MCMC algorithm and right column corresponds to the DIS-block MCMC algorithm.

4.3.3.4 Conclusions

The conclusions from simulation study G are summarized as follows.

- Under all scenarios of the simulation study, the DIS-block algorithm has better mixing compared to the standard-block algorithm. Specifically, for the considered datasets of the simulation study, the DIS-block algorithm is from 1.41 to 6.57 times more efficient than the standard-block algorithm, and 3.35 times on average.

- The scale of the outbreak, quantified by N, does not appear to have an evident effect on the comparative performance of the two algorithms suggesting that the advantage of the DIS-block algorithm would be similar in small and large-scale outbreaks.

- The severity of the outbreak, quantified by R_0, appears to have an effect, suggesting that the advantage of the DIS-block algorithm, over the standard-block algorithm, would be greater for larger values of R_0.

- The shape parameter, of the infectious period distribution ν, appears to have an effect, in the sense that the advantage of the DIS-block algorithm appears slightly increased for larger values of ν, but the evidence for this is not conclusive.

Overall, the DIS-block algorithm exhibits considerable improvement in mixing, compared to the standard-block algorithm. Taking into account the fact that the computational cost associated with the two algorithms is very similar (see the last paragraph in section 4.3.2.2), the DIS-block algorithm appears to be a much more preferable choice in practice.

4.4 Discussion

4.4.1 Addressing chapter aims

This chapter developed two novel MCMC algorithms based on newly defined proposal mechanisms for the infection component. The first algorithm, referred to as the IS-1d algorithm, is based on an 1-dimensional update step of the infection component whereas the second algorithm, referred to as the DIS-block algorithm, conducts a block update step of the infection component. A key idea behind both algorithms, is the use of individual-specific parameters in the proposal distributions for the infection times. These individual-specific parameters, allow the proposal distributions the flexibility required to capture the pattern sometimes exhibited in the target distribution, where the infectious periods of individuals are not homogeneous (see section 4.2.1).

Extensive simulation studies suggested that both of these algorithms are more efficient than their analogue algorithms, found in the currently available published literature. That is, the IS-1d algorithm is more efficient compared to the currently existing 1-dimensional algorithms, and, the DIS-block algorithm is more efficient compared to the currently existing block update algorithms. A particularly appealing fact, is that the computational cost associated with the algorithms developed is very similar to that of their analogue existing algorithms, i.e. the improvement in mixing does not come with any additional computational cost. From a practical standpoint, more impactful are the results regarding the DIS-block algorithm, since in general block update steps are more efficient than their 1-dimensional counterparts. According to simulation study G, the improvement in mixing offered by the DIS-block algorithm is substantial, with the DIS-block algorithm on occasions being up to 6.5 times more efficient than the standard-block algorithm.

4.4.2 Limitations

All MCMC algorithms in this chapter, when relevant (that is when the infectious period distribution was assumed to be Gamma(ν, λ)), assumed that the shape parameter ν was known, as opposed to it being an unknown parameter to be estimated from the data. This is a limitation, because in a practical situation a user would ideally want to allow all parameters to be estimated from the data, without worrying about specifying values for unknown quantities. As mentioned in section 4.2.4.1, the decision to treat ν as known was taken in order to avoid mixing issues that are induced in the instance that ν is unknown. Note however, that this decision was taken consistently for all compared algorithms and therefore it is unlikely that any of the algorithms gained an advantage from it, i.e. it is unlikely that treating ν as unknown would affect the comparative performance of the algorithms.

4.4.3 General remarks

The approach followed throughout this chapter was to compare like with like, namely 1d-update algorithms to other 1d-update algorithms and block-update algorithms to other block-update algorithms. This is because block update algorithms have inherently better mixing than their 1-dimensional counterparts; the latter only update one infection time per update step, whereas the former update many. A good way to put this into perspective is to look at the plots that visualize the movement of a sampler, such as figure 4.14, and notice how small the area of the proposed move is, when the block step size is 15, compared to 100 or 250; using this as reference, it is not hard to appreciate how small the area of the proposed move is when the block step size is 1. Nonetheless, it should be noted that, 1-dimensional update algorithms could benefit from efficient coding, which updates the value of the likelihood at each iteration rather than recalculating it from scratch, and can therefore become computationally cheaper. However, even when taking into account the reduced runtime, 1-dimensional update algorithms are still generally less efficient

than block update algorithms.

The algorithms developed in this chapter use burn-in iterations to tune either parameters in the proposal distributions or the block step size or both. In practice, it is most sensible, to assess the quality of the burn-in sample in question before using it to conduct such tuning. The general idea is that the tuning will be efficient, as long as the associated burn-in sample provides a somewhat adequate representation of the posterior distribution. To this end, in order to allow the chain to move away from its initial state and closer to stationarity, it could be safer to discard a number of the first burn-in iterations of the sample in question, or, equivalently, run an additional batch of burn-in iteration and discard it, prior to running the burn-in sample iterations that will be used to conduct tuning.

4.4.4 Further work

Recall from section 4.3.2.1, that the procedure to propose infections in the DIS-block algorithm can be seen as having two steps. First, infectious periods are drawn from a Gamma(ν_k, λ_k) distribution, independently, for each $k \in \boldsymbol{b}$, where $\boldsymbol{b}$ the chosen set of individuals for which infection times are proposed. Second, the drawn infectious periods are scaled by a factor of $1/d^{(s)}$, where $d^{(s)}$ is such that $\mathrm{E}(B^*) = B^{(s)}$. Note that, in this procedure, the infectious periods of individuals not in $\boldsymbol{b}$ remain fixed at their current values. An idea worth pursuing, would be to modify the second step of the above procedure and to scale, by a factor of $1/d^{(s)}$ (where $d^{(s)}$ would again be specified so that $\mathrm{E}(B^*) = B^{(s)}$), the infectious periods of all individuals, rather than only those in $\boldsymbol{b}$. It would be interesting to investigate if such a modification could improve the ability of the proposal distribution to capture the pattern sometimes exhibited in the target distribution, where the infectious periods of individuals are not homogeneous (see section 4.2.1).

Another interesting approach would be to make $d^{(s)}$ a random variable, as opposed to being deterministically specified at each iteration s. For example, one could set $d^{(s)}$ to have a Normal distribution, $N(\boldsymbol{x}^{(s)}, \sigma^2)$, with mean $\boldsymbol{x}^{(s)}$ and some variance σ^2, where $\boldsymbol{x}^{(s)}$ would be specified, at each iteration s, so that $\mathrm{E}(B^*) = B^{(s)}$ and σ^2 would be a tuning parameter. The motivation behind such an approach, would be to gain more control over the area of the proposed move of the sampler, perhaps by tuning σ^2 in an optimal sense, in an attempt to further improve its efficiency.

Chapter 5

Discussion

5.1 Addressing thesis aims

This thesis successfully developed novel methods for both model assessment and inference for stochastic epidemic models, based on partially observed data. The development of all methods took into consideration the peculiarities of the epidemic setting, where data are partially observed and highly correlated, and epidemic outbreaks are realized only once.

Chapter 2 developed two new model assessment methods, based on the posterior predictive distribution of removal curves, namely the distance method and the position-time method. The distance method (see section 2.5), assesses the plausibility of the observed removal curve, under its posterior predictive distribution, by calculating distances between removal curves. The position-time method (see section 2.6), conducts this assessment at a sequence of suitably selected time points. Particularly appealing is the fact that, both methods provide visual and quantitative assessment of model fit, with easily interpretable outputs. For example, the position-time method allows for summaries (over time) of essentially any interesting event (with respect to the posterior predictive distribution) to be calculated, such as

the proportion of time that the observed removal curve in question spends in any (inverse) quantile interval of its posterior predictive distribution. Also appealing, is the fact that the two methods are different in the way that they utilize the information from the removal curves, and thus they provide complementary types of assessment regarding the fit of a model. The performance of both the distance and the position-time methods highly benefits from the use of time shifting (see section 2.4), an application that successfully removes the undesired noise exhibited at the initial stages of an epidemic (where epidemic processes typically behave like branching processes), and allows for a more informative comparison between removal curves. A computational appeal of the methods is that, unlike most of the currently existing methods (see e.g. Alharthi (2016)), they do not require the creation of replications that have exactly the same final size as the observed; a procedure which is computationally intensive to perform. More specifically, the methods can be applied using matched replications (i.e. replications having exactly the same final size as the observed) or unmatched (major outbreak) replications (i.e. replications that fall in the major outbreak part of the posterior predictive distribution of the final size). This is made possible by an automated procedure (see section 2.3) which classifies each replication from the posterior predictive distribution as a minor or a major outbreak. Extensive simulation studies suggested that both methods can successfully assess important aspects of epidemic models, namely the infectious period assumption (see section 2.7.1) and the infection rate form assumption (see section 2.8.1). Particularly desirable, from a practical point of view, is that the simulation studies suggested that the higher the scale of the outbreak the better the performance of the methods.

Chapter 3 developed a new classical hypothesis test for assessing the population mixing assumption of epidemic models, in the case that population structure information is available. The test is based on household labels of individuals and utilizes the idea that, if there is a two-level-mixing effect (i.e. higher infectivity within households than between households), then events of individuals belonging to

the same household should happen closer in time rather than further apart. What makes the development and implementation of the test possible, is the fact that, under the assumption of a homogeneously mixing population, the discrete random vector of household labels has a known sampling distribution that is independent of any model parameters. The test has an easy ordinal interpretation, where the lower the observed value of the test statistic and its corresponding p-value are, the more the evidence against the hypothesis of homogeneous mixing, and in favour of the hypothesis of two-level-mixing. The performance of the test was examined via an extensive simulation study (see section 3.3) and by applying it to real data (see section 3.4). In both cases, the test exhibited excellent performance; when applied to simulated data (generated under various simulation scenarios) the test demonstrated a systematic ability to succesfully assess the population mixing assumption and, when applied to real data, the test yielded a conclusion that was in line with previous analyses in the literature. Very appealing, from a practical standpoint, is the fact that the test is computationally cheap and simple to perform, as it does not involve any model fitting. What this suggests is that, in practice, it would be very useful to conduct the test before any models are fitted to data, and use the result of the test as a guide in choosing a suitable model for the data in question.

Chapter 4 developed two new MCMC algorithms, namely the IS-1d MCMC algorithm (see section 4.2.2) and the DIS-block MCMC algorithm (see section 4.3.2), by considering new proposal mechanisms for the update step of the infection component. As the names suggest, the IS-1d MCMC algorithm is based on an 1-dimensional update step of the infection component, while the DIS-block algorithm on a block update step. Both algorithms benefit from the use of individual-specific parameters in the proposal distributions for the infection times. By using individual-specific parameters, the proposal distributions gain the ability to mimic the patterns of nonhomogeneity, with respect to the infectious periods of individuals, sometimes exhibited in the target distribution. The performance of both algorithms, as far

as mixing and efficiency is concerned, was examined via extensive simulation studies (see sections 4.2.3, 4.2.4 and 4.3.3), where each of the two algorithms was compared to its analogue currently existing MCMC algorithm. In all comparisons, the newly developed algorithm exhibited improvement in mixing compared to its existing counterpart. More noteworthy were the results regarding the comparison of block update algorithms, since block update steps are in general more efficient than their 1-dimensional counterparts. As suggested by the relevant simulation study, the DIS-block algorithm can offer a substantial improvement in mixing compared to the current optimally performing block update algorithm; for the considered datasets of the simulation study, the effective sample size ratio of the DIS-block algorithm over its comparator, ranged from 1.41 to 6.57 and was 3.35 on average. What must be taken into account is that this improvement in mixing does not come with any additional computational cost, since the cost associated with running the DIS-block algorithm is very similar to that of its comparator.

5.2 Limitations

Limitations specifically related to the work of each of chapters 2, 3 and 4 have already been discussed at the relevant sections of the chapters (see sections 2.10.2, 3.5.2 and 4.4.2, respectively). What might be considered as a general limitation of this thesis is that the methods developed were mostly applied on simulated rather than real data. Although, there is no doubt that there is a benefit in illustrating methods on real datasets, the fact that the methods were mostly applied on simulated data should not undermine their practical utility. For example, in the context of model assessment, the use of simulated data served the purpose of determining whether an assumed model was correctly specified or not and therefore allowed for effective examination of the performance of the methods; in the sense that a model assessment method works well if it detects misspecification when there is, or, does not detect misspecification when there is not. This type of examination is hardly as effective

when using real data, since in that case the true data generating process is typically unknown and there is no natural way of telling whether a model assessment method outputs what it is supposed to or not.

5.3 Contribution

The main contribution of this thesis is the development of novel methods, for both model assessment and inference of stochastic epidemics models, that can readily be applied by practitioners in the field. Regarding model assessment, the methods developed, offer a much needed immediate addition to the rather sparse currently existing toolkit. As far as inference, the block update algorithm developed, provides substantial improvement in efficiency, compared to the currently existing algorithms. A less immediate contribution, but perhaps equally important, is that, in the process of developing these methods, it was revealed how and why the peculiarities of the epidemic setting should be a matter of careful consideration, when designing new methodology. What was also revealed, that was not clear before, was how and why some of the currently existing methods underperform. The hope is that knowledge of this kind can help guide the development of new methods in the future.

Bibliography

Bibliography

Adrakey, H. K., Streftaris, G., Cunniffe, N. J., Gottwald, T. R., Gilligan, C. A., and Gibson, G. J. (2017). Evidence-based controls for epidemics using spatio-temporal stochastic models in a Bayesian framework. *Journal of The Royal Society Interface*, 14(136):20170386. 1

Alharthi, M. (2016). *Bayesian Model Assessment for Stochastic Epidemic Models*. PhD thesis, University of Nottingham. 68, 74, 78, 84, 90, 107, 130, 131, 133, 155, 156, 159, 179, 213, 245, 293

Andersson, H. and Britton, T. (2000). *Stochastic Epidemic Models and Their Statistical Analysis*. Springer New York. arXiv:1411.2624. 27, 28, 29, 31, 32, 44, 54, 55, 70, 86

Auranen, K., Arjas, E., Leino, T., and Takala, A. K. (2000). Transmission of pneumococcal carriage in families: A latent Markov process model for binary longitudinal data. *Journal of the American Statistical Association*, 95(452):1044–1053. 77

Bailey, N. T. J. (1975). *The mathematical theory of infectious diseases and its applications*. Griffin, London, 2nd edition. 29, 44, 70, 206

Ball, F. (1986). A unified approach to the distribution of total size and total area under the trajectory of infectives in epidemic models. *Advances in Applied Probability*, 18(2):289–310. 69, 170

Ball, F. and Donnelly, P. (1995). Strong approximations for epidemic models. *Stochastic Processes and their Applications*, 55(1):1–21. 54, 97

Ball, F., Mollison, D., and Scalia-Tomba, G. (1997). Epidemics with two levels of mixing. *The Annals of Applied Probability*, 7(1):46–89. 28, 60, 69

Bartlett, M. S. (1949). Some evolutionary stochastic processes. *Journal of the Royal Statistical Society. Series B (Methodological)*, 11(2):211–229. 44

Becker, N. (1989). *Analysis of Infectious Disease Data (Chapman & Hall/CRC Monographs on Statistics and Applied Probability)*. Chapman and Hall/CRC. 29, 31, 70

Bernardo, J. M. and Smith, A. F. M., editors (1994). *Bayesian Theory*. John Wiley & Sons, Inc. 6

Blum, J. R., Chernoff, H., Rosenblatt, M., and Teicher, H. (1958). Central limit theorems for interchangeable processes. *Canadian Journal of Mathematics*, 10:222–229. 211

Boys, R. J. and Giles, P. R. (2007). Bayesian inference for stochastic epidemic models with time-inhomogeneous removal rates. *Journal of Mathematical Biology*, 55(2):223–247. 74, 207

Britton, T. (1997a). Limit theorems and tests for within family clustering in epidemic models. *Communications in Statistics - Theory and Methods*, 26(4):953–976. 213, 214

Britton, T. (1997b). A test of homogeneity versus a specified heterogeneity in an epidemic model. *Mathematical Biosciences*, 141(2):79–99. 213

Britton, T. (1997c). A test to detect within-family infectivity when the whole epidemic process is observed. *Scandinavian Journal of Statistics*, 24(3):315–330. 213

Britton, T. (1997d). Tests to detect clustering of infected individuals within families. *Biometrics*, 53(1):98. 213

Britton, T., Kypraios, T., and O'Neill, P. D. (2011). Inference for epidemics with three levels of mixing: Methodology and application to a measles outbreak. *Scandinavian Journal of Statistics*, pages no–no. 28, 30

Britton, T. and O'Neill, P. D. (2002). Bayesian inference for stochastic epidemics in populations with random social structure. *Scandinavian Journal of Statistics*, 29(3):375–390. 36, 77

Casella, G. and George, E. I. (1992). Explaining the Gibbs sampler. *The American Statistician*, 46(3):167. 16

Cauchemez, S., Carrat, F., Viboud, C., Valleron, A. J., and Boëlle, P. Y. (2004). A Bayesian MCMC approach to study transmission of influenza: application to household longitudinal data. *Statistics in Medicine*, 23(22):3469–3487. 77

Chernoff, H. and Teicher, H. (1958). A central limit theorem for sums of interchangeable random variables. *The Annals of Mathematical Statistics*, 29(1):118–130. 211

Clancy, D. and O'Neill, P. D. (2008). Bayesian estimation of the basic reproduction number in stochastic epidemic models. *Bayesian Analysis*, 3(4):737–757. 29, 207

Clancy, D., O'Neill, P. D., and Pollett, P. K. (2001). Approximations for the long-term behaviour of an open-population epidemic model. *Methodology And Computing In Applied Probability*, 3(1):75–95. 29

Daley, D. J. and Gani, J. (1999). *Epidemic Modelling*. Cambridge University Press. 27

Demiris, N. and O'Neill, P. D. (2006). Computation of final outcome probabilities for the generalised stochastic epidemic. *Statistics and Computing*, 16(3):309–317. 94

Eichner, M. and Dietz, K. (2003). Transmission potential of smallpox: Estimates based on detailed data from an outbreak. *American Journal of Epidemiology*, 158(2):110–117. 207, 208

Ferguson, N. M., Laydon, D., Nedjati-Gilani, G., Imai, N., Ainslie, K., Baguelin, M., Bhatia, S., Boonyasiria, A., Cucunubá, Z., Cuomo-Dannenburg, G., Dighe, A., Dorigatti, I., Fu, H., Gaythorpe, K., Green, W., Hamlet, A., Hinsley, W., Okell, L. C., van Elsland, S., Thompson, H., Verity, R., Volz, E., Wang, H., Wang, Y., Walker, P. G., Walters, C., Winskill, P., Whittaker, C., Donnelly, C. A., Riley, S., and Ghani, A. C. (2020). Report 9: Impact of non-pharmaceutical interventions (NPIs) to reduce COVID-19 mortality and healthcare demand. Technical report, Imperial College London. 1

Freeman, M. F. and Tukey, J. W. (1950). Transformations related to the angular and the square root. *The Annals of Mathematical Statistics*, 21(4):607–611. 75

Gardner, A., Deardon, R., and Darlington, G. (2011). Goodness-of-fit measures for individual-level models of infectious disease in a Bayesian framework. *Spatial and Spatio-temporal Epidemiology*, 2(4):273–281. 73, 86

Gelfand, A. E. and Smith, A. F. M. (1990). Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association*, 85(410):398–409. 9

Gelman, A. (2013). Two simple examples for understanding posterior p-values whose distributions are far from uniform. *Electronic Journal of Statistics*, 7(0):2595–2602. 26, 73, 83

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2013). *Bayesian Data Analysis*. Chapman & Hall/CRC Texts in Statistical Science. CRC Press, third edition. 14, 15, 23, 24, 25, 26, 75, 84

Gelman, A., Meng, X.-L., and Stern, H. (1996). Posterior predictive assessment of model fitness via realized discrepancies. *Statistica Sinica*, 6(4):733–760. 25, 84

Gibson, G. and Renshaw, E. (1998). Estimating parameters in stochastic compartmental models using Markov chain methods. *Mathematical Medicine and Biology*, 15(1):19–40. 2, 33, 76, 77

Gibson, G. J., Otten, W., Filipe, J. A. N., Cook, A., Marion, G., and Gilligan, C. A. (2006). Bayesian estimation for percolation models of disease spread in plant populations. *Statistics and Computing*, 16(4):391–402. 71

Gibson, G. J., Streftaris, G., and Thong, D. (2018). Comparison and assessment of epidemic models. *Statistical Science*, 33(1):19–33. 3, 72, 73, 83, 87

Gilks, W., Richardson, S., and Spiegelhalter, D. (1996). *Markov Chain Monte Carlo in Practice*. Chapman & Hall/CRC Interdisciplinary Statistics. CRC Press, first edition. 8, 10, 12, 19, 20, 22, 24, 26, 75

Haccou, P., Jagers, P., and Vatutin, V. A. (2005). *Branching Processes: Variation, Growth, and Extinction of Populations*. Cambridge Studies in Adaptive Dynamics. Cambridge University Press, first edition. 95

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109. 9

Held, L., Hens, N., O'Neill, P. D., and Wallinga, J., editors (2019). *Handbook of Infectious Disease Data Analysis*. Taylor & Francis Ltd. 5, 8, 35, 42

Hoff, P. D. (2009). *A First Course in Bayesian Statistical Methods*. Springer New York. 8, 22

Jewell, C. P., Kypraios, T., Neal, P., and Roberts, G. O. (2009). Bayesian analysis for emerging infectious diseases. *Bayesian Analysis*, 4(3):465–496. 29, 33, 72, 78, 130, 214, 245

Keeling, M. J. (2001). Dynamics of the 2001 UK foot and mouth epidemic: Stochastic dispersal in a heterogeneous landscape. *Science*, 294(5543):813–817. 1

Kendall, D. G. (1956). Deterministic and stochastic epidemics in closed populations. In *Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability, Volume 4: Contributions to Biology and Problems of Health*, pages 149–165, Berkeley, Calif. University of California Press. 30

Knock, E. S. and Kypraios, T. (2014). Bayesian non-parametric inference for infectious disease data. arXiv:1411.2624. 78

Kucharski, A. J., Klepac, P., Conlan, A., Kissler, S. M., Tang, M., Fry, H., Gog, J., and Edmunds, J. (2020a). Effectiveness of isolation, testing, contact tracing and physical distancing on reducing transmission of SARS-CoV-2 in different settings. *medRxiv*. 1

Kucharski, A. J., Russell, T. W., Diamond, C., Liu, Y., Edmunds, J., Funk, S., Eggo, R. M., Sun, F., Jit, M., Munday, J. D., Davies, N., Gimma, A., van Zandvoort, K., Gibbs, H., Hellewell, J., Jarvis, C. I., Clifford, S., Quilty, B. J., Bosse, N. I., Abbott, S., Klepac, P., and Flasche, S. (2020b). Early dynamics of transmission and control of COVID-19: a mathematical modelling study. *The Lancet Infectious Diseases*, 20(5):553–558. 1

Kypraios, T. (2007). *Efficient Bayesian inference for partially observed stochastic epidemics and a new class of semi-parametric time series models*. PhD thesis, Lancaster University. 2, 14, 29, 32, 33, 35, 36, 78, 79, 130, 217, 245

Kypraios, T., Neal, P., and Prangle, D. (2017). A tutorial introduction to Bayesian inference for stochastic epidemic models using Approximate Bayesian Computation. *Mathematical Biosciences*, 287:42–53. 32, 106, 183, 207

Kypraios, T. and O'Neill, P. D. (2018). Bayesian nonparametrics for stochastic epidemic models. *Statistical Science*, 33(1):44–56. 78

Lau, M. S. Y., Marion, G., Streftaris, G., and Gibson, G. J. (2014). New model diagnostics for spatio-temporal systems in epidemiology and ecology. *Journal of The Royal Society Interface*, 11(93):20131093–20131093. 72

Lee, P. M. (2012). *Bayesian Statistics: An Introduction, 4th Edition*. Wiley. 5

Lekone, P. E. and Finkenstädt, B. F. (2006). Statistical inference in a stochastic epidemic SEIR model with control intervention: Ebola as a case study. *Biometrics*, 62(4):1170–1177. 73, 86

McKinley, T., Cook, A. R., and Deardon, R. (2009). Inference in epidemic models without likelihoods. *The International Journal of Biostatistics*, 5(1). 32

McKinley, T. J., Vernon, I., Andrianakis, I., McCreesh, N., Oakley, J. E., Nsubuga, R. N., Goldstein, M., and White, R. G. (2018). Approximate Bayesian computation and simulation-based inference for complex stochastic epidemic models. *Statistical Science*, 33(1):4–18. 32, 183

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092. 9

Neal, P. and Roberts, G. (2005). A case study in non-centering for data augmentation: Stochastic epidemics. *Statistics and Computing*, 15(4):315–327. 2, 33, 36, 78, 79, 217

Neal, P. J. and Roberts, G. O. (2004). Statistical inference and model selection for the 1861 Hagelloch measles epidemic. *Biostatistics*, 5(2):249–261. 34, 78, 217

Nguyen, C. and Carlson, J. M. (2016). Optimizing real-time vaccine allocation in a stochastic SIR model. *PLOS ONE*, 11(4):e0152950. 1

O'Neill, P. and Wen, C. (2012). Modelling and inference for epidemic models featuring non-linear infection pressure. *Mathematical Biosciences*, 238(1):38–48. 55, 56, 60, 77

O'Neill, P. D. (1996). Strong approximations for some open population epidemic models. *Journal of Applied Probability*, 33(2):448–457. 29

O'Neill, P. D. (2010). Introduction and snapshot review: Relating infectious disease transmission models to data. *Statistics in Medicine*, 29(20):2069–2077. 3, 27, 30, 70, 73, 78, 183, 216

O'Neill, P. D. and Becker, N. G. (2001). Inference for an epidemic when susceptibility varies. *Biostatistics*, 2(1):99–108. 34, 40, 77, 79, 80, 217

O'Neill, P. D. and Roberts, G. O. (1999). Bayesian inference for partially observed stochastic epidemics. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 162(1):121–129. 2, 29, 31, 33, 34, 42, 47, 76, 77, 207, 217

Papaspiliopoulos, O., Roberts, G. O., and Skold, M. (2003). Non-centered parameterisations for hierarchical models and data augmentation. *Bayesian Statistics*, 7:307–326. 72, 78

Parry, M., Gibson, G. J., Parnell, S., Gottwald, T. R., Irey, M. S., Gast, T. C., and Gilligan, C. A. (2014). Bayesian inference for an emerging arboreal epidemic in the presence of control. *Proceedings of the National Academy of Sciences*, 111(17):6258–6262. 73

R Core Team (2019). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. 82, 186, 219

Retkute, R., Jewell, C. P., Boeckel, T. P. V., Zhang, G., Xiao, X., Thanapongtharm, W., Keeling, M., Gilbert, M., and Tildesley, M. J. (2018). Dynamics of the 2004 avian influenza h5n1 outbreak in Thailand: The role of duck farming, sequential model fitting and control. *Preventive Veterinary Medicine*, 159:171–181. 29

Rida, W. N. (1991). Asymptotic properties of some estimators for the infection rate in the general stochastic epidemic model. *Journal of the Royal Statistical Society. Series B (Methodological)*, 53(1):269–283. 32

Robert, C. (2007). *The Bayesian Choice: From Decision-Theoretic Foundations to Computational Implementation (Springer Texts in Statistics)*. Springer. 5

Robert, C. P. and Casella, G. (2004). *Monte Carlo Statistical Methods*. Springer New York. 8, 12, 14, 17, 18

Roberts, G. O. and Rosenthal, J. S. (2001). Optimal scaling for various Metropolis-Hastings algorithms. *Statistical Science*, 16(4):351–367. 19

Ross, S. M. (2009). *Introduction to Probability Models*. Academic Press, tenth edition. 35, 65, 338

Sellke, T. (1983). On the asymptotic distribution of the size of a stochastic epidemic. *Journal of Applied Probability*, 20(2):390–394. 71

Severo, N. C. (1969). Generalizations of some stochastic epidemic models. *Mathematical Biosciences*, 4(3-4):395–402. 55

Snijders, T. (2001). Hypothesis testing: Methodology and limitations. In *International Encyclopedia of the Social & Behavioral Sciences*, pages 7121–7127. Elsevier. 212

Stockdale, J. E., Kypraios, T., and O'Neill, P. D. (2017). Modelling and Bayesian analysis of the Abakaliki smallpox data. *Epidemics*, 19:13–23. 78, 207, 208

Streftaris, G. and Gibson, G. J. (2004a). Bayesian analysis of experimental epidemics of foot–and–mouth disease. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 271(1544):1111–1117. 29

Streftaris, G. and Gibson, G. J. (2004b). Bayesian inference for stochastic epidemics in closed populations. *Statistical Modelling: An International Journal*, 4(1):63–75. 33, 77

Streftaris, G. and Gibson, G. J. (2012). Non-exponential tolerance to infection in epidemic systems–modeling, inference, and assessment. *Biostatistics*, 13(4):580–593. 71

Tanner, M. A. and Wong, W. H. (1987). The calculation of posterior distributions by data augmentation. *Journal of the American Statistical Association*, 82(398):528–540. 11

Thompson, D. and Foege, W. (1968). *Faith Tabernacle Smallpox Epidemic, Abakaliki, Nigeria*. World Health Organization. 207, 208, 209

Weber, N. C. (1980). A martingale approach to central limit theorems for exchangeable random variables. *Journal of Applied Probability*, 17(3):662–673. 211

Xiang, F. and Neal, P. (2014). Efficient MCMC for temporal epidemics via parameter reduction. *Computational Statistics & Data Analysis*, 80:240–250. 2, 29, 33, 43, 79, 80, 217, 218, 220, 236, 253, 254, 255, 256, 257, 258, 261, 265, 280

Yuan, E. C., Alderson, D. L., Stromberg, S., and Carlson, J. M. (2015). Optimal vaccination in a stochastic epidemic model of two non-interacting populations. *PLOS ONE*, 10(2):e0115826. 1

Appendix

Appendix A

Tables and Figures

A.1 Examples from chapter 2

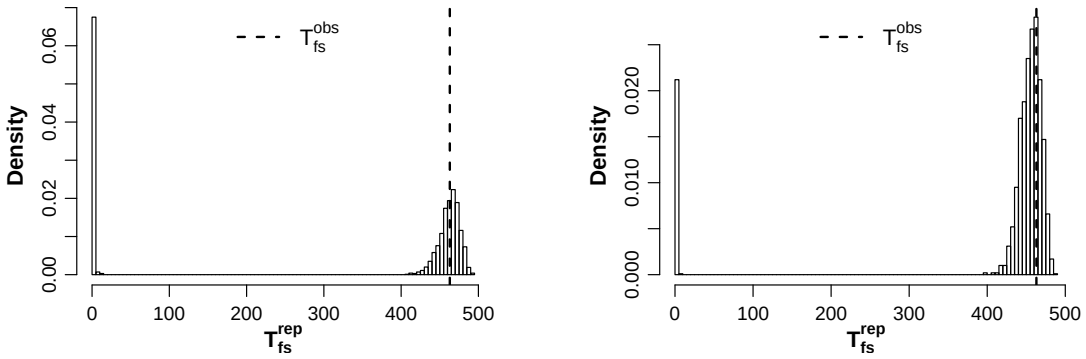


Figure A.1: Histograms of 2000 replications from the posterior predictive distribution of the final size T_{fs}^{rep}, with the observed final size $T_{fs}^{obs} = 463$ (black, dashed line) imposed, for the example in sections 2.5.5 and 2.6.3, figures 2.5 and 2.6. Left and right histograms correspond to the Exp-HM and the Gamma-HM models, respectively.

A.2 Examples from chapter 4

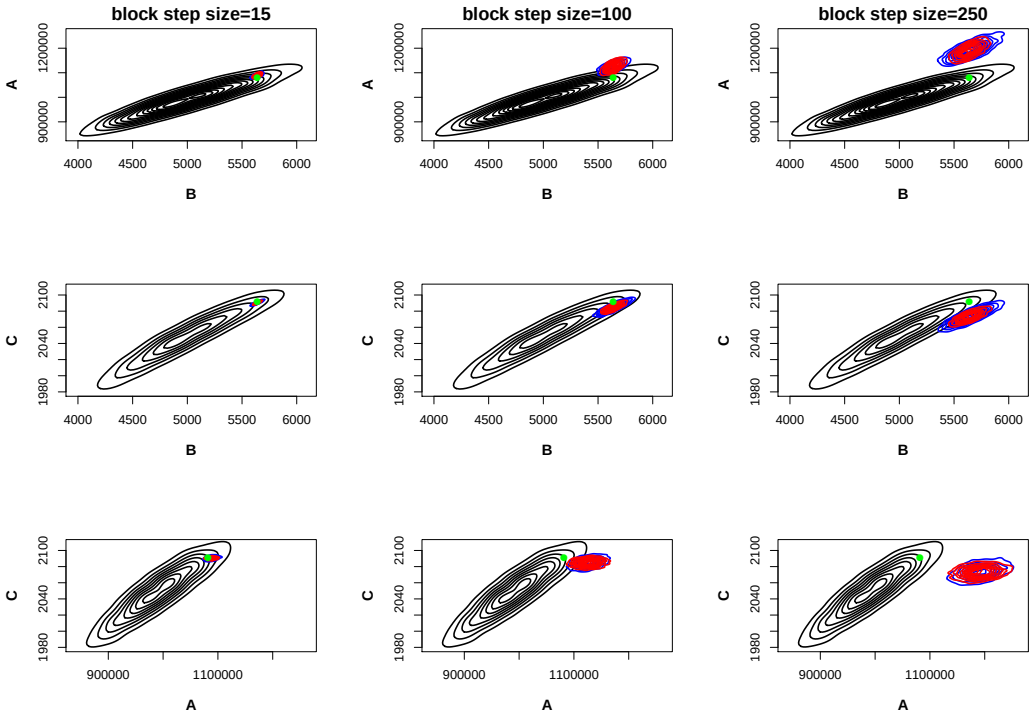


Figure A.2: Bivariate target posterior densities (black, solid contours), bivariate standard-block proposal densities (blue, solid contours) and bivariate P_1 proposal densities (red, solid contours), for the vector (A, B, C). Imposed (green, circle) is the current state. Columns (left to right) correspond to block step size values of 15, 100 and 250, respectively. The dataset is generated from a Gamma-HM model ($N = 500$, $R_0 = 2.5$, $\nu = 5$) and the number of infections is $n = 448$.

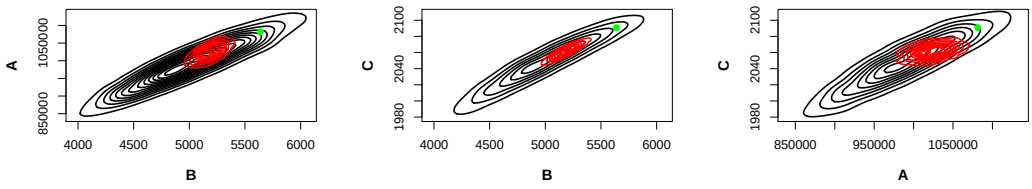


Figure A.3: Bivariate target posterior densities (black, solid contours) and bivariate P_2 proposal densities (red, solid contours), for the vector (A, B, C). The block step size is $m = 448$. Imposed (green, circle) is the current state. The dataset is generated from a Gamma-HM model ($N = 500$, $R_0 = 2.5$, $\nu = 5$) and the number of infections is $n = 448$.

A.3 Simulation Study A

Table A.1: Number of datasets for which the matching procedure was completed over number of total datasets, for the Exp-HM model for simulation study A. For each dataset, the number of required matched replications was 500 and the computational time allowed for achieving them was 15 hours. Simulation conditions for each scenario are given in table 2.5.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	24/24	24/24	24/24	23/24
Scenario 2	24/24	24/24	24/24	23/24
Scenario 3	24/24	24/24	24/24	24/24
Scenario 4	24/24	18/24	3/24	0/24

Table A.2: Number of datasets for which the matching procedure was completed over number of total datasets, for the Gamma-HM model for simulation study A. For each dataset, the number of required matched replications was 500 and the computational time allowed for achieving them was 15 hours. Simulation conditions for each scenario are given in table 2.5.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	24/24	24/24	24/24	24/24
Scenario 2	24/24	24/24	24/24	24/24
Scenario 3	24/24	24/24	24/24	24/24
Scenario 4	24/24	11/24	0/24	0/24

Table A.3: Number of datasets for which the matching procedure was completed over number of total datasets, for the Constant-HM model for simulation study A. For each dataset, the number of required matched replications was 500 and the computational time allowed for achieving them was 15 hours. Simulation conditions for each scenario are given in table 2.5.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	24/24	24/24	22/24	20/24
Scenario 2	24/24	24/24	23/24	24/24
Scenario 3	24/24	24/24	24/24	24/24
Scenario 4	24/24	13/24	0/24	0/24

Table A.4: Median (95% quantile interval) final size (mid) ppp-value for the Exp-HM model for simulation study A. Simulation conditions for each scenario are given in table 2.5.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.52 (0.41, 0.61)	0.49 (0.43, 0.59)	0.49 (0.44, 0.55)	0.48 (0.44, 0.60)
Scenario 2	0.52 (0.44, 0.62)	0.55 (0.46, 0.69)	0.57 (0.49, 0.66)	0.60 (0.49, 0.71)
Scenario 3	0.51 (0.43, 0.60)	0.54 (0.45, 0.65)	0.59 (0.48, 0.67)	0.63 (0.53, 0.75)
Scenario 4	0.69 (0.55, 0.92)	0.84 (0.61, 0.99)	0.97 (0.79, 1)	1 (0.96, 1)

Table A.5: Median (95% quantile interval) final size (mid) ppp-value for the Gamma-HM model for simulation study A. Simulation conditions for each scenario are given in table 2.5.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.55 (0.41, 0.74)	0.57 (0.48, 0.71)	0.55 (0.45, 0.71)	0.61 (0.46, 0.67)
Scenario 2	0.51 (0.44, 0.63)	0.48 (0.41, 0.64)	0.49 (0.40, 0.56)	0.48 (0.44, 0.56)
Scenario 3	0.48 (0.40, 0.58)	0.49 (0.41, 0.59)	0.49 (0.42, 0.55)	0.50 (0.45, 0.58)
Scenario 4	0.75 (0.40, 0.96)	0.90 (0.35, 0.99)	1 (0.64, 1)	1 (0.37, 1)

Table A.6: Median (95% quantile interval) final size (mid) ppp-value for the Constant-HM model for simulation study A. Simulation conditions for each scenario are given in table 2.5.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.47 (0.26, 0.74)	0.54 (0.32, 0.73)	0.54 (0.26, 0.79)	0.62 (0.02, 0.75)
Scenario 2	0.44 (0.38, 0.59)	0.48 (0.37, 0.58)	0.47 (0.12, 0.59)	0.50 (0.42, 0.59)
Scenario 3	0.45 (0.38, 0.57)	0.47 (0.38, 0.55)	0.45 (0.30, 0.54)	0.51 (0.45, 0.60)
Scenario 4	0.67 (0.25, 0.95)	0.90 (0.21, 1)	1 (0.34, 1)	1 (0.13, 1)

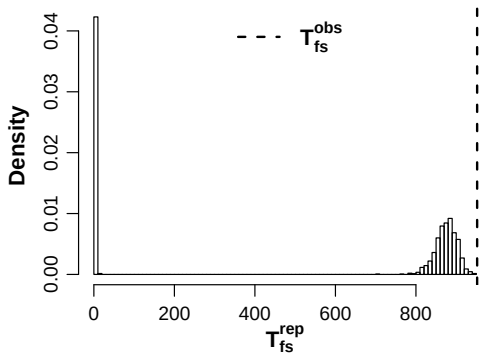


Figure A.4: Histogram of 2000 replications from the posterior predictive distribution of the final size T_{fs}^{rep} of the Exp-HM model, with the observed final size $T_{fs}^{obs} = 952$ (black, dashed line) imposed, for a typical dataset of round 4 ($N = 1000$) in scenario 4 (data generated from a HPP) of simulation study A.

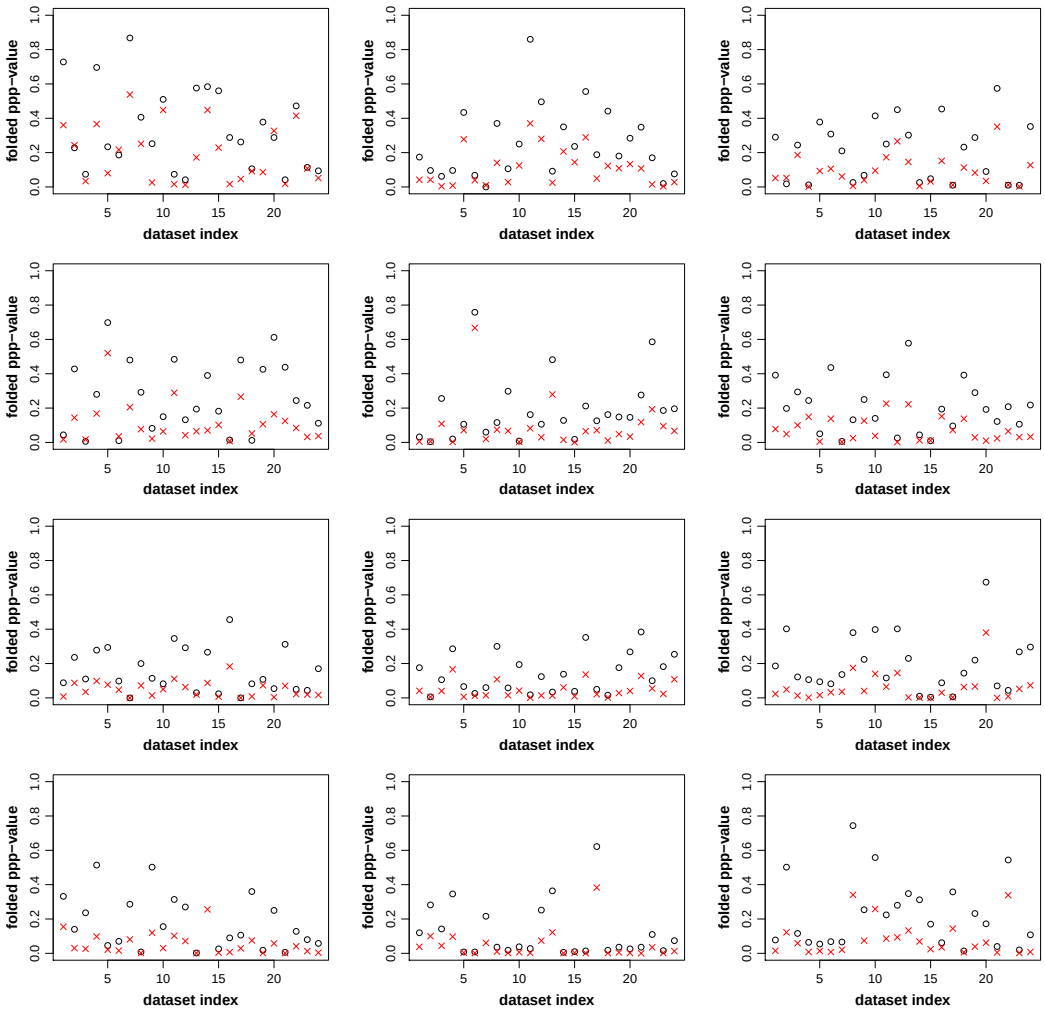


Figure A.5: Folded ppp-value from the distance method (d_{L_2} distance shifting, d_{L_2} distance function) against dataset index using matched (black circles) and unmatched (red crosses) replications, under correct specification, from simulation study A. Data are generated from the fitted model. Columns (left to right) correspond to the Exp-HM, the Gamma-HM and the Constant-HM models, respectively. Rows (top to bottom) correspond to N values of 100, 200, 500 and 1000, respectively.

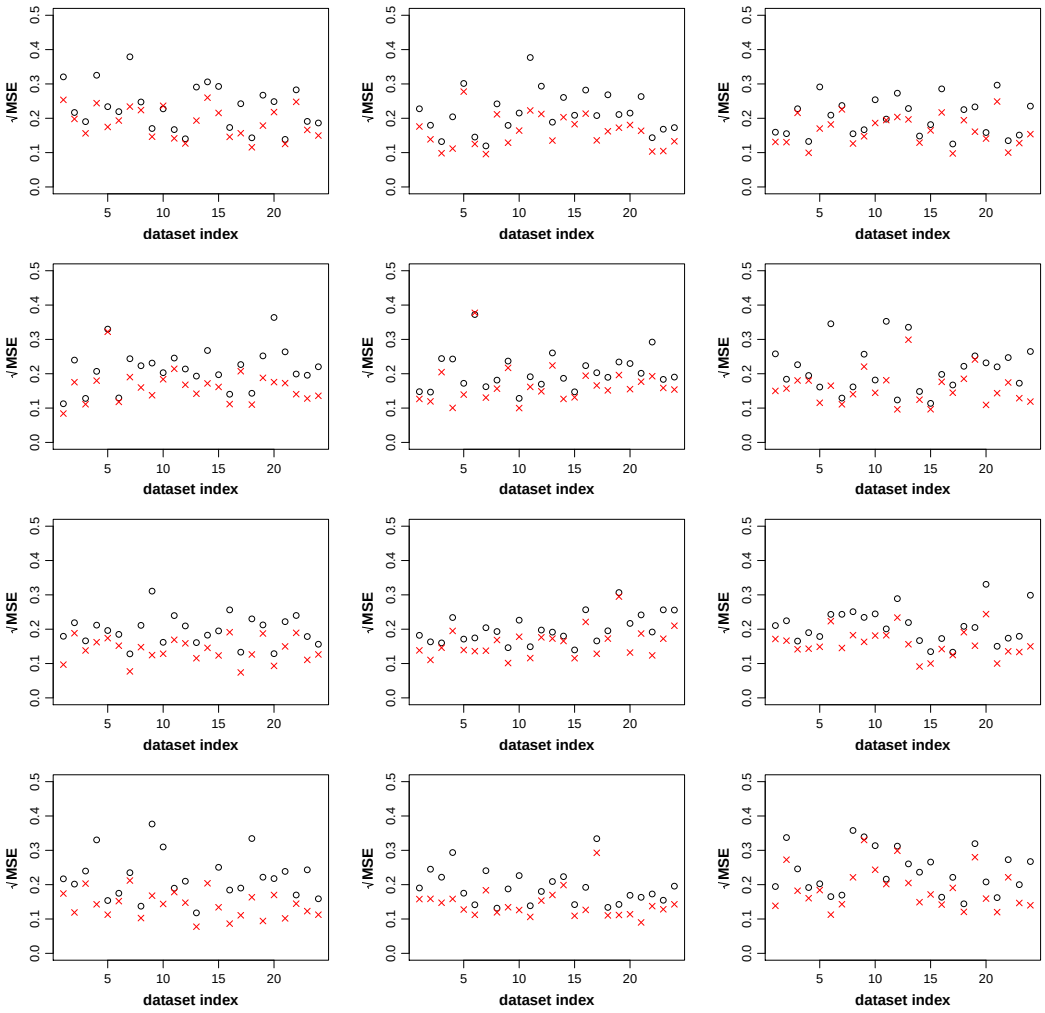


Figure A.6: $\sqrt{\text{MSE}}$ from the position-time method (d_{L_2} distance shifting) against dataset index using matched (black circles) and unmatched (red crosses) replications, under correct specification, from simulation study A. Data are generated from the fitted model. Columns (left to right) correspond to the Exp-HM, the Gamma-HM and the Constant-HM models, respectively. Rows (top to bottom) correspond to N values of 100, 200, 500 and 1000, respectively.

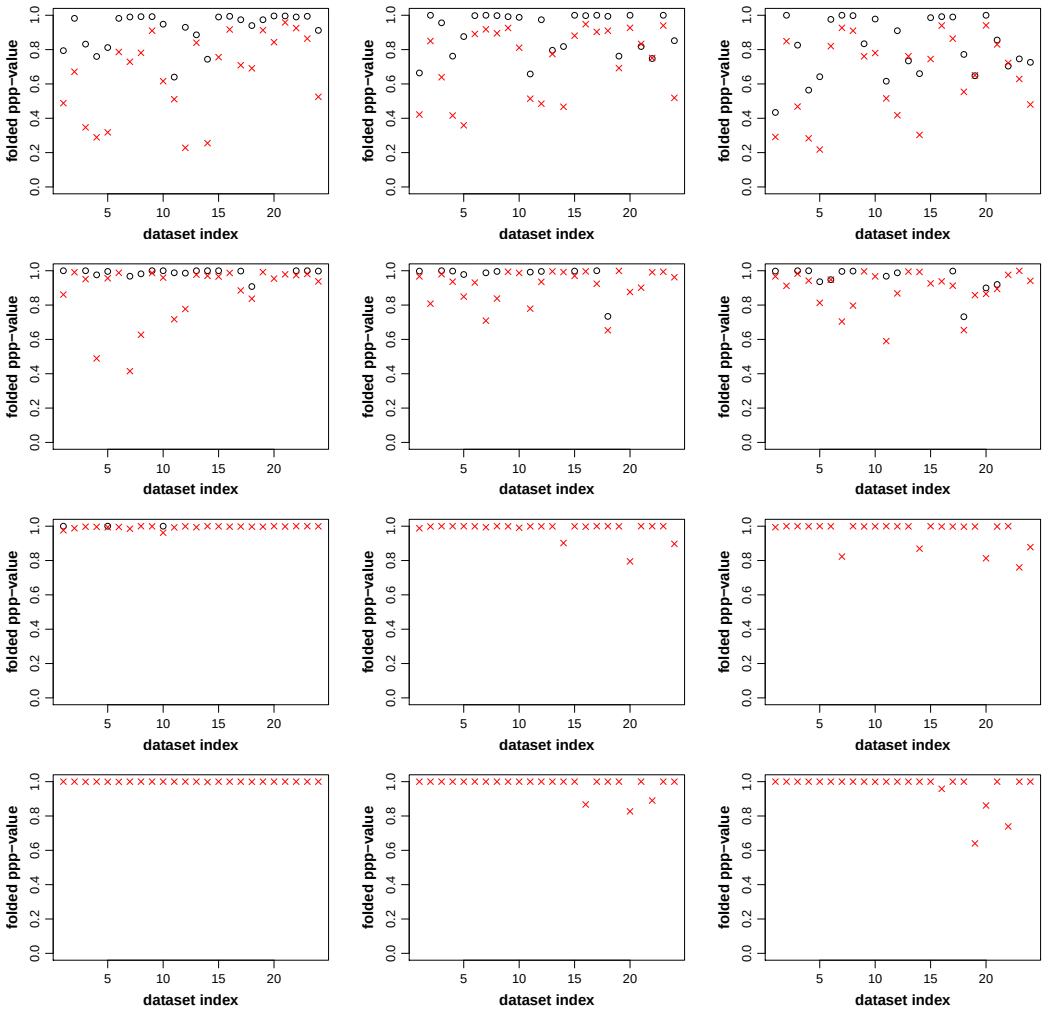


Figure A.7: Folded ppp-value from the distance method (d_{L_2} distance shifting, d_{L_2} distance function) against dataset index using matched (black circles) and unmatched (red crosses) replications, under clear misspecification, for simulation study A. Data are generated from the HPP. Columns (left to right) correspond to the Exp-HM, the Gamma-HM and the Constant-HM models, respectively. Rows (top to bottom) correspond to N values of 100, 200, 500 and 1000, respectively.

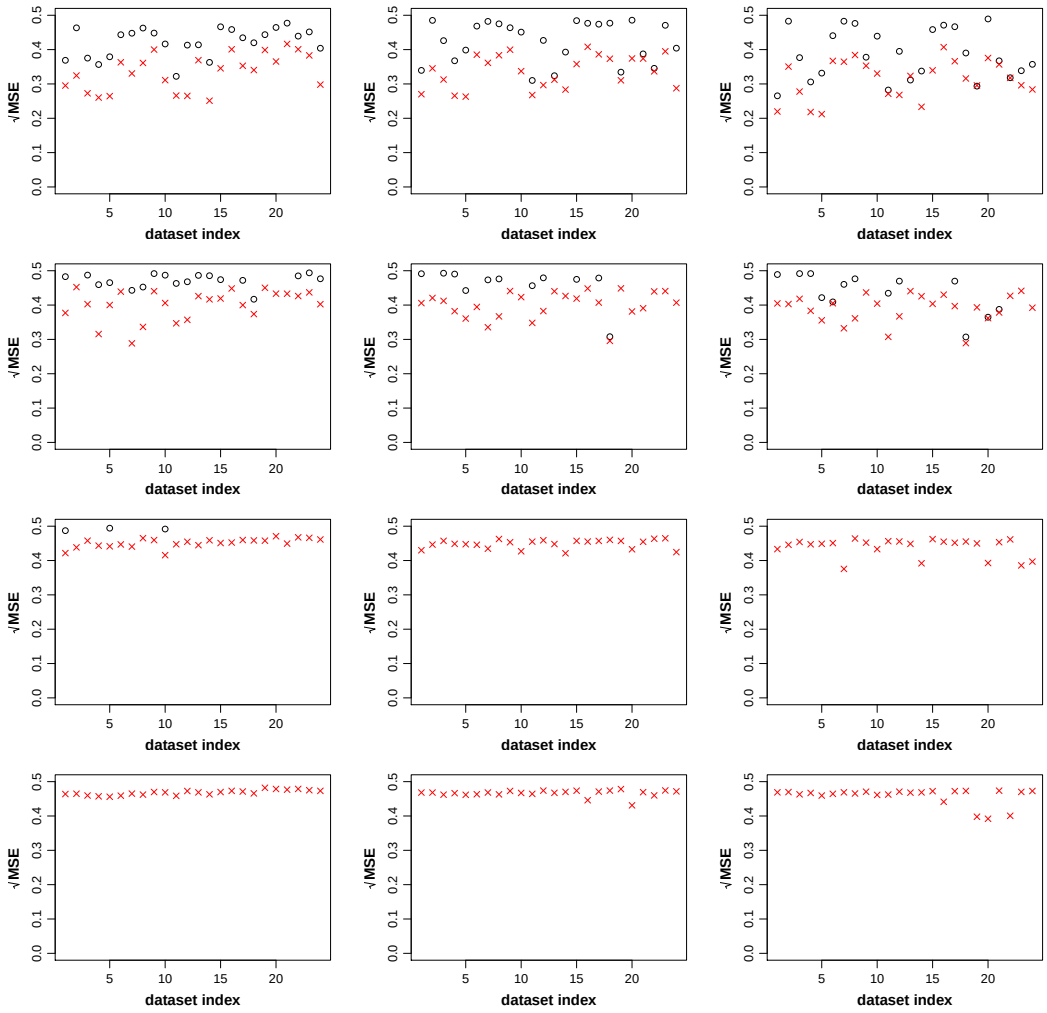


Figure A.8: $\sqrt{\text{MSE}}$ from the position-time method (d_{L_2} distance shifting) against dataset index using matched (black circles) and unmatched (red crosses) replications, under clear misspecification, for simulation study A. Data are generated from the HPP. Columns (left to right) correspond to the Exp-HM, the Gamma-HM and the Constant-HM models, respectively. Rows (top to bottom) correspond to N values of 100, 200, 500 and 1000, respectively.

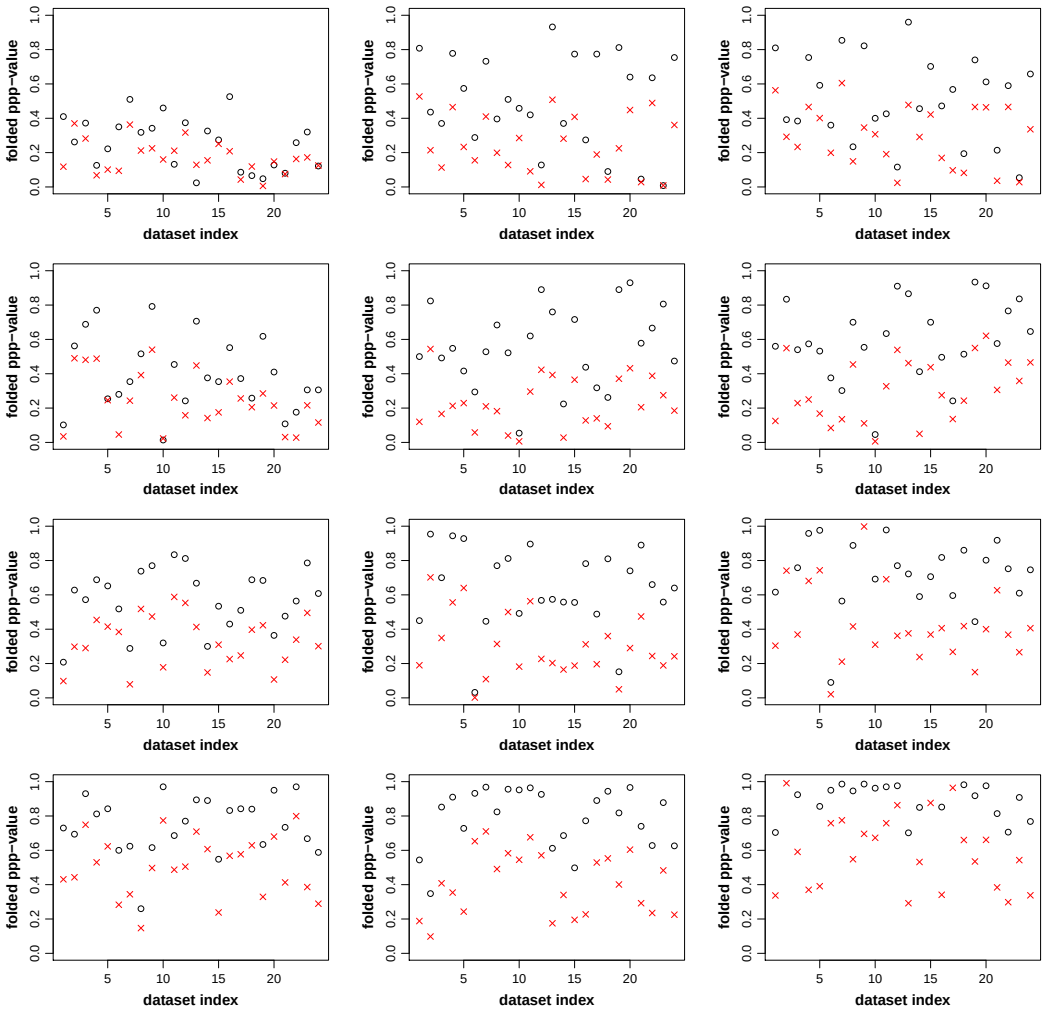


Figure A.9: Folded ppp-value from the distance method (d_{L_2} distance shifting, d_{L_2} distance function) against dataset index using matched (black circles) and unmatched (red crosses) replications, under (less clear) misspecification, from simulation study A. Left column corresponds to data generated from the Constant-HM and fitted model being the Exp-HM, middle column corresponds to data generated from the Exp-HM and fitted model being the Gamma-HM and right column corresponds to data generated from the Exp-HM and fitted model being the Constant-HM. Rows (top to bottom) correspond to N values of 100, 200, 500 and 1000, respectively.

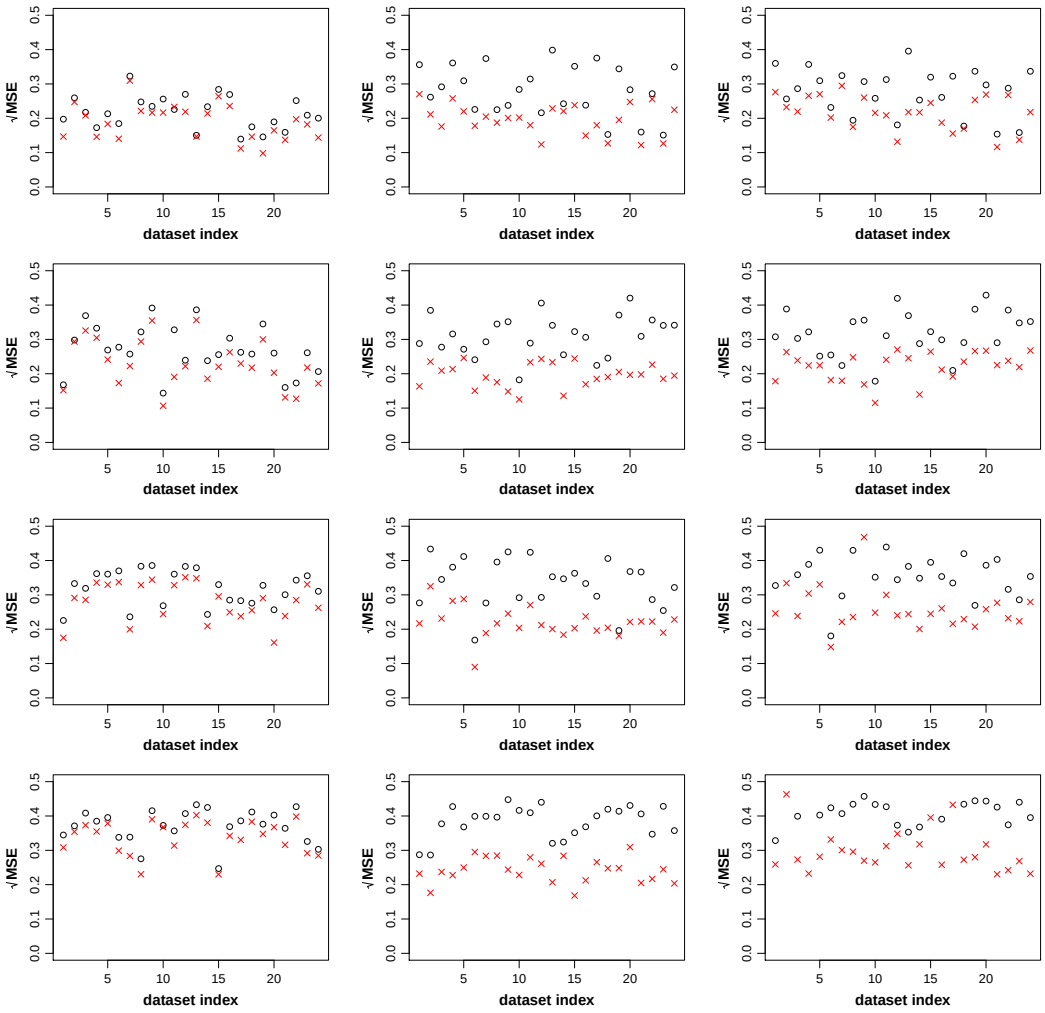


Figure A.10: $\sqrt{\text{MSE}}$ from the position-time method (d_{L_2} distance shifting) against dataset index using matched (black circles) and unmatched (red crosses) replications, under (less clear) misspecification, from simulation study A. Left column corresponds to data generated from the Constant-HM and fitted model being the Exp-HM, middle column corresponds to data generated from the Exp-HM and fitted model being the Gamma-HM and right column corresponds to data generated from the Exp-HM and fitted model being the Constant-HM. Rows (top to bottom) correspond to N values of 100, 200, 500 and 1000, respectively.

A.4 Simulation Study B

Table A.7: Number of datasets for which the matching procedure was completed over number of total datasets, for the Exp-HM model for simulation study B. For each dataset, the number of required matched replications was 500 and the computational time allowed for achieving them was 15 hours. Simulation conditions for each scenario are given in table 2.18.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	24/24	24/24	24/24	23/24
Scenario 2	24/24	24/24	22/24	7/24

Table A.8: Number of datasets for which the matching procedure was completed over number of total datasets, for the Exponential-NL model for simulation study B. For each dataset, the number of required matched replications was 500 and the computational time allowed for achieving them was 15 hours. Simulation conditions for each scenario are given in table 2.18.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	24/24	24/24	23/24	2/24
Scenario 2	24/24	24/24	24/24	23/24

Table A.9: Median (95% quantile interval) final size (mid) ppp-value for the Exp-HM model for simulation study B. Simulation conditions for each scenario are given in table 2.18.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.52 (0.41, 0.61)	0.49 (0.43, 0.59)	0.49 (0.44, 0.55)	0.48 (0.44, 0.60)
Scenario 2	0.49 (0.35, 0.58)	0.52 (0.46, 0.66)	0.63 (0.53, 0.92)	0.81 (0.62, 0.99)

Table A.10: Median (95% quantile interval) final size (mid) ppp-value for the Exponential-NL model for simulation study B. Simulation conditions for each scenario are given in table 2.18.

	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
Scenario 1	0.45 (0.34, 0.59)	0.33 (0.21, 0.54)	0.09 (0.04, 0.20)	0 (0, 0.05)
Scenario 2	0.49 (0.37, 0.69)	0.50 (0.36, 0.76)	0.43 (0.12, 0.96)	0.44 (0.08, 0.99)

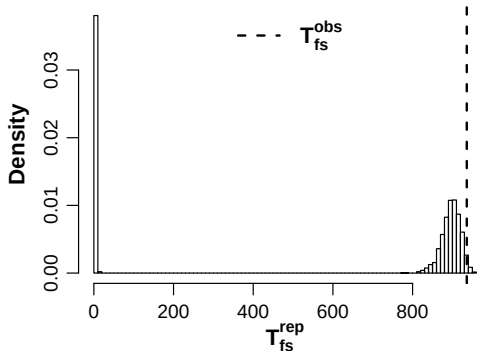


Figure A.11: Histogram of 2000 replications from the posterior predictive distribution of the final size T_{fs}^{rep} of the Exp-HM model, with the observed final size $T_{fs}^{obs} = 936$ (black, dashed line) imposed, for a typical dataset of round 4 ($N = 1000$) in scenario 2 (data generated from an Exp-NL model) of simulation study B.

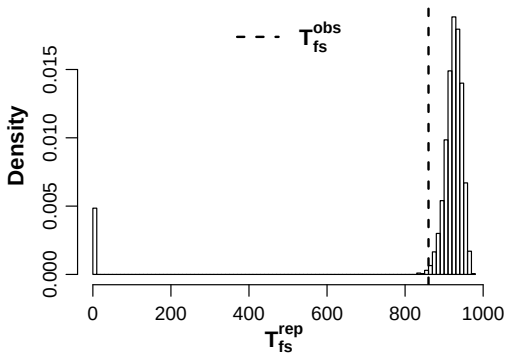


Figure A.12: Histogram of 2000 replications from the posterior predictive distribution of the final size T_{fs}^{rep} of the Exp-NL model, with the observed final size $T_{fs}^{obs} = 860$ (black, dashed line) imposed, for a typical dataset of round 4 ($N = 1000$) in scenario 1 (data generated from an Exp-HM model) of simulation study B.

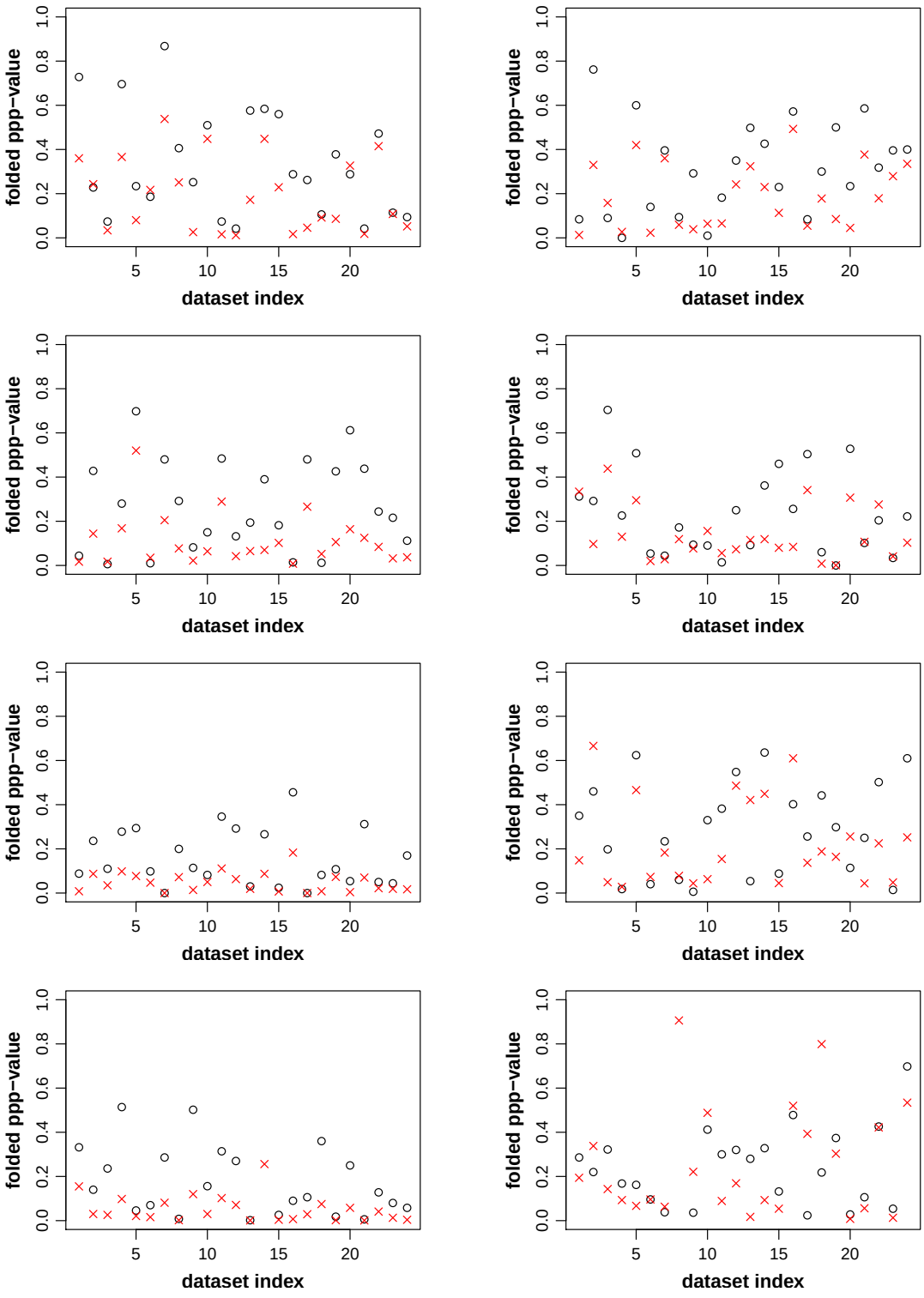


Figure A.13: Folded ppp-value from the distance method (d_{L_2} distance shifting, d_{L_2} distance function) against dataset index using matched (black circles) and unmatched (red crosses) replications, under correct specification, from simulation study B. Data are generated from the fitted model. Left and right columns corresponds to the Exp-HM and the Exp-NL models, respectively. Rows (top to bottom) correspond to N values of 100, 200, 500 and 1000, respectively.

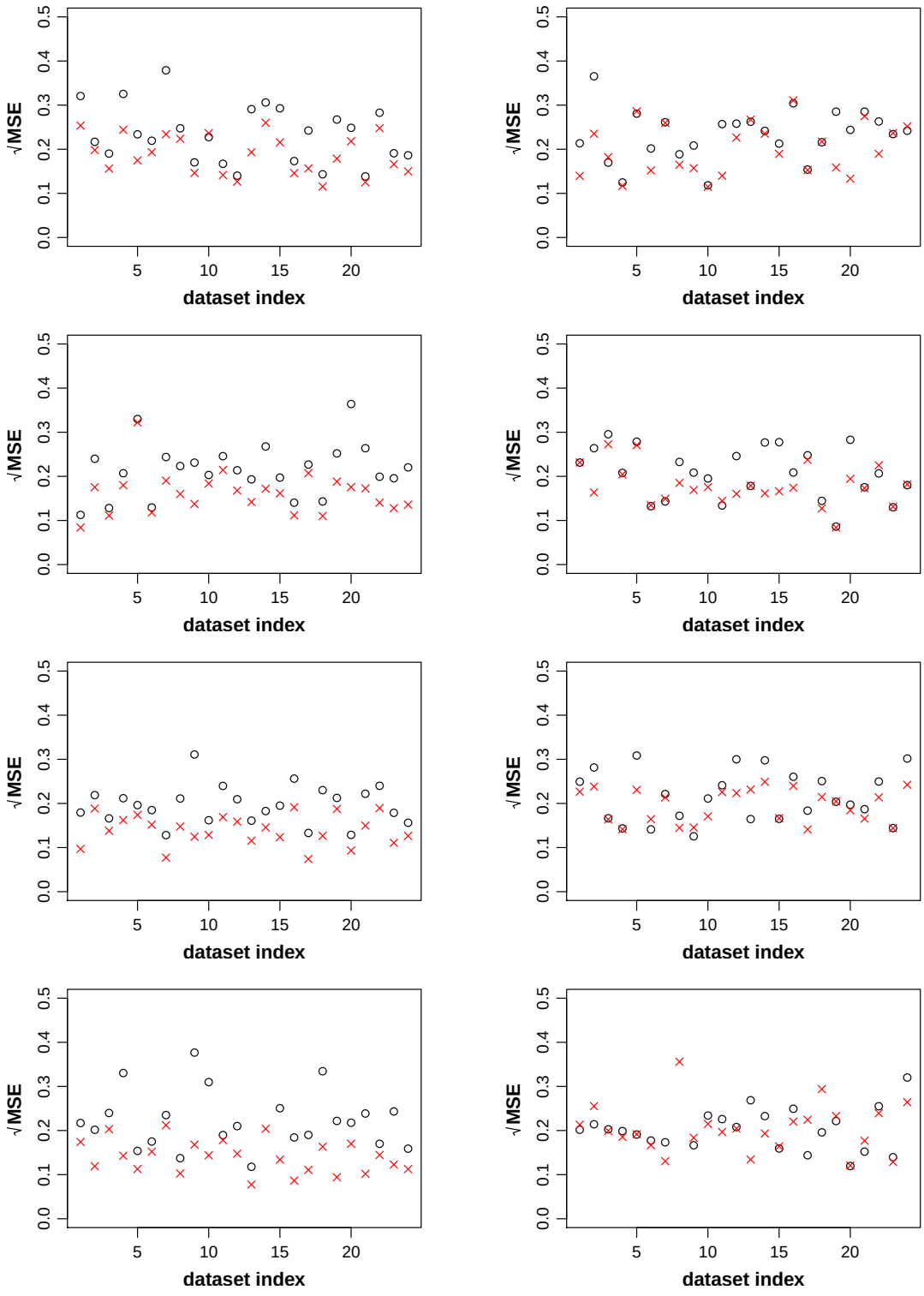


Figure A.14: $\sqrt{\text{MSE}}$ from the position-time method (d_{L_2} distance shifting) against dataset index using matched (black circles) and unmatched (red crosses) replications, under correct specification, from simulation study B. Data are generated from the fitted model. Left and right columns correspond to the Exp-HM and the Exp-NL models, respectively. Rows (top to bottom) correspond to N values of 100, 200, 500 and 1000, respectively.

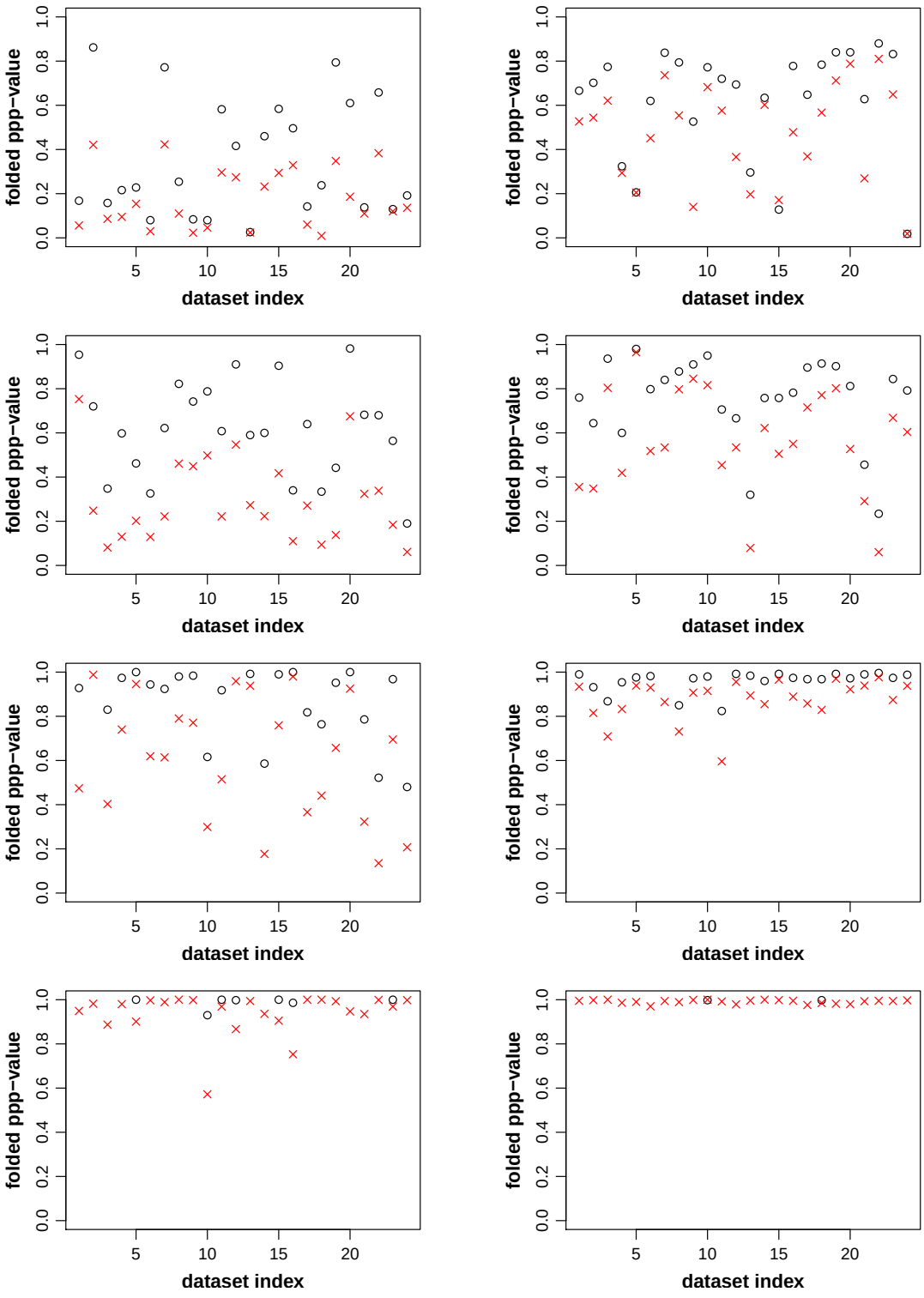


Figure A.15: Folded ppp-value from the distance method (d_{L_2} distance shifting, d_{L_2} distance function) against dataset index using matched (black circles) and unmatched (red crosses) replications, under misspecification, from simulation study B. Left column corresponds to data generated from the Exp-NL and fitted model being the Exp-HM and right column corresponds to data generated from the Exp-HM and fitted model being the Exp-NL. Rows (top to bottom) correspond to N values of 100, 200, 500 and 1000, respectively.

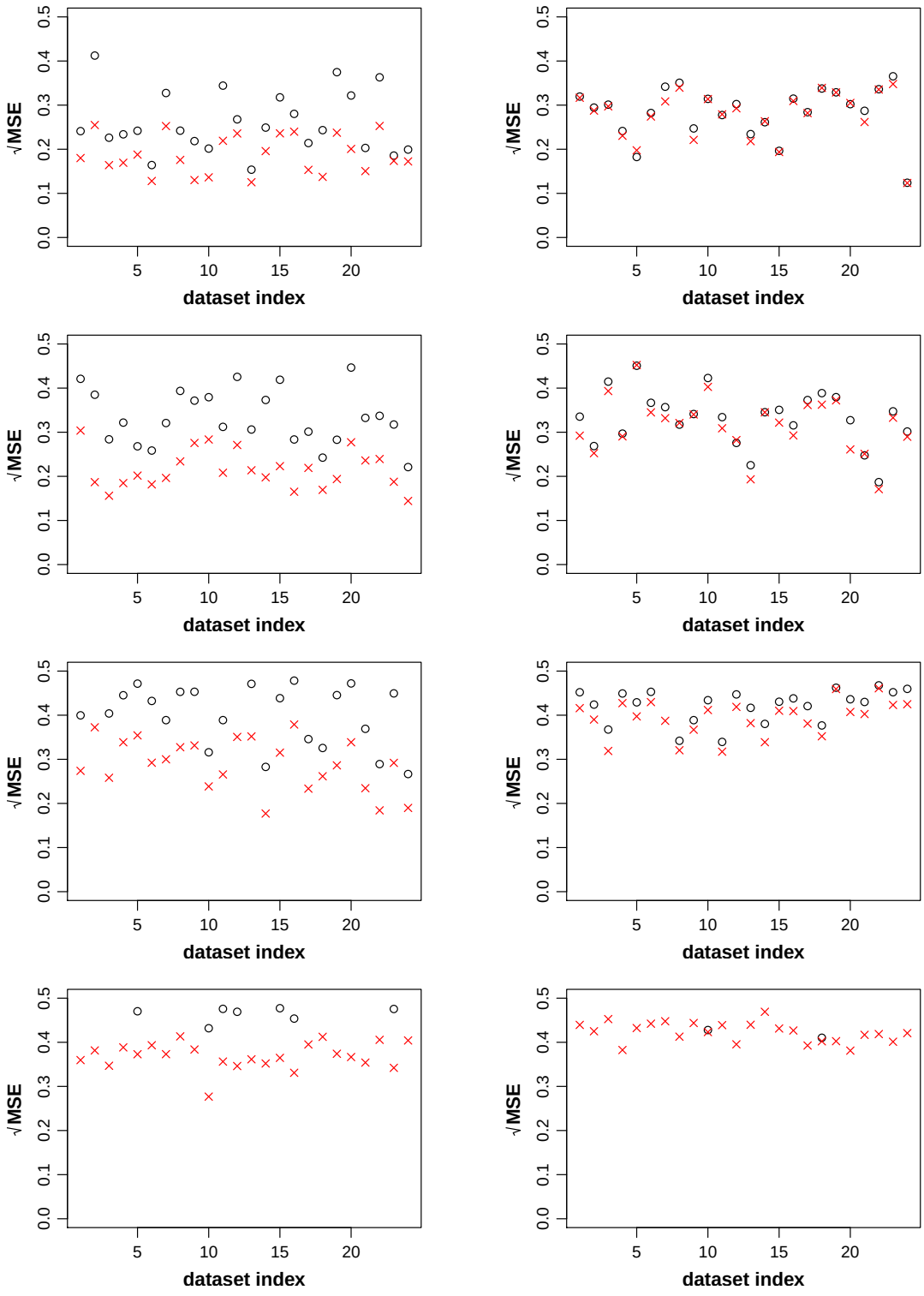


Figure A.16: $\sqrt{\text{MSE}}$ from the position-time method (d_{L_2} distance shifting) against dataset index using matched (black circles) and unmatched (red crosses) replications, under misspecification, from simulation study B. Left column corresponds to data generated from the Exp-NL and fitted model being the Exp-HM and right column corresponds to data generated from the Exp-HM and fitted model being the Exp-NL. Rows (top to bottom) correspond to N values of 100, 200, 500 and 1000, respectively.

A.5 Simulation Study C

Table A.11: Number of datasets for which the matching procedure was completed over number of total datasets, for the Constant-2L model for simulation study C. For each dataset, the number of required matched replications was 500 and the computational time allowed for achieving them was 15 hours. Simulation conditions for each scenario are given in table 2.27.

	$N = 99$	$N = 199$	$N = 499$
Scenario 1	24/24	24/24	23/24
Scenario 2	24/24	24/24	22/24
Scenario 3	24/24	23/24	21/24
Scenario 4	24/24	24/24	23/34

Table A.12: Number of datasets for which the matching procedure was completed over number of total datasets, for the Constant-HM model for simulation study C. For each dataset, the number of required matched replications was 500 and the computational time allowed for achieving them was 15 hours. Simulation conditions for each scenario are given in table 2.27.

	$N = 99$	$N = 199$	$N = 499$
Scenario 1	24/24	24/24	24/24
Scenario 2	24/24	24/24	24/24
Scenario 3	24/24	24/24	24/24
Scenario 4	24/24	24/24	23/34

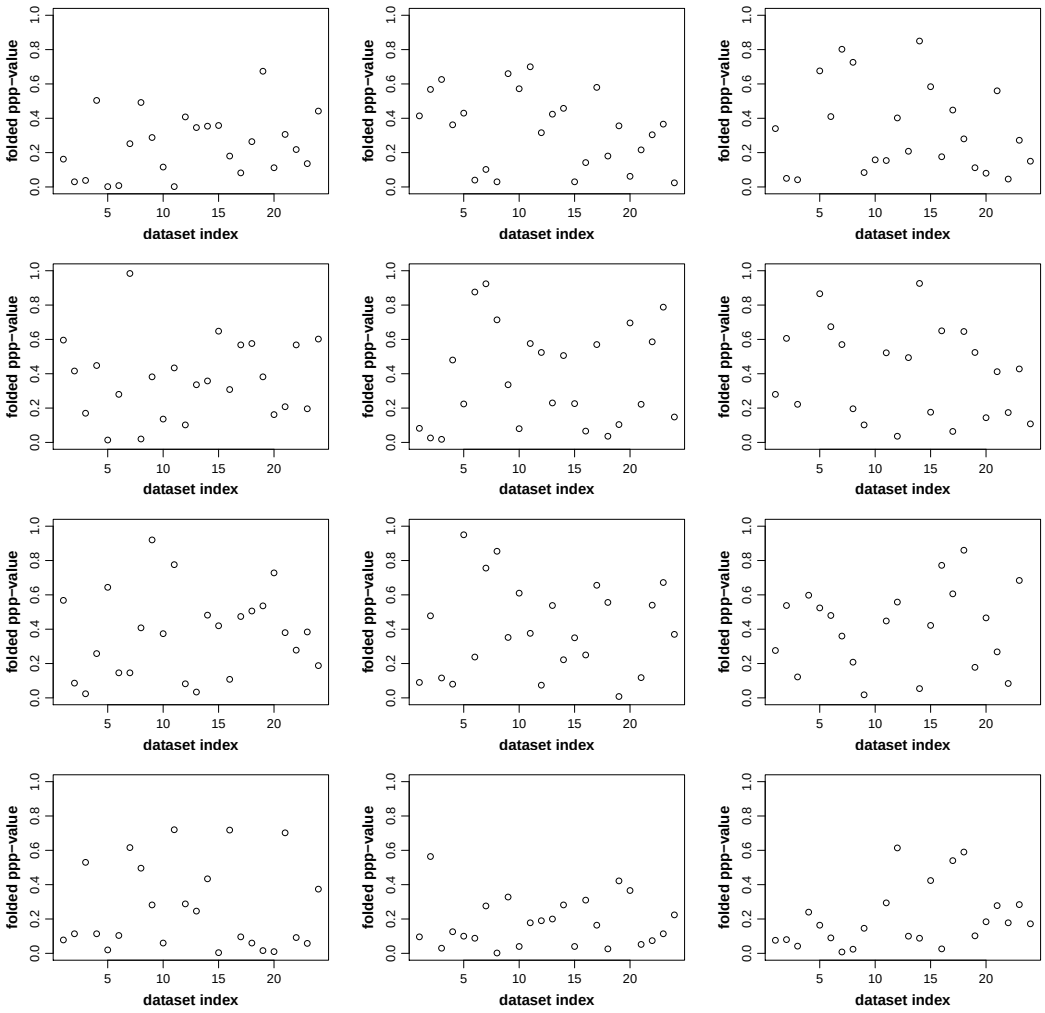


Figure A.17: Folded ppp-value from the distance method (d_{L_2} distance shifting, d_{L_2} distance function) against dataset index using matched replications, under correct specification for the Constant-2L model, from simulation study C. Data are generated from the fitted model. Rows (top to bottom) correspond to R_0^H values of 1, 2, 5 and 20, respectively. Columns (left to right) correspond to N values of 99, 199 and 499, respectively.

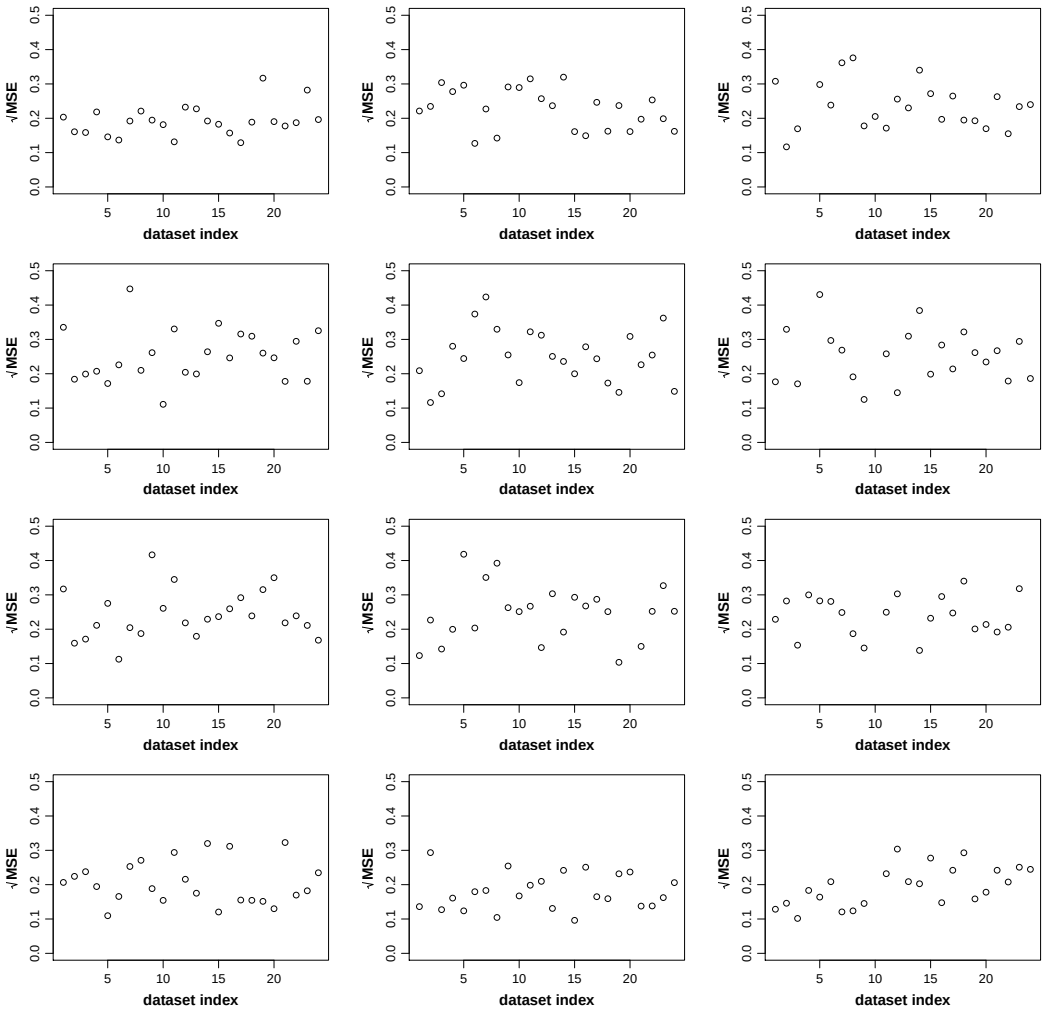


Figure A.18: $\sqrt{\text{MSE}}$ from the position-time method (d_{L_2} distance shifting) against dataset index using matched replications, under correct specification for the constant-2L model, from simulation study C. Data are generated from the fitted model. Rows (top to bottom) correspond to R_0^H values of 1, 2, 5 and 20, respectively. Columns (left to right) correspond to N values of 99, 199 and 499, respectively.

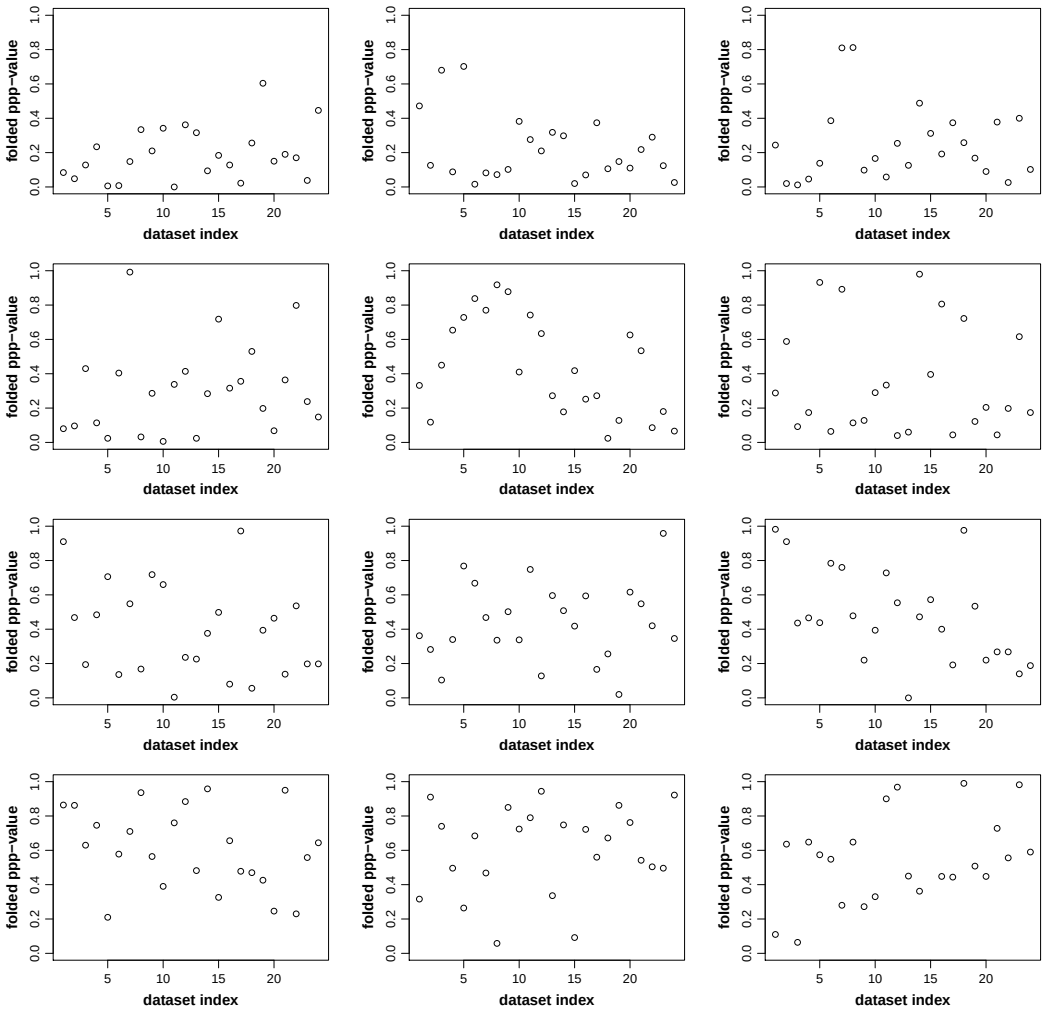


Figure A.19: Folded ppp-value from the distance method (d_{L_2} distance shifting, d_{L_2} distance function) against dataset index using matched replications, under misspecification for the constant-HM model, from simulation study C. Data are generated from the constant-2L model. Rows (top to bottom) correspond to R_0^H values of 1, 2, 5 and 20, respectively. Columns (left to right) correspond to N values of 99, 199 and 499, respectively.

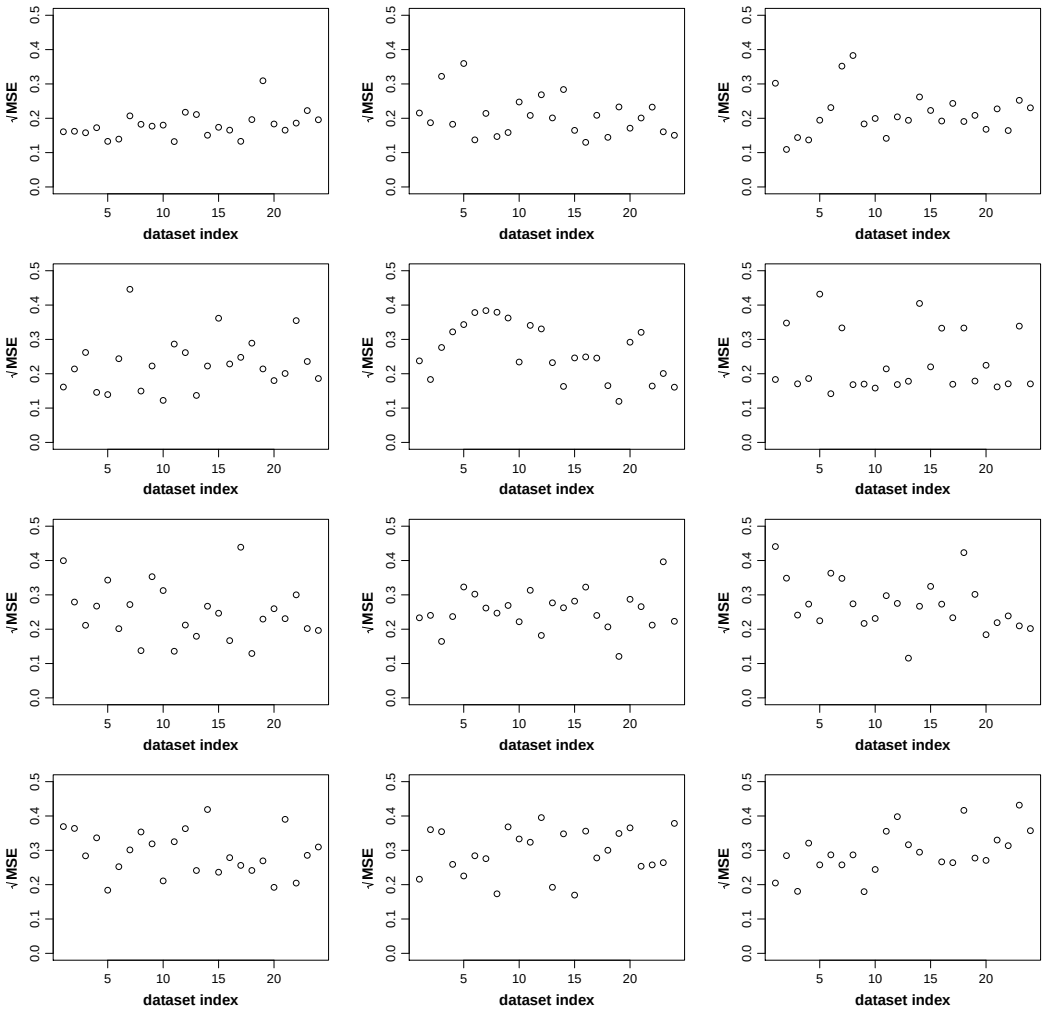


Figure A.20: $\sqrt{\text{MSE}}$ from the position-time method (d_{L_2} distance shifting) against dataset index using matched replications, under misspecification for the constant-HM model, from simulation study C. Data are generated from the constant-2L model. Rows (top to bottom) correspond to R_0^H values of 1, 2, 5 and 20, respectively, respectively. Columns (left to right) correspond to N values of 99, 199 and 499, respectively.

Table A.13: Median (95% quantile interval) final size (mid) ppp-value for the constant-HM model for simulation study C.

	$\boldsymbol{N = 99}$	$\boldsymbol{N = 199}$	$\boldsymbol{N = 499}$
Scenario 1	0.44 (0.36, 0.51)	0.48 (0.36, 0.61)	0.52 (0.42, 0.61)
Scenario 2	0.50 (0.38, 0.64)	0.54 (0.36, 0.67)	0.52 (0.40, 0.64)
Scenario 3	0.47 (0.37, 0.65)	0.51 (0.39, 0.70)	0.54 (0.34, 0.74)
Scenario 4	0.45 (0.27, 0.68)	0.54 (0.33, 0.75)	0.54 (0.41, 0.91)

A.6 Simulation Study D

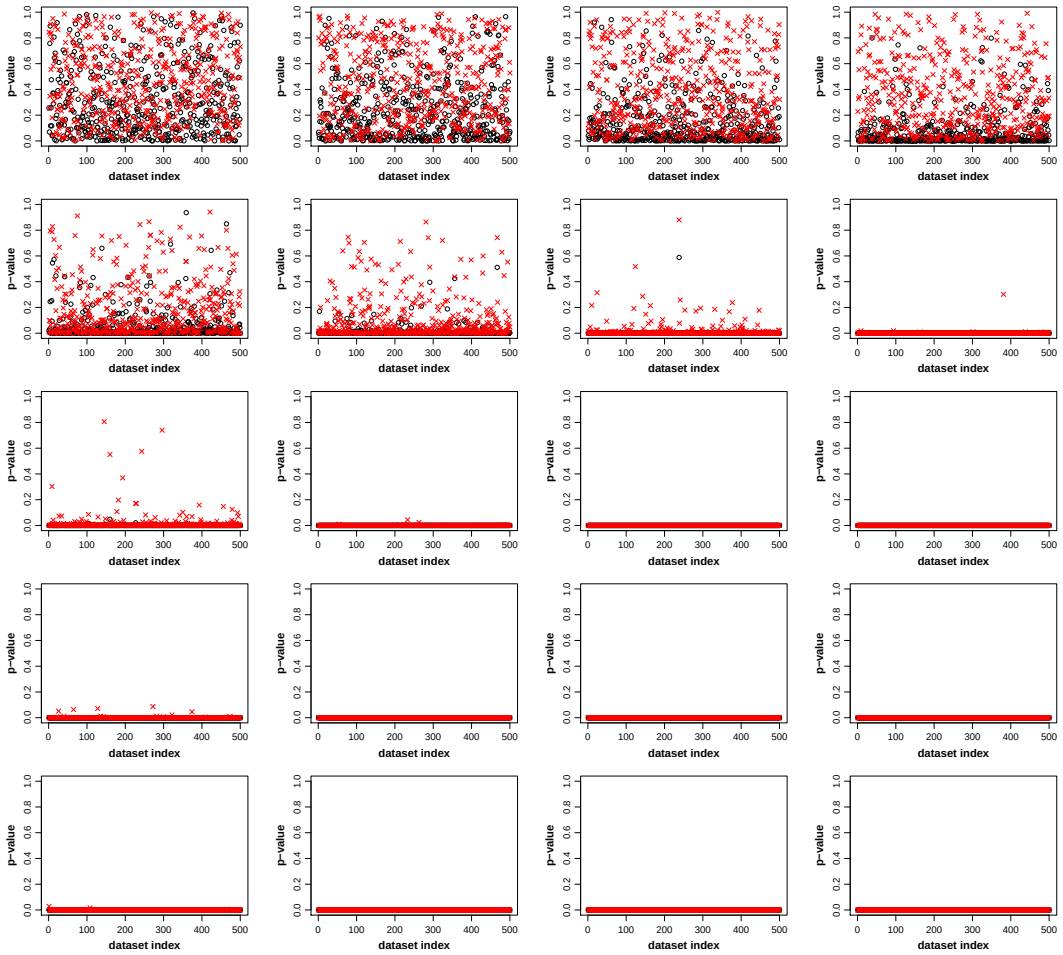


Figure A.21: p-value from the household labels test based on observing infection times, p-value$_{\boldsymbol{i}}$ (black circles), and based on observing removal times, p-value$_{\boldsymbol{r}}$ (red crosses), against dataset index, from simulation study D. Rows (top to bottom) correspond to R_0^H values of 0.5, 1, 2, 5 and 20, respectively. Columns (left to right) correspond to N values of 99, 199, 499 and 999, respectively.

A.7 Simulation Study E

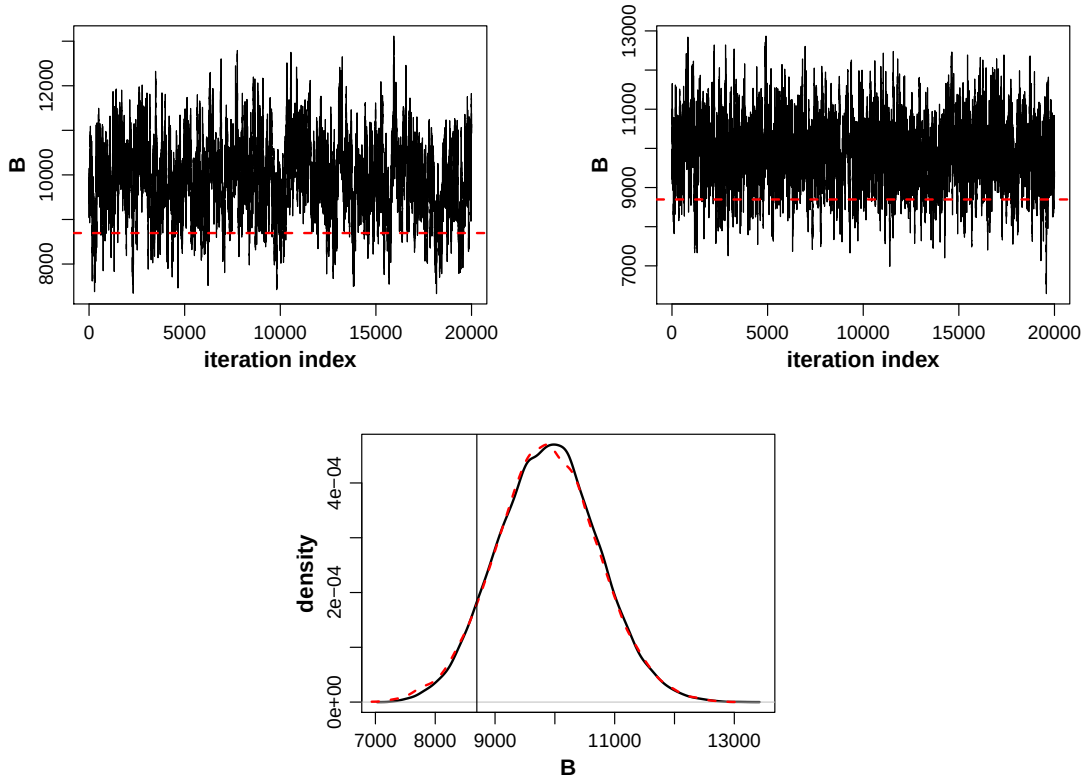


Figure A.22: MCMC convergence diagnostic plots for dataset 4 ($R_0 = 2.5, N = 1000$) of simulation study E. The simulation and run conditions are described in sections 4.2.3.2 and 4.2.3.3, respectively. Top plots are trace plots for $B = \sum_{k=1}^{n}(r_k - i_k)$. Left plot corresponds to the standard-1d MCMC algorithm and right plot corresponds to the IS-1d MCMC algorithm. Imposed (red, dashed, horizontal line) is the true vale of B. Bottom plot is the posterior density of B, based on the MCMC sample of the standard-1d MCMC algorithm (black, solid line) and the IS-1d MCMC algorithm (red, dashed line). Imposed (black, solid, vertical line) is the true vale of B.

A.8 Simulation Study F

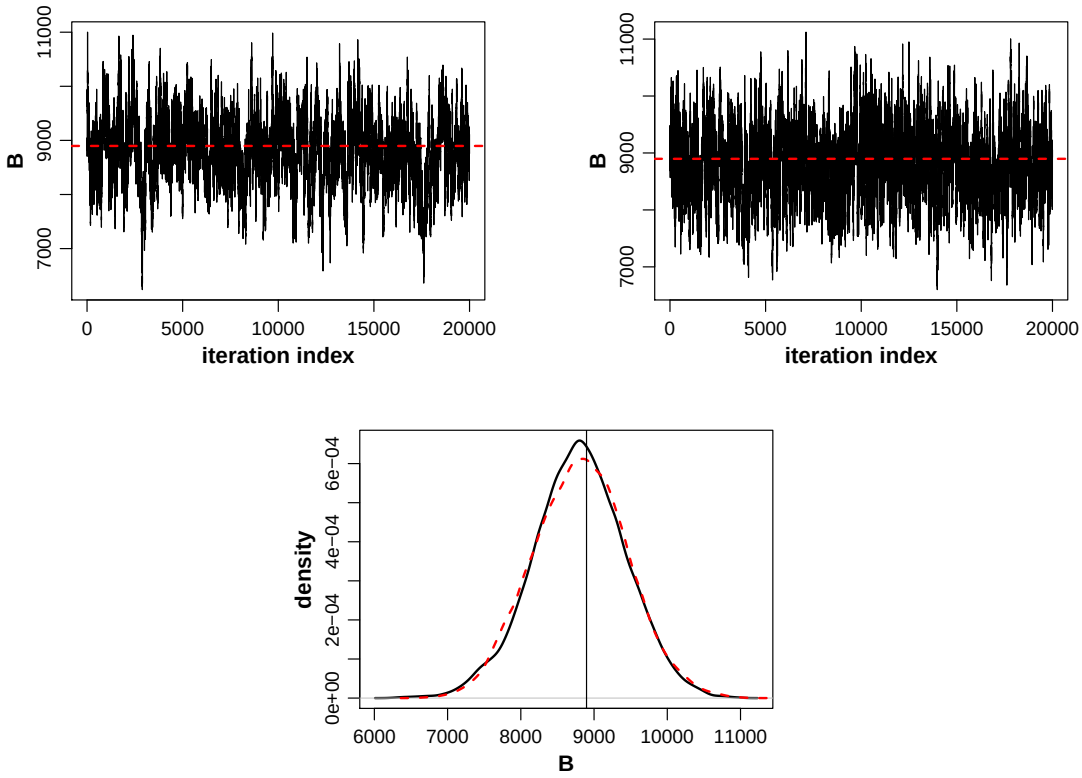


Figure A.23: MCMC convergence diagnostic plots for dataset 6 ($R_0 = 2.5, \nu = 2, N = 1000$) of simulation study F. The simulation and run conditions are described in section 4.2.4.1. Top plots are trace plots for $B = \sum_{k=1}^{n}(r_k - i_k)$. Left plot corresponds to the standard-1d MCMC algorithm and right plot corresponds to the IS-1d MCMC algorithm. Imposed (red, dashed, horizontal line) is the true vale of B. Bottom plot is the posterior density of B, based on the MCMC sample of the standard-1d MCMC algorithm (black, solid line) and the IS-1d MCMC algorithm (red, dashed line). Imposed (black, solid, vertical line) is the true vale of B.

A.9 Simulation Study G

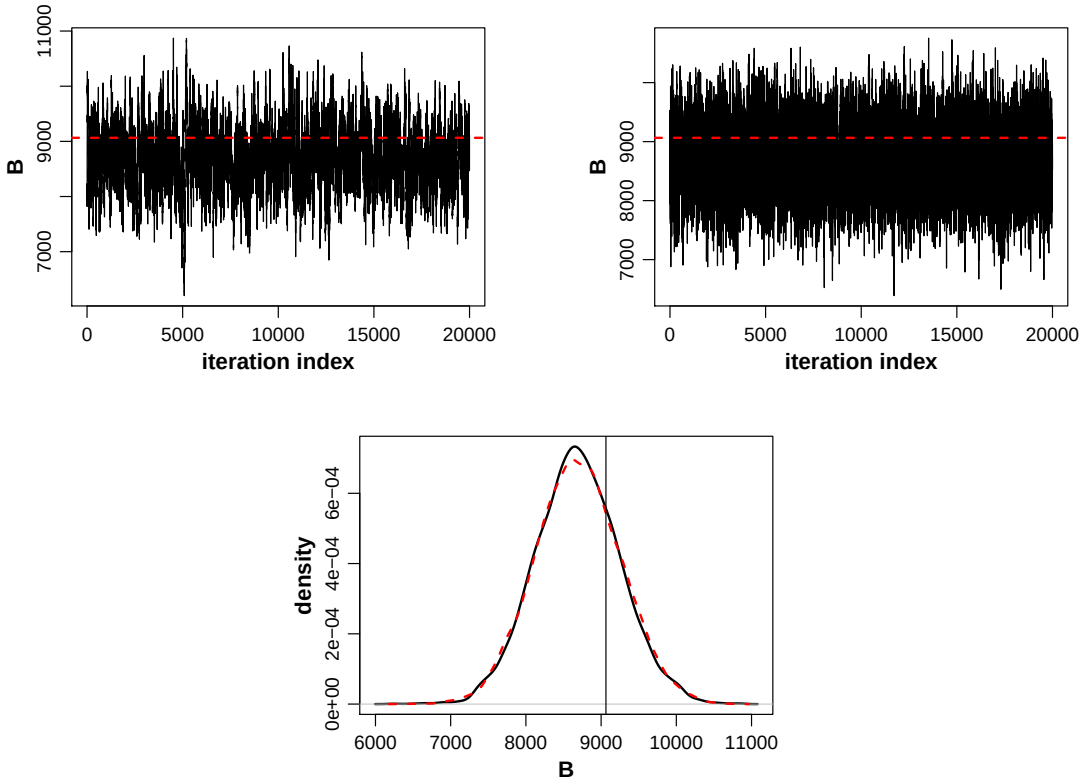


Figure A.24: MCMC convergence diagnostic plots for dataset 12 ($R_0 = 2.5, \nu = 5, N = 1000$) of simulation study G. The simulation and run conditions are described in section 4.3.3.2. Top plots are trace plots for $B = \sum_{k=1}^{n}(r_k - i_k)$. Left plot corresponds to the standard-block MCMC algorithm and right plot corresponds to the DIS-block MCMC algorithm. Imposed (red, dashed, horizontal line) is the true vale of B. Bottom plot is the posterior density of B, based on the MCMC sample of the standard-block MCMC algorithm (black, solid line) and the DIS-block MCMC algorithm (red, dashed line). Imposed (black, solid, vertical line) is the true vale of B.

Appendix B

Supplementary Material

B.1 Homogeneous Poisson process

B.1.1 Definition

There are more than one (equivalent) ways for defining the homogeneous Poisson Process (HPP); see any standard applied probability and stochastic process textbook such as (Ross, 2009, chapter 5). Let $X_1, X_2, ...$ be i.i.d. random variables, having an Exp(ρ) distribution, corresponding to times between events that occur randomly in time. Define $S_n = \sum_{k=1}^{n} X_k$ for $n \geq 1$ and set $S_0 = 0$, i.e. S_n is the waiting time until the n^{th} event. Let $N_t = \max\{n : S_n \leq t\}$, i.e. N_t is the total number of events up to time t. Then the counting stochastic process $\{N_t\}_{t\in\mathbb{R}}$ is a HPP of rate ρ.

B.1.2 Likelihood

Suppose that a HPP of rate ρ is realized in a time window $[T_{\text{on}}, T_{\text{off}}]$ and the observed time-ordered event times are $\boldsymbol{r} = (r_1, r_2, ..., r_n)$. Then the likelihood of the HPP is given by

$$\pi(\boldsymbol{r}|T_{\text{on}}, T_{\text{off}}, \rho) = \rho^n \exp\big(-\rho(T_{\text{off}} - T_{\text{on}})\big). \tag{B.1}$$

B.1.3 Bayesian inference and MCMC algorithm

In the case that the HPP is used to model removal times of an epidemic outbreak (as it is used in the context of this thesis), the time window is unknown and thus T_{on} and T_{off} are unknown parameters that must be estimated from the data. The target posterior density is $\pi(T_{\text{on}}, T_{\text{off}}, \rho|\boldsymbol{r}) \propto \pi(\boldsymbol{r}|T_{\text{on}}, T_{\text{off}}, \rho)\pi(T_{\text{on}}, T_{\text{off}}, \rho)$, where $\pi(T_{\text{on}}, T_{\text{off}}, \rho)$ is the joint prior density of the parameters T_{on}, T_{off} and ρ.

Assuming prior independence for the three parameters, Exponential prior distributions are put on $r_1 - T_{\text{on}}$ and $T_{\text{off}} - r_n$. This ensures that T_{on} and T_{off} have the desired support, i.e. T_{on} is before the first removal time r_1 and T_{off} is after the last removal time r_n. For ρ, a Gamma prior distribution is used. More specifically, the prior distribution assignment is done as follows.

$$
\begin{aligned}
r_1 - T_{\text{on}} &\sim \text{Exp}(\gamma_{T_{\text{on}}}) \\
T_{\text{off}} - r_n &\sim \text{Exp}(\gamma_{T_{\text{off}}}) \\
\rho &\sim \text{Gamma}(\nu_\rho, \lambda_\rho).
\end{aligned}
$$

The specific choice of prior distributions results in conjugancy for the three full conditional distributions and sampling from the target posterior distribution can easily be achieved via an MCMC algorithm with three Gibbs steps, as in Algorithm 21 below.

Whenever the HPP model is fitted in this thesis, the prior distribution parameters are set so that all three parameters have uninformative $\text{Exp}(10^{-3})$ prior distributions, i.e. the prior parameters are set as $\gamma_{T_{\text{on}}} = \gamma_{T_{\text{off}}} = \lambda_\rho = 10^{-3}$, $\nu_\rho = 1$.

Algorithm 21 MCMC algorithm for the HPP model

1. Suppose the current state is $(T_{\text{on}}^{(s)}, T_{\text{off}}^{(s)}, \rho^{(s)})$
2. Sample $r_1 - T_{\text{on}}^{(s+1)} \sim \pi(r_1 - T_{\text{on}} \mid \boldsymbol{r}, \rho^{(s)}) \equiv \text{Exp}(\rho^{(s)} + \gamma_{T_{\text{on}}})$ using a Gibbs step
3. Sample $T_{\text{off}}^{(s+1)} - r_n \sim \pi(T_{\text{off}} - r_n \mid \boldsymbol{r}, \rho^{(s)}) \equiv \text{Exp}(\rho^{(s)} + \gamma_{T_{\text{off}}})$ using a Gibbs step
4. Sample $\rho^{(s+1)} \sim \pi(\rho \mid \boldsymbol{r}, T_{\text{on}}^{(s+1)}, T_{\text{off}}^{(s+1)}) \equiv \text{Gamma}(n + \nu_\rho, T_{\text{off}}^{(s+1)} - T_{\text{on}}^{(s+1)} + \lambda_\rho)$ using a Gibbs step
5. Set the next state as $(T_{\text{on}}^{(s+1)}, T_{\text{off}}^{(s+1)}, \rho^{(s+1)})$.

B.2 Probability mass function of $\boldsymbol{g}^{\boldsymbol{e}^{sam}} \sim H_0$

Consider the notation of sections 3.2.1 and 3.2.2. The random vector $\boldsymbol{g}^{\boldsymbol{e}^{sam}} = (g_1^{\boldsymbol{e}^{sam}}, g_2^{\boldsymbol{e}^{sam}}, ..., g_n^{\boldsymbol{e}^{sam}}) \sim H_0$ has support $g_k^{\boldsymbol{e}^{sam}} \in \{1, 2, ..., l\}$, $k = 1, 2, ..., n$, and for household label vector $\boldsymbol{g}^{\boldsymbol{e}} = (g_1^{\boldsymbol{e}}, g_2^{\boldsymbol{e}}, ..., g_n^{\boldsymbol{e}})$, corresponding to time-ordered event times $\boldsymbol{e} = (e_1, e_2, ..., e_n)$, its joint probability mass function (p.m.f.) $f_{\boldsymbol{g}^{\boldsymbol{e}^{sam}}}(\boldsymbol{g}^{\boldsymbol{e}})$ is given by

$$\begin{aligned} f_{\boldsymbol{g}^{\boldsymbol{e}^{sam}}}(\boldsymbol{g}^{\boldsymbol{e}}) &= P(\boldsymbol{g}^{\boldsymbol{e}^{sam}} = \boldsymbol{g}^{\boldsymbol{e}}) = P(g_1^{\boldsymbol{e}^{sam}} = g_1^{\boldsymbol{e}}, g_2^{\boldsymbol{e}^{sam}} = g_2^{\boldsymbol{e}}, ..., g_n^{\boldsymbol{e}^{sam}} = g_n^{\boldsymbol{e}}) \\ &= P(g_1^{\boldsymbol{e}^{sam}} = g_1^{\boldsymbol{e}}) P(g_2^{\boldsymbol{e}^{sam}} = g_2^{\boldsymbol{e}} \mid \mathcal{H}_{e_2^-}) \dots P(g_n^{\boldsymbol{e}^{sam}} = g_n^{\boldsymbol{e}} \mid \mathcal{H}_{e_n^-}) \end{aligned} \tag{B.2}$$

where, the first of the above terms (the marginal p.m.f. of $g_1^{\boldsymbol{e}^{sam}}$) is calculated as $P(g_1^{\boldsymbol{e}^{sam}} = m) = \frac{C_m}{C}$, for $m = 1, 2, ..., l$, and the remaining terms (the marginal p.m.f. of $g_k^{\boldsymbol{e}^{sam}}$, conditioned on $\mathcal{H}_{e_k^-}$, $k = 2, 3, ..., n$) as $P(g_k^{\boldsymbol{e}^{sam}} = m \mid \mathcal{H}_{e_k^-}) = \frac{C_m - \nu_{\mathcal{H}_{e_k^-}}^{(m)}}{C-(k-1)}$, for $m = 1, 2, ..., l$, $k = 2, 3, ..., n$, where $\mathcal{H}_{t^-}$ denotes the history of the process up to time t^- (where t^- is the time just before time t) and $\nu_{\mathcal{H}_{t^-}}^{(m)}$ denotes the number of times that the label of household m appears in $\boldsymbol{g}^{\boldsymbol{e}}$, up to time t^-.