Identifying Clusters in Bayesian Disease Mapping

Craig Anderson

A Dissertation Submitted to the

University of Glasgow

for the degree of

Doctor of Philosophy

School of Mathematics & Statistics

December 2014

Abstract

This thesis develops statistical methodology for disease mapping, an increasingly important field of spatial epidemiology. Disease mapping has applications in public health by allowing for identification of areas which are at high risk of particular health problems. Such approaches are generally based on areal data, which involves partitioning the study region into a set of non-overlapping areal units and recording counts of disease cases within each areal unit. The majority of approaches assume a spatially smooth risk surface, but this may not be realistic, and there has been recent interest in developing methodology which allows for discontinuities in this structure. This can be done by identifying clusters of areal units with similar disease risks, and allowing for discontinuities between these clusters. The work presented in this thesis develops models to identify such clusters and also estimate disease risk. Three Bayesian hierarchical models are proposed; the first two are based on spatial data at a single time point, while the third extends into the spatio-temporal domain by modelling across multiple time points. Each model is applied to respiratory hospital admission data from the Greater Glasgow and Clyde Health Board area in order to identify clusters which have high disease risk.

Acknowledgements

I would like to thank Dr Duncan Lee and Dr Nema Dean for their guidance, support and friendship over the duration of my PhD, both have helped make my experience an almost entirely positive one. Without them, this work would never have been completed. Thanks to Linsay, for her constant support, her understanding during the more difficult spells, and for always keeping me positive and cheerful. I would also like to thank my mum and dad without whom I would never have been here to complete this work. Their continued emotional, moral and indeed financial support have been invaluable in getting me to where I am now.

I would also like to thank the Carnegie Trust for their generous funding which made this work possible.

Declaration

I have prepared this thesis myself; no section of it has been submitted previously as part of any application for a degree. I carried out the work reported in it, except where otherwise stated.

The work presented in Chapters 4 and 5 has been published in the Biostatistics journal with the title *"Identifying clusters in Bayesian disease mapping"* (Volume 15, p457-469), and is jointly authored by Dr Duncan Lee and Dr Nema Dean. I delivered an invited talk on this work at the GEOMED Conference in Sheffield, UK in 2013. The work presented in Chapters 4 and 6 was also presented at the COMPSTAT conference in Geneva, Switzerland in 2014, and was published in the conference proceedings with the title *"Bayesian cluster detection via adjacency modelling"*.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The population level risk of a particular disease can often vary across geographical regions, and it is of great interest to governments, health authorities and policy makers to explore these variations in disease risk in order to identify possible underlying reasons for these differences. Such analysis has the potential to identify previously unidentified environmental exposures which may be responsible for the different disease risk levels in different areas. One of the earliest examples of this was the 1854 cholera outbreak in Soho, London, which was responsible for more than 600 deaths. At the time, it was believed that such diseases were transmitted via air, but physician John Snow produced one of the first known disease maps (Figure 1.1) which showed that the cholera cases were grouped around a water pump on Broad Street (Snow (1855)). These findings proved to be crucial in developing the understanding that diseases such as cholera are spread by polluted water supplies, which led to modernisation of water supplies and sanitation systems in London

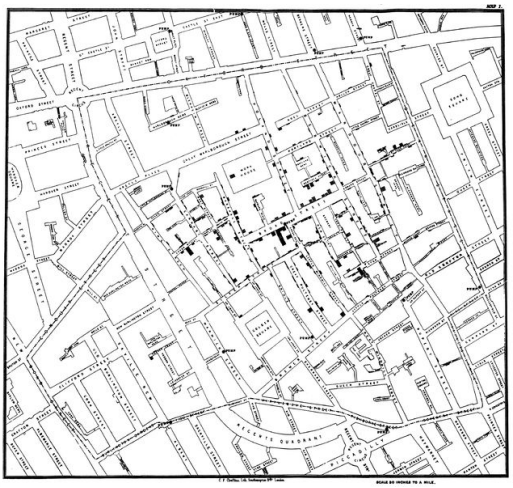

Figure 1.1: The original map of cholera cases in Soho, London, constructed by John Snow in 1854, and reproduced from Snow (1855).

and across the world. Later, Palm (1890) studied the geographical spread of rickets, and identified that the disease was more prevalent in areas with less sunlight. His findings would eventually lead to an understanding that rickets was caused by a vitamin D deficiency, one of the causes of which was a lack of exposure to sunlight.

In general, the location itself (i.e. a set of geographical co-ordinates) is unlikely to affect the risk of any particular disease; there is no reason why one set of co-ordinates would inherently have more risk than another. Instead, geographical location is generally a proxy measure for differences in the characteristics of the areas. These differences could be in terms of physical geography (e.g. temperature, sunlight, altitude), environmental factors (e.g. air quality, water quality), or population behaviour (e.g. diet, exercise, smoking prevalence, alcohol consumption). Identifying differences in disease

risk across a geographical region can prompt further investigation into the underlying reasons for the differences, which can lead to health breakthroughs such as those identified by Snow (1855) and Palm (1890). In addition, the identification of high risk areas allows health authorities to focus extra resources on these areas in an attempt to change the population behaviours which contribute to increased disease risk.

Most disease mapping approaches are based on the geographical region being partitioned into areal units, with the disease risks being estimated for each of these areas. This is because individual level data would breach patient confidentiality, and because governments are more interested in risk levels for populations as a whole. Each areal unit will have different population demographics, so comparisons between areal units are usually based on the standardised incidence ratio (SIR), which is the number of observed cases in the area divided by the number of cases expected for the area based on its population demographics. Much of the modern methodology for estimating disease risk relies on conditional autoregressive (CAR) models (Besag et al. (1991)), which assume spatial autocorrelation between neighbouring areas based on the idea that nearby areas are likely to have more in common than those which are further apart. This is because neighbouring areas are more likely to share similar socio-economic characteristics in terms of deprivation and population behaviour. These models assume that this level of spatial autocorrelation is constant across the entire spatial region, but there are many cases where this is not realistic, and there is increasing interest in developing models which allow for discontinuities in this spatial autocorrelation pattern.

Some of these models have the aim of identifying groups of areal units that exhibit substantially different risks compared to their neighbours by partitioning the areal units into a set of disease risk clusters.

The main aim of this thesis will be to develop new spatial and spatio-temporal methodology which can simultaneously identify disease clusters and estimate disease risk. This will then be applied to data for the Greater Glasgow and Clyde Health Board area to identify areal units which are at high risk of respiratory disease. The majority of risk estimation is based on the assumption of a spatially smooth surface via CAR models, while clustering is generally based on identifying substantial differences between neighbouring areas via methods such as SaTScan (Kulldorff (1997)). These are therefore two inherently different and conflicting aims, because if neighbouring areas are smoothed towards each other then it is not particularly sensible to look for clusters in that smoothed risk surface. It is therefore of interest to develop methodology which can carry out both smoothing and clustering simultaneously. Two separate problems are tackled within this thesis; the first is to develop a modelling approach for a single time point, and the second is to develop a spatio-temporal model which can identify changes in the disease risk pattern over multiple time points.

The single time point problem is outlined in Chapters, 4, 5 and 6, and involves estimating the disease clusters and risk levels at a particular point in time. Such modelling approaches allow health authorities to identify areas which had high (or low) disease risk in the particular year being studied and

can therefore be used to drive public health policy across the study region. We propose a novel spatial agglomerative hierarchical clustering approach which uses prior data to produce a set of potential cluster structures, and then develop two distinct Bayesian modelling approaches for selecting the best of these cluster structures. The first approach fits a separate Poisson log-linear model for each possible cluster structure and compares them via the Deviance Information Criterion (DIC). This model assigns different mean risk levels to each cluster via a fixed effect and allows disease risk to follow a spatially smooth pattern within a cluster whilst having a disjoint jump between clusters. The second approach consists of a single Poisson log-linear model with the optimal cluster structure estimated as a parameter within that model. This model uses a set of random effects which allow the disease risk to be correlated for pairs of neighbouring areal units within a cluster, but conditionally independent for pairs of areal units in different clusters.

The spatio-temporal problem is outlined in Chapter 7, and involves estimating disease clusters and risk levels over multiple time points. This allows health authorities to model trends in disease risk as well as identifying areas which are at high (or low) risk on average, and more resources can be focused on areal units with an increasing disease risk level. We propose a novel spatio-temporal Bayesian modelling approach which divides the areal units into clusters based on both their average risk (intercept) and the change in their disease risk over time (slope). This model has separate sets of correlated random effects for the intercept and slope, and additional cluster-specific fixed effects for the intercept and slope terms.

The remainder of this thesis is divided into seven chapters. Chapter 2 provides an overview of the existing statistical methodology which will be used in this thesis as well as the related literature. Chapter 3 introduces disease mapping and provides a critique of the existing disease mapping literature, focusing on the standard methods used in both spatial and spatio-temporal contexts and highlighting some of the deficiencies therein which are addressed within this thesis. In Chapter 4, a novel spatial agglomerative clustering approach is developed and is applied to respiratory hospital admission data for the Greater Glasgow and Clyde Health Board area to produce a set of potential cluster structures for disease risk. These cluster structures are used in Chapters 5 and 6, where two alternative Bayesian modelling approaches are developed for simultaneously estimating disease risk and the spatial cluster structure within the Glasgow region. The model developed in Chapter 5 uses a mean-based (fixed effects) approach while that in Chapter 6 uses a variance-based (random effects) approach. Chapter 7 discusses a new Bayesian spatio-temporal model which identifies the change in spatial pattern over time. Finally, Chapter 8 summarises the work contained within this thesis and discusses the implications for future disease mapping research. The Bayesian model presented in Chapter 5 was based on integrated nested Laplace approximations, while the models in Chapters 6 and 7 are based on Markov chain Monte Carlo algorithms, all of which were written in the R statistical language (R Development Core Team (2008)). Part of the algorithm for model in Chapter 7 was written in the C++ language using the Rccp package (Eddelbuettel and François (2011)).

Chapter 2

Statistical background

This chapter outlines the statistical theory and methodology used and developed throughout this thesis, and also provides an overview of the existing literature within these areas of statistics. Section 2.1 introduces Bayesian statistics, which is the statistical framework employed throughout this thesis. The concepts of prior and posterior distributions are introduced, and methods of inference for Bayesian approaches are discussed. Section 2.2 explores generalised linear models (GLMs) and their uses, with a particular focus on the Poisson GLMs which are used in the spatial modelling approaches in this thesis. Spatial modelling is introduced in Section 2.4, and this will form the basis of the methodology developed in Chapters 5 and 6. Section 2.5 gives a brief outline of spatio-temporal modelling, which will be the focus of the methodology developed in Chapter 7. Clustering approaches form part of the new modelling methodology developed in Chapters 5, 6 and 7, and the concept of clustering is outlined in Section 2.6.

2.1 Bayesian modelling

2.1.1 Introduction to Bayesian statistics

In any statistical modelling approach we have a vector of observed data, $\boldsymbol{Y} = (Y_1, \ldots, Y_n)$, which are believed to have come from a probability model, $f(\boldsymbol{Y}|\boldsymbol{\theta})$, with a set of unknown parameters $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)$. The aim of statistical modelling is to use the data to infer the best possible estimate of the values of these unknown parameters. Under the likelihood approach, the parameters are estimated as the value, $\hat{\boldsymbol{\theta}}$, which maximises the likelihood function, denoted by $L(\boldsymbol{\theta}|Y) = \prod_{i=1}^{n} f(Y_i|\boldsymbol{\theta})$ where $Y_1, \ldots, Y_n$ are assumed to be independent. Under this framework, it is assumed that the unknown true values of the model parameters $\boldsymbol{\theta}$ are fixed, with inference based on a point estimate $\hat{\boldsymbol{\theta}}$ (e.g. the maximum likelihood estimator) and the uncertainty of that estimate specified by a $c\%$ confidence interval. The definition of these intervals is that if the data were repeatedly sampled and an interval constructed each time, then $c\%$ of these intervals would contain the "true" value of the parameter.

An alternative to the likelihood framework is the Bayesian approach to statistics, which has its roots in Bayes' Theorem, developed by Thomas Bayes in the 18th century (Bayes (1764)). Bayes' Theorem is a mathematical formulation of the natural idea that our estimates should change in light of observed evidence, and is defined as follows for events A and B:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)},$$

where $P(A|B)$ is the conditional probability of the event A happening given that event B has occurred, and $P(A), P(B)$ are the probabilities of events A and B occurring.

This can be adapted to provide a basis of inference for model parameters. As with the likelihood approach, the data $\boldsymbol{Y}$ are used to estimate the likely values of the parameters $\boldsymbol{\theta}$, but in the Bayesian case the parameters are treated as random and can therefore have probability distributions assigned to them. Our uncertainty about the parameter values can therefore be expressed in advance by assigning each parameter a distribution known as a prior, $f(\boldsymbol{\theta})$, which is discussed in more detail in Section 2.1.2. The prior beliefs about the parameter values can then be updated in light of the observed data, $\boldsymbol{Y}$, via the data likelihood, $f(\boldsymbol{Y}|\boldsymbol{\theta})$, in order to determine a posterior distribution, $f(\boldsymbol{\theta}|\boldsymbol{Y})$ for each parameter using an adaptation of Bayes' Theorem as follows:

$$f(\boldsymbol{\theta}|\boldsymbol{Y}) = \frac{f(\boldsymbol{Y}|\boldsymbol{\theta})f(\boldsymbol{\theta})}{f(\boldsymbol{Y})},$$

where $f(\boldsymbol{\theta})$ is the joint prior distribution for the parameters $\boldsymbol{\theta}$, and $f(\boldsymbol{Y})$ is the marginal distribution of the observed data, $\boldsymbol{Y}$. However, the distribution $f(\boldsymbol{Y})$ can often be difficult to estimate, and since it has no dependence on $\boldsymbol{\theta}$, the posterior distribution can instead be expressed up to a constant of proportionality as

$$f(\boldsymbol{\theta}|\boldsymbol{Y}) \propto f(\boldsymbol{Y}|\boldsymbol{\theta})f(\boldsymbol{\theta}),$$

which is the product of the likelihood function and the prior distribution.

Point estimates of $\boldsymbol{\theta}$ are taken to be a central value of the posterior distribution, with the posterior mean or median generally used. Unlike the likelihood approach, the posterior distribution can be interpreted to provide probabilistic statements about the model parameters, $\boldsymbol{\theta}$. Uncertainty is estimated via a $c\%$ credible interval, which has the interpretation that the parameter will lie within the interval with probability $\frac{c}{100}$.

The choice between the likelihood and Bayesian inference is the subject of much philosophical debate amongst statisticians. One of the key differences comes in terms of the way a probability is interpreted within each framework. Under the likelihood approach, probabilities are considered to be fixed values which are representations of the relative frequency of an event occurring over a large number of repeatable events, while under the Bayesian framework, a probability is interpreted as the (often subjective) plausibility of a particular statement being true, or a particular outcome occurring, and can be updated in the face of evidence. The differences between these approaches can be outlined by the simple example of a coin toss, where the probability of a head is 0.5. Under the likelihood approach, the interpretation of that probability would be that, given an infinite number of coin tosses, 50% of coins will land on a head. Under the Bayesian framework, this probability

would represent a belief in the absence of any evidence to the contrary, that either outcome is equally likely. Despite the philosophical differences between these approaches, in practice it is often possible to fit the same model in both frameworks and obtain similar results. The modelling approaches developed within this thesis will be outlined in a Bayesian setting.

2.1.2 Prior distributions

The concept of a prior distribution was briefly introduced in Section 2.1, and forms a crucial part of Bayesian inference. A prior distribution, $f(\boldsymbol{\theta})$ represents all of the information which is known about the parameters $\boldsymbol{\theta}$, in advance of observing the data $\boldsymbol{Y}$. This prior distribution could be based on information from previous studies on similar data sets or an estimate from an expert in the field, or it could simply be used to represent a position of prior ignorance. Prior distributions can take a variety of forms depending on the type of model and data being used; it is possible to choose a univariate prior for each individual parameter (assuming independence between the parameters), that is $f(\boldsymbol{\theta}) = \prod_{j=1}^{p} f(\theta_j)$, or a single multivariate prior for all parameters together, or, as will be the case in this thesis, a combination of multivariate and univariate prior distributions. The parameters of these prior distributions are known as hyperparameters.

The choice of prior distribution will influence the posterior distribution obtained, so it is important to make a sensible choice of prior in order to produce a sensible estimate for the parameters. This choice is not always

straightforward; in some cases we may have little or no intuition about the value of the parameter in advance of observing the data. In such cases, it is possible to represent our lack of prior knowledge by assigning a weakly informative prior which will have a negligible effect on the posterior, thus allowing the posterior distribution $f(\boldsymbol{\theta}|\boldsymbol{Y})$ to be driven by the data rather than the choice of prior. Examples of weakly informative priors include a Gaussian distribution with a very large variance ($\theta_k \sim$ N(0,1000)) for real valued parameters, a uniform distribution which covers the entire possible range of values ($\theta_k \sim$ Uniform(0,1) prior for a parameter on the unit interval) and a weakly informative Gamma or uniform prior on the positive real line for a variance parameter. Completely non-informative priors can take the form of distributions without a finite density, for example Uniform$(-\infty, \infty)$. These are known as improper priors, but care should be taken when using them because in many cases they can lead to an improper posterior distribution which makes inference impossible. Another form of prior is the Jeffreys prior (Jeffreys (1946)), which is designed to be invariant under reparameterisation. These Jeffreys priors are of the form $f(\boldsymbol{\theta}) \propto \sqrt{\det I(\boldsymbol{\theta})}$ where $I(\boldsymbol{\theta})$ is the Fisher information, defined as

$$I(\boldsymbol{\theta}) = E\left[\left(\frac{\partial}{\partial \boldsymbol{\theta}} \log f(\boldsymbol{Y};\boldsymbol{\theta})\right)^2 \middle| \boldsymbol{\theta}\right] = \int \left(\frac{\partial}{\partial \boldsymbol{\theta}} \log f(\boldsymbol{Y};\boldsymbol{\theta})\right)^2 f(\boldsymbol{Y};\boldsymbol{\theta}) dy$$

.

In the models developed throughout this thesis we make use of both informative and weakly informative priors.

In some cases it may be possible to use a conjugate prior; these are prior distributions which result in the posterior distribution following the same distributional form as the prior. These types of priors are convenient because the posterior distribution will be part of a standard distributional family and are therefore straightforward to evaluate as a closed-form expression. For example, if we have a single observation $Y \sim N(\mu, \tau)$ then an example of a conjugate prior for the precision parameter τ would be $\tau \sim \text{Gamma}(\alpha, \beta)$. Here, the posterior distribution, conditional on μ, would be $f(\tau|\mu, Y) \sim \text{Gamma}(\frac{1}{2} + \alpha, \frac{1}{2}(Y - \mu)^2 + \beta)$.

2.1.3 Inference

Bayesian modelling relies on the ability to compute posterior distributions in order to provide estimates for each of the model parameters. Some of these posterior distributions are straightforward to compute; for example, as discussed in Section 2.1.2, distributions with a conjugate prior usually have a posterior distribution which follows a standard distributional form. In many cases, however, the computation required is more complex and a more advanced approach is required to calculate the posterior distribution. These advanced methods commonly make use of some form of numerical simulation, generally by drawing a sample of parameter values from an approximation of the posterior distribution $f(\boldsymbol{\theta}|\boldsymbol{Y})$ to enable estimation of the distributions of the model parameters.

Markov chain Monte Carlo simulation

Markov chain Monte Carlo (McMC) simulation is by far the most common of these simulation approaches for evaluating the posterior distribution when the likelihood is tractable. McMC simulation works by constructing a Markov chain with properties which allow it to converge to the desired joint posterior distribution $f(\boldsymbol{\theta}|\boldsymbol{Y})$ after a finite number of iterations. A set of starting parameters, $\boldsymbol{\theta}^{(0)} = (\theta_1^{(0)}, \ldots, \theta_p^{(0)})$ are defined, usually via the priors, and these should be updated iteratively until convergence is achieved. In some cases, multiple Markov Chains are run from different starting points in order to measure convergence. In order to update the parameters, the parameter vector is partitioned into a set of b blocks, with $\boldsymbol{\theta} = (\boldsymbol{\zeta}_1, \ldots, \boldsymbol{\zeta}_b)$. Here, $\boldsymbol{\zeta}_k = (\theta_{r+1}, \ldots, \theta_{r+s})$ where θ_r is the final parameter in block $\boldsymbol{\zeta}_{k-1}$ and s is the number of parameters in block $\boldsymbol{\zeta}_k$. Each block of parameters is updated in turn, with new values proposed for each parameter at each iteration of the algorithm. These proposed values can either be accepted or rejected, and the percentage of proposals accepted for each block over the full set of iterations is known as the acceptance rate. The case where $b = 1$ corresponds to updating all parameters at once, which is likely to be faster computationally but will lead to lower acceptance rates. The case where $b = p$ corresponds to all parameters being updated individually, which has higher acceptance rates but is computationally slower. The optimal design therefore often lies somewhere between these two, allowing the acceptance rate to remain reasonably high whilst allowing for faster computational speed. Typically, sets of parameters with similar characteristics can be updated in the same block. The appropriate level of blocking depends on the context of the problem.

In this thesis, convergence is determined via visual assessment of trace plots, where it is considered that convergence is achieved when the trace plot looks weakly stationary. It is, however, noted that an objective criteria for checking convergence for multiple Markov chains was proposed by Gelman and Rubin (1992), based on a weighted average of within-chain and between-chain variances. It is also important to ensure that the Markov chain has the opportunity to explore the entire parameter space in order to provide a good estimate of the posterior distribution. This is known as mixing of the Markov chain, and can be monitored via the acceptance rates for each parameter or block. A low acceptance rate indicates that too few of the proposal parameter values are being accepted due to too much exploration beyond the support of the posterior density. An acceptance rate which is too high means that too many proposal parameter values are being accepted, due to the chain not exploring the full posterior density. Mixing can also be monitored by visual assessment of trace plots; regular movement of the chain corresponds to good mixing, while long periods without movement suggests poor mixing.

Figure 2.1 displays the trace plots for independent Markov chains for the same model parameter with two different starting values. The results from the second independent chain with a different starting value reinforce the results of the first chain, making it very unlikely that the first chain had converged on a local mode. These chains both converge on a very similar posterior distribution despite starting relatively far apart, which suggests

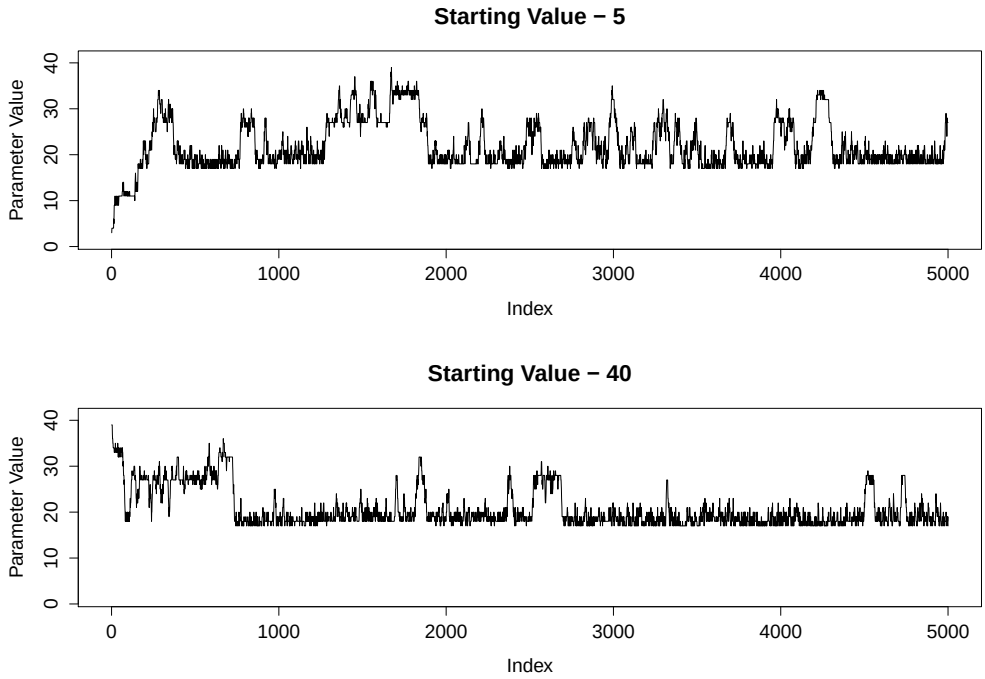

Figure 2.1: Comparison of two Markov-chains with different starting values

that the distributions obtained are more likely to be a good estimate of the true posterior. This illustrates that, where it is computationally possible, it is preferable to run multiple Markov chains in order to assess convergence.

The parameter estimates obtained before convergence should be disregarded; this is known as a "burn-in" period. After the chain has converged on the desired joint distribution, we are able to draw samples from the posterior distribution to enable estimation of the model parameters, $\boldsymbol{\theta}$. It should be noted that the nature of the Markov Chain means that correlation exists between consecutive draws from the posterior distribution; this autocorrelation means that the samples are not independent of each other. It is possible to produce

independent samples via the process of thinning, which involves storing only every kth draw (after burn-in) from the posterior distribution and discarding all others. However, thinning also increases the computational time required to obtain a set of d draws from the posterior distribution, and it has been postulated by Link and Eaton (2012) that thinning is inefficient and usually unnecessary. Thinning is not used in this thesis.

The McMC simulation within this thesis will be carried out using Gibbs sampling (Geman and Geman (1984)) and the Metropolis-Hastings algorithm (Hastings (1970)). In each of these approaches the set of parameters, $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)$ are updated in blocks, based on the current value of the other parameters. Gibbs sampling is used to simulate from the posterior distribution of a block of parameters where that block has a known full conditional distribution $f(\zeta_k|\zeta_1, \ldots, \zeta_{k-1}, \zeta_{k+1}, \ldots, \zeta_b, Y)$ with a form which lends itself to straightforward sampling. Such cases regularly occur when a conjugate prior is used and the full conditional posterior distribution comes from a standard statistical family. The Gibbs sampling algorithm for drawing d samples from the posterior distribution is as follows:

Gibbs Sampling Algorithm

1. Choose a set of initial values $\boldsymbol{\theta}^{(0)} = (\theta_1^{(0)}, \ldots \theta_p^{(0)})$ which will represent the starting point of the Markov chain.

2. For each iteration $i = 1, \ldots, d$, draw a sample $\boldsymbol{\zeta}_k^{(i)}$ for each of the $k = 1, \ldots, b$ blocks in turn from the conditional distribution $f(\boldsymbol{\zeta}_k | \boldsymbol{\zeta}_1^{(i)}, \ldots, \boldsymbol{\zeta}_{k-1}^{(i)}, \boldsymbol{\zeta}_{k+1}^{(i-1)}, \ldots, \boldsymbol{\zeta}_b^{(i-1)}, Y)$. This means that each block of parameters is sampled conditional on the current value of each of the other blocks.

In cases where the conditional distribution is not straightforward to sample from, simulation is generally carried out using the more complex Metropolis-Hastings algorithm. At each iteration of the Metropolis-Hastings algorithm, a potential new value of the model parameter is generated from a specified proposal distribution, and the likelihoods under the current and proposed value are compared in order to decide whether this proposed value should be accepted or rejected. The Metropolis-Hastings algorithm for drawing d samples from the posterior distribution is as follows:

Metropolis-Hastings Algorithm

1. Choose a set of initial values $\boldsymbol{\theta}^{(0)} = (\theta_1^{(0)}, \ldots \theta_p^{(0)})$ which will represent the starting point of the Markov chain.

2. For each iteration $i = 1, \ldots, d$, draw a sample $\boldsymbol{\zeta}_k^{(i)}$ for each of the $k = 1, \ldots, b$ blocks of parameters using the following steps.

   (a) Generate a set of proposed parameter values $\boldsymbol{\zeta}_k^*$ from the proposal distribution $g(\boldsymbol{\zeta}_k^*|\boldsymbol{\zeta}_k^{(i-1)})$. This proposal distribution is typically based on the current value of the parameters in the block, $\boldsymbol{\zeta}_k^{i-1}$.

   (b) Accept the proposed set of values $\boldsymbol{\zeta}_k^*$ with probability

$$p = \min\left\{1, \frac{f(\boldsymbol{\zeta}_k^*|\boldsymbol{Y})g(\boldsymbol{\zeta}_k^{(i-1)}|\boldsymbol{\zeta}_k^*)}{f(\boldsymbol{\zeta}_k^{(i-1)}|\boldsymbol{Y})g(\boldsymbol{\zeta}_k^*|\boldsymbol{\zeta}_k^{(i-1)})}\right\},$$

   and reject it with probability $1 - p$.

   (c) If the proposal is accepted then set $\boldsymbol{\zeta}_k^{(i)} = \boldsymbol{\zeta}_k^*$, and if the proposal is rejected then set $\boldsymbol{\zeta}_k^{(i)} = \boldsymbol{\zeta}_k^{(i-1)}$.

The Metropolis algorithm (Metropolis et al. (1953)) is a special case of this algorithm where the proposal distribution is symmetric; that is $g(\boldsymbol{\zeta}_k^{(i-1)}|\boldsymbol{\zeta}_k^*) = g(\boldsymbol{\zeta}_k^*|\boldsymbol{\zeta}_k^{(i-1)})$. The Metropolis algorithm therefore follows the algorithm outlined above with the exception that in step 2(b), the acceptance probability is equal to $p = \min\left\{1, \frac{f(\boldsymbol{\zeta}_k^*|\boldsymbol{Y})}{f(\boldsymbol{\zeta}_k^{i-1}|\boldsymbol{Y})}\right\}$.

For all three algorithms, the first m samples are discarded due to the burn-in period, leaving a set of $d-m$ simulated draws, $\{\theta_k^{(m+1)}, \ldots \theta_k^{(d)}\}$ from the posterior distribution of each of the k model parameters. This set of draws can then be used to make a variety of inferential statements about the parameter, such as the probability of the parameter being above a certain value. Where appropriate, this entire posterior distribution can be reproduced as part of the results of the analysis (eg Figure 6.2), but in many cases it is not practical to report results in terms of a large set of draws from a posterior distribution. Instead it is often necessary to condense this set of draws into a single point estimate, or to produce a credible interval for the parameter value. Such an approach does discard a great deal of information, but has the advantage of providing a single "take home" estimate which can be more easily digested by non-statisticians. The point estimate of the parameter value is generally taken to be a central value (usually the mean or median) of these $d-m$ simulated draws. For example, if the posterior mean was used then the parameter value would be estimated as $\hat{E}(\theta_k|\boldsymbol{Y}) = \frac{1}{d-m}\sum_{i=m+1}^{d}\theta_k^{(i)}$. A 95% credible interval is easily obtained by taking the 2.5th and 97.5th percentiles of the simulated posterior draws as the lower and upper bounds respectively.

Integrated nested Laplace approximation

An increasingly common alternative to McMC is Integrated nested Laplace approximation (INLA, Rue et al. (2009)). This approach is used to estimate approximations to the univariate full conditional posterior distributions, thus eliminating the need to simulate from the posterior. The key advantage of this approach is its speed compared to McMC simulation; inference using

INLA has been shown by Schrödle et al. (2011) to produce almost identical results to McMC simulation in a much quicker time. This approach is used where the model has a latent Gaussian Markov Random field, with the parameters of interest being latent variables which are not observed directly, but are instead inferred from other observed variables.

Consider the following hierarchical model, which will appear in Section 3.3.

$$\begin{aligned} Y_i &\sim \text{Poisson}(\mu_i) \qquad i = 1, \ldots, n, \\ \log(\mu_i) &= \boldsymbol{x_i^T \beta} + \phi_i. \end{aligned}$$

Here, $\boldsymbol{\phi} = (\phi_1, ,\phi_n)$ is a set of random effects, which can be considered to be the set of latent variables. Let $\boldsymbol{\omega}$ be the set of hyperparameters relating to $\boldsymbol{\phi}$, then the marginal posterior for each variable ϕ_i is as follows:

$$\pi(\phi_i|\boldsymbol{Y}) = \int_{\omega} \int_{\boldsymbol{\phi}^{-i}} \pi(\boldsymbol{\phi}, \boldsymbol{\omega}|\boldsymbol{Y}) d\boldsymbol{\phi}_{-i} d\boldsymbol{\omega},$$

where $\boldsymbol{\phi}^{-i}$ is the vector $\boldsymbol{\phi}$ with element ϕ_i removed. This can be rewritten as

$$\pi(\phi_i|\boldsymbol{Y}) = \int_{\boldsymbol{\omega}} \pi(\phi_i|\boldsymbol{\omega}, \boldsymbol{Y}) \pi(\boldsymbol{\omega}|\boldsymbol{Y}) d\boldsymbol{\omega}. \tag{2.1}$$

INLA involves the construction of a nested approximation of (2.1), which requires approximations of $\pi(\boldsymbol{\omega}|\boldsymbol{Y})$ and $\pi(\phi_i|\boldsymbol{\omega}, \boldsymbol{Y})$. Here, $\pi(\boldsymbol{\omega}|\boldsymbol{Y})$ can be

approximated using the following Laplace approximation

$$\tilde{\pi}(\boldsymbol{\omega}|\boldsymbol{Y}) \propto \left.\frac{\pi(\boldsymbol{\phi},\boldsymbol{\omega},\boldsymbol{Y})}{\tilde{\pi}_G(\boldsymbol{\phi}|\boldsymbol{\omega},\boldsymbol{Y})}\right|_{\boldsymbol{\phi}=\boldsymbol{\phi}^*(\boldsymbol{\omega})},$$

where $\tilde{\pi}_G(\boldsymbol{\phi}|\boldsymbol{\omega},\boldsymbol{Y})$ is the Gaussian approximation to the full conditional distribution of $\boldsymbol{\phi}$ and $\boldsymbol{\phi}^*(\boldsymbol{\omega})$ is the mode of the full conditional distribution of $\boldsymbol{\phi}$ for a given value of $\boldsymbol{\omega}$.

The authors in Rue et al. (2009) propose using a Laplace approximation of $\pi(\phi_i|\boldsymbol{\omega},\boldsymbol{Y})$, which takes the following form:

$$\tilde{\pi}_{LA}(\phi_i|\boldsymbol{\omega},\boldsymbol{Y}) \propto \left.\frac{\pi(\boldsymbol{\phi},\boldsymbol{\omega},\boldsymbol{Y})}{\tilde{\pi}_G(\boldsymbol{\phi}_{-i}|\phi_i,\boldsymbol{\omega},\boldsymbol{Y})}\right|_{\boldsymbol{\phi}_{-i}=\boldsymbol{\phi}^*_{-i}(\phi_i,\boldsymbol{\omega})},$$

where $\tilde{\pi}_G(\boldsymbol{\phi}_{-i}|\phi_i,\boldsymbol{\omega},\boldsymbol{Y})$ is a Gaussian approximation to $\boldsymbol{\phi}_{-i}|\phi_i,\boldsymbol{\omega},\boldsymbol{Y}$ and $\boldsymbol{\phi}^*_{-i}(\phi_i,\boldsymbol{\omega})$ is its mode for a given value of $\boldsymbol{\omega}$.

The equation (2.1) can therefore be approximated via numerical integration as

$$\tilde{\pi}(\phi_i|\boldsymbol{Y}) = \sum_k \tilde{\pi}(\phi_i|\omega_k,\boldsymbol{Y})\tilde{\pi}(\omega_k|\boldsymbol{Y})\Delta_k$$

where Δ_k is a weight assigned to each ω_k, based on the strategy chosen for selecting the ω_k. For more details on the evaluation of these approximations, see Rue et al. (2009) or Schrödle and Held (2010). INLA can be applied

to a number of disease mapping approaches, and some examples of these implementations can be found in Ugarte et al. (2014).

Although in most cases similar results will be obtained by McMC and INLA inference, it should be noted that there are fundamental differences in the way that posterior distributions are estimated. McMC can sample directly from a joint posterior distribution, while INLA uses a closed form expression to estimate the marginal posterior distributions. Consider the example of a linear model, $y = ax + b$ given data (x=1, y=1), where priors $p(a) =$ Uniform(5,5) and $p(b) =$ Uniform(5,5) are assumed. Under McMC inference, the joint posterior distribution $p(a, b|x, y)$ can be directly estimated, and will take the form of a straight line which passes through the points (a=0, b=1) and (a=1, b=0). On the other hand, INLA estimates the marginal distributions $p(a|x, y)$ and $p(b|x, y)$ separately, and both of these will be identical to the prior. Clearly in this example, the posterior obtained from McMC inference is preferable. Care must therefore be taken when using INLA, to ensure that it is an appropriate inferential method for the model being studied.

2.2 Generalised linear models

In the statistical modelling considered in this thesis, the data generally consists of a response variable $\boldsymbol{Y} = (Y_1, \ldots, Y_n)$ and a set of covariate data $X = (\boldsymbol{x}_1^T, \ldots, \boldsymbol{x}_n^T)$, where $\boldsymbol{x}_i^T = (x_{i1}, \ldots x_{ip})$ is the set of p covariate values relating to observation i and $\boldsymbol{x}_1$ is a vector of ones for the intercept term. The aim of the modelling approach is to estimate a set of regression parameters $\boldsymbol{\beta} = (\beta_1 \ldots, \beta_p)$ which best describe the relationship between the response and these covariates. The simplest modelling approach is the linear model, which represents a linear relationship between the covariate data and the response, and takes the form:

$$\begin{aligned} Y_i &\sim \mathrm{N}(\mu_i, \sigma^2) \qquad i = 1, \ldots, n, \\ \mu_i &= \boldsymbol{x_i^T \beta}, \end{aligned}$$

where each response Y_i is assumed to be an independent normal random variable with mean μ_i and variance σ^2. In the Bayesian framework, we can assign a prior $\boldsymbol{\beta} \sim \mathrm{N}(0, \sigma_b^2)$ to represent a lack of any strong prior belief about the intercept, and a conjugate prior $\sigma^2 \sim \mathrm{InvGamma}(\alpha, \psi 1)$, and then the full conditionals are given as:

$$\begin{aligned} f(\boldsymbol{\beta}|\sigma^2, \boldsymbol{Y}) &\propto \prod_{i=1}^{n} \mathrm{N}(Y_i|\boldsymbol{x_i^T}\boldsymbol{\beta}, \sigma^2) \times \prod_{j=1}^{p} \mathrm{N}(\beta|0, \sigma_b^2) \\ &\propto \prod_{i=1}^{n} \exp\left(-\frac{(Y_i - \boldsymbol{x_i^T}\boldsymbol{\beta})^2}{2\sigma^2}\right) \times \prod_{j=1}^{p} \exp\left(-\frac{\beta_j^2}{2\sigma_b^2}\right) \end{aligned}$$

$$\begin{aligned} f(\sigma^2|\boldsymbol{\beta}, \boldsymbol{Y}) &\propto \prod_{i=1}^{n} \mathrm{N}(Y_i|\boldsymbol{x_i^T}\boldsymbol{\beta}, \sigma^2) \times \prod_{j=1}^{p} \mathrm{InvGamma}(\sigma^2|\alpha, \psi) \\ &\propto (\sigma^2)^{-\frac{n}{2}} \exp\left(\frac{\sum_{i=1}^{n}(Y_i - \boldsymbol{x_i^T}\boldsymbol{\beta})^2}{\sigma^2}\right) \times (\sigma^2)^{-(\alpha+1)} \exp\left(\frac{\psi}{\sigma^2}\right) \\ &\propto (\sigma^2)^{-(\alpha+\frac{n}{2}+1)} \exp\left(\frac{\sum_{i=1}^{n}(Y_i - \boldsymbol{x_i^T}\boldsymbol{\beta})^2 + \psi}{\sigma^2}\right) \\ &\sim \mathrm{InvGamma}\left(\alpha + \frac{n}{2}, \psi + \sum_{i=1}^{n}(Y_i - \boldsymbol{x_i^T}\boldsymbol{\beta})^2\right) \end{aligned}$$

A generalised linear model (GLM, Nelder and Wedderburn (1972)) is an extension of this linear model form, which allows for more flexibility in the modelling approach. Under this approach, there no longer needs to be a direct linear relationship between the response and the covariates, and the response variable, $\boldsymbol{Y}$ can be a set of independent random variables from any exponential family distribution, f. The exponential family is the set of statistical distributions which, for some random variable Y and some parameter θ, take the form $f(y|\theta) = \exp(a(y) + b(\theta) + c(y)d(\theta))$ where, a, b, c, d are a set of known functions. Members of this exponential family include the Gaussian, Binomial, Exponential and Poisson distributions.

A generalised linear model takes the form:

$$\begin{aligned} Y_i &\sim f(\mu_i) \qquad i = 1, \ldots, n, \\ g(\mu_i) &= \eta_i = \boldsymbol{x_i^T \beta}. \end{aligned} \tag{2.2}$$

Here, $\eta_i = \boldsymbol{x_i^T \beta}$ is known as the linear predictor, and $g()$ is a known monotonic invertible function called a link function. Common examples of the link function $g()$ include log, square root and logit transformations. Note that the linear model outlined above is a special case of the GLM, which is obtained where the link function is simply the identity function $g(\mu_i) = \mu_i$ and $f(Y_i|\mu_i) = \mathrm{N}(\mu_i, \sigma^2)$.

2.2.1 Poisson GLM

The modelling methodology developed within this thesis is based on count data, which can be represented by a Poisson distribution. The Poisson distribution is a member of the exponential family, and therefore a generalised linear model can be applied to these data. The response data from the Poisson distribution can only take a non-negative value, so the log is a suitable and commonly used link function which ensures that the model always fits non-negative values. The basic Poisson GLM can be specified as follows:

$$\begin{aligned} Y_i &\sim \mathrm{Poisson}(\mu_i) \qquad i = 1, \ldots, n, \\ \log(\mu_i) &= \boldsymbol{x_i^T \beta}. \end{aligned} \tag{2.3}$$

In the Bayesian framework, if we assign the prior $\boldsymbol{\beta} \sim \mathrm{N}(0, \sigma_b^2)$ then the full conditional is given as:

$$\begin{aligned} f(\boldsymbol{\beta}|\sigma^2, \boldsymbol{Y}) &\propto \prod_{i=1}^{n} \mathrm{Poisson}(\boldsymbol{x_i^T\beta}) \times \prod_{j=1}^{p} \mathrm{N}(\beta|0, \sigma_b^2) \\ &\propto \prod_{i=1}^{n} \exp\left(-\boldsymbol{x_i^T\beta}\right) (\boldsymbol{x_i^T\beta})^{Y_i} \times \prod_{j=1}^{p} \exp\left(-\frac{\beta_j^2}{2\sigma_b^2}\right) \end{aligned}$$

2.3 Model Comparison

In cases where multiple statistical models are being considered, it is necessary to have a method for determining which model provides the most appropriate fit to the data. The aim of any modelling approach is to provide a model which is able to provide the most accurate explanation possible for the observed data, and also in many cases to produce the most adequate predictions for future data. A model selection approach based purely on maximising the likelihood would be flawed, because adding extra parameters tends to increase the likelihood even if these extra parameters lead to overfitting. It is therefore necessary to consider approaches which provide a balance between maximising the likelihood and avoiding overparameterisation.

One such approach is the Akaike Information Criterion (AIC, Akaike (1973)), which contains a term for the number of parameters in order to penalise modelling approaches which overparameterise. The AIC is computed as AIC $= -2\log(\hat{L}) + 2k$, where $\hat{L}$ is the maximum likelihood of the model and k is

the number of independent model parameters. When comparing two or more models, the model with the lowest AIC value should be preferred. Adding an extra unnecessary parameter would have a negligible impact on the maximum likelihood, but would increase the second part of the AIC score by 2, and therefore the simpler model would be preferred.

A similar approach is the Bayesian Information Criterion (BIC, Schwarz (1978)), which is computed as $\text{BIC} = -2\log(\hat{L}) + k\log(n)$. Again, when comparing two or more models, the model with the lowest BIC value should be preferred. In this case, adding an extra unnecessary parameter would increase the second part of the BIC score by $\log n$ rather than by 2 in the AIC. This means that for cases where $n > 100$, the BIC penalises the number of parameters more strongly than the AIC.

An alternative comparison method is the Deviance Information Criterion (DIC, Spiegelhalter et al. (2002)), which is based on the model deviance. The DIC is defined as $\text{DIC} = \bar{D} + p_d$, where $\bar{D} = E[-2\log(\hat{L})]$ is the mean posterior deviance and p_d is the effective number of parameters. When comparing two or more models, the model with the lowest DIC value should be preferred. Similar to the BIC, this approach penalises models which have superfluous parameters, and favours approaches which provide a sensible fit to the data while minimising the number of parameters. An comparison of these model comparison approaches in a number of scenarios is outlined in Gelman et al. (2014).

2.4 Spatial modelling

2.4.1 Introduction

Spatial data are any form of statistical data which have geographical locations attached. Spatial data come in three main forms; point-referenced data, areal data and point pattern data. Point-referenced data consist of a set of observations of the response and/or covariates taken at a set of precise spatial locations. An example of point-referenced data would be the amount of rainfall recorded at a set of weather stations within a particular geographical space. The overall rainfall pattern for the entire space could then be estimated based on the data obtained at the set of fixed points. Areal data involves the entire geographical space being partitioned into a set of non-overlapping subregions known as areal units; e.g. a country being divided into a set of electoral wards or council regions. The areal data take the form of aggregated summaries for each individual areal unit; e.g. the number of hospital admissions for patients living in an areal unit. Areal data are particularly common for applications in health, because confidentiality issues sometimes prevent the specific locations of disease cases or hospital admissions being recorded, while patient anonymity can be preserved via aggregated data. Point pattern data are a form of spatial data where the actual location itself is the feature of interest, the aim is to identify the locations where a particular event occurs. An example of this would be the locations of oak trees within a national park. A wide range of spatial modelling methodology has been developed for each of these types of data, but in this thesis we will focus on developing methodologies for areal modelling.

2.4.2 Extending a Poisson GLM to allow for spatial autocorrelation

The study region $\mathcal{A}$ is partitioned into n non-overlapping areal units $\mathcal{A} = \{\mathcal{A}_1, \ldots, \mathcal{A}_n\}$, and a response Y_i is observed in each of those areal units to give a set of response data $\boldsymbol{Y} = (Y_1, \ldots, Y_n)$. Areal count data are commonly modelled by extending the Poisson log-linear model (2.3) to account for the spatial pattern of the data. The data is likely to contain spatial autocorrelation, where correlation exists between pairs of areal units which are close to each other geographically. The spatial pattern of the data is modelled by a combination of covariate data, $X = (\boldsymbol{x_1^T}, \ldots, \boldsymbol{x_n^T})$ and a set of random effect terms, $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_n)$. These random effect terms account for the unexplained spatial autocorrelation induced into the disease data by unmeasured confounding variables. The spatial models used with count data $\boldsymbol{Y}$ are typically Poisson GLMs of the form introduced in Section 2.2.1 and are outlined as follows:

$$\begin{aligned} Y_i &\sim \text{Poisson}(\mu_i) \qquad i = 1, \ldots, n, \\ \log(\mu_i) &= \boldsymbol{x_i^T \beta} + \phi_i. \end{aligned} \tag{2.4}$$

The random effects $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_n)$ are typically modelled using a conditional autoregressive (CAR) prior. These models can be specified by a set of univariate full conditional distributions of the form $f(\phi_i|\boldsymbol{\phi_{-i}})$, where $\boldsymbol{\phi_{-i}} = (\phi_1, \ldots, \phi_{i-1}, \phi_{i+1}, \ldots, \phi_n)$. The spatial autocorrelation between these random effect terms is accounted for by a binary neighbourhood matrix W,

where $w_{ij} = 1$ if areal units $(\mathcal{A}_i, \mathcal{A}_j)$ share a common border (denoted $i \sim j$) and $w_{ij} = 0$ otherwise. An area is not considered to share a border with itself, so $w_{ii} = 0$ for all i.

The level of spatial autocorrelation within a set of areal data can be tested via Moran's I value (Moran (1950)), which is defined as follows:

$$I = \frac{n}{\sum_i \sum_j w_{ij}} \frac{\sum_i \sum_j w_{ij}(Y_i - \bar{Y})(Y_j - \bar{Y})}{\sum_i (Y_i - \bar{Y})^2}.$$

A Moran's I value close to 1 corresponds to strong positive spatial autocorrelation, a value close to -1 corresponds to strong negative spatial autocorrelation and 0 corresponds to complete spatial randomness. A permutation test can be carried out to test the null hypothesis that no spatial autocorrelation exists. The observed values can be randomly allocated to the n areal units, and the Moran's I value calculated for this allocation. This random permutation is then repeated multiple times. The randomly allocated data will have no underlying spatial autocorrelation present, and any observed correlation under Moran's I corresponds to random error. The true observed Moran's I value can then be compared to this set of random permutations to identify whether any true underlying spatial autocorrelation is present in the observed data.

A number of different conditional autoregressive models have been proposed, and three of the most common are outlined below.

Intrinsic CAR

The simplest CAR prior is the intrinsic model proposed by Besag et al. (1991), which is given by

$$\phi_i|\boldsymbol{\phi}_{-i} \sim \mathrm{N}\left(\frac{\sum_{j=1}^{n} w_{ij}\phi_j}{\sum_{j=1}^{n} w_{ij}}, \frac{1}{\tau(\sum_{j=1}^{n} w_{ij})}\right) \quad i = 1, \ldots, n. \tag{2.5}$$

where τ is a conditional precision parameter. The conditional expectation of ϕ_i is the mean of the random effects in neighbouring areal units, while the precision is proportional to the number of neighbouring units. This precision formulation is sensible, because you would expect the precision to be higher when you have more neighbouring areas and therefore more information to estimate the value of ϕ_i. This set of conditional distributions correspond to a multivariate Gaussian distribution, with zero vector mean but an improper precision matrix given by $Q = \tau(\mathrm{diag}(W\mathbf{1}) - W)$, where $W\mathbf{1}$ is a vector containing the number of neighbours for each areal unit. One drawback of this model is the lack of a parameter to control the strength of the spatial autocorrelation; if you multiplied $\boldsymbol{\phi}$ by 10 then the precision τ would decrease, but the spatial structure does not change. This means that the intrinsic model is only sensible in cases where the spatial autocorrelation in the data is strong; it is not sensible for cases where there is weak or moderate spatial autocorrelation across the study region because the model would tend to produce an overly smooth estimated risk surface in these cases. This formulation of the precision will make sense if strong spatial autocorrelation is present, because an increased number of neighbours means that more information is available to estimate the random effect value. However, in cases where weaker spa-

tial autocorrelation is present, this formulation is less sensible, because an increase in the number of neighbours would not necessarily lead to a huge increase in the amount of information available to estimate the random effect value.

Besag-York-Mollié (BYM) model

Besag et al. (1991) also proposed an alternative spatial modelling approach, where they combine the intrinsic CAR prior model given in (2.5) with a set of independent random effects. This model is of the form:

$$\begin{aligned} Y_i &\sim \text{Poisson}(\mu_i) \qquad i = 1, \ldots, n, \\ \log(\mu_i) &= \boldsymbol{x_i^T \beta} + \phi_i + \gamma_i, \end{aligned} \tag{2.6}$$

where $\boldsymbol{\phi}$ is a set of structured random effects which follow the intrinsic CAR model (2.5) and γ is a set of unstructured, independent random effects $\gamma_i \sim \text{N}(0, \sigma)$. This combination of two sets of random effects can model different levels of spatial autocorrelation by varying the relative values of the parameters $\boldsymbol{\phi}$ and $\boldsymbol{\gamma}$. Strong spatial autocorrelation can be modelled using larger values of $\boldsymbol{\phi}$ and smaller values of $\boldsymbol{\gamma}$, while weaker correlation can be modelled with larger values of $\boldsymbol{\gamma}$ and smaller values of $\boldsymbol{\phi}$. The main drawback of this approach is that these random effects cannot easily be estimated separately, it usually is only possible to identify the sum $\phi_i + \gamma_i$ for each area.

Leroux CAR

The issue of accounting for the possibility of weaker spatial autocorrelation was addressed by Leroux et al. (1999), who proposed the following CAR model:

$$\phi_i|\boldsymbol{\phi}_{-i} \sim \mathrm{N}\left(\frac{\rho\sum_{j=1}^{n} w_{ij}\phi_j}{\sum_{j=1}^{n} w_{ij}\rho + 1 - \rho}, \frac{1}{\tau(\sum_{j=1}^{n} w_{ij}\rho + 1 - \rho)}\right) \quad i = 1, \ldots, n. \tag{2.7}$$

Here, ρ controls for the level of spatial autocorrelation present in the data. A value of $\rho = 1$ corresponds to the intrinsic model (2.5), while $\rho = 0$ corresponds to a completely spatially smooth model with a constant mean, 0, and precision, τ. This increased flexibility can therefore enable the random effects to model a wider range of spatial autocorrelation than the intrinsic approach.

Lee CAR

Both the intrinsic and Leroux models are globally smooth; that is they assume a constant level of spatial smoothness across the entire study region with the partial correlation between (ϕ_i, ϕ_j) conditional on the remaining random effects $\boldsymbol{\phi}_{-ij}$ given by

$$\mathrm{Corr}[\phi_i, \phi_j|\boldsymbol{\phi}_{-ij}] = \frac{\rho w_{ij}}{\sqrt{(\rho\sum_{k=1}^{n} w_{ik} + 1 - \rho)(\rho\sum_{l=1}^{n} w_{jl} + 1 - \rho)}}. \tag{2.8}$$

For the Leroux CAR, a value of ρ close to 1 will lead to strong spatial autocorrelation between all pairs of adjacent areas for which $w_{ij} = 1$, while if ρ is close to 0 then there will be lower spatial autocorrelation across the study region. Thus ρ controls the level of spatial smoothness globally across the region. This may not be realistic in practice, because you may expect different levels of spatial autocorrelation in different areas of the study region. This has been addressed by Lee et al. (2014), who proposed a localised conditional autoregressive model which offers more flexibility in the way the random effects are modelled by allowing for discontinuities in the spatial autocorrelation surface.

Here, elements of the neighbourhood matrix relating to adjacent areas, $\{w_{ij}|i \sim j\}$, are treated as binary random quantities which are no longer fixed at 1; if w_{ij} is estimated as 0 for neighbouring areal units i and j, then that corresponds to a boundary between the areal units as (ϕ_i, ϕ_j) are conditionally independent given the random effects. The matrix W is defined as a set of edges, and under this terminology, $w_{ij} = 0, i \sim j$, means that an edge has been removed. A joint prior distribution for an extended set of random effects $\widetilde{\boldsymbol{\phi}}$ and the neighbourhood matrix, W is proposed as $f(W, \widetilde{\boldsymbol{\phi}}) = f(\widetilde{\boldsymbol{\phi}}|W)f(W)$. Here, the extended random effects vector takes the form $\widetilde{\boldsymbol{\phi}} = (\boldsymbol{\phi}, \phi^*)$, with ϕ^* being a global random effect which is potentially common to all areas and can be used to account for a discontinuity in the spatial structure. Equation (2.5) shows that if all edges are removed for area i then $\sum_{j=1}^{n} w_{ij} = 0$, resulting in an infinite mean and variance for $\phi_i|\boldsymbol{\phi_{-i}}$, and the extra global random effect ϕ^* is added to prevent this. An extended $(n+1) \times (n+1)$

neighbourhood matrix $\widetilde{W}$ is specified for this vector $\widetilde{\boldsymbol{\phi}}$, which takes the form

$$\widetilde{W} = \begin{pmatrix} W & \boldsymbol{w}_* \\ \boldsymbol{w}_*^T & 0 \end{pmatrix},$$

where $\boldsymbol{w}_* = (w_1*, \ldots, w_n*)$ and $w_i* = I[\sum_{i\sim j}(1 - w_{ij}) > 0]$. Here $i \sim j$ denotes that areas i and j are neighbours, and $I[.]$ denotes an indicator function, which sets $w_i* = 1$ if any entry in row i of the neighbourhood matrix W is changed from a 1 to a 0. If row i of the neighbourhood matrix W remains unchanged then $w_i^* = 0$.

The full conditionals of $f(\widetilde{\boldsymbol{\phi}}|W)$ are given by

$$\begin{aligned} \phi_i|\widetilde{\boldsymbol{\phi}}_{-i} &\sim \mathrm{N}\left(\frac{\sum_{j=1}^n w_{ij}\phi_j + w_{i*}\phi_*}{\sum_{j=1}^n w_{ij} + w_{i*} + \epsilon}, \frac{1}{\tau(\sum_{j=1}^n w_{ij} + w_{i*} + \epsilon)}\right), \quad (2.9) \\ \phi_*|\widetilde{\boldsymbol{\phi}}_{-*} &\sim \mathrm{N}\left(\frac{\sum_{j=1}^n w_{j*}\phi_j}{\sum_{j=1}^n w_{j*} + \epsilon}, \frac{1}{\tau(\sum_{j=1}^n w_{j*} + \epsilon)}\right). \end{aligned}$$

Here, ϵ is added to ensure that the precision matrix $\widetilde{Q}$ is invertible in the multivariate form of this prior, $\boldsymbol{\phi} \sim \mathrm{N}(\mathbf{0}, \frac{1}{\tau}\widetilde{Q}^{-1})$. $\widetilde{Q}$ often defined as $\widetilde{Q} = \mathrm{diag}(\widetilde{W}\mathbf{1}) - \widetilde{W}$, but this form is singular, and thus the inverse cannot be computed as required. Instead, the authors set $\widetilde{Q} = \mathrm{diag}(\widetilde{W}\mathbf{1}) - \widetilde{W} + \epsilon I$, which is diagonally dominant and therefore invertible.

Under this CAR prior, the conditional expectation for an area is a weighted average of the random effects in neighbouring areas and the global random

effect ϕ^*, with binary weights based on the extended neighbourhood matrix $\widetilde{W}$. If no discontinuities are introduced (i.e. the neighbourhood matrix W is unchanged from the original), then this simplifies to the intrinsic model (2.5), while if discontinuities are introduced between every pair of neighbouring areas (i.e. the neighbourhood matrix W is changed to a zero matrix) then this simplifies to a set of independent random effects with mean ϕ_* and precision τ.

The neighbourhood matrix W is considered as a single random quantity, and is represented by $\widetilde{W} \sim$ discrete Uniform $(\widetilde{W}^{(0)}, \widetilde{W}^{(1)}, \ldots, \widetilde{W}^{(m)})$. In the model proposed by Lee et al. (2014), $\widetilde{W}^{(m)}$ corresponds to all possible edges being retained in W, while $\widetilde{W}^{(0)}$ corresponds to all edges being removed. Here, a move from $\widetilde{W}^{(k)}$ to $\widetilde{W}^{(k-1)}$ corresponds to one edge being removed from W. This set of potential matrices, $(\widetilde{W}^{(0)}, \widetilde{W}^{(1)}, \ldots, \widetilde{W}^{(m)})$, is elicited from a set of disease data from a time period prior to the study period. This localised conditional autoregressive model will be used in Chapter 6 of this thesis, but instead of $(\widetilde{W}^{(0)}, \ldots, \widetilde{W}^{(m)})$ being defined by the number of edges removed, they will be defined by the number of clusters present.

2.5 Spatio-temporal modelling

The spatial modelling approaches introduced in Section 2.4 use data at a single fixed point in time to identify the spatial pattern in the data across n areal units. However in some cases, data are collected across T time points at each of the n areal units, and spatio-temporal modelling approaches have

been developed in order to estimate the trends in both space and time across the dataset. The aim of such spatio-temporal modelling approaches is usually to identify changes in the spatial pattern of the response over time, or to compare the temporal trends in the response across different parts of the study region. Spatio-temporal models can be developed for each of the three types of spatial data introduced in Section 2.4, but again this thesis will focus on the methodology for areal data. A review of the spatio-temporal disease mapping literature is given in Section 3.6, but a more detailed summary of two of the most important models is given here.

2.5.1 Bernardinelli model

One of the first spatio-temporal models for areal data was that proposed by Bernardinelli et al. (1995), who suggested a Poisson GLM with the linear predictor containing separate terms for space and time as well as a space-time interaction effect which allows for different temporal trends in different areas. The observed spatio-temporal response data takes the form $\boldsymbol{Y} = (\boldsymbol{Y_{1.}}, \ldots, \boldsymbol{Y_{n.}})$ where $\boldsymbol{Y_{i.}} = (Y_{i1}, \ldots, Y_{iT})$ is the set of T observations for area i. This model takes the form:

$$\begin{aligned} Y_{it} &\sim \text{Poisson}(\mu_{it}) \qquad i = 1, \ldots, n, \quad t = 1, \ldots, T \\ \log(\mu_{it}) &= (\alpha + \phi_i) + (\beta + \delta_i)t, \end{aligned} \tag{2.10}$$

where α is a global intercept term common to all areas, ϕ_i is the area effect for area i, β is the time effect and δ_i represents the space-time interaction

term. The random effect terms $\boldsymbol{\phi}$ and $\boldsymbol{\delta}$ can both take either an unstructured or structured form. The structured form is given by the intrinsic CAR prior outlined in (2.5), while the unstructured form is a set of independent random effects drawn from a $\mathrm{N}(0, \sigma)$ distribution.

The intercept for area i can be obtained by the sum $\alpha + \phi_i$, while the slope (or trend) for area i is the sum $\beta + \delta_i$. In order to ensure identifiability, the model is parameterised so that $\sum \phi_i = 0$ and $\sum \delta_i = 0$, thus allowing straightforward interpretation of the random effects, $\boldsymbol{\phi}$ and $\boldsymbol{\delta}$. Here ϕ_i is the difference between the global intercept term, α, and the area-specific intercept. $\phi_i > 0$ means that area i has a greater than average intercept, while $\phi_i < 0$ means it has a lower than average intercept. Likewise δ_i is the difference between the global slope term, β, and the area-specific slope, with $\delta_i > 0$ meaning that area i has a steeper than average slope, while $\delta_i < 0$ means it has a shallower than average slope.

2.5.2 Knorr-Held model

An alternative spatio-temporal modelling approach was proposed by Knorr-Held (2000), who introduced a space-time interaction term. Here the response is modelled by a binomial GLM with a logit link, as follows:

$$\begin{aligned} Y_{it} &\sim \text{Binomial}(n_{it}, \pi_{it}) \\ \log\left(\frac{\pi_{it}}{1-\pi_{it}}\right) &= \alpha_i + \phi_i + \beta_t + \delta_t + \gamma_{it}, \end{aligned} \tag{2.11}$$

where α_i and ϕ_i are area-specific terms which account for the spatial structure of the data, β_t and δ_t are time-specific terms which account for the temporal structure of the data and γ_{it} is a space-time interaction term which accounts for unexplained differences in the spatial pattern at different time points. Here, $\boldsymbol{\phi}$ is a set of structured random effects which follow a CAR model such as the intrinsic model (2.5), and $\boldsymbol{\alpha}$ is a set of unstructured spatially independent effects. Similarly, $\boldsymbol{\delta}$ is a set of structured effects modelled by an approach where neighbouring time points tend to be alike, such as a first order random walk, while $\boldsymbol{\beta}$ is a set of unstructured effects which are independent over time.

There are four possible options for the space-time interaction term γ_{it}, one for each possible combination of spatial and temporal effects. If the expected interaction is between the two unstructured effects $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ then all interaction terms γ_{it} are independent. This is appropriate if the interaction term is included to account for unexplained effects which have no spatial or temporal structure. For an interaction between the unstructured spatial effect $\boldsymbol{\alpha}$ and the structured temporal effect $\boldsymbol{\delta}$ then the interaction terms $\boldsymbol{\gamma}_i = (\gamma_{i1}, \ldots, \gamma_{iT})$ follow a random walk independent of space. This is appropriate if the interaction term is included to account for temporal trends which are different across different areal units, but which do not follow any spatial structure. For an interaction between the structured spatial effect $\boldsymbol{\phi}$ and the unstructured temporal effects $\boldsymbol{\beta}$, the interaction terms $\boldsymbol{\gamma}_t = (\gamma_{1t}, \ldots, \gamma_{nt})$ can be modelled by CAR models. This is appropriate where the interaction term is included to account for spatial trends which vary from time point to time point, but

which do not have any temporal structure. Finally, for an interaction between the two structured effects $\boldsymbol{\phi}$ and $\boldsymbol{\delta}$, the interaction terms γ_{it} can be modelled as $\gamma_{it}|\gamma_{-it} \sim \mathrm{N}(\mu_{it}, \tau_{it})$. The mean and precision are computed as follows:

$$\begin{aligned} \mu_{it} &= \frac{1}{2}(\gamma_{i,t-1} + \gamma_{i,t+1}) + \frac{\sum_{j=1}^{n} w_{ij}\gamma_{jt}}{\sum_{j=1}^{n} w_{ij}} + \frac{\sum_{j=1}^{n} w_{ij}(\gamma_{i,t-1} + \gamma_{i,t+1})}{2\sum_{j=1}^{n} w_{ij}} \\ \tau_{it} &= 2\kappa \sum_{j=1}^{n} w_{ij} \end{aligned} \tag{2.12}$$

where κ is a temporal precision term. This interaction is appropriate where the temporal trends are different across different areal units, but are more likely to be similar for neighbouring areal units.

2.6 Clustering

2.6.1 Introduction

Clustering is a field of statistics relating to the task of partitioning a set of objects into a number of groups (or clusters) based on some specified characteristic(s). A number of different clustering approaches have been proposed, but they all share the common goal of producing a set of clusters where objects within a cluster are similar to each other and objects in different clusters are different from each other. Clustering approaches have applications in a number of fields, including genetics (e.g. identifying similar gene characteristics in humans), taxonomy (e.g. dividing animals or plants into species groups) and medical imaging (e.g. identifying particular types of tissue in MRI scans). The similarity between two objects is measured by some function of the data relating to these objects, and some examples of ways of calculating this similarity are given in Section 2.6.2 below. One of the most challenging aspects of a clustering approach is deciding how many clusters the data should be partitioned into. In some cases the user will have an existing belief about the appropriate number of clusters and can make a decision based on that belief, but on many other occasions there will be no obvious reason to select a particular number over another *a priori*. In Section 2.6.3, we will discuss some model-based clustering approaches which can be used to address this problem.

2.6.2 Hierarchical agglomerative clustering

One of the most commonly used clustering approaches is hierarchical agglomerative clustering (Hastie et al. (2001)). Under this approach, one must initially consider each data point as its own singleton cluster, and then join together the two least dissimilar clusters at each stage to form a larger cluster. This process is repeated until only one cluster containing all data points remains.

Consider a set of n objects $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_n\}$ described by data $\boldsymbol{\psi} = (\boldsymbol{\psi_1}, \ldots, \boldsymbol{\psi_n})$, where $\boldsymbol{\psi_i}$ is a vector of data relating to object i. The hierarchical clustering algorithm will have n steps, and the cluster structures obtained at each step are denoted by $\{\mathcal{C}_1, \ldots, \mathcal{C}_n\}$. Here $\mathcal{C}_k = \{\mathcal{C}_k(1), \ldots, \mathcal{C}_k(k)\}$ partitions the n objects $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_n\}$ into k spatially contiguous groups, where $\mathcal{C}_k(j)$ is the set of objects in the jth cluster in the cluster solution.

In order to identify clusters, it is necessary to compute the distance between two objects, and the most common choice of distance measurement is the Euclidean distance. Let $\boldsymbol{\psi_i} = (\psi_{i1}, \ldots, \psi_{ip})$ and $\boldsymbol{\psi_j} = (\psi_{j1}, \ldots, \psi_{jp})$ be the covariate information relating to areas i and j respectively. Then the Euclidean distance between the areas is defined as $d_{ij} = ||\boldsymbol{\psi_i} - \boldsymbol{\psi_j}|| = \sqrt{\sum_{r=1}^{p}(\psi_{jr} - \psi_{ir})^2}$. One alternative distance measure is the Manhattan distance, which is computed as $d_{ij} = ||\boldsymbol{\psi_i} - \boldsymbol{\psi_j}|| = \sum_{r=1}^{p} |\psi_{jr} - \psi_{ir}|$. It is also necessary to compute the dissimilarity between two clusters which contain multiple objects. For a configuration with k clusters the dissimilarity, d_{ij},

between clusters i ($\mathcal{C}_k(i)$) and j ($\mathcal{C}_k(j)$) can be measured by a number of metrics called linkage methods, which are usually based on some function of the distances between objects within the clusters. Three of the most common linkage methods are single linkage, centroid linkage and Ward's linkage, and these are defined as follows:

- Single linkage measures the dissimilarity as the shortest distance between two clusters, that is $d_{ij} = \min\{||\boldsymbol{\psi}_f - \boldsymbol{\psi}_g|| : \mathcal{S}_f \in \mathcal{C}_k(i), \mathcal{S}_g \in \mathcal{C}_k(j)\}$, where $||.||$ denotes a distance metric.

- Centroid linkage measures the dissimilarity as the distance between the average of the two clusters, that is $d_{ij} = ||\bar{\mathcal{C}}_k(i) - \bar{\mathcal{C}}_k(j)||$, where $\bar{\mathcal{C}}_k(i) = (1/n_i) \sum_{f:\mathcal{S}_f \in \mathcal{C}_k(i)} \boldsymbol{\psi}_f$, and n_i is the number of objects in cluster $\mathcal{C}_k(i)$.

- Ward's Linkage measures the dissimilarity as the increase in the error sum of squares (ESS) when joining two smaller clusters into a larger cluster, that is $d_{ij} = \text{ESS}(\mathcal{C}_k(i,j)) - [\text{ESS}(\mathcal{C}_k(i)) + \text{ESS}(\mathcal{C}_k(j))]$, where $\mathcal{C}_k(i,j) = \mathcal{C}_k(i) \cup \mathcal{C}_k(j)$ and $\text{ESS}(\mathcal{C}_k(i)) = \sum_{f:\mathcal{S}_f \in \mathcal{C}_k(i)} ||\boldsymbol{\psi}_f - \bar{\mathcal{C}}_k(i)||^2$.

The hierarchical agglomerative clustering algorithm is outlined as follows:

Algorithm

1. Choose a distance metric and linkage method.
2. Construct $\mathcal{C}_n = \{\mathcal{C}_n(1), \ldots, \mathcal{C}_n(n)\}$, an initial cluster structure where each object is in its own singleton cluster.
3. Repeat the following steps for $h = n, \ldots, 2$, where step h produces $\mathcal{C}_{h-1}$ from $\mathcal{C}_h$.
   (a) Compute the $h \times h$ distance matrix D, whose klth element is given by
   $$D_{kl} = \begin{cases} d_{kl} & \text{if } k > l \\ \infty & \text{otherwise,} \end{cases}$$
   where d_{kl} is the distance between clusters $(\mathcal{C}_h(k), \mathcal{C}_h(l))$ under the selected linkage method.
   (b) Set $\{i, j\} = \arg_{k,l} \min(D_{kl})$, that is the identifiers of the two clusters that have the minimum dissimilarity as measured by the linkage method. In case of ties, $\{i, j\}$ is randomly selected from these.
   (c) Compute
   $$\mathcal{C}_{h-1} = \{\mathcal{C}_h(1), \ldots, \mathcal{C}_h(i-1), \mathcal{C}_{h-1}(i), \mathcal{C}_h(i+1), \ldots, \mathcal{C}_h(j-1), \mathcal{C}_h(j+1), \ldots, \mathcal{C}_h(h)\},$$
   where $\mathcal{C}_{h-1}(i) = \mathcal{C}_h(i) \cup \mathcal{C}_h(j)$.

This algorithm produces a set of n cluster structures, $\{\mathcal{C}_1, \ldots, \mathcal{C}_n\}$, and a decision must be made about which of these is the most appropriate structure

to fit the data. Ideally we want each cluster to contain data points which are similar to each other, but different to those in each of the other clusters. If too many clusters are created, then there is the possibility that similar data points are kept apart as a result, while if too few clusters are created then it may be that data points which are very different end up in the same cluster. There is no single "best" method for determining the number of clusters, and a number of approaches have been proposed. The simplest method would be to subjectively choose the number of clusters using plotting tools, but this could obviously lead to biased or non-optimal results even if the choice is made very carefully. A numerical method is preferred, and two such methods are proposed in Chapters 5 and 6. Alternative objective approaches include the Gap statistic (Tibshirani et al. (2001)), the Calinski-Harabasz Index (Calinski and Harabasz (1974)) and the silhouette statistic (Rousseeuw (1987)).

2.6.3 Model-based clustering

Heuristic clustering algorithms such as hierarchical agglomerative clustering generally provide a quick and reasonable estimate of the cluster structure, but these methods have drawbacks in terms of the choice of the number of clusters, as was highlighted in Section 2.6.2. An alternative approach is model-based clustering, an overview of which is given by Fraley and Raftery (2002). In model-based clustering, it is assumed that the data have been generated from a finite mixture of probability distributions, with each component distribution corresponding to a different cluster, and the aim is to estimate the parameters of these underlying probability distributions.

For a set of data $\boldsymbol{\xi} = (\boldsymbol{\xi_1}, \ldots, \boldsymbol{\xi_n})$, a finite mixture model with G components (or clusters) has the likelihood:

$$L(\boldsymbol{\theta_1}, \ldots, \boldsymbol{\theta_n}, \pi_1, \ldots, \pi_n | \boldsymbol{\xi}) = \prod_{i=1}^{n} \sum_{k=1}^{G} \pi_k f_k(\boldsymbol{\xi_i} | \boldsymbol{\theta_k}),$$

where $f_k()$ is the distribution of the kth component, and $\boldsymbol{\theta_k}$ are the parameters of that distribution. π_k is the prior or mixing probability of an observation belong to the kth component, with $\pi_k \geq 0, \sum_{k=1}^{G} \pi_k = 1$. In most cases, one considers a mixture of multivariate Normal distributions, $f_k(\boldsymbol{\xi_i} | \boldsymbol{\theta_k}) \sim \mathrm{N}(\boldsymbol{\mu_k}, \Sigma_k^{-1})$. This approach differs from the hierarchical clustering approach outlined in Section 2.6.2 in that this is a model-based parametric approach rather than an algorithmic approach. Here the parameters $\boldsymbol{\theta_k} = (\boldsymbol{\mu_k}, \Sigma_k^{-1})$ where $\boldsymbol{\mu_k}$ is the vector of means and Σ_k^{-1} is the precision matrix.

This likelihood can be maximised using the expectation-maximisation algorithm (Dempster et al. (1977)). In this context, we consider the complete dataset to be $\boldsymbol{Y_i} = (\boldsymbol{\xi_i}, z_i)$ where the hidden or latent variable is $\boldsymbol{z_i} = (z_{i1}, \ldots, z_{iG})$ with

$$z_{ik} = \begin{cases} 1 & \text{if } \xi_i \text{ comes from component } k \\ 0 & \text{otherwise.} \end{cases}$$

If we assume that the density of $\boldsymbol{\xi_i}$ given z_i is specified as $\prod_{k=1}^{G} f_k(\boldsymbol{\xi_i} | \boldsymbol{\theta_k})^{z_{ik}}$ and that each $\boldsymbol{z_i}$ is an independent and identically distributed from a multi-

nomial distribution with probabilities $(\pi_1, \ldots, \pi_G)$, then the complete-data log-likelihood is given by

$$l(\theta_k, \pi_k, z_{ik}|\boldsymbol{\xi}) = \sum_{i=1}^{n}\sum_{k=1}^{G} z_{ik} \log\left(\pi_k f_k(\boldsymbol{\xi_i}|\boldsymbol{\theta_k})\right).$$

The EM algorithm is an two stage iterative process consisting of the expectation (or E) step and the maximisation (or M) step. In the E step, a conditional expectation, $\hat{z}_{ik}$ is computed based on the data and the current set of parameter estimates, then in the M step, the complete-data log-likelihood, $l(\boldsymbol{\theta}_k, \pi_k, \hat{z}_{ik}|\boldsymbol{\xi})$ is maximised with respect to the model parameters. The two steps of the algorithm at iteration t are as follows:

E Step

$$\hat{z}_{ik}^{(t)} \leftarrow \frac{\hat{\pi}_k^{(t-1)} f_k(\boldsymbol{\xi}_i|\hat{\boldsymbol{\mu}}_k^{(t-1)}, \hat{\Sigma}_k^{-1(t-1)})}{\sum_{j=1}^{G} \hat{\pi}_j^{(t-1)} f_j(\boldsymbol{\xi}_i|\hat{\boldsymbol{\mu}}_j^{(t-1)}, \hat{\Sigma}_j^{-1(t-1)})}$$

M Step

$$\begin{aligned}
n_k^{(t)} &\leftarrow \sum_{i=1}^{n} \hat{z}_{ik}^{(t)} \\
\hat{\pi}_k^{(t)} &\leftarrow \frac{n_k^{(t)}}{n} \\
\hat{\boldsymbol{\mu}}_k^{(t)} &\leftarrow \frac{\sum_{i=1}^{n} \hat{z}_{ik}^{(t)} \boldsymbol{\xi_i}}{n_k^{(t)}} \\
\hat{\Sigma}_k^{-1(t)} &\leftarrow \text{depends on the model being used, see Celeux and Govaert (1995).}
\end{aligned}$$

These steps should be repeated alternately until convergence is achieved.

Within this thesis, the "mclust" package in the R computer language is used for model-based clustering, and this package uses the lack-of-progress criterion to identify convergence. Here, convergence is determined to have been achieved when the difference between consecutive iterations is within a certain tolerance level. Another alternative for determining convergence in this context is the Aitken acceleration criterion, which is discussed in McLachlan and Peel (2004).

In practice, the optimal number of model components G can be estimated by comparing a number of models with different numbers of components and then choosing the one which performs best on a particular model comparison criterion. Further detail on model comparison criteria is outlined in Section 2.3. This approach is used for the posterior classification step for the BYM in the simulation studies in Chapters 5 and 6.

2.6.4 Cluster comparison

In order to evaluate the performance of any clustering method, it is necessary to have a metric for comparing two cluster structures to determine how similar they are. Such a metric allows the user to test their approach on a dataset with a known cluster structure, comparing that true structure to the one obtained by their method. The most common of these cluster comparison metrics is the Rand Index, proposed in Rand (1971). Assume that objects $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_n\}$ are partitioned into two different cluster structures $\mathcal{C}_k = \{\mathcal{C}_k(1), \ldots, \mathcal{C}_k(k)\}$ and $\mathcal{D}_l = \{\mathcal{D}_l(1), \ldots, \mathcal{D}_l(l)\}$ with the same notation

as in Section 2.6.2, and compute the following:

a - the number of pairs of objects $\mathcal{S}_1, \ldots, \mathcal{S}_n$ that are in the same cluster in structure $\mathcal{C}_k$ and in the same cluster in structure $\mathcal{D}_l$.
b - the number of pairs of objects $\mathcal{S}_1, \ldots, \mathcal{S}_n$ that are in different clusters in structure $\mathcal{C}_k$ and in different clusters in structure $\mathcal{D}_l$.
c - the number of pairs of objects $\mathcal{S}_1, \ldots, \mathcal{S}_n$ that are in the same cluster in structure $\mathcal{C}_k$ and in different clusters in structure $\mathcal{D}_l$.
d - the number of pairs of objects $\mathcal{S}_1, \ldots, \mathcal{S}_n$ that are in the different cluster in structure $\mathcal{C}_k$ and in the same cluster in structure $\mathcal{D}_l$.

Then the Rand Index, R, can be calculated as follows:

$$R = \frac{a+b}{a+b+c+d}.$$

A value of 1 indicates complete agreement between the two cluster configurations, while a value of 0 indicates that no pair of areal units are classified in the same way under both configurations. This index will be used in the simulation studies in Chapters 4, 5, 6 and 7 to test the cluster structures produced by the proposed methods against a known "true" cluster structure.

Chapter 3

Disease mapping

3.1 Introduction

Disease risk varies geographically as a result of many factors, including differences in environmental exposures, and cultural and behavioural differences between the inhabitants of different areas. One of the most important reasons for these differences is poverty, with a recent Audit Scotland report (Audit Scotland (2012)) finding that people in deprived areas have higher rates of coronary heart disease, mental health problems, obesity, alcohol and drug misuse, diabetes and some types of cancer. This has been attributed to negative lifestyle choices in these deprived areas, including increased smoking and alcohol consumption, poorer diets and less exercise. The extent and pattern of such health inequalities are illustrated via disease maps, which are produced by partitioning the study region into n non-overlapping areal units such as electoral wards or census tracts, and then computing and mapping the

disease risk for the population living in each areal unit. The disease incidence is displayed visually via a choropleth map of the study region, where areas are shaded on a scale which relates to their level of disease risk (for example, see Figure 4.2). This visualisation of disease risk allows for easier comparison of risks across the area, and allows the user to identify features of interest on the map. The key benefit of such maps is that they allow public health officials to identify areal units that exhibit elevated disease risks, which in turn enables interventions to be appropriately targeted at the communities at greatest need. Such interventions can take the form of a vaccination programme, or a public awareness campaign about potential risk factors. This thesis will focus on developing novel disease mapping methodology, and so an overview of the disease mapping literature is outlined within this chapter.

3.2 Data

Disease mapping studies are based on data relating to a study region $\mathcal{A}$, which is partitioned into n non-overlapping areal units $\mathcal{A} = \{\mathcal{A}_1, \ldots, \mathcal{A}_n\}$. The nature of the data collection processes generally means that these areal units take the form of some sort of pre-existing administrative unit, such as postcode areas, electoral wards, council regions or even counties. The applications outlined in Chapters 4, 5, 6 and 7 of this thesis are based on the Greater Glasgow and Clyde Health Board, where the areal units are $n = 271$ administrative regions known as Intermediate Geographies (IGs).

Health data tends to consist of aggregated disease counts for these areal

units, because this allows patient anonymity to be maintained. If the specific locations of disease cases or hospital admissions were recorded then this could potentially allow individual patients to be identified, which would be a breach of confidentiality. The data are denoted by $\boldsymbol{Y} = (Y_1, \ldots, Y_n)$, where Y_i represents the number of observed disease cases within areal unit i. These data are obtained from governmental agencies or health authorities; for example, in Scotland a variety of health data is made publicly available by the Scottish government via the Scottish Neighbourhood Statistics website (www.sns.gov.uk). For most modern data, the diseases are classified by the tenth revision of World Health Organisation's International Classification of Disease (World Health Organization (1993)).

The naive approach would be to model disease risk based purely on these disease counts, but this fails to take account of the fact that different areas could have vastly different population demographics. The differences in disease counts could be as a result of the demographic differences rather than some underlying difference in disease risk. For example, if elderly people are at higher risk of respiratory hospital admissions, then areas which have a higher percentage of elderly people are likely to have a higher number of respiratory admissions. In order to account for these demographic differences, a set of expected disease counts, $\boldsymbol{E} = (E_1, \ldots, E_n)$, can be constructed, where E_i is the expected number of disease cases in area i. These expected counts can be constructed via external standardisation, based on the age and sex demographics of the population within the areal units. One should construct a set of m strata based on age and sex, and then compute $E_i = \sum_{j=1}^{m} N_{ij} r_j$,

where N_{ij} is the population in area i and strata j, and r_j is the overall disease rate for strata j.

3.3 Model

Based on these expected counts, the simplest measure of disease risk is the standardised incidence ratio (SIR), which is given for area i as $\text{SIR}_i = \frac{Y_i}{E_i}$. An SIR value greater than 1 indicates that there is a higher than expected disease risk within the areal unit, while a value less than 1 represents a lower than expected disease risk. For example, an SIR value of 1.1 corresponds to a disease risk which is 10% higher than the average, while a value of 0.9 corresponds to a disease risk which is 10% lower than the average. A plot of these SIR values can be used to give a simple visual guide about the relative levels of disease risk across the study region, but as a technique for modelling the disease risk it has disadvantages. In cases where the disease being studied is rare, or the population of the study region is small, some areal units may have low values of E_i, and the ratio $\frac{Y_i}{E_i}$ would be susceptible to small random fluctuations in the value of Y_i. In the most extreme case where $E_i = 0$, the ratio could not be computed at all. Also, this approach operates independently of space, and therefore does not take into account the spatial autocorrelation which could be present in the data.

It is therefore more common to take a Bayesian modelling approach to disease mapping by adopting a spatial model of the form introduced in Section 2.4. Typically, these spatial models are Poisson GLMs of the form introduced in

Section 2.2.1 and are outlined as follows:

$$\begin{aligned} Y_i &\sim \text{Poisson}(E_i R_i) \qquad i = 1, \ldots, n, \\ \log(R_i) &= \boldsymbol{x_i^T \beta} + \phi_i. \end{aligned} \tag{3.1}$$

Here, covariate information, $\boldsymbol{x_i^T \beta}$, and a set of random effects, $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_n)$, are used to estimate the disease risk. The set of random effects is used to account for the spatial autocorrelation present in the data, which has not been accounted for by the covariate information. These random effects allow each areal unit to borrow information from its neighbours, thus reducing the chance of the estimates being affected by a small E_i value as was the case with the SIR. The random effects $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_n)$ are typically modelled using a conditional autoregressive (CAR) prior (see Section 2.4.2). As discussed in Section 2.4.2, the spatial autocorrelation between these random effect terms is controlled by a binary neighbourhood matrix W, where $w_{ij} = 1$ if areal units $(\mathcal{A}_i, \mathcal{A}_j)$ share a common border (denoted $i \sim j$) and $w_{ij} = 0$ otherwise. These Bayesian modelling approaches can therefore produce estimates of the disease risk which take into account the spatial nature of the data to produce spatially smoothed estimates of disease risk.

3.4 Boundary Detection

In Section 2.4.2, a number of conditional autoregressive priors for the random effects $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_n)$ were introduced. The intrinsic (Besag et al. (1991)),

BYM (Besag et al. (1991)) and Leroux (Leroux et al. (1999)) models were discussed in this section, and it was noted that all three share the common assumption that there is a constant level of spatial smoothness across the entire study region. In practice, there will be many situations where this assumption of constant spatial smoothness does not hold, and the level of spatial autocorrelation varies across the study region. Some areas of the study region may display strong spatial smoothness, while other areas exhibit weak (or no) spatial autocorrelation. In the context of disease mapping, the latter could represent pairs of neighbouring areas which exhibit vastly different disease risks, and various reasons for these differences are discussed in Mitchell and Lee (2014). It may therefore be more realistic to adopt a modelling approach which allows for non-constant spatial autocorrelation across the study region by introducing discontinuities (or boundaries) in the spatial structure between pairs of neighbouring areas which do not exhibit similar traits.

A number of models have been proposed for identifying these boundaries in the disease risk surface. The majority of these treat the elements of the neighbourhood matrix $\{w_{ij}|i \sim j\}$ as binary random quantities, where estimating $w_{ij} = 0$ corresponds to identifying a boundary between $(\mathcal{A}_i, \mathcal{A}_j)$. If $w_{ij} = 0$ then that implies that (ϕ_i, ϕ_j) are conditionally independent and should not be smoothed over in the modelling process. One of the first examples of this approach came from Lu et al. (2007), who proposed a logistic regression model for $\{w_{ij}|i \sim j\}$ using a measure of dissimilarity between $(\mathcal{A}_i, \mathcal{A}_j)$ as the covariate. However, this results in an excessively large num-

ber of parameters, which led Lee and Mitchell (2012) to treat $\{w_{ij}|i \sim j\}$ as a deterministic function of a small number of parameters and the areal level measure of dissimilarity. The same authors (Lee and Mitchell (2013)) also proposed iteratively re-estimating $\{w_{ij}|i \sim j\}$ and the remaining model parameters conditional on each other until a convergence criterion is reached, where $\{w_{ij}|i \sim j\}$ was updated deterministically based on the other model parameters. However, this approach has the drawback of being unable to quantify the level of uncertainty on $w_i j$. An alternative model proposed by Lee et al. (2014) uses an extended random effects vector with a global random effect which is potentially common to all areas. This model was discussed in Section 2.4.2. This approach does have the limitation of requiring prior data, though such data does generally tend to be available for disease incidence. Li et al. (2011) took a different approach, by fitting multiple models with different W specifications and thus different potential sets of boundaries to the data, and using the Bayesian Information Criterion (BIC) to choose the best model. This approach only allows one boundary to be removed at a time, which limits the scope of the model. However, the model comparison approach will be revisited in a different guise in Chapter 5 of this thesis. These approaches all produce what are known as "open boundaries", which are a set of potentially disjoint boundary segments that do not necessarily enclose an areal unit or group of units. However, in many applications the aim is to identify distinct spatially cohesive groups of areal units that exhibit substantially different risks compared to their neighbours, and this approach requires "closed boundaries" which entirely enclose a group of areal units. These closed boundaries can be used to partition the study region into a set of non-overlapping clusters of areal units with similar levels of disease risk.

This thesis will focus on developing methodology to identify clusters in a disease mapping context.

3.5 Clustering

A number of approaches have been proposed for identifying clusters within a disease mapping context. One of the first and still most widely used cluster detection approaches are scan statistics (Kulldorff (1997)), which identify clusters of areal units that exhibit an elevated risk of disease. Their popularity in part stems from the availability of the SaTScan software, which makes it straightforward for others to implement this approach. However, scan statistics merely identify high-risk clusters, and do not estimate the spatial pattern in disease risk. This approach would not be suitable for the many applications for which the estimation of the disease risk pattern is one of the main aims of the study. A number of hierarchical modelling approaches have been proposed which simultaneously model the spatial pattern of disease risk and estimate the cluster structure within the data. Knorr-Held and Rasser (2000) proposed a Bayesian model where the areal units are partitioned into a set of spatially contiguous clusters. A set of cluster centres are selected, and then the remaining areal units are allocated to clusters based on their distance from these cluster centres. The number of clusters and the locations of the cluster centres are not fixed in advance, and are instead estimated by the model. Another approach was suggested by Green and Richardson (2002), who propose a mixture model with an unknown number of components (or clusters). Here, the allocation of the areal units into clusters is based on

the Potts model (Wu (1982)), which is commonly used in image processing. An area is more likely to be allocated to a particular cluster if that cluster already contains neighbouring areal units, and the model has a parameter which controls the strength of this spatial dependence for allocation. As with Knorr-Held and Rasser (2000), the number of clusters is not fixed and is estimated as part of the modelling approach. The inference for both approaches requires reversible-jump Markov chain Monte Carlo methods (Green (1995)), which allows the number of parameters to vary in the model. Such inference can be computationally complex and as a result may be beyond the scope of most epidemiologists, particularly since no publicly available software exists to allow others to implement these approaches.

An alternative was proposed by Charras-Garrido et al. (2012), based on a set of disease risk classes (or clusters) which are naturally ordered by the level of disease risk within the class. Each areal unit is allocated to one of these risk clusters, with a penalty introduced for neighbouring areas which are in different classes, based on the distance between these risk levels. These penalties are smaller for smaller distances between classes, making it more likely for neighbouring areas to have the same or similar disease risk. Inference for this model is carried out via a Monte Carlo Expectation-Maximisation algorithm with post-hoc classification. A Bayesian analogue to the Kulldorff (1997) approach was proposed by Wakefield and Kim (2013), with the focus being the identification of a small number of high or low risk circular clusters. A list of all possible clusters is defined by taking each area in turn and continually adding the geographically closest neighbouring area until a maximum

cluster size is reached. The data is then used to determine which (if any) of these possible cluster configurations can be considered as a high (or low) disease risk cluster. However, the authors acknowledge that this approach is restricted by the requirement for circular clusters, which may not necessarily be realistic in practice. These methods have been designed specifically with cluster detection in mind, but it should be noted that it is also possible to identify clusters in the risk surfaces estimated by other non-clustering approaches by carrying out a post-hoc clustering step. Such an approach is discussed in Charras-Garrido et al. (2013), which proposes fitting model (3.1) to the data and then carrying out post-hoc clustering on the resulting estimated risk surface to identify possible risk clusters. Charras-Garrido et al. (2013) also provides a comparison of some of the common cluster detection approaches.

One of the main differences between these approaches is that Knorr-Held and Rasser (2000) and Wakefield and Kim (2013) force the clusters to be spatially contiguous, while Green and Richardson (2002) and Charras-Garrido et al. (2012) do not. It should, however, be noted that it is straightforward to induce spatial contiguity in the sets of clusters produced by the latter methods by simply relabelling the non-contiguous parts as new clusters. In all of these approaches, the disease risk is assumed to be constant within a cluster, which has the advantage that it partitions the relative risk into risk classes/clusters which are easy to interpret for epidemiologists. However, for real data it is likely that disease risk varies within a cluster, and in Chapters 5 and 6 we propose methodology which allows for such within cluster variation.

3.6 Spatio-temporal disease mapping

There has been increasing interest in extending disease mapping models to identify patterns and trends in space and time simultaneously. One of the first such models was suggested by Bernardinelli et al. (1995), who proposed a generalised linear model where the linear predictor has separate linear trends for each areal unit, as introduced in Section 2.5.1. The space and space-time terms can either be structured (i.e. modelled via conditional autoregressive models) or unstructured (i.e. a set of independent random effects). An alternative approach was outlined by Waller et al. (1997), who proposed an extension of the BYM (Besag et al. (1991)) model where the disease risk pattern is estimated by a combination of a set of spatially dependent random effects modelled by a CAR prior and a set of independent random effects. Here, there is no attempt at smoothing over time, which may not be realistic in practice. Knorr-Held (2000) proposed an approach consisting of a pair of area-specific effects, one structured (via a CAR model) and one unstructured, a pair of time-specific effects, one structured (via a random walk) and one unstructured, as well as an additional space-time interaction term. This model is outlined in Section 2.5.2.

MacNab and Dean (2001) proposed a generalised additive mixed model for estimating disease risk via a combination of a CAR model for the spatial pattern and a set of smooth functions known as B-splines (de Boor (1972)) for the temporal trends. Here, a fixed effect is used to model the global temporal trend, while random effects are used to model the localised trends for

individual areal units. Inference for this model is carried out via penalised quasi-likelihood (Breslow and Clayton (1993)), though the authors note that this approach is not ideal in terms of estimating model uncertainty. This is addressed by Ugarte et al. (2008), who compare a number of estimators of prediction error to account for uncertainty, and recommend a bootstrap adjusted Empirical Bayes variance estimator (MacNab et al. (2004)). The same authors (Ugarte et al. (2010)) also outlined a model based on P-splines (Eilers and Marx (1996)) and derive the mean square error of the predictor in order to compute confidence intervals for the risks. Congdon and Southall (2005) instead propose modelling spatio-temporal components via autoregressive time series models (Chib (1993)) with a set of space-time errors, each of which depends on the error at the previous time point. Bohning (2003) suggested modelling the disease risk as a mixture of Poisson distributions (see Section 2.6.3), and outlined two possible approaches for constructing spatio-temporal mixtures. The first approach identifies a separate mixture model at each time period, thus meaning that at each time point we may have a different set of Poisson distributions from which the mixture is drawn. The second approach fits a single mixture model such that the same set of Poisson distributions exists across all time points, though areas can move between these mixture components at different time points. The author prefers the latter method because it allows easier interpretation of changes in the cluster structure; in this case disease clusters for different time points are directly comparable since the mixture component remains the same at all time points. The identification of clusters of disease risk in space and time allows straightforward interpretation of changes in disease risk over time, and will be the focus of the spatio-temporal approach outlined in Chapter 7.

Chapter 4

A new spatially adapted hierarchical agglomerative clustering algorithm.

4.1 Introduction

One of the main aims of disease mapping is to identify similarities and differences in risk across the study area in terms of the disease being studied. This is particularly true where the aim is to identify clusters of areas exhibiting elevated risks that differ greatly compared to neighbouring regions. These aims are similar to the motivations of clustering as described in Section 2.6, and thus it is sensible to utilise and extend clustering methodology in a disease mapping setting.

In a disease mapping context each individual areal unit can be considered as a clustering object, and it is then possible to cluster these areal units to produce groups of areal units which exhibit similar risks, thus identifying groups of high or low risk areas. Such an approach has advantages in terms of guiding health policies in the larger area; it allows for easy identification of the high risk areas where further interventions and/or investment in health care and education are required. The identification of low risk clusters could also be useful in terms of providing insight into the aetiological factors which cause particular areal units to have a low risk, which could provide possible solutions for high risk areas.

The aim over the next three chapters will be to outline new methodology which allows for the estimation of the spatial pattern in disease risk, whilst simultaneously detecting the spatial extent of high or low risk clusters. The methodology brings together hierarchical agglomerative clustering techniques and conditional autoregressive models in a two-stage approach. The first stage is a spatially-adjusted hierarchical agglomerative clustering algorithm, which is used to elicit a set of n candidate cluster configurations containing between 1 and n clusters. The second stage utilises a Bayesian modelling approach in order identify the most appropriate cluster structure and also estimate disease risk. In this chapter we will introduce the hierarchical agglomerative clustering approach used in the first stage of our method, where the aim is to produce a set of potential cluster structures which respect the spatial contiguity of the study region. Chapters 5 and 6 will propose two different modelling approaches for the second stage of our approach.

In order to cluster areal units we must choose a numerical measure for each area, so that dissimilarities between pairs of areas can be calculated. The most obvious measure is the observed disease risk value represented by the SIR, however this is not appropriate in the context of the two-stage modelling approaches used in Chapters 5 and 6. These approaches estimate a set of potential cluster structures in the first stage and model disease risk based on the potential cluster structures and then select the best cluster structure in the second stage. The approach in Stage 2 makes use of the SIR values in order to estimate risk, so using the SIR values in the cluster estimation would involve using the data twice. Therefore we apply our proposed clustering algorithm to disease data from a time period prior to the study period. For example if the data being analysed are disease risk data from 2014, then the clustering algorithm could be applied to disease risk data from 2011, 2012 and 2013. Unless substantial urban regeneration has taken place over this short time period, it is likely that the previous years will exhibit a similar spatial risk pattern to the period being studied. Such an approach is only appropriate for chronic diseases; it would not be suitable for epidemic diseases where the risk pattern changes more rapidly. An alternative would be to estimate the cluster structure using covariate information which has a strong correlation with the disease risk, such as using smoking data to model lung cancer risk.

The rest of this chapter is organised as follows. Section 4.2 gives a recap of existing clustering methods and outlines the hierarchical agglomerative

clustering algorithm. Section 4.3 introduces our novel spatial agglomerative clustering approach, while section 4.4 tests this method and also compares the proposed linkage methods. Section 4.5 outlines an application of our methodology, based on respiratory hospital admissions in the Greater Glasgow area in 2011. Finally, section 4.6 discusses the advantages of this method and outlines how it will be used in the methods which will be introduced in Chapters 5 and 6.

4.1.1 Notation

The following notation will be used in the context of the clustering approaches outlined in this chapter as well as in the next two chapters.

Let $(\boldsymbol{Y}^{(1)}, \boldsymbol{E}^{(1)}), \ldots, (\boldsymbol{Y}^{(q)}, \boldsymbol{E}^{(q)})$ denote the observed and expected disease counts for the q time intervals (usually years) preceding the study period. We use these earlier data to elicit a set of n potential cluster configurations for the study data, which are denoted here by $\{\mathcal{C}_1, \ldots, \mathcal{C}_n\}$. Here, the areal units are used as the clustering objects, and $\mathcal{C}_k = \{\mathcal{C}_k(1), \ldots, \mathcal{C}_k(k)\}$ partitions the n areal units $\mathcal{A} = \{\mathcal{A}_1, \ldots, \mathcal{A}_n\}$ into k spatially contiguous groups, where $\mathcal{C}_k(j)$ is the jth cluster. These n candidate structures are then used in the modelling approaches developed in Chapters 5 and 6.

The data are clustered on the log standardised incidence ratio scale, that is $\ln\left(\frac{\boldsymbol{Y}^{(j)}}{\boldsymbol{E}^{(j)}}\right)$, because it corresponds to the linear predictor scale in (2.2). Let

$\boldsymbol{\psi} = \left[\ln\left(\frac{\boldsymbol{Y}^{(1)}}{\boldsymbol{E}^{(1)}}\right), \ldots, \ln\left(\frac{\boldsymbol{Y}^{(q)}}{\boldsymbol{E}^{(q)}}\right)\right]$ be the $n \times q$ matrix whose columns comprise $\ln\left(\frac{\boldsymbol{Y}^{(j)}}{\boldsymbol{E}^{(j)}}\right)$ for $j = 1, \ldots, q$. The ith row is given by $\boldsymbol{\psi}_i = \left[\ln\left(\frac{Y_i^{(1)}}{E_i^{(1)}}\right), \ldots, \ln\left(\frac{Y_i^{(q)}}{E_i^{(q)}}\right)\right]$, the vector of q values for areal unit $\mathcal{A}_i$, and it is these vectors upon which the clustering algorithm will be applied. Note that if any of the expected values are zero, that is $E_i^{(j)} = 0$, then a small constant must be added to prevent dividing by zero.

4.2 Recap of clustering methods

As discussed in Section 2.6, the aim of clustering is to divide a set of objects into a set of disjoint groups (clusters) based on their characteristics. Objects which are similar should be in the same cluster, while objects which are different should be kept apart; in other words there should be homogeneity within a cluster but heterogeneity between clusters. Many different approaches can be used to partition the objects into clusters, but one of the most common is hierarchical agglomerative clustering. Hierarchical agglomerative clustering is an iterative process where we start by considering each object as its own singleton cluster. At each iteration of the algorithm, the two least dissimilar clusters are joined together to form one larger cluster until eventually we end up with a single cluster which contains every object.

The hierarchical clustering algorithm is outlined in Section 2.6.2. This algorithm produces a set of n potential cluster structures, $\{\mathcal{C}_1, \ldots, \mathcal{C}_n\}$, containing between 1 and n clusters. The agglomerative nature of the clustering

means that there is an inherent nesting within these cluster structures; for example $\mathcal{C}_{(h-1)}$ only differs from $\mathcal{C}_h$ in that two of the clusters in the latter have been joined together in the former. The algorithm does not distinguish between the n cluster structures produced, and each of them must therefore be considered as a candidate to be the most appropriate structure for the data.

4.3 Spatial clustering

Traditional clustering methods such as the agglomerative hierarchical clustering approach outlined in Section 4.2 group together objects based purely on the dissimilarities between areas, and therefore do not take into account the spatial context of the data. Applying such methods to areal data would allow us to identify clusters of areal units which have similar disease risk, but there would be no spatial restrictions on these clusters. Under such an approach, a cluster could contain areas which are geographically distant from each other. In many spatial applications such an approach will not be sensible, because the specific aim will be to identify groups of *neighbouring* areal units which have similar risks. Spatially contiguous clusters are easier to interpret for a non-statistician, because the study area will be partitioned into clear regions which each share similar levels of disease risk. Such applications allow health boards or government agencies to identify regions of high (or low) risk for a particular disease in order to make some form of medical intervention or policy decision within that region. In order to produce spatially contiguous clusters it is necessary to extend the traditional clustering

approaches to account for the neighbourhood structure which exists within the data.

4.3.1 Spatial agglomerative hierarchical clustering approach

Our aim is to develop a clustering algorithm which can be applied to disease risk data to produce spatially contiguous clusters. We achieve this by extending the agglomerative hierarchical clustering algorithm described in Section 4.2 to take account of the spatial structure of the data. Our novel spatial agglomerative hierarchical clustering approach introduces an extra restriction to the joining step of the algorithm by only allowing two clusters to be joined together if they share at least one common border.

A set of n potential cluster structures $\{\mathcal{C}_1, \ldots, \mathcal{C}_n\}$ are produced, each containing between 1 and n spatially contiguous clusters. As in Section 4.2, an inherent nesting exists within the cluster structures with $\mathcal{C}_{(h-1)}$ only differing from $\mathcal{C}_h$ in that two of the clusters in the latter have been joined together in the former. This algorithm does not distinguish between the n cluster structures produced, and each of them must therefore be considered as a candidate to be the most appropriate structure for the data. Different methods of deciding which candidate is suitable will be explored in Chapters 5 and 6.

The spatial hierarchical agglomerative clustering algorithm proposed here is

as follows:

Algorithm

1. Choose a distance metric and linkage method.

2. Construct $\mathcal{C}_n = \{\mathcal{C}_n(1), \ldots, \mathcal{C}_n(n)\}$, an initial cluster structure where each areal unit $\mathcal{A} = \{\mathcal{A}_1, \ldots, \mathcal{A}_n\}$ is in its own singleton cluster.

3. Repeat the following steps for $h = n, \ldots, 2$, where step h produces $\mathcal{C}_{h-1}$ from $\mathcal{C}_h$.

   (a) Compute the $h \times h$ distance matrix D, whose klth element is given by

   $$D_{kl} = \begin{cases} d_{kl} & \text{if } k \sim l \,\&\, k > l \\ \infty & \text{otherwise,} \end{cases}$$

   where d_{kl} is the distance between clusters $(\mathcal{C}_h(k), \mathcal{C}_h(l))$ under the selected linkage method and distance metric, and $k \sim l$ means that the clusters contain at least one pair of areas that share a common border.

   (b) Set $\{i, j\} = \arg\min(D_{kl})$, that is the identifiers of the two clusters that have the minimum dissimilarity as measured by the linkage method. In case of ties, $\{i, j\}$ is randomly selected from these.

   (c) Compute

   $$\mathcal{C}_{h-1} = \{\mathcal{C}_h(1), \ldots, \mathcal{C}_h(i-1), \mathcal{C}_{h-1}(i), \mathcal{C}_h(i+1), \ldots, \mathcal{C}_h(j-1), \mathcal{C}_h(j+1), \ldots, \mathcal{C}_h(h)\},$$

   where $\mathcal{C}_{h-1}(i) = \mathcal{C}_h(i) \cup \mathcal{C}_h(j)$.

4.4 Simulation study to test linkage methods

4.4.1 Aim

A simulation study was carried out to assess the quality of the clustering algorithm outlined in Section 4.3.1, and to compare the performance of three common linkage methods, single, centroid and Ward's. These linkage methods are explained in detail in Section 2.6.2.

4.4.2 Data Generation

Clustered disease data were generated under the template shown in Figure 4.1, which consists of 19 clusters of different sizes. There is a large cluster shaded in light grey and 18 smaller clusters shaded in either white or dark grey, some of which are singletons. A set of cluster means, $\boldsymbol{\mu_C} = (\mu_{C_1}, \ldots, \mu_{C_n})$ is constructed by multiplying the cluster values by a constant C, where larger values of C represent larger differences between the clusters.

In order for our simulated data to reflect the true Glasgow data, each of the simulated data sets consist of the study data plus three sets of "prior" data, with the "prior" data being used for the clustering. To allow for the fact that the log risk surfaces for the study and prior data sets are unlikely to be identical, uniform random noise was added to the random effects from the three prior data sets, which corresponds to multiplicative random noise on the risk scale. To provide a suitable analogue with real data across three

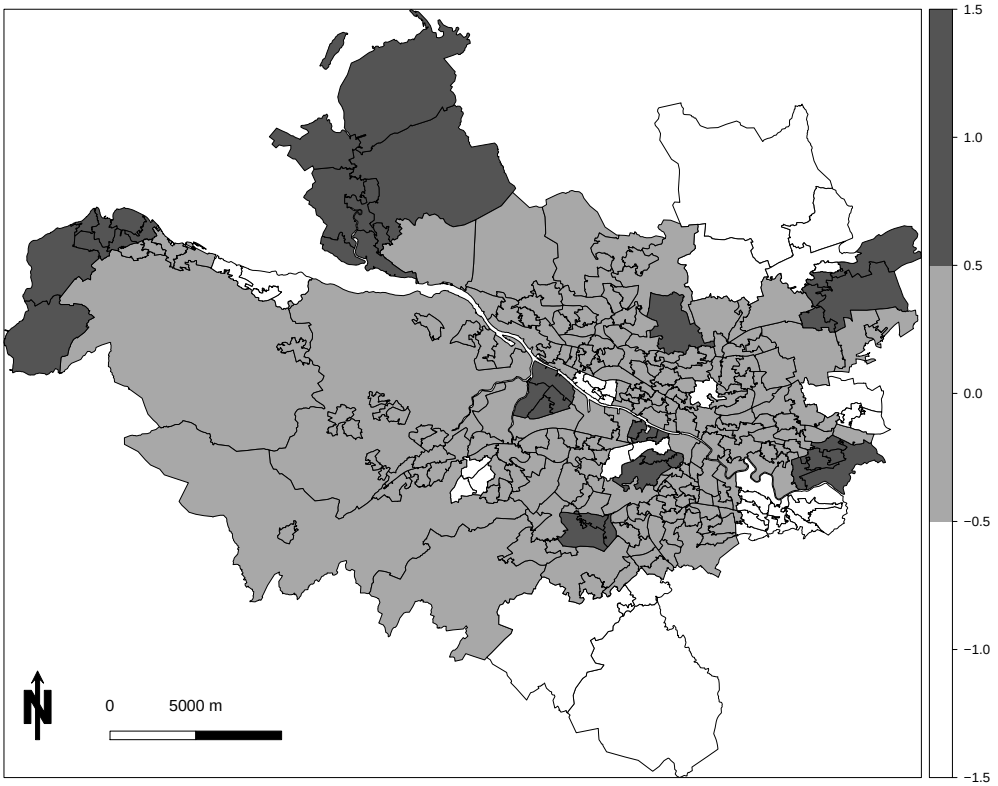

Figure 4.1: Plot of the simulated cluster structure in the Greater Glasgow area.

different years, different levels of noise were added to the three prior data sets, with larger noise added to the data which were further away in time. The uniform random noise for the three prior data sets were on the following intervals $[-0.05, 0.05]$, $[-0.1, 0.1]$ and $[-0.15, 0.15]$, and were chosen to match the correlations between the study and prior data sets for the real Glasgow respiratory admissions data.

The data were generated from the model below (similar to model (3.1) with the simplification that no covariates are included):

$$
\begin{aligned}
Y_{ij}|E_i, R_{ij} &\sim \text{Poisson}(E_i R_{ij}) \qquad i = 1, \ldots, n, j = 1, \ldots, 4, \\
\ln(R_{ij}) &= \phi_i + u_{ij}. \\
u_{i1} &= 0, \\
u_{i2} &\sim \text{Uniform}(-0.05, 0.05), \\
u_{i3} &\sim \text{Uniform}(-0.1, 0.1), \\
u_{i4} &\sim \text{Uniform}(-0.15, 0.15), \\
\phi_i &\sim \text{N}(\boldsymbol{\mu_C}, Q^{-1}).
\end{aligned} \tag{4.1}
$$

where $\boldsymbol{Y_{\cdot 1}}$ represents the "study" data and $\boldsymbol{Y_{\cdot 2}}, \boldsymbol{Y_{\cdot 3}}$, and $\boldsymbol{Y_{\cdot 4}}$ represent the "prior" data.

The random effects were generated from a multivariate Gaussian distribution with a spatially correlated precision matrix, $Q = \rho * (\text{diag}(W\mathbf{1}) - W) + (1 - \rho)I_n$, where $W\mathbf{1}$ is a vector containing the number of neighbours for each areal unit and I_n is an $n \times n$ identity matrix. This precision matrix corresponds to the Leroux CAR prior outlined in Section 2.4.2. The mean, μ_C, of the random effects, $\boldsymbol{\phi}$, follows a piecewise constant mean function which is based on the template shown in Figure 4.1. The values in Figure 4.1 are multiplied by C, where larger values of C represent larger differences between the clusters, which should thus be easier to identify. Values of $C = 0.5, 1$ are used in this study; $C = 1$ corresponds to a case where there are large differences between the clusters while $C = 0.5$ corresponds to a more difficult case where there are smaller differences. Examples of the "study" data simulated under each of these values of C are provided in Figure 4.2.

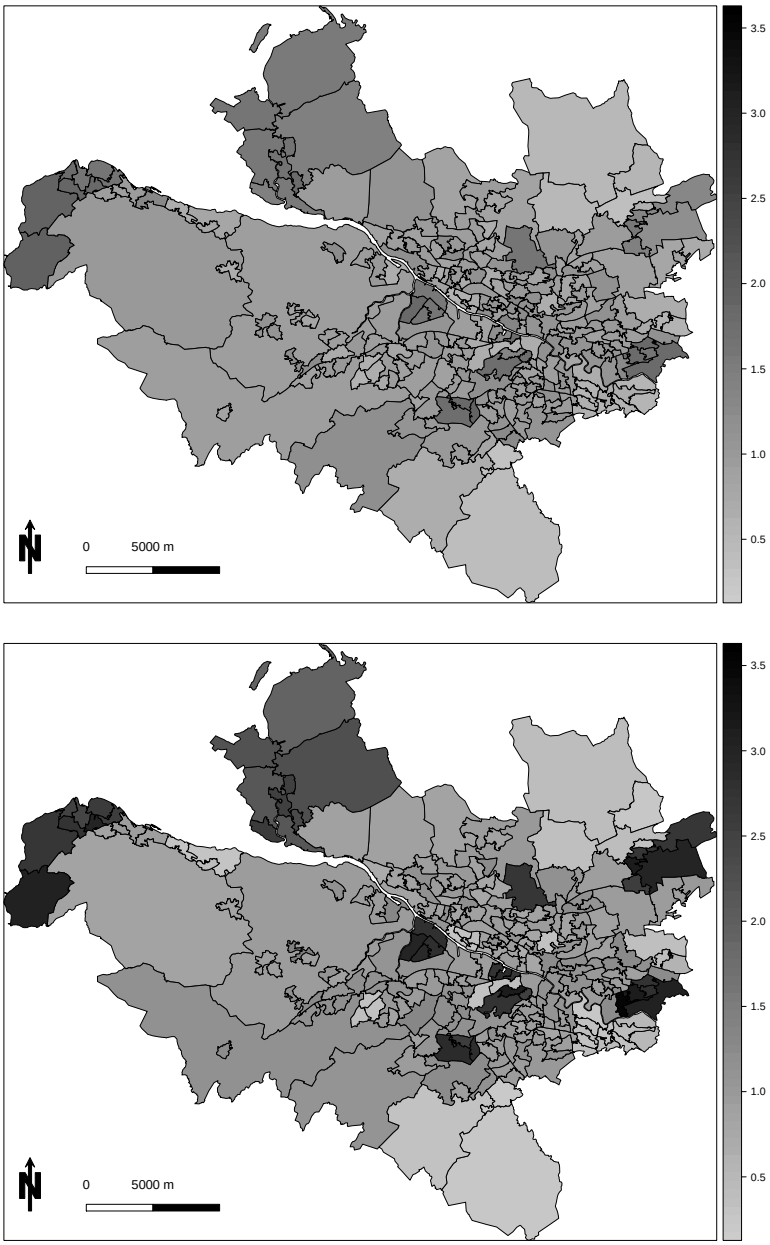

Figure 4.2: Clustered disease data were generated in order to test the quality of the clustering algorithm. The top panel shows the data generated with $C = 0.5$ while the bottom panel shows the data generated with $C = 1$.

4.4.3 Results

One hundred datasets were simulated for each value of C, and all three linkage methods were applied to each dataset. For each of these simulations we recorded the following:

- *True Matches* - Whether or not the "true" cluster structure was included amongst the candidate clusterings.
- *Number of Clusters* - The number of clusters relating to the maximum Rand Index.
- *Maximum Rand Index* - How close the best candidate clustering was to the true clustering. This was done by calculating the Rand Index between the "true" clustering and each of the candidate clusterings, and identifying the maximum Rand Index for each simulation. For details on the Rand Index, see Section 2.6.
- *Rand 19* - How close the candidate clustering containing 19 clusters was to the true clustering. This was done by calculating the Rand Index between the "true" clustering and the candidate clustering containing 19 clusters.

Table 4.1 displays the number of true matches for each linkage method under the two values of C. For $C = 1$, the true cluster structure was identified in the majority of cases for both centroid (99%) and Ward's linkage (88%), but single linkage only managed to identify the true structure on 2% of occasions. For $C = 0.5$, the true cluster structure was only identified in a very small

	Mean Diff.	Single Linkage	Centroid Linkage	Ward's Linkage
True Matches	$C = 0.5$	0 (0.000)	8 (0.273)	4 (0.197)
	$C = 1$	2 (0.141)	99 (0.100)	88 (0.327)
No. of Clusters	$C = 0.5$	13 (11.2)	20 (2.11)	17 (3.33)
	$C = 1$	42 (17.0)	19 (0.10)	19 (0.60)
Max Rand	$C = 0.5$	0.658 (0.043)	0.989 (0.011)	0.972 (0.086)
	$C = 1$	0.836 (0.092)	1 (0.000005)	1 (0.018)
Rand 19	$C = 0.5$	0.623 (0.049)	0.978 (0.023)	0.799 (0.109)
	$C = 1$	0.693 (0.085)	1 (0.0005)	1 (0.072)

Table 4.1: Results of the simulation study to test the clustering algorithm.

number of simulations (8% for centroid linkage, 4% for Ward's linkage and 0% for single linkage). This is not necessarily surprising, as in this case of $C = 0.5$ the cluster structure in the data is not that strong, and the algorithm only needs to make one wrong move for the true cluster structure to be missed. For a fairer assessment of the performance of the algorithm we must therefore look at the maximum Rand Index to see how close we get to identifying the true cluster structure for each linkage method.

The top panel of Figure 4.3 shows the number of clusters corresponding to the maximum Rand Index in each case, with the dashed line indicating the true number of clusters (19). For $C = 1$, both centroid and Ward's linkage produce a median of 19 clusters, with centroid linkage (0.100) having a lower standard deviation than Ward's linkage (0.603). In the more difficult case

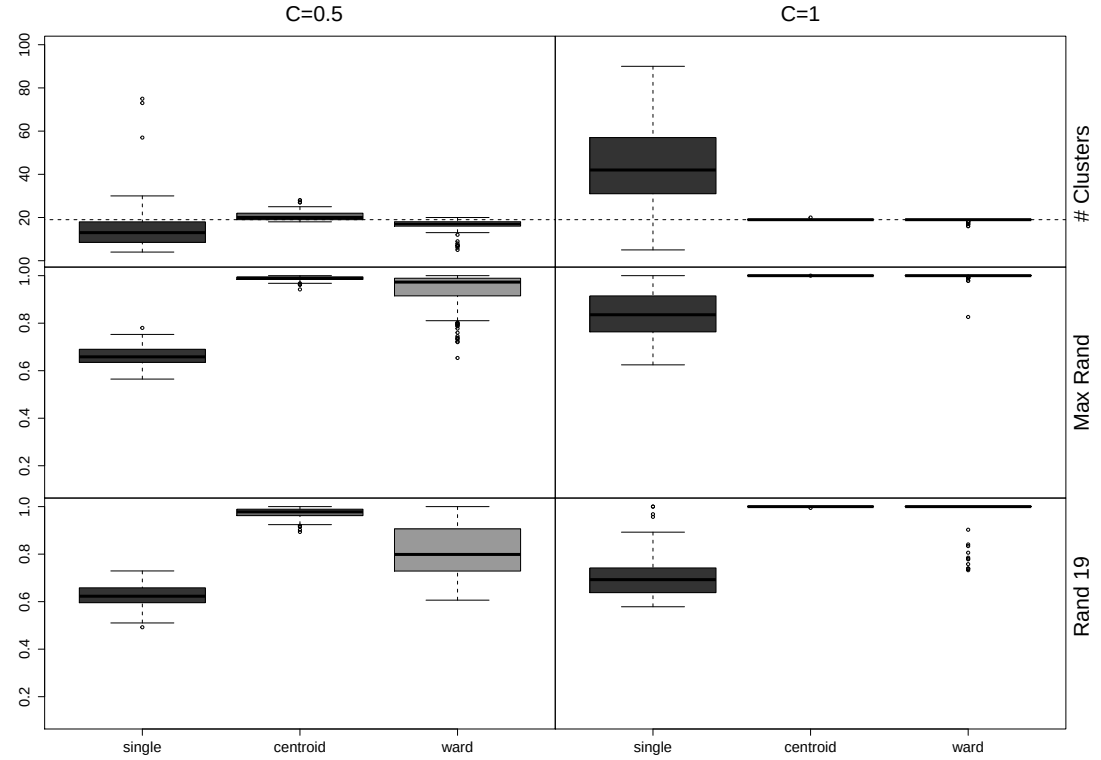

Figure 4.3: A comparison of the three linkage methods in the simulation study. The left column contains the results for $C = 0.5$ while the right column contains the results for $C = 1$. The top row displays the number of clusters relating to the maximum Rand Index, the middle row contains the maximum Rand Index value and the bottom row contains the Rand Index for the clustering with 19 clusters.

with $C = 0.5$, centroid linkage is again shown to be the most accurate, with a median of 20 clusters compared to 17 clusters for Ward's linkage and 13 for single linkage, and a standard deviation of 2.11 compared to 3.33 for Ward's linkage and 11.25 for single linkage.

The middle row of Figure 4.3 displays the maximum Rand Index score obtained under each linkage method for each value of C. For $C = 1$, the figure shows that the median value for both centroid and Ward's linkage is 1, the maximum score possible, which is unsurprising since both methods correctly identified the true cluster structure in the majority of cases. However a comparison of standard deviations shows that Ward's linkage (0.018) has a larger spread than centroid linkage (0.000005). Single linkage has a median value of 0.836 and a standard deviation of 0.092, suggesting that it performs reasonably well, but is less successful than the other two linkage methods in identifying accurate cluster structures. In the more difficult case where $C = 0.5$, the median Rand Index is still relatively high for centroid linkage (0.989) and Ward's linkage (0.972). This suggests that even in cases where a correct match is not obtained, these linkage methods are still able to identify cluster structures which are very similar to the true clustering. Again centroid linkage (0.011) has a much lower standard deviation than Ward's linkage (0.086), a feature which is displayed graphically by the longer tails for Ward's linkage in Figure 4.3.

The bottom panel of Figure 4.3 shows that similar results were obtained for the Rand Index values for 19 clusters. In the case of $C = 1$, both centroid and Ward's linkage give a median Rand 19 value of 1, which is unsurprising

since the correct cluster structure (containing 19 clusters) was obtained in the majority of cases for both and thus the Rand Index for all of these will have a value of 1. Once more, centroid linkage (0.0005) displays a lower standard deviation than Ward's linkage (0.072), which shows that centroid linkage is producing more consistent results. For $C = 0.5$, centroid linkage (0.978) performs much better than Ward's linkage (0.799) and single linkage (0.623) in terms of median Rand 19, which suggests that centroid linkage will produce the most accurate cluster structures for the true number of clusters. Centroid linkage also performs best in terms of standard deviation.

Taking these criteria into account, it appears that centroid linkage is the most effective linkage method, and it will therefore be used for all further applications of clustering within this thesis. The Rand Index values for centroid linkage show that even in a difficult case ($C = 0.5$), the cluster structures produced are close to the true cluster structures, which suggests that the algorithm itself is effective in producing accurate cluster structures.

4.5 Real data example

In order to illustrate the kind of clusters produced, we apply our spatial agglomerative hierarchical algorithm to Glasgow respiratory admissions data for 2011. The study region is the Greater Glasgow and Clyde Health Board area, which contains the city of Glasgow in the east and the river Clyde estuary in the west. Glasgow is the largest city in Scotland, with a population

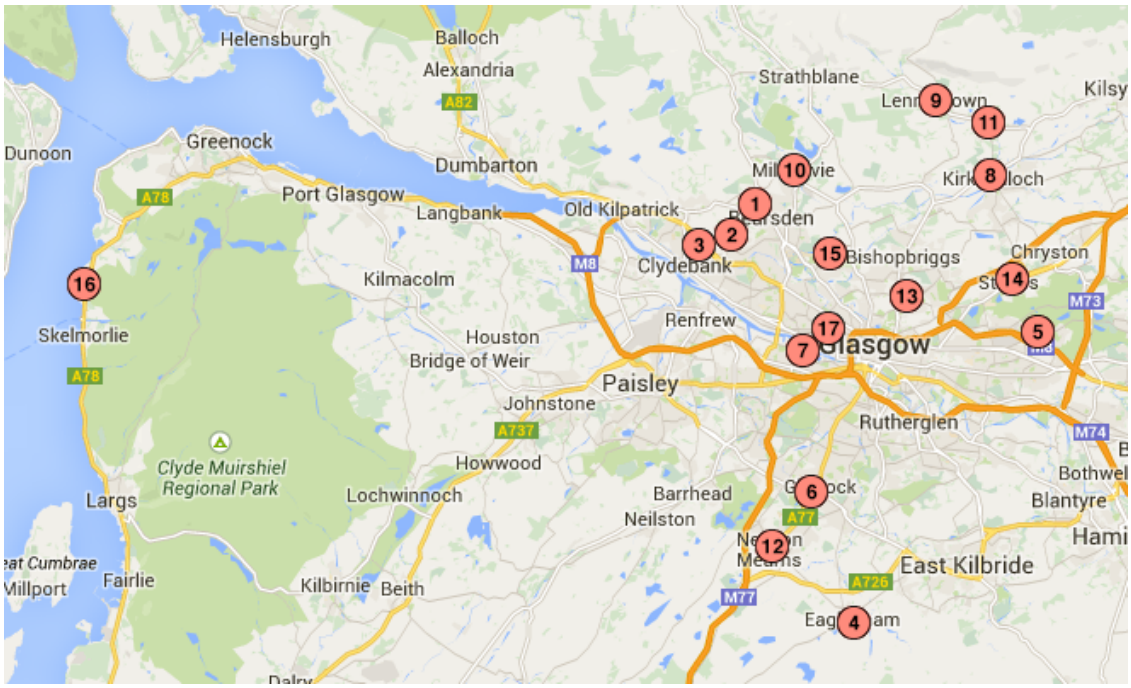

Figure 4.4: A map of Glasgow and the surrounding areas, with locations mentioned in this thesis identified. A key can be found in Table 4.2.

Number	Area	Number	Area	Number	Area
1	Bearsden	7	Govan	13	Springburn
2	Drumchapel	8	Kirkintilloch	14	Stepps
3	Drumry	9	Lennoxtown	15	Summerston
4	Eaglesham	10	Milngavie	16	Wemyss Bay
5	Easterhouse	11	Milton of Campsie	17	West End
6	Giffnock	12	Newton Mearns		

Table 4.2: Key for Figure 4.4.

of around 600,000 people. The health board is split into $n = 271$ administrative units known as Intermediate Geographies (IGs), containing populations of between 2,244 and 10,877 people with a median value of 4,239. Figure 4.4 contains a map of Glasgow and the surrounding areas, with pins in the map to identify each location which is mentioned in this thesis. Table 4.2 provides a key for this map, with the numbers in the table corresponding to those in the pins in Figure 4.4.

The disease data were obtained from the Scottish Neighbourhood Statistics website (http://www.sns.gov.uk/Downloads/AdHocChoose.aspx) by selecting the following drop downs in turn: "Intermediate Geography - 2006 Health Board - Greater Glasgow & Clyde - Health - Hospital Admissions - Respiratory Disease - Respiratory Disease, both sexes". For each of the 271 areal units, we use ten years of observed data from 2002-2011, which consists of a count of respiratory admissions for each areal unit in each year. The expected respiratory admissions can be obtained by first downloading the disease rates from ISD Scotland from the link (http://www.isdscotland.org/Health-Topics/Hospital-Care/Diagnoses) and then computing expected disease risk for each area using external standardisation as described in Section 3.2. The response data, $\boldsymbol{Y} = (Y_1, \ldots, Y_n)$, are based on the 2011 data, where Y_i is the number of hospital admissions with a primary diagnosis of respiratory disease in areal unit i in 2011, which corresponds to the International Classification of Disease tenth revision codes J00-J99 and R09.1. The expected values, $\boldsymbol{E} = (E_1, \ldots, E_n)$, are the expected hospital admission numbers for each areal unit in 2011. Here, for illustrative purposes, the algorithm is ap-

plied to the log of the SIR for these data for 2011, represented by the vector $\boldsymbol{v}$ where $v_i = \ln(\text{SIR}_i) = \frac{Y_i}{E_i}$.

Figures 4.5 and 4.6 display a selection of the cluster structures produced by the algorithm, namely those containing 5, 10, 20 and 30 clusters. Each plot displays the 2011 SIR values for each areal unit in greyscale, with the clusters indicated by white dots. The agglomerative nature of the clustering means that the structures with fewer clusters were formed by merging the clusters in the structures which have more clusters; for example the clusters in the 5 cluster structure were formed by merging clusters in the 10 cluster structure. This makes it inevitable that there will be many similarities between the different structures, and many features will be present in each of the plots.

The 5 cluster structure in the top panel of Figure 4.5 consists of one very large cluster plus a small cluster in the south-east and three singleton clusters. One of the singleton clusters is a high-risk areal unit which is substantially different from its neighbours, while the other two are low risk areal units which are slightly different to neighbouring areas. The small cluster in the south-east appears to display lower risk than many of its neighbours. These features are all present in the 10 cluster structure in the bottom panel of 4.5, while the very large cluster is divided into a number of smaller clusters. Three small low-risk clusters have been identified, including one just north of the river which contains the affluent West End. It has also identified a large low-risk cluster to the north of the city and another small cluster with a slightly higher risk to the north-east.

The 20 cluster structure in the top panel of Figure 4.6 picks out a number of smaller features, including a small low-risk cluster to the east of the city and two small higher-risk clusters to the north. The 30 cluster structure in the bottom panel of Figure 4.6 identifies many additional features, most notably the high-risk cluster north of the river, containing Drumry and Drumchapel. In each case, the majority of clusters appear to be sensible, with visual differences in risk between most pairs of neighbouring clusters.

4.6 Discussion

Clustering of disease maps allows for the identification of groups of areal units which exhibit similar risks, and therefore provides a crucial tool in the detection groups of high or low risk areas. Knowledge of the extent and the location of such clusters is extremely valuable for government agencies and health boards because it allows them to pinpoint high risk areas which can then be the focus of targeted health interventions. In this chapter we have introduced a new spatial agglomerative clustering algorithm for areal disease risk data. The algorithm uses data from a time period prior to the study period to produce a set of n cluster structures, each of which partitions the study region into spatially contiguous clusters with similar disease risk. The agglomerative nature of the clustering algorithm means that there is a natural ordering of these n cluster structures; the first structure contains 1 cluster and the last structure contains n clusters. The algorithm proposed

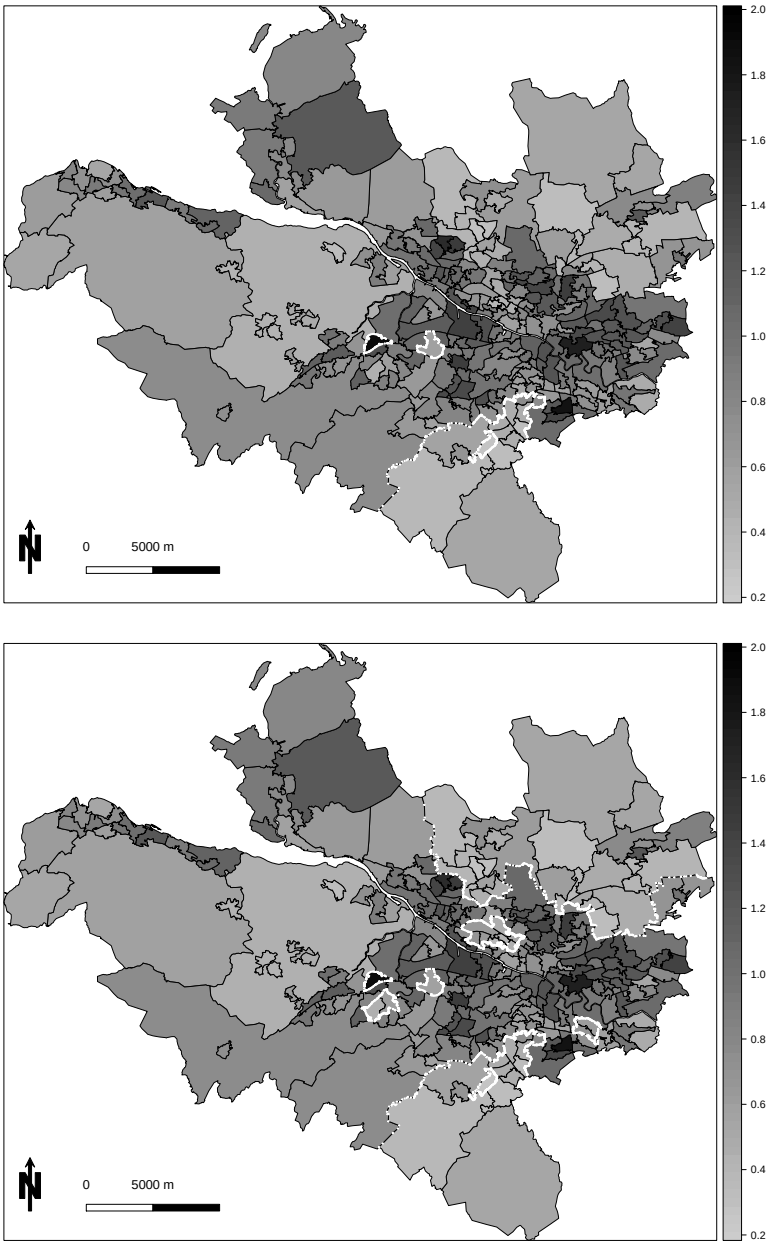

Figure 4.5: Clustering was carried out on the 2008-2010 SIR values, and the cluster structures containing 5 and 10 clusters are plotted on the top and bottom panels respectively. Each plot displays the 2011 SIR values (greyscale) with clusters indicated by white dots.

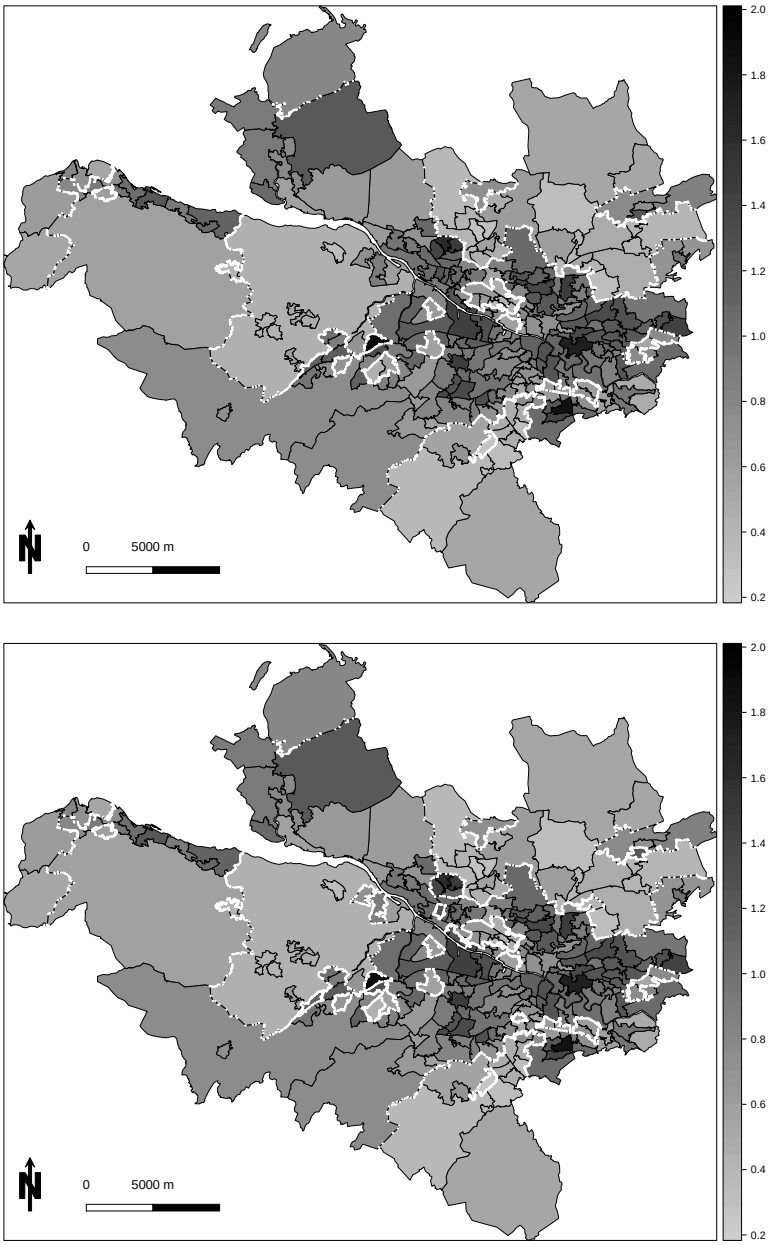

Figure 4.6: Clustering was carried out on the 2008-2010 SIR values, and the cluster structures containing 20 and 30 clusters are plotted on the top and bottom panels respectively. Each plot displays the 2011 SIR values (greyscale) with clusters indicated by white dots.

here is reliant on the existence of data from prior time periods, which are generally readily available for most disease mapping examples. In the case that such data does not exist, an alternative set of data would be required to estimate the cluster structure; one such alternative would be to use covariate information which has a strong correlation with the disease risk.

The algorithm outlined here could be used as a standalone exploratory technique for the identification of high and low risk clusters, as outlined in Section 4.5. However, the main motivation for developing the algorithm was as Stage 1 of a two stage model for simultaneously estimating the spatial pattern in disease risk and detecting the spatial extent of high or low risk clusters. In Chapters 5 and 6 we will propose two different Bayesian modelling approaches for identifying the most appropriate cluster structure and estimating the disease risk. These approaches provide two alternative options for Stage 2 of the two stage model, and both of these modelling approaches involve some form of comparison of the potential cluster structures. The set of all possible spatially contiguous clusterings for the study region $\mathcal{A}$ is very large, and it would be extremely computationally intensive to compare all of these. By using our algorithm to elicit a set of n candidate cluster configurations, we reduce the computational burden to a manageable level and can thus carry out the comparisons required in Chapters 5 and 6. The simulation study showed that the algorithm was successful in identifying either the correct cluster structure or a cluster structure which was very similar to the true structure, and thus we can be confident that the set of n candidate cluster configurations will contain sensible clusterings for the disease risk data.

Chapter 5

Identifying spatial clusters using a mean (fixed effects) based approach.

5.1 Introduction

The spatial surface of disease risk is commonly modelled as being spatially smooth, but as discussed in Section 2.4.2 this may not always reflect the true spatial pattern of the data. There may be pairs of areal units which exhibit vastly different disease risks despite being close geographically, often as a result of contrasting risk-inducing behaviours within the populations of the areas. Section 3.4 discussed a number of existing methods for identifying and modelling these spatial discontinuities, but many of these methods produce open rather than closed boundaries, or require computationally complex

modelling approaches such as reversible jump McMC simulation.

Here we will propose a new modelling approach for identifying spatial clusters (discontinuities) in the disease risk pattern. Our proposed model divides the study area into a set of spatially contiguous clusters of areal units based on the similarity of their disease risks, but also estimates a separate (albeit correlated) risk in each individual areal unit within a cluster. Our modelling approach is in two stages, the first of which uses the spatially-adjusted hierarchical clustering algorithm introduced in Chapter 4 to elicit a set of n candidate cluster configurations containing between 1 and n clusters. In this chapter we introduce a Bayesian model for the second stage of our modelling approach, where the aim is to select the best of the cluster structures proposed in Stage 1, and simultaneously estimate the disease risk across the study region.

A spatial cluster model represents a belief that a group of neighbouring areas exhibit a different level of underlying disease risk than other neighbouring areas. Our approach allows for this by assigning different mean risk levels to each cluster via a fixed effect, based on the cluster structures obtained in Chapter 4. Specifically, the model proposed in this chapter combines the smooth intrinsic CAR model with a piecewise constant cluster model, thus allowing disease risk to follow a spatially smooth pattern within a cluster whilst having a disjoint jump between clusters. In Chapter 6 we introduce an alternative Bayesian model for Stage 2, which accounts for the discontinuities by modelling the correlation structure in the random effects rather

than introducing mean level fixed effects. These two modelling approaches will be compared via a simulation study in Chapter 6.

The rest of this chapter is organised as follows. Section 5.2 outlines the fixed effect model proposed here and links it to the clustering algorithm in Chapter 4, while section 5.3 tests this model against an existing method using simulated data. In Section 5.4, additional simulation studies are carried out to assess the sensitivity of the model in a wider range of situations, before Section 5.5 outlines an application of our methodology, based on respiratory hospital admissions in the Greater Glasgow area in 2011. Finally, section 5.6 discusses the advantages of this modelling approach as well as discussing how it will be developed in Chapters 6 and 7.

5.2 Fixed effect model

We propose a two-stage approach for estimating the spatial pattern in disease risk and identifying high or low risk clusters. The first stage uses the clustering algorithm described in Chapter 4 to produce a set of candidate cluster structures $\{\mathcal{C}_1, \ldots, \mathcal{C}_n\}$, with each containing a different number of clusters between 1 and n. These candidate cluster structures are all considered as potentially being the structure which best fits the data, and we must compare these structures to decide which is the optimal clustering of the areal units. Thus, the second stage of our modelling process involves a model comparison approach. For each of those n cluster configurations in turn, we fit a separate Bayesian hierarchical model to the study data. Since the only variable

changing across these models is the choice of cluster structure, this modelling procedure can be considered as a comparison of the potential cluster structures. The Deviance Information Criterion (DIC) for each of these models can be compared, with the cluster structure corresponding to the model with the lowest DIC being selected as the most appropriate cluster structure for the data. The DIC is outlined in more detail in Section 2.3.

The proposed model is as follows, for a given cluster structure $\mathcal{C}_k$ containing k clusters:

$$\begin{aligned} Y_i|E_i, R_i &\sim \text{Poisson}(E_i R_i) \qquad i = 1, \ldots, n, \\ \ln(R_i) &= \phi_i + \sum_{j=1}^{k} I[\mathcal{A}_i \in \mathcal{C}_k(j)]\alpha_j, \\ \alpha_j &\sim \text{N}(0, 100) \qquad j = 1, \ldots, k, \\ \phi_i|\boldsymbol{\phi}_{-i} &\sim \text{N}\left(\frac{\sum_{j=1}^{n} w_{ij}\phi_j}{\sum_{j=1}^{n} w_{ij}}, \frac{1}{\tau(\sum_{j=1}^{n} w_{ij})}\right), \\ \tau &\sim \text{Gamma}(1, 1), \end{aligned} \tag{5.1}$$

This model allows disease risk to evolve smoothly within a cluster whilst having a disjoint multiplicative jump between clusters. This is achieved by combining the smooth intrinsic CAR model (2.5) for $\boldsymbol{\phi}$ with a piecewise constant cluster model. The former is equivalent to writing the multivariate formulation $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_n) \sim \text{N}\left(\mathbf{0}, \frac{Q^{-1}}{\tau}\right)$, where Q is a singular precision matrix given by $Q = \text{diag}(W\mathbf{1}) - W$ where $W\mathbf{1}_i = \sum_{j=1}^{n} w_{ij}$. The simpler intrinsic prior is preferred to the Leroux prior because the localised structure

in the data is captured by the fixed effects in the model. The piecewise constant cluster model is defined by $\sum_{j=1}^{k} I[\mathcal{A}_i \in \mathcal{C}_k(j)]\alpha_j$ on the linear predictor scale. Here, $I[.]$ denotes an indicator function, so that $I[\mathcal{A}_i \in \mathcal{C}_k(j)]$ equals 1 if areal unit $\mathcal{A}_i$ lies in cluster j and is 0 otherwise. Thus this piecewise constant cluster model is essentially a single categorical covariate with k levels, where each cluster represents a different level. Therefore α_j is the mean risk level in cluster j. We note that when areal unit $\mathcal{A}_i$ is in a singleton cluster, then this model essentially includes an indicator variable for that areal unit, resulting in the fitted value equalling the observed value. We considered modelling the cluster parameters $(\alpha_1, \ldots, \alpha_k)$ as random rather than fixed effects, that is a model such as $\alpha_j \sim \mathrm{N}(0, \sigma^2)$, but an initial simulation study showed that this resulted in poor performance in terms of cluster identification. In order to reduce the computational time, we only fit the model for cluster structures containing $1 : m$ clusters, where m is a sensible upper limit for the number of clusters you would expect to find. Finally, the hyperparameters $(1, 1)$ in the gamma prior will be varied in the simulation study, to gauge the sensitivity of the results.

Inference for the above model is implemented using integrated nested Laplace approximations (INLA), because fitting the m models corresponding to $\{\mathcal{C}_1, \ldots, \mathcal{C}_m\}$ would be computationally prohibitive using Markov chain Monte Carlo (McMC) methods. Inference using INLA has been shown by Schrödle et al. (2011) to produce almost identical results to McMC simulation. The model above does not include additional covariates other than the factor variable representing the cluster structure, because the goal of the analysis

is to identify clusters in the disease risk surface, not in the residual surface after adjusting for covariate factors.

5.3 Simulation study

5.3.1 Aim

A simulation study was conducted to establish the efficacy of the two-stage modelling approach outlined in the previous section. The template for the study was the set of 271 Intermediate Geographies comprising the Greater Glasgow and Clyde Health Board, which is the same study region as the clustering simulation study presented in Section 4.4. A study was conducted comparing the two-stage approach proposed here with an existing alternative, and the results of this simulation study are outlined below.

5.3.2 Data Generation

Clustered disease data were generated according to the template shown in Figure 4.1. The template consists of 19 clusters of different sizes, which include the large cluster shaded in light grey and the 18 smaller clusters shaded in either white or dark grey, some of which are singletons. Disease data were generated under this template in the same way as described in Chapter 4.4. The random effects were generated from a multivariate Gaussian distribution with a spatially correlated precision matrix, given by $Q = \rho * (\text{diag}(W\mathbf{1}) - W) + (1 - \rho)I_n$, which corresponds to the CAR model

proposed by Leroux et al. (1999). Here $W\mathbf{1}$ is a vector containing the number of neighbours for each areal unit and I_n is an $n \times n$ identity matrix. The value of ρ controls the level of spatial autocorrelation in the data, and here we set $\rho = 0.99$ which corresponds to strong spatial smoothness. Clustered disease data were obtained by specifying a piecewise constant mean function for $\boldsymbol{\phi}$, which follows the template shown in Figure 4.1.

The values in Figure 4.1 are multiplied by C, where larger values of C represent larger differences between the clusters, which should thus be easier to identify. Values of $C = 0, 0.5, 1$ are used in this study; $C = 1$ corresponds to a case where there are large differences between the clusters, $C = 0.5$ corresponds to a more difficult case where there are smaller differences and $C = 0$ corresponds to a spatially smooth risk surface where one would hope to identify a single cluster covering the entire study region. The top panel of Figure 5.1 displays the simulated risk data for $C = 0$ on the same scale as those with $C = 0.5$ and $C = 1$ in Figure 4.2, and it is clear that the data are spatially smooth. In order to get an idea of the extent of the random fluctuations within the simulated data with $C = 0$, the bottom panel of Figure 5.1 displays the simulated data on an alternative scale. Here we can see that there are only very small differences across the study region, and that the distribution of high and low risk areas appears to be completely random. For the analyses described in this section the expected disease counts are set equal to those from the respiratory disease motivating application. However, a sensitivity analysis assessing the robustness of our methodology to changing $\mathbf{E}$ is presented in Section 5.4.

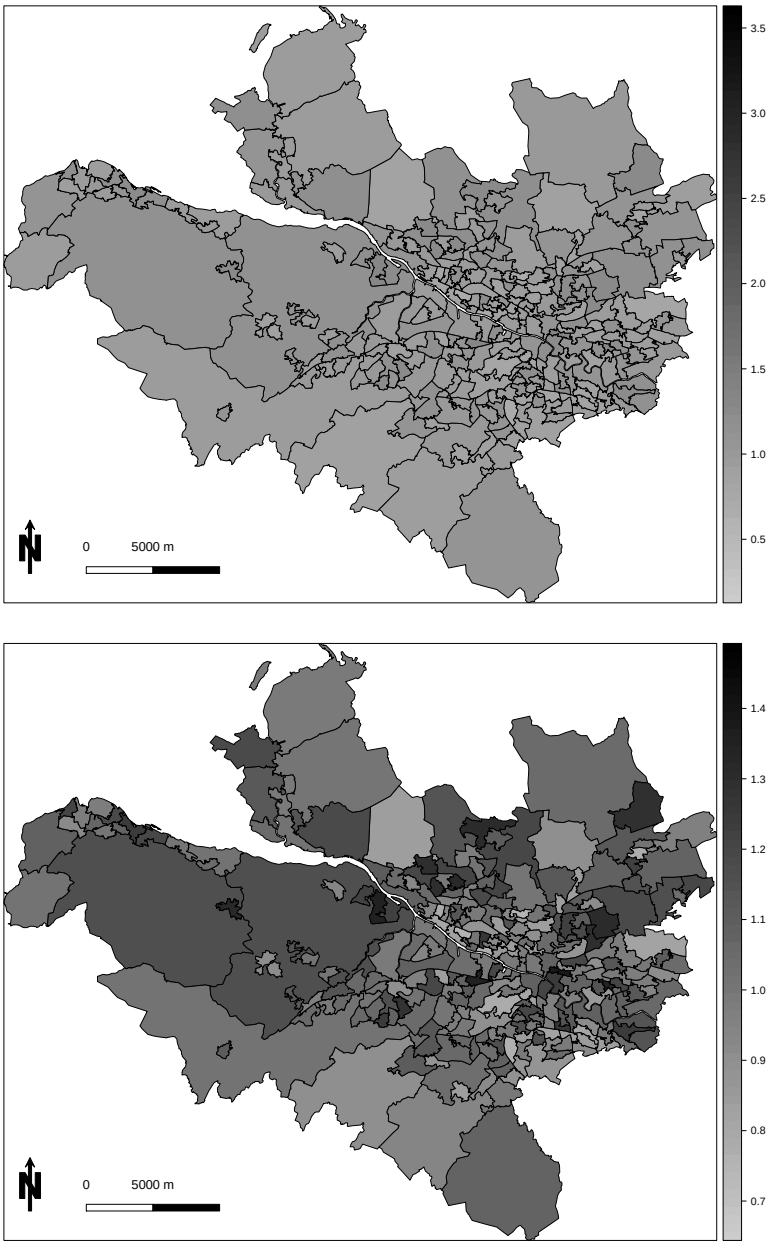

Figure 5.1: Both panels show the data simulated under C=0. The top panel is on the same scale as those in Figure 4.2, while the bottom panel displays them on an alternative scale to illustrate the differences across the study area.

Five hundred datasets were generated for each of the three scenarios ($C = 0, 0.5, 1$), and the model proposed here was compared against the Besag-York-Mollié (BYM, Besag et al. (1991)) model, which is commonly used in disease mapping and was outlined in Section 2.4.2. To identify clusters in the fitted risk surface the posterior classification approach described in Charras-Garrido et al. (2012) and Charras-Garrido et al. (2013) was implemented. However, this approach does not produce spatially contiguous clusters, so a further post-processing step was implemented using the to partition the clusters identified into spatially contiguous groups. This is achieved by taking each cluster in turn and identifying sets of adjacent areal units within that cluster. We note that we have not compared our approach to a method such as Knorr-Held and Rasser (2000), because software to implement these complex estimation methods is not publicly available, and also because they use different inferential frameworks which may affect the results. In contrast, the BYM model was implemented using INLA, which is the inferential approach adopted here. However, we note that by taking such an approach, we are comparing our method which looks for clusters first and then carries out smoothing with an alternative which smooths first and then clusters. A comparison to another clustering first approach, such as that proposed by Kulldorff (1997), may have been appropriate and may be considered in future. Such an approach could involve using the scan statistic to identify a cluster structure which can then be fitted in our fixed effect model. The results of the simulation study in Section 4.4 show that centroid linkage always outperforms single and Ward's linkage methods, and thus centroid linkage will be used here to obtain the set of candidate cluster structures in Stage 1.

5.3.3 Results

The results of the study are outlined in Table 5.1 and summarised in Figure 5.2, which displays a comparison of the relative performances of the approach proposed here and the BYM model with post-hoc clustering, using three different metrics. The accuracy of the risk surfaces estimated by both approaches is quantified by their root mean square error (RMSE), while the correctness of the estimated cluster structures is quantified by both the number of clusters identified and the Rand Index between the true and estimated cluster structures. The latter is a measure of the similarity between two cluster structures and lies in the interval $[0, 1]$. A value of 1 indicates complete agreement between the two cluster configurations, a value of 0 indicates that no pair of areal units are classified in the same way under both configurations and a value of 0.5 is equivalent to random guessing. For more information on the Rand Index, see Section 2.6.

The top panel of Figure 5.2 shows boxplots of the numbers of clusters estimated by each method in the 500 simulated data sets, where the true values of 1 (when $C = 0$) and 19 (when $C = 0.5, 1$) are represented by dashed lines. The middle panel displays boxplots of the Rand index for all simulated data sets, while the bottom panel shows the RMSE values for the estimated risk surface. The top panel shows that when $C = 0$ both methods estimate the correct number of clusters on average, but our method has a lower standard deviation of 1.42 compared to 6.07 for the BYM model. Likewise, for $C = 1$ both approaches correctly identify 19 clusters, but our model has a standard

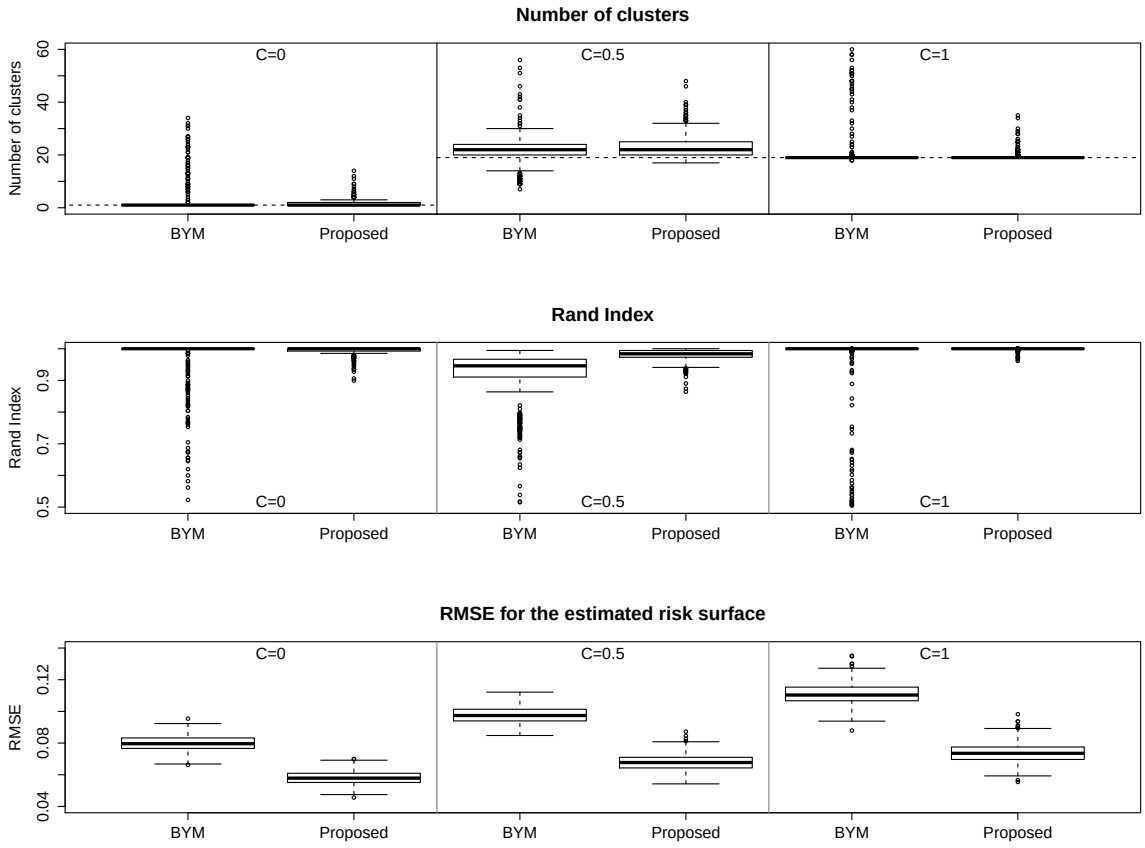

Figure 5.2: Summary of the simulation study results. The top, middle and bottom panels display boxplots of the estimated number of clusters, the Rand Index and the root mean square error of the estimated risk surface for the BYM model and the model proposed here. The results relate to $C = 0$ (left), $C = 0.5$ (middle) and $C = 1$ (right). In the top panel the dashed lines represent the true number of clusters.

	Mean Difference	BYM	Proposed Model
Number of Clusters	$C = 0$	1 (6.07)	1 (1.42)
	$C = 0.5$	22 (6.10)	22 (4.56)
	$C = 1$	19 (8.14)	19 (1.79)
Rand Index	$C = 0$	1 (0.123)	1 (0.012)
	$C = 0.5$	0.946 (0.088)	0.984 (0.019)
	$C = 1$	1 (0.101)	1 (0.004)
RMSE	$C = 0$	0.080 (0.005)	0.058 (0.004)
	$C = 0.5$	0.097 (0.005)	0.068 (0.005)
	$C = 1$	0.110 (0.007)	0.074 (0.006)

Table 5.1: Results of the simulation study to compare the proposed model to the BYM.

deviation of 1.79 compared to 8.14 for the BYM approach. In both cases, this shows that while both methods get the number of clusters correct on average, the approach introduced here gets closer on average than the BYM method in those cases where the correct number of clusters is not identified. When $C = 0.5$ the median values are slightly high at 22 for both models, but again our model has the lower standard deviation of 4.56 compared to 6.10 for the BYM.

From the top row of Figure 5.2, it is apparent that both our model and the BYM model tend to overestimate the number of clusters present. In general, overestimation of the number of clusters is likely to come from a split in a true

cluster, while an underestimation of the number of clusters would require the joining of two or more true clusters. In the case where $C = 1$, the differences between the true clusters are so pronounced that it is very unlikely any of these could be incorrectly joined together, and therefore underestimation is very unlikely. However, it is possible that the sampling variation in the disease counts Y_i within a cluster could cause the model to incorrectly identify a split in the cluster. In the case where $C = 0.5$, underestimation is slightly more likely, but the differences are still pronounced enough that joining two true clusters is unlikely. Overestimation is more likely than before in this case because the random noise is now larger relative to the true differences between clusters. For $C = 0$ there is a much more straightforward reason for the overestimation - the true number of clusters is 1 so it would be impossible to underestimate.

The median Rand Index values are equal to 1 for both models when $C = 1$, but our model has a lower standard deviation of 0.004 compared to 0.101 for the BYM model. Similarly, for $C = 0$, both models have a median Rand Index of 1, but our standard deviation is 0.012 while the BYM has a standard deviation of 0.123. This again suggests that while both models are correct on average, our model performs gets closer to the true cluster structure in those cases where the correct structure is not obtained. For $C = 0.5$ the median values are 0.984 for the model proposed here and 0.946 for the BYM model, and our model also has a lower standard deviation of 0.019 compared to 0.088 for the BYM approach.

Finally, the figure shows that the RMSE is always lower using the method proposed here compared with the BYM model, with reductions in the median of 27.3% ($C = 0$), 30.5% ($C = 0.5$), and 33.3% ($C = 1$) respectively. For $C = 0.5$ and $C = 1$, this is likely to be because the BYM model allows for (incorrect) smoothing between clusters, something which our proposed model does not enforce. This means the smoothing in our model is likely to be more accurate, and thus the estimation of risk is improved. For $C = 0$, the difference in performance between the models is likely to be as a result of the additional non-smooth parameter in the BYM model inducing random noise, causing inaccurate estimation compared to our model which has no such parameter.

5.4 Sensitivity Analyses

Three additional simulation studies were carried out to further test the efficacy of the model and to test the model's sensitivity to the choice of prior distribution and the type of disease data used. Section 5.4.1 presents a sensitivity analysis to the choice of prior distribution for the precision τ. Section 5.4.2 presents additional simulations summarising model performance when diseases of different prevalence are considered (via different size expected disease counts $\mathbf{E}$). Section 5.4.3 contains a comparison of the proposed model with a simplification that only includes the piecewise constant cluster model and not the spatially smooth random effects.

5.4.1 Sensitivity of the prior for τ

The model proposed uses a Gamma(1,1) prior for the precision parameter τ in the CAR model. To assess the effect of this choice of prior on the model fit and the number of clusters selected, we compare our choice of hyperparameters with two alternative choices - Gamma(0.5,0.0005) and Gamma(0.001,0.001). One hundred datasets of clustered disease data were simulated as described in Section 5.3, except that only the value of $C = 0.5$ is used in this study because this represents the most difficult case. Our Bayesian log-linear model was applied to the data using three different choices of prior Gamma distributions for τ. Model 1 used the prior Gamma(1,1), Model 2 used the prior Gamma (0.5,0.0005) and Model 3 used the prior Gamma (0.001,0.001). For each model, the accuracy of the risk surfaces estimated is quantified by root mean square error (RMSE), while the correctness of the estimated cluster structures is quantified by both the number of clusters identified and the Rand Index.

The results of this simulation study are displayed in Figure 5.3 and show that there is little difference between the three models in any of the selected criteria. Models 2 and 3 have slightly better RMSE scores than Model 1, with median values of 0.0596 and 0.595 compared to 0.662. However, Model 1 is slightly more successful than the other two models in terms of identifying the correct number of clusters, with a median cluster number of 21 compared to 22 for both models 2 and 3. Model 1 also appears to perform better in terms of the maximum Rand Index, with a median score of 0.988 compared

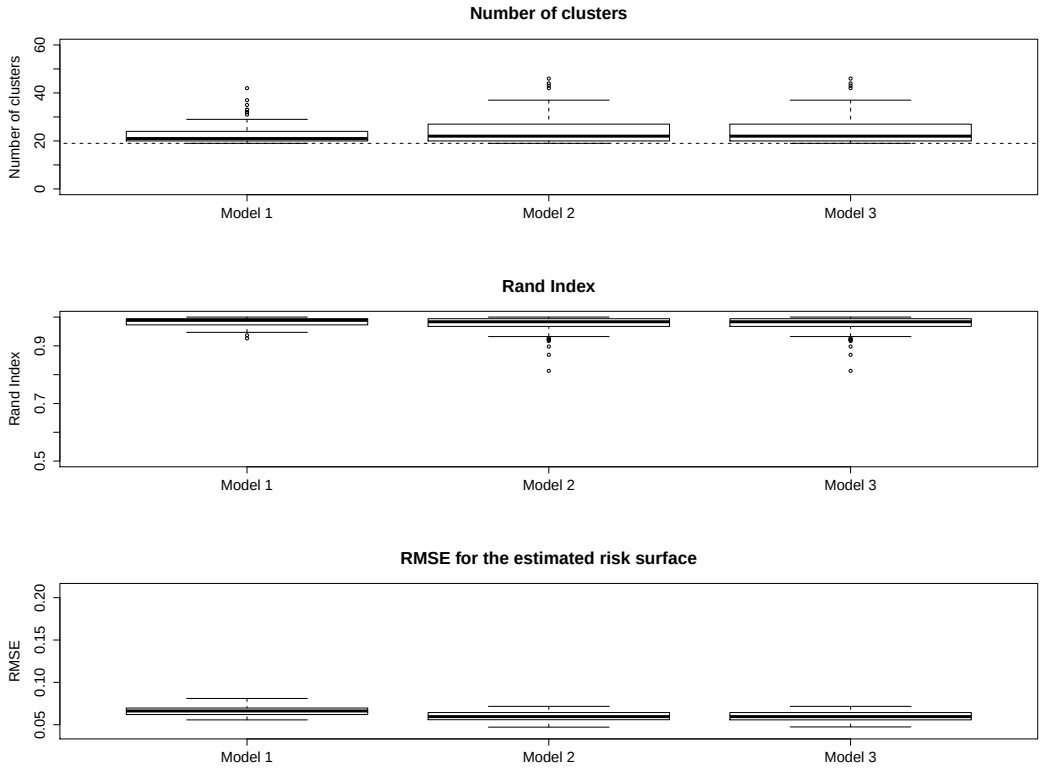

Figure 5.3: Summary of the simulation study results from changing the hyperparameters of the Gamma prior distribution for τ. The top panel displays boxplots of the chosen number of clusters under each model, the middle panel displays boxplots of the maximum Rand Index obtained under each model and the bottom panel displays the RMSE for each model.

to 0.983 for both models 2 and 3. In addition, the boxplots of the maximum Rand Index scores show that models 2 and 3 have a longer tail than model 1, meaning that there is slightly more variability in the Rand Index scores for models 2 and 3. All observed differences between the three models are small, and it does not appear that changing the hyperparameters of the Gamma

distribution for τ has any substantial effect on the ability of the model to identify the correct cluster structure or estimate the disease risk for each area.

5.4.2 Sensitivity to disease prevalence

The Greater Glasgow respiratory disease data that motivated the methodology presented here has expected counts E between 49 and 180, with a median value of 92. These counts are displayed in a histogram in Figure 5.4. To assess the impact of the prevalence of the disease on model performance, we apply both the model proposed here and the BYM model to disease data where the expected counts E are drawn as uniform random variables in the intervals: (A) - [10; 25], (B) - [50; 100] and (C) - [150; 250]. The simulated data are generated as described in Section 5.3, and as before we only consider the most challenging case of $C = 0.5$ here. The results of this analysis are displayed in Figure 5.5 below. The format of the figure is the same as that of Figure 5.2, and compares scenarios A (left), B (middle) and C (right). The results from the figure are based on 100 simulated data sets.

The top panel shows the number of clusters estimated by each model under each of the three scenarios, with the dashed line indicating the true number of clusters (19). For scenario A, the median number of clusters is 14.0 for the BYM model compared with 32.5 for our model, and for scenario B, the median number is 21.0 for the BYM and 22.0 for our model. However for scenario C our model performs better, with a median of 20.0 compared to

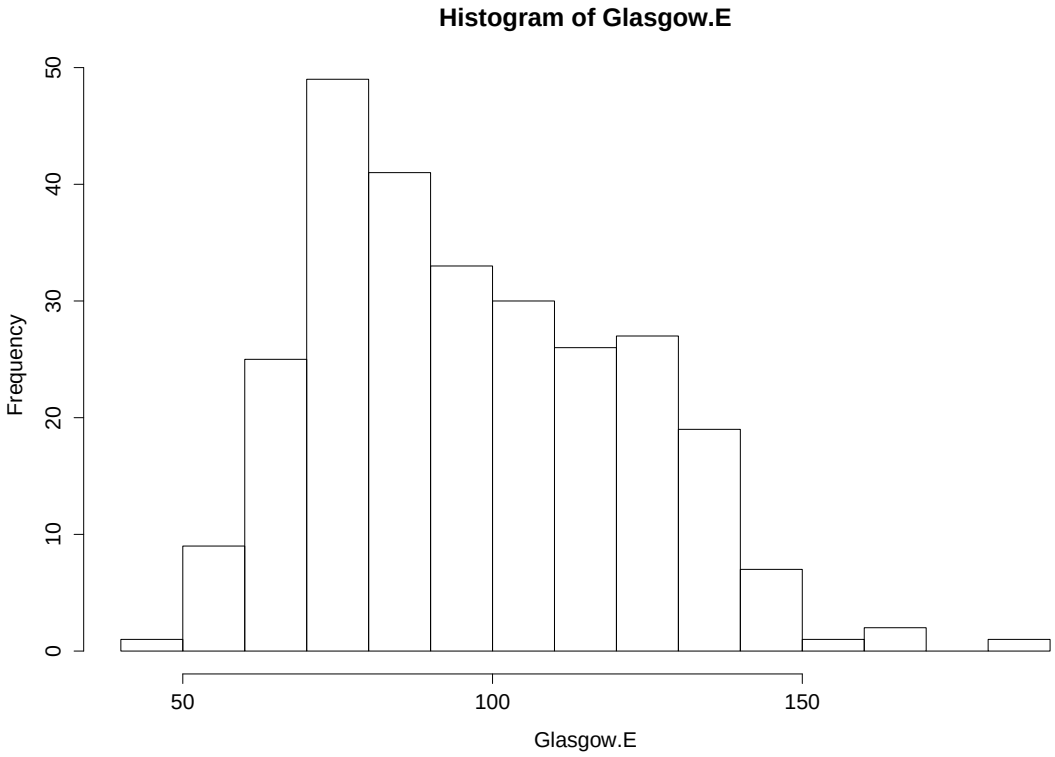

Figure 5.4: Histogram of the expected number of respiratory disease cases for Intermediate Geographies in Glasgow in 2011.

21.0 for the BYM. More crucially, our model performs better under all three scenarios in terms of the Rand Index. Under scenario A, our proposed model has a median Rand Index of 0.84 compared to 0.70 for the BYM. For scenario B, our model has a median Rand value of 0.98 compared to 0.92 for the BYM, and for scenario C our model has a median value of 0.99 compared to 0.98 for the BYM. These results suggest that while the BYM model is able to get closer to the correct number of clusters, the clusters it estimates are less accurate than those of our model in all three cases. Our model also performs better under RMSE in all three cases; under scenario A the median RMSE

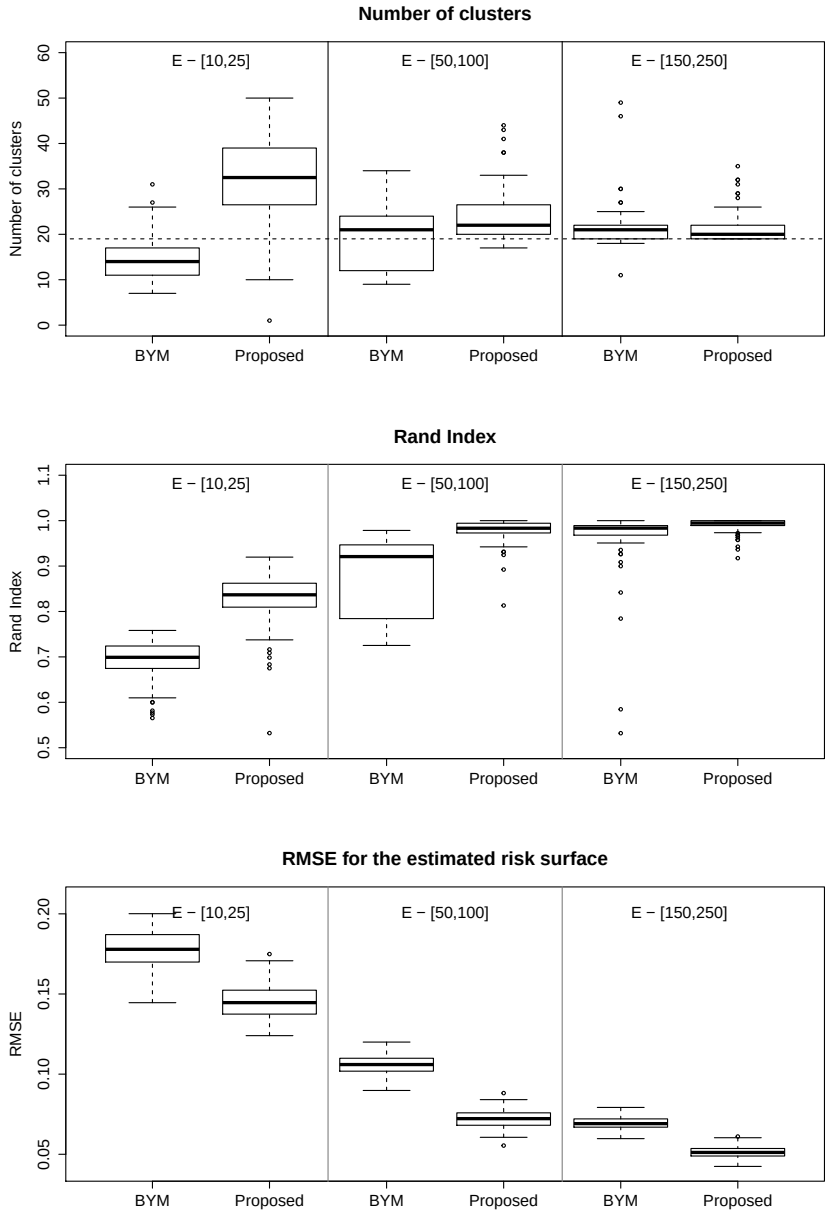

Figure 5.5: Summary of the simulation study results from changing $\mathbf{E}$. The top, middle and bottom panels display boxplots of the estimated number of clusters, the Rand Index and the root mean square error of the estimated risk surface for the BYM model and the model proposed here. The results relate to $\mathbf{E} = [10, 25]$ (left), $\mathbf{E} = [50, 100]$ (middle) and $\mathbf{E} = [150, 250]$ (right). In the top panel the dashed lines represent the true number of clusters.

is 0.14 for our model and 0.18 for the BYM, under scenario B the median is 0.07 for our model and 0.11 for the BYM, and under scenario C the median is 0.05 for our model and 0.07 for the BYM.

Both models perform better under all three criteria when the expected disease counts are higher. This occurs because $\mathbf{Y} = \mathbf{E}\exp(\boldsymbol{\phi})$, so that a fixed difference in risk (as measured by $\exp(\boldsymbol{\phi})$) between two neighbouring areas is made more prominent in terms of the size of the difference in $\mathbf{Y}$ by multiplying it by larger values of $\mathbf{E}$. As a result, disease clusters are easier to identify for more prevalent diseases with larger values of $\mathbf{E}$. The model proposed here performs well for scenarios B and C, while it performs rather less well for small values of E in the region [10; 25]. However, the BYM model performs even worse in this case, except for estimating the correct number of clusters.

5.4.3 Comparison with a cluster only model

This simulation study compares our combined fixed effect and CAR model with an alternative model which uses only a fixed effect and assumes a constant risk within each cluster. One hundred datasets of clustered disease data were simulated as described in Section 5.3 and centroid linkage methods were applied to each dataset to produce a set of candidate clusterings. As before only $C = 0.5$ is considered here. Two models were applied to the data; Model 1, which combines a fixed effect term with a CAR model and Model 2, which uses only a fixed effect term. For each model, the accuracy of

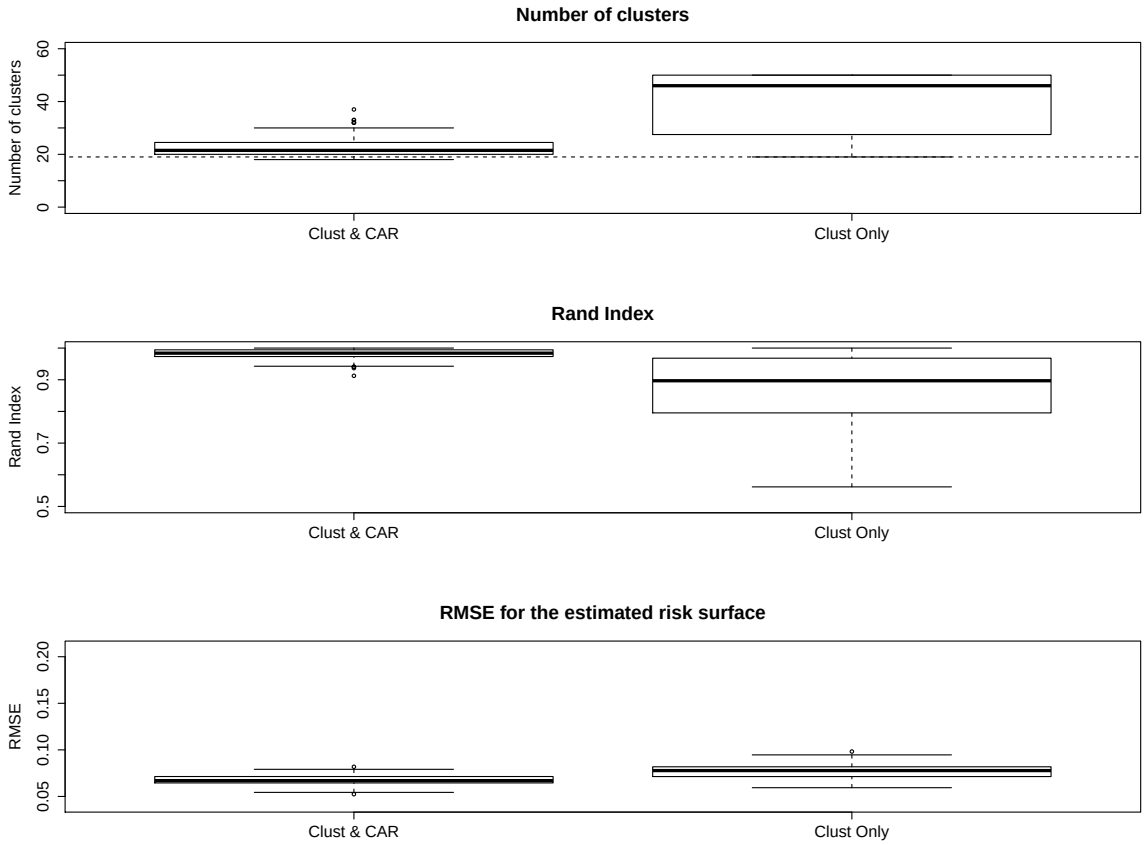

Figure 5.6: Summary of the simulation study results for the cluster plus CAR model and the cluster only model. The top panel displays boxplots of the chosen number of clusters under each model, the middle panel displays boxplots of the maximum Rand Index obtained under each model and the bottom panel displays the RMSE for each model.

the risk surfaces estimated is quantified by root mean square error (RMSE), while the correctness of the estimated cluster structures is quantified by both the number of clusters identified and the Rand Index.

The results of the study are shown in Figure 5.6, with our combined model (Model 1) performing better in each of the three criteria. The median number of clusters identified by Model 1 is 21.5 compared with 46 for model 2. Model 1 also has a higher maximum Rand Index than Model 2, with a median of 0.983 compared to 0.897, and a much smaller standard deviation (0.017 compared to 0.112). In addition, Model 1 has a lower RMSE than Model 2 (0.067 compared with 0.078). These results are likely to be a result of the cluster only model not allowing for smoothing within a cluster, which means that it can be adversely affected by within cluster sampling variation in the disease counts Y_i. Our model is able to smooth over this random noise within a cluster, while the cluster only model is unable to do so is therefore more likely to overestimate the number of clusters on account of small differences between areas within a true cluster. This lack of smoothing also makes the estimation less accurate for the cluster only model because the requirement for constant risk within a cluster does not reflect the true pattern of disease risk. Based on these results, it is clear that the combination of a fixed effect and CAR model performs better than simply having a fixed effect with a constant risk within each cluster.

5.5 Application to real data

This section continues the analysis of the respiratory hospitalisation risk data presented in Chapter 4.

5.5.1 Study design

As in the previous chapter, the study region is the Greater Glasgow and Clyde Health Board area, and we use the respiratory admission data introduced in Section 4.5 Figure 4.4 contains a map of Glasgow and the surrounding areas, with pins in the map to identify each location which is mentioned in this thesis. Table 4.2 provides a key for this map, with the numbers in the table corresponding to those in the pins in Figure 4.4.

The response data, $\boldsymbol{Y} = (Y_1, \ldots, Y_n)$, are based on the 2011 data, where Y_i is the number of hospital admissions with a primary diagnosis of respiratory disease in areal unit i in 2011. The expected values, $\boldsymbol{E} = (E_1, \ldots, E_n)$, are the expected hospital admission numbers for each areal unit in 2011. The top panel of Figure 5.8 displays the Standardised Incidence Ratio (SIR) for respiratory hospital admission, which is the ratio of the observed to the expected numbers of cases. The figure shows that there are regions of high risk in the east of the city and directly south of the river, which contain the heavily deprived neighbourhoods of Easterhouse and Govan. In contrast, areas in the centre (just north of the river) and far south of the study region exhibit much lower risks, which are the affluent West End and Giffnock districts of the city. In addition, there are a number of areas where a discontinuity in disease risk appears to exist.

5.5.2 Results

The two-stage clustering model proposed in Section 5.2 was applied to these data, where the prior elicitation step was based on respiratory disease data from 2008 to 2010. The fitted risk surfaces for these data sets exhibit similar spatial patterns to the 2011 study data, with Pearson's correlation coefficients of 0.86 (2010 data), 0.84 (2009 data) and 0.82 (2008 data) respectively. Model (5.1) was applied to the data with between 1 and 100 clusters, and Figure 5.7 shows the DIC values for these models. The model with 33 clusters minimises the DIC, while only models with between 32 and 38 clusters are within 4 of this minimum DIC value.

The estimated risk surface (grey-scale) and cluster structure (white dots) are displayed in the bottom panel of Figure 5.8, which has the same scale as the SIR plot in the top panel of that figure. The majority of the clusters identified appear to exhibit different risks compared with neighbouring areal units, although there are a small number of exceptions such as the small singleton cluster to the far west of the study region. The likely reason for this is the slight overestimation of the number of clusters as illustrated by the simulation study in Figure 5.2 (top panel with $C = 0.5$), a problem that is shared by the posterior classification approach based on the BYM model.

Three prominent features of the risk map are highlighted A, B and C. Cluster A is the low-risk West End of Glasgow, which is one of the most affluent parts of the city. The large high-risk cluster denoted by B contains a number

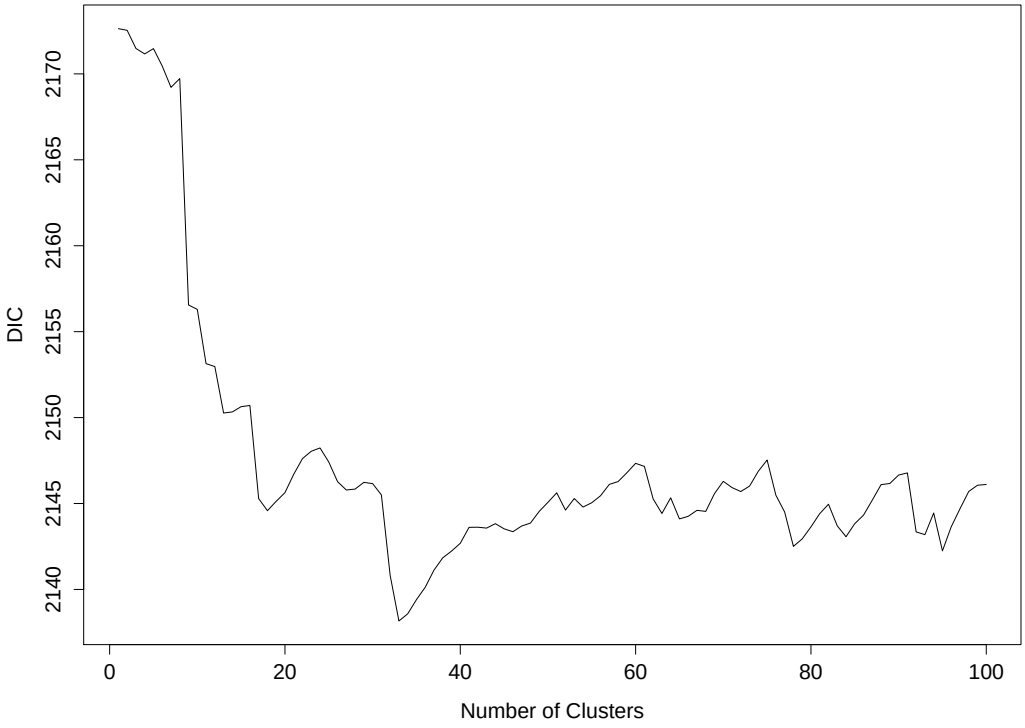

Figure 5.7: Plot of the Deviance Information Criterion for models with between 1 and 100 clusters.

of the most deprived neighbourhoods of Glasgow, including Easterhouse in the east and Springburn and Summerston in the North. Finally, cluster C is the deprived suburb of Drumchapel, which exhibits elevated risks compared with the affluent Bearsden area to the north east. The main driver of these cluster configurations is socio-economic deprivation, which is well known to have a large effect on population health. The high-risk areas in Figure 5.8 typically exhibit high levels of socio-economic deprivation, where as low-risk areas are more affluent. One could of course include a covariate measuring deprivation in the regression model to account for this, but while it would

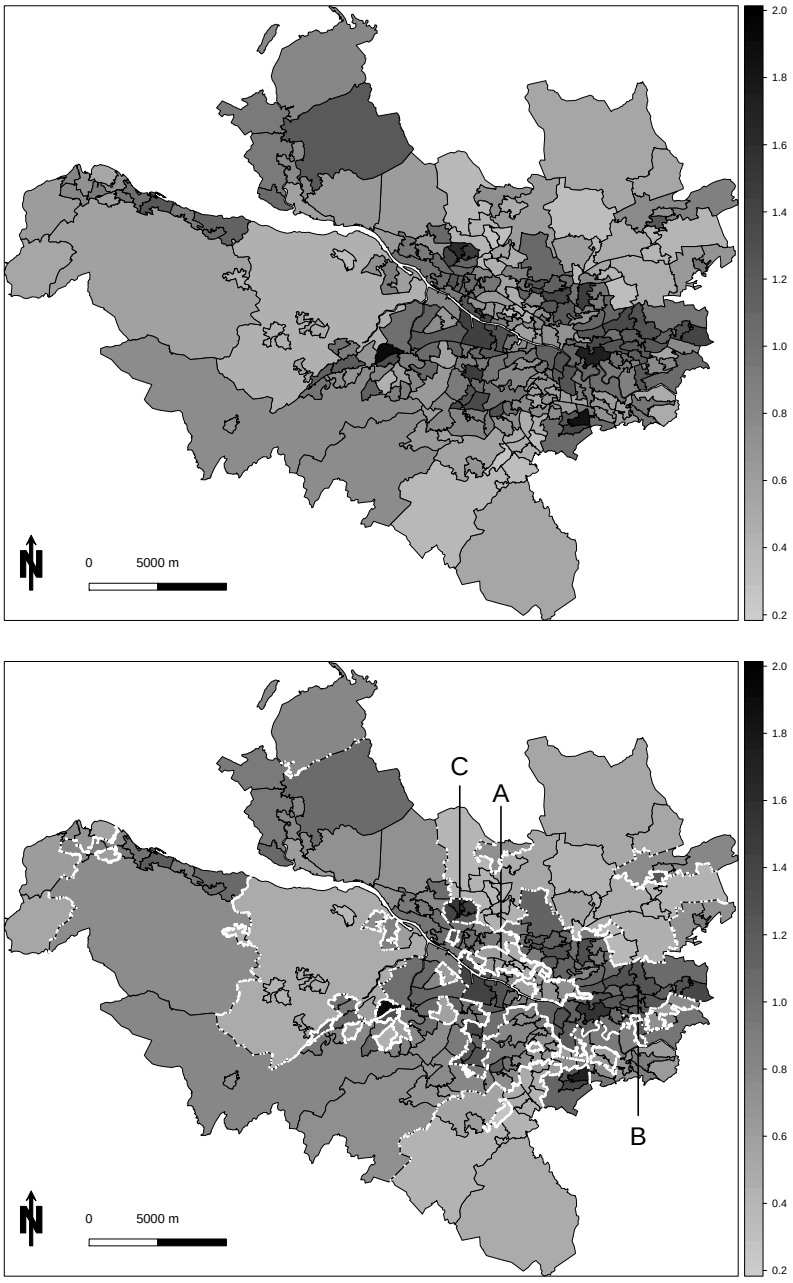

Figure 5.8: The top panel displays the standardised incidence ratio (grey-scale) for respiratory disease hospitalisation in 2011 in Greater Glasgow. The bottom plot displays the estimated risk surface (grey-scale) from the model with 33 clusters (white dots). The labels A, B and C represent prominent clusters that have been identified.

explain the spatial pattern in respiratory disease risk it would not be able to identify the spatial extent of the high-risk clusters.

5.6 Discussion

The aim of this modelling approach is to identify discontinuities in the spatial pattern of disease risk, which corresponds to the identification of clusters exhibiting both elevated and reduced risks. The methodology we have developed is a two-stage approach, which is a fusion of the spatially-adapted hierarchical agglomerative clustering techniques described in Chapter 4 with conditional autoregressive models described in this chapter. In Stage 1, a set of candidate cluster structures for the study data are obtained by applying the clustering approach introduced in Chapter 4 to data quantifying disease risk prior to the study period. Then, in Stage 2, separate spatial random effects models are applied to the study data for each candidate cluster structure, and the choice of the best cluster structure is treated as a model comparison problem. The Bayesian hierarchical models fitted in the second stage represent disease risk with a linear combination of a spatially smooth intrinsic CAR model and a piecewise constant cluster model, which allows disease risk to evolve smoothly within a cluster with a disjoint multiplicative jump between clusters. Removing the CAR component of the model would assume a constant disease risk within a cluster, which is unlikely to be true in general, and which the simulation study in Section 5.4.3 showed resulted in poorer estimation.

The model comparison approach adopted here does not estimate the cluster structure simultaneously with the risk surface as is done by Knorr-Held and Rasser (2000), which ignores the uncertainty about the number of clusters in the estimation procedure. However, this two-stage approach is easy to implement and makes the identification of the 'final' cluster structure straightforward, which is not always the case for approaches such as Knorr-Held and Rasser (2000) which may produce a different cluster structure for each McMC iteration.

The simulation studies presented in Sections 5.3 and 5.4 showed that our model generally performs well, in particular outperforming the BYM model with a posterior classification step. Improved performance was observed for both risk estimation and cluster identification, which is most likely to be because our approach attempts to estimate the cluster structure in the data. In contrast, the posterior classification approach estimates a smooth risk surface using the BYM model, and attempts to identify clusters from that smoothed surface. This approach is inherently flawed because it is attempting to find non-smooth patterns in data which has been smoothed. The studies we have conducted also suggest that our method performs well for diseases with moderate to large numbers of cases, but that when the number of cases in each areal units less than 25 it, like other methods, is likely to be less accurate at identifying the correct cluster structure.

There is scope to extend this method in two main ways. The approach proposed here models spatial discontinuities (clusters) in risk via the mean

function using a piecewise constant fixed effect, which contrasts with the majority of the open boundary literature which achieves this by modelling the correlation structure in the random effects (see Lu et al. (2007) and Lee and Mitchell (2012)). Adapting Stage 2 of the approach proposed here to identify clusters via the correlation structure of the random effects is a natural extension, and this will be explored in Chapter 6. The second avenue for developing this approach is to extend these methods into the spatio-temporal domain, thus allowing policy makers to identify whether a health intervention has had an effect in reducing disease risk in a high risk cluster; this will be explored in Chapter 7.

Chapter 6

Identifying spatial clusters using a variance (random effects) based approach.

6.1 Introduction

In Chapter 5, a method for identifying spatial clusters via a mean-based approach was introduced. This approach consisted of a two-stage model, where the first stage was the spatial clustering method introduced in Chapter 4 and the second stage was a Bayesian model selection algorithm. The latter combined the smooth intrinsic CAR model with a piecewise constant cluster model, which was iteratively fitted with between 1 and n clusters. This approach accounted for spatial discontinuities in the risk surface by assigning different mean risk levels to each cluster via a fixed effect term. In this

chapter, an alternative Bayesian model for Stage 2 is introduced; this model accounts for the discontinuities by modelling the correlation structure in the random effects rather than introducing mean level fixed effects.

As in Chapter 5, this methodology allows for the estimation of the spatial pattern in disease risk, whilst simultaneously detecting the spatial extent of high or low risk clusters. In doing so, the cluster structure is accounted for when estimating disease risk, so that high risk clusters are not smoothed towards their geographical neighbours that do not exhibit elevated risks. The methodology proposed here follows the same basic approach as that introduced in the previous chapter. In Stage 1, the spatial hierarchical agglomerative clustering algorithm is applied to disease data preceding the study period to elicit n candidate cluster configurations containing between 1 and n clusters, and then in Stage 2 the optimal structure is selected and disease risk is estimated. In the previous chapter, the optimal cluster structure was chosen via a model comparison procedure, but the approach introduced in this chapter consists of a single model with the optimal cluster structure estimated as a parameter within that model. The advantage of this is that the uncertainty in the cluster structure can be quantified, and propagated through the disease risk model. This Bayesian model is an extension of the Poisson log-linear model originally developed by Lee et al. (2014). The other major difference between this approach and that introduced in Chapter 5 is that here Markov chain Monte Carlo (McMC) simulation methods are used to estimate both the optimal cluster structure and disease risk, whereas INLA was used for parameter estimation in Chapter 5.

The remainder of this chapter is organised as follows. Section 6.2 outlines the random effect model proposed here and how it combines with the clustering algorithm in Chapter 4 to form the overall model. Section 6.3 uses simulated data to test this method against an existing method, and also against the method proposed in Chapter 5. Section 6.4 outlines an application of this methodology, based on respiratory hospital admissions in the Greater Glasgow area in 2011. Finally, section 6.5 discusses the advantages of this modelling approach compared to existing methods, including that introduced in Chapter 5.

6.2 Methodology

6.2.1 Proposed model

We propose a two-stage approach for estimating the spatial pattern in disease risk and identifying high or low risk clusters. The first stage uses the clustering algorithm described in Chapter 4 to produce a set of candidate cluster structures $\{\mathcal{C}_1, \ldots, \mathcal{C}_n\}$, with each containing a different number of clusters between 1 and n. In the second stage we propose a hierarchical Bayesian model for the disease data, which can simultaneously select the optimal cluster configuration from the candidates elicited in Stage 1 and also estimate disease risk.

The best cluster structures for these data from the set of n candidates $\{\mathcal{C}_1, \ldots, \mathcal{C}_n\}$ elicited from Stage 1 are estimated together with disease risk, by extending the Poisson log-linear model in Section 2.2.1. This approach takes advantage of the natural ordering of the cluster structures, by considering the number of clusters as a univariate parameter within the model. The mechanism for implementing a given cluster structure is the neighbourhood matrix W, which is altered so that $w_{ij} = 1$ if areal units $(\mathcal{A}_i, \mathcal{A}_j)$ share a border and are in the same cluster, and $w_{ij} = 0$ otherwise. If two adjacent areal units are in the same cluster (ie $w_{ij} = 1$), their random effects are partially correlated and their values are smoothed towards each other, while if they are in different clusters (ie $w_{ij} = 0$), they are conditionally independent and are not smoothed over. There is a one-to-one relationship between a given cluster structure and the value of W, and the n candidate values of W are denoted by $(W_1, \ldots, W_n)$. Here W_1 corresponds to a single cluster and thus equals W, the original adjacency structure of the region. This value enforces global spatial smoothing across the region, as no high or low risk clusters have been identified. In contrast, W_n corresponds to all n areal units being assigned to their own cluster of size one, and thus W_n is the zero matrix. This value thus corresponds to independent random effects with no spatial smoothing constraints.

The intrinsic CAR prior given by $\phi_i|\boldsymbol{\phi}_{-i} \sim \mathrm{N}\left(\frac{\sum_{j=1}^n w_{ij}\phi_j}{\sum_{j=1}^n w_{ij}}, \frac{1}{\tau(\sum_{j=1}^n w_{ij})}\right)$ is not appropriate here, since our model could produce a neighbourhood matrix W in which an areal unit has no neighbours due to it being a singleton cluster. If this was areal unit i, this would cause $\sum_{j=1}^n w_{ij} = 0$, yielding an infinite

mean and variance in the above CAR prior. Instead, we use the localised CAR model outlined in Lee et al. (2014), where an extended random effects vector $\widetilde{\boldsymbol{\phi}} = (\boldsymbol{\phi}, \phi^*)$ is used, with ϕ^* being a global random effect which is potentially common to all areas and prevents the infinite mean and variance problem outlined above. An extended $(n+1)\times(n+1)$ neighbourhood matrix $\widetilde{W}$ is specified for this vector $\widetilde{\boldsymbol{\phi}}$, which takes the form

$$\widetilde{W} = \begin{pmatrix} W & \boldsymbol{w}_* \\ \boldsymbol{w}_*^T & 0 \end{pmatrix},$$

where $\boldsymbol{w}_* = (w_1*, \ldots, w_n*)$ and $w_i* = I[\sum_{i\sim j}(1 - w_{ij}) > 0]$. Here, $I[.]$ denotes an indicator function, which sets $w_i* = 1$ if any entry in row i of the neighbourhood matrix W is changed from a 1 to a 0 due to a neighbouring area being in a different cluster. Otherwise, $w_i^* = 0$. Based on this extended neighbourhood matrix, $\widetilde{\boldsymbol{\phi}}$ is modelled as $\widetilde{\boldsymbol{\phi}} = \mathrm{N}(\mathbf{0}, \frac{Q(\widetilde{W},\epsilon)^{-1}}{\tau})$ with the precision matrix

$$Q(\widetilde{W}, \epsilon) = \mathrm{diag}(\widetilde{W}1) - \widetilde{W} + \epsilon I. \tag{6.1}$$

This corresponds to the intrinsic CAR model for the extended random effects vector $\widetilde{\boldsymbol{\phi}}$, with a small positive constant added to the diagonal of the matrix to ensure that it is invertible. The invertibility of $Q(\widetilde{W}, \epsilon)$ is required as its determinant is computed when updating W, and Lee et al. (2014) suggest that the results are insensitive to small values of ϵ and set $\epsilon = 0.001$, so we also choose this value for ϵ. The full conditionals of this extended CAR model are given by

$$\begin{aligned}
\phi_i|\widetilde{\boldsymbol{\phi}}_{-i} &\sim \mathrm{N}\left(\frac{\sum_{j=1}^n w_{ij}\phi_j + w_{i*}\phi_*}{\sum_{j=1}^n w_{ij} + w_{i*} + \epsilon}, \frac{1}{\tau(\sum_{j=1}^n w_{ij} + w_{i*} + \epsilon)}\right), \quad (6.2)\\
\phi_*|\widetilde{\boldsymbol{\phi}}_{-*} &\sim \mathrm{N}\left(\frac{\sum_{j=1}^n w_{j*}\phi_j}{\sum_{j=1}^n w_{j*} + \epsilon}, \frac{1}{\tau(\sum_{j=1}^n w_{j*} + \epsilon)}\right).
\end{aligned}$$

This means that the conditional expectation for an area is a weighted average of the random effects in neighbouring areas and the global random effect ϕ^*, with binary weights based on the current choice of W matrix. Here, $(\widetilde{W}_1, \ldots, \widetilde{W}_n)$ is the set of extended neighbourhood matrices corresponding to $(W_1, \ldots, W_n)$, so that $\widetilde{W}_j$ is the matrix corresponding to the cluster structure with j clusters. Given this extended CAR prior, the overall Bayesian hierarchical model we propose is given by

$$\begin{aligned}
Y_i|E_i, R_i &\sim \text{Poisson}(E_i R_i) \text{ for } i = 1, \ldots, n,\\
\ln(R_i) &= \beta_0 + \phi_i,\\
\widetilde{\boldsymbol{\phi}} &\sim N\left(\mathbf{0}, \frac{Q(\widetilde{W}, \epsilon)^{-1}}{\tau}\right),\\
\widetilde{W} &\sim \text{Discrete}(\widetilde{W}_1, \ldots, \widetilde{W}_n; \pi_1, \ldots, \pi_n),\\
\pi_j &= \frac{\exp(-j\theta)}{\sum_{i=1}^n \exp(-i\theta)}, \quad (6.3)\\
\beta_0 &\sim N(0, 1000),\\
\theta &\sim \text{Uniform}(0, 1),\\
\tau &\sim \text{Gamma}(0.001, 0.001).
\end{aligned}$$

A gamma prior with small shape and scale parameter values is specified for τ in an attempt to be non-informative about its value. However, we assess the

sensitivity of this choice in Section 6.4.3 by comparing it to other Gamma specifications. Initially, a discrete uniform prior was considered for $\widetilde{W}$, but it may not be appropriate to give equal prior probability to cluster structures with extremely large numbers of clusters, as the spatial autocorrelation present in the data suggests the number of clusters will be relatively small. Therefore our prior probabilities for $(\widetilde{W_1}, \ldots, \widetilde{W_n})$ are given by $(\pi_1, \ldots, \pi_n)$ where $\pi_j = \frac{\exp(-j\theta)}{\sum_{i=1}^{n} \exp(-i\theta)}$, with an additional parameter θ being introduced to control the relative sizes of $(\pi_1, \ldots, \pi_n)$. When $\theta = 0$, $\widetilde{W}$ has a discrete uniform prior, while $\theta = 1$ corresponds to a scaled exponential weighting which gives larger prior weight to values of W corresponding to fewer clusters.

The estimated number of clusters in the data could be represented by a central value from the posterior distribution of $\widetilde{W}$, with uncertainty estimated via a 95% credible interval. Here we will select the number of clusters using the posterior mode, because it is the most commonly occurring cluster structure in the McMC algorithm, but the median could also be used. The mean would not be sensible because the number of clusters follows a discrete distribution and requires an integer value. The posterior median is used to give a point estimate for each of the other model parameters.

6.2.2 Inference via McMC

Inference for this model is carried out using a McMC algorithm, using a combination of Gibbs sampling and Metropolis-Hastings steps. The algorithm produces posterior distributions for each of the model parameters, and the

full conditionals for the parameters of this McMC algorithm are as follows:

β_0 - intercept term

The full conditional for β_0 is as follows:

$$\begin{aligned} f(\beta_0|\boldsymbol{Y}) &\propto \prod_{i=1}^{n} \text{Poisson}(Y_i|\beta_0) \times \text{N}(\beta_0|0, 1000) \\ &\propto \prod_{i=1}^{n} \exp\Big(- E_i \exp(\beta_0 + \phi_i)\Big)\Big(\exp(\beta_0 + \phi_i)\Big)^{Y_i} \times \exp\left(-\frac{\beta_0^2}{2000}\right) \end{aligned}$$

A Metropolis algorithm is used to update this parameter, with a proposal β_0^* drawn from the distribution $\beta_0^* \sim \text{N}(\beta_0^{(i)}, v_\beta)$, where $\beta_0^{(i)}$ is the current state of the chain. The acceptance probability of a move from $\beta_0^{(i)}$ to β_0^* is given by $\min\left(1, \frac{f(\beta_0^*|\boldsymbol{Y})}{f(\beta_0^{(i)}|\boldsymbol{Y})}\right)$. The proposal variance v_β can be altered within the algorithm to maintain an acceptance rate between 40% and 80%.

$\boldsymbol{\phi}$ - random effects

The full conditional for $\boldsymbol{\phi}$ is as follows:

$$\begin{aligned} f(\boldsymbol{\phi}|\boldsymbol{Y}) &\propto \prod_{i=1}^{n} \text{Poisson}(Y_i|\phi_i) \times \text{N}\left(\widetilde{\boldsymbol{\phi}}|\mathbf{0}, \frac{Q(\widetilde{W}, \epsilon)^{-1}}{\tau}\right) \\ &\propto \prod_{i=1}^{n} \Big(E_i \exp(\beta_0 + \phi_i)\Big)^{Y_i} \exp\Big(- E_i \exp(\beta_0 + \phi_i)\Big) \times \\ &\quad \exp\left(-\frac{1}{2}\tau(\boldsymbol{\phi}^T Q(\widetilde{W}, \epsilon)\boldsymbol{\phi})\right) \end{aligned}$$

Updating the entire vector, $\boldsymbol{\phi}$, in a single step would lead to a low acceptance probability, while updating each ϕ_j individually would be computationally intensive, so in this case we adopt an intermediate strategy of updating in blocks of size b. Each block $\boldsymbol{\phi_{rs}} = (\phi_r, \ldots, \phi_s)$ was updated in turn, conditional on $\boldsymbol{\phi_{-rs}} = (\phi_1, \ldots, \phi_{r-1}, \phi_{s+1}, \ldots, \phi_n)$. In order to carry out this step, $\boldsymbol{\phi}$ is partitioned as follows:

$$\boldsymbol{\phi} = \begin{pmatrix} \boldsymbol{\phi_{rs}} \\ \boldsymbol{\phi_{-rs}} \end{pmatrix} \sim \mathrm{N}\left[\begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix}, \tau^{-1} \begin{pmatrix} Q_{rs,rs} & Q_{rs,-rs} \\ Q_{-rs,rs} & Q_{-rs,-rs} \end{pmatrix} \right].$$

Under this notation, $f(\boldsymbol{\phi_{rs}}|\boldsymbol{\phi_{-rs}}) \sim \mathrm{N}(\bar{\boldsymbol{\phi}}_{\boldsymbol{rs}}, \Sigma_{rs})$, where $\bar{\boldsymbol{\phi}}_{\boldsymbol{rs}} = -Q_{rs,rs}^{-1} Q_{rs,-rs} \boldsymbol{\phi_{-rs}}$ and $\Sigma_{rs} = \frac{Q_{rs,rs}^{-1}}{\tau}$. The full conditional distribution is as follows:

$$\begin{aligned} f(\boldsymbol{\phi_{rs}}|\boldsymbol{\phi_{-rs}}, \boldsymbol{Y}) &\propto \prod_{i=r}^{s} \mathrm{Poisson}(Y_i|\phi_i) \times \mathrm{N}(\boldsymbol{\phi_{rs}}|\bar{\boldsymbol{\phi}}_{\boldsymbol{rs}}, \Sigma_{rs}) \\ &\propto \prod_{i=r}^{s} \Big(E_i \exp(\beta_0 + \phi_i) \Big)^{Y_i} \exp\Big(- E_i \exp(\beta_0 + \phi_i) \Big) \times \\ &\quad \exp\left(-\frac{1}{2} (\boldsymbol{\phi_{rs}} - \bar{\boldsymbol{\phi}}_{\boldsymbol{rs}})^T \Sigma_{rs}^{-1} (\boldsymbol{\phi_{rs}} - \bar{\boldsymbol{\phi}}_{\boldsymbol{rs}}) \right) \end{aligned}$$

A Metropolis algorithm is used to update the blocks, with a proposal $\boldsymbol{\phi}_{\boldsymbol{rs}}^*$ drawn from the distribution $\boldsymbol{\phi}_{\boldsymbol{rs}}^* \sim \mathrm{N}(\boldsymbol{\phi_{rs}}^{(i)}, \Sigma_{rs})$. The acceptance probability of a move from $\boldsymbol{\phi_{rs}}^{(i)}$ to $\boldsymbol{\phi}_{\boldsymbol{rs}}^*$ is given by $\min\left(1, \frac{f(\boldsymbol{\phi}_{\boldsymbol{rs}}^*|\boldsymbol{\phi_{-rs}}, \boldsymbol{Y})}{f(\boldsymbol{\phi_{rs}}^{(i)}|\boldsymbol{\phi_{-rs}}, \boldsymbol{Y})}\right)$.

ϕ_* - global random effect

The additional random effect ϕ_* is entirely conditional on the values of the vector $\boldsymbol{\phi}$ and the current W matrix. It is evaluated at each iteration of the McMC algorithm by Gibbs sampling from the following distribution:

$$\phi_*|\widetilde{\boldsymbol{\phi}}_{-*} \sim \mathrm{N}\left(\frac{\sum_{j=1}^{n} w_{j*}\phi_j}{\sum_{j=1}^{n} w_{j*}+\epsilon}, \frac{1}{\tau(\sum_{j=1}^{n} w_{j*}+\epsilon)}\right).$$

$\widetilde{W}$ - neighbourhood matrix term

The full conditional for $\widetilde{W}$ is as follows:

$$\begin{aligned} f(\widetilde{W}|\widetilde{\boldsymbol{\phi}}) &\propto \mathrm{N}\left(\widetilde{\boldsymbol{\phi}}|\mathbf{0}, \frac{Q(\widetilde{W},\epsilon)^{-1}}{\tau}\right) \times \mathrm{P}(\widetilde{W}=\widetilde{W}_j) \\ &\propto |Q(\widetilde{W},\epsilon)|^{\frac{1}{2}} \exp\left(-\frac{1}{2}\tau(\widetilde{\boldsymbol{\phi}}^T Q(\widetilde{W},\epsilon)\widetilde{\boldsymbol{\phi}})\right) \times \prod_{j=1}^{n}\left(\frac{\exp(-j\theta)}{\sum_{i=1}^{n}\exp(-i\theta)}\right)^{I(\widetilde{W}=\widetilde{W}_j)} \end{aligned}$$

The set of n potential $\widetilde{W}$ matrices are selected during Stage 1 of the model, and the matrices themselves remain unchanged during Stage 2 of the model. The Bayesian model in Stage 2 is used to select which of these predetermined matrices provides the best fit to the data, and therefore this stage of the McMC algorithm is used to update the choice of matrix. The set of $\widetilde{W}$ matrices, $(\widetilde{W}_1, \ldots, \widetilde{W}_n)$ have a natural ordering, which allows us to propose a $\widetilde{W}^*$ which is close to the current matrix $\widetilde{W}^{(i)}$ in the sequence. Here, we propose a maximum of s steps in either direction from the current matrix.

Where $\widetilde{W}_j^{(i)}$ is the current choice of $\widetilde{W}$ matrix, we propose from the set ($\widetilde{W}_{j-s}^{(i)}$, $\ldots$, $\widetilde{W}_{j-1}^{(i)}$, $\widetilde{W}_{j+1}^{(i)}$, $\ldots$, $\widetilde{W}_{j+s}^{(i)}$) with equal probability of selecting each matrix. Note that if our current value is close to an endpoint (ie either 1 or n) then some of these theoretical proposal matrices may not exist in practice. Therefore, if $j \leq s$ or $j > n - s$ then the number of proposal matrices is reduced, and the associated probabilities are adjusted accordingly. In our analysis, a value of $s = 2$ was chosen to maintain an acceptance rate between 40% and 80%.

The endpoint scenario discussed above means that the proposal distribution is not necessarily symmetric, and we must therefore use a Metropolis-Hastings algorithm to update the choice of $\widetilde{W}$ matrix. The acceptance probability of a move from $\widetilde{W}^{(i)}$ to $\widetilde{W}^*$ is given by $\min\left(1, \frac{f(\widetilde{W}^*|\widetilde{\boldsymbol{\phi}})P(\widetilde{W}^{(i)}|\widetilde{W}^*)}{f(\widetilde{W}^{(i)}|\widetilde{\boldsymbol{\phi}}P(\widetilde{W}^*|\widetilde{W}^{(i)}))}\right)$, where $P(\widetilde{W}^*|\widetilde{W}^{(i)})$ is the probability of proposing $\widetilde{W}^*$ given that the current state is $\widetilde{W}^{(i)}$.

θ - parameter for controlling the cluster selection weight

$$\begin{aligned} f(\theta|\boldsymbol{\pi}) &\propto f(\widetilde{W}|\pi_1, \ldots, \pi_n) \times \text{Uniform}(\theta|0,1) \\ &\propto \frac{\exp(-j\theta)}{\sum_{i=1}^{n}\exp(-i\theta)} \times I_{[\theta\in[0,1]]} \end{aligned}$$

A Metropolis algorithm is used to update this parameter, with a proposal θ^* drawn from the distribution $\theta^* \sim \text{N}(\theta^{(i)}, v_\theta)$, with the condition that $\theta^* \in (0,1)$. The acceptance probability of a move from $\theta^{(i)}$ to θ^* is given by

$\min\left(1, \frac{f(\theta^*|\boldsymbol{\pi})}{f(\theta^{(i)}|\boldsymbol{\pi})}\right)$. The proposal variance v_θ can be altered within the algorithm to maintain an acceptance rate between 40% and 80%.

τ - precision hyperparameter for random effects

$$\begin{aligned} f(\tau|\widetilde{\phi}) &\propto \mathrm{N}\left(\widetilde{\boldsymbol{\phi}}|\mathbf{0}, \frac{Q(\widetilde{W},\epsilon)^{-1}}{\tau}\right)\mathrm{Gamma}(\tau|0.001, 0.001) \\ &\propto |\tau|^{\frac{n+1}{2}}\exp\left(-\frac{1}{2}(\widetilde{\boldsymbol{\phi}}^T Q(\widetilde{W},\epsilon)\widetilde{\boldsymbol{\phi}})\tau\right) \times \tau^{0.001-1}\exp(-0.001\tau) \\ &\propto |\tau|^{-(\frac{n+1}{2}+0.001)-1}\exp\left(-(\frac{1}{2}\widetilde{\boldsymbol{\phi}}^T Q(\widetilde{W},\epsilon)\widetilde{\boldsymbol{\phi}} + 0.001)\tau\right) \\ &\sim \mathrm{Gamma}\left(\frac{n+1}{2} + 0.001, \frac{1}{2}\widetilde{\boldsymbol{\phi}}^T Q(\widetilde{W},\epsilon)\widetilde{\boldsymbol{\phi}} + 0.001\right) \end{aligned}$$

This full conditional distribution can be sampled from using Gibbs sampling, so a Metropolis step is not required in this case. To update τ we simply draw from the posterior Gamma distribution.

6.3 Simulation study

6.3.1 Aim

A simulation study was conducted to establish the efficacy of the two-stage modelling approach outlined in the previous section. The template for the study was the set of 271 Intermediate Geographies comprising the Greater Glasgow and Clyde Health Board, which is the study region for the motivating application presented in Section 6.4. A study was conducted comparing

the two-stage approach proposed here with that proposed in Chapter 5 and also an existing alternative, the BYM model as described in Section 2.4.2 with post-hoc clustering.

6.3.2 Data Generation

Clustered disease data were generated in the same way as described in Chapter 5. The simulated data consists of 19 clusters of different sizes, with risk data generated via the model outlined in Chapter 5 so that each cluster has one of three levels of disease risk. The size of the differences between these levels, and thus the extent of the differences between clusters, is controlled by multiplying the piecewise mean values by a value C prior to generating the simulated data. Larger values of C represent larger differences between the clusters, which should thus be easier to identify. Values of $C = 0, 0.5, 1$ are used in this study; $C = 1$ corresponds to a case where there are large differences between the clusters, $C = 0.5$ corresponds to a more difficult case where there are smaller differences and $C = 0$ corresponds to a spatially smooth risk surface where one would hope to identify a single cluster covering the entire study region. Examples of the data generated under each of these cases can be found in Figures 4.2 and 5.1. For the analyses described in this section the expected disease counts are set equal to those from the respiratory disease motivating application.

Five hundred datasets were generated for each of the three scenarios ($C = 0, 0.5, 1$), and the model proposed here was compared against two alterna-

tives. The first was the fixed effects model introduced in Chapter 5 and the second was the Besag-York-Mollié (BYM, Besag et al. (1991)) which is commonly used in disease mapping. In the case of the BYM model, the posterior classification approach described in Charras-Garrido et al. (2012) and Charras-Garrido et al. (2013) was implemented to identify the clusters, which is based on a Gaussian mixture model-based clustering approach. Further details of this approach are given in Section 2.6.3. However, this clustering approach does not produce spatially contiguous clusters, so a further post-processing step was implemented to partition the clusters identified into spatially contiguous groups. This is achieved by taking each cluster in turn and identifying sets of adjacent areal units within that cluster. In common with Chapter 5, we note that we have not compared our approach to a method such as Knorr-Held and Rasser (2000), because software to implement these complex estimation methods is not publicly available.

6.3.3 Results

The results of the study are outlined in Table 6.1 and summarised in Figure 6.1, which displays a comparison of the relative performances of our approach and the two alternatives using three different metrics. The accuracy of the risk surfaces estimated by each approach is quantified by their root mean square error (RMSE), while the correctness of the estimated cluster structures is quantified by both the number of clusters identified and the Rand Index between the true and estimated cluster structures. The latter is a measure of the similarity between two cluster structures and lies in the interval $[0, 1]$. It is computed as the proportion of pairs of areal units classified either

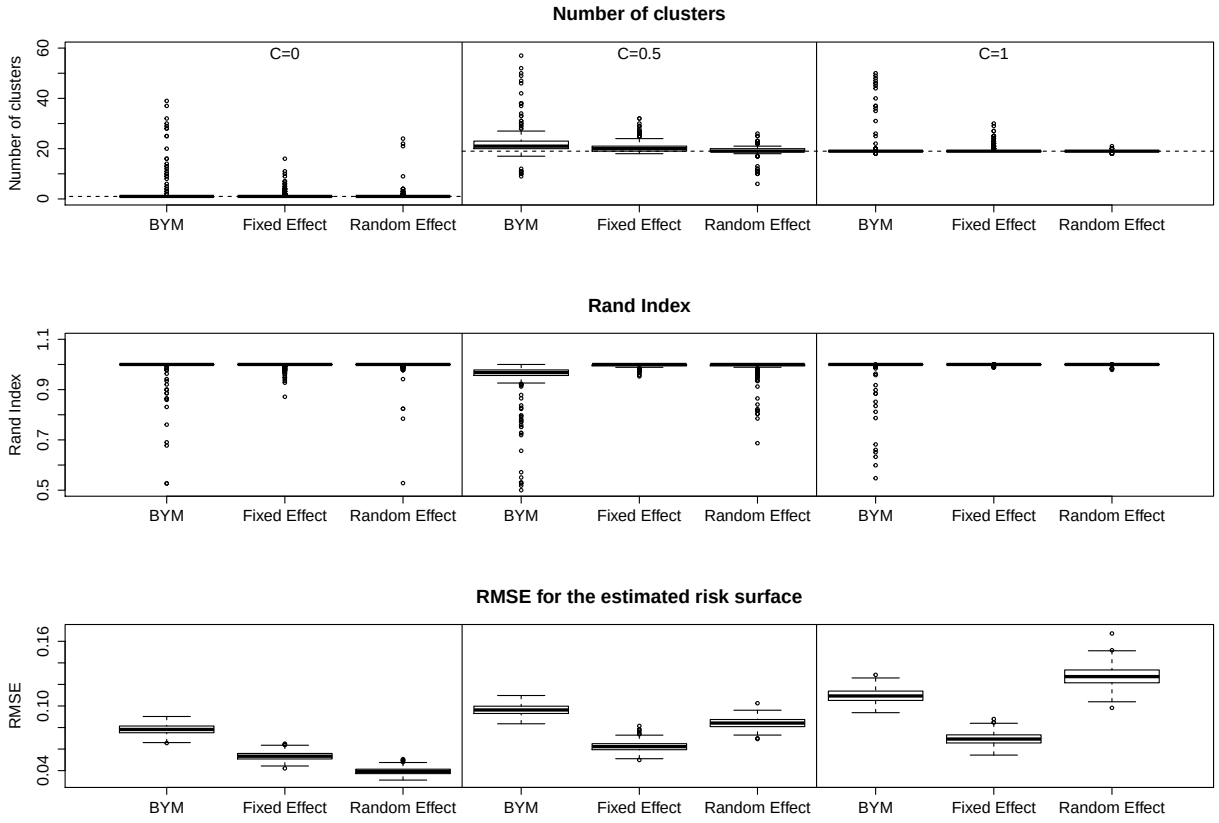

Figure 6.1: Summary of the simulation study results. The top, middle and bottom panels display boxplots of the estimated number of clusters, the Rand Index and the root mean square error of the estimated risk surface for each model in turn. The results relate to $C = 0$ (left panels), $C = 0.5$ (middle panels) and $C = 1$ (right panels). Within each panel, the boxplots relate to the BYM model (left), fixed effects model (middle) and our proposed random effects model (right). In the top panel the dashed lines represent the true number of clusters.

	Mean Difference	BYM	Fixed Effect	Random Effect
No of Clusters	$C = 0$	1 (4.35)	1 (1.31)	1 (6.74)
	$C = 0.5$	21 (4.50)	20 (2.31)	19 (1.68)
	$C = 1$	19 (3.98)	19 (1.36)	19 (0.18)
Rand Index	$C = 0$	1 (0.088)	1 (0.010)	1 (0.047)
	$C = 0.5$	0.9679 (0.065)	0.9999 (0.007)	0.9999 (0.030)
	$C = 1$	1 (0.045)	1 (0.002)	1 (0.002)
RMSE	$C = 0$	0.078 (0.005)	0.053 (0.004)	0.039 (0.003)
	$C = 0.5$	0.096 (0.005)	0.062 (0.004)	0.084 (0.005)
	$C = 1$	0.109 (0.006)	0.069 (0.006)	0.127 (0.009)

Table 6.1: Results of the simulation study to compare the proposed random effects model to the fixed effects model from Chapter 5 and the BYM.

in the same or in different clusters by both methods, that is the proportion of pairwise agreements between the two methods. A value of 1 indicates complete agreement between the two cluster configurations and a value of 0 indicates that no pair of areal units are classified in the same way under both configurations. For more information on the Rand Index, see Section 2.6.

The top panel of Figure 6.1 shows boxplots of the numbers of clusters estimated by each method, where the true values of 1 (when $C = 0$) and 19 (when $C = 0.5, 1$) are represented by dashed lines. The middle panel displays boxplots of the Rand index for all simulated data sets, while the bottom panel shows the RMSE values for the estimated risk surface. The

top panel shows that when $C = 0$ all three methods estimate the correct number of clusters on average. The random effects method proposed here has the largest standard deviation with a value of 6.74 compared with 4.35 for the BYM model and 1.31 for the fixed effects approach, though this can be explained by two large outliers under our approach. When $C = 0.5$ the median values are slightly high for the BYM model (21) and the fixed effects model (20), while the model proposed here estimates the correct number of clusters on average (19). Additionally our model has the lowest standard deviation in this scenario with a value of 1.68 compared to 4.50 for the BYM approach and 2.31 for the fixed effects model. When $C = 1$, all three models estimate the correct number of clusters on average, but again the model proposed here has the lowest standard deviation (0.18) compared with the BYM (3.98) and fixed effects (1.36) approaches.

A median Rand Index of 1 is obtained for all three models when $C = 0$ or $C = 1$, while when $C = 0.5$ we obtained medians of 0.9679 for the BYM model and 0.9999 for both the fixed effects model and the random effects model proposed here. In addition, there are a number of datasets for which the BYM model produces poor results in terms of both of these metrics, while there are fewer of these in the other two approaches.

Finally, the bottom panel shows that the model proposed here performs the best of the three in terms of RMSE when $C = 0$, with a median of 0.039 compared with 0.078 and 0.053 for the BYM and fixed effects approaches respectively, but performs poorest of the three for RMSE when $C = 1$, with

a median of 0.127 compared with 0.109 and 0.069. It seems likely that our model performs worse than the BYM model in this respect because our model has a single set of random effects, while the BYM model has two sets to share the modelling burden in this extreme case. Thus the independent random effects are able to capture the jumps in risk between clusters. In the more realistic case of $C = 0.5$ our method outperformed the BYM in terms of RMSE because it allows more flexibility in terms of localised smoothing.

6.4 Application to real data

This section continues the analysis of the respiratory hospitalisation risk data presented in Chapters 4 and 5.

6.4.1 Study design

As in the previous chapter, the study region is the Greater Glasgow and Clyde Health Board area, and we use the respiratory admission data introduced in Section 4.5. Figure 4.4 contains a map of Glasgow and the surrounding areas, with pins in the map to identify each location which is mentioned in this thesis. Table 4.2 provides a key for this map, with the numbers in the table corresponding to those in the pins in Figure 4.4.

The response data, $\boldsymbol{Y} = (Y_1, \ldots, Y_n)$, are based on the 2011 data, where Y_i is the number of hospital admissions with a primary diagnosis of respiratory disease in areal unit i in 2011. The expected values, $\boldsymbol{E} = (E_1, \ldots, E_n)$, are the expected hospital admission numbers for each areal unit in 2011. The top panel of Figure 5.8 displays the Standardised Incidence Ratio (SIR) for respiratory hospital admission, which is the ratio of the observed to the expected numbers of cases.

6.4.2 Results

The two-stage clustering model proposed in Section 6.2 was applied to these data, where the clustering step used respiratory disease data from 2008 to 2010. The fitted risk surfaces for these data sets exhibit similar spatial patterns to the 2011 study data, with Pearson's correlation coefficients of 0.86 (2010 data), 0.84 (2009 data) and 0.82 (2008 data) respectively. Markov chain Monte Carlo inference was used to obtain these results, with 5000 samples used for burn-in and a further 5000 used for the inference.

Figure 6.2 displays the posterior probabilities for the different numbers of clusters, and the optimal cluster structure was chosen to be that corresponding to the mode cluster number, which in this case was 18. Our method has the advantage of being able to quantify the uncertainty in the number of clusters identified, and Figure 6.2 shows that the 95% credible interval for this ranges between 17 and 29. In addition, the median cluster number was

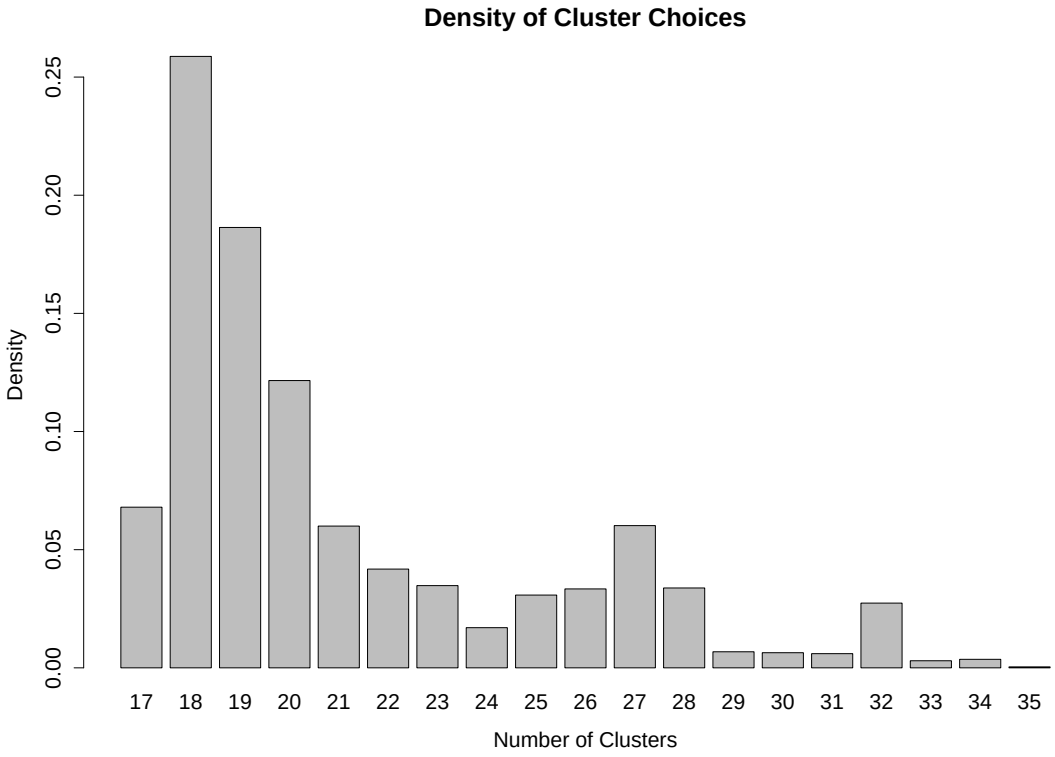

Figure 6.2: Plot of the posterior probability of each cluster configuration.

19. Note that due to the agglomerative nature of the clustering algorithm in Chapter 4, the clusters in the 17 cluster configuration are also present in the 29 cluster configuration but have been augmented by splitting off of a further 12 clusters.

The estimated risk surface (greyscale) and cluster structure (white dots) for the configuration with 18 clusters are displayed in the top panel of Figure 6.3. In the majority of cases, there do appear to be differences in risks between neighbouring clusters, and two of the more prominent clusters have been labelled A and B on the map. The low risk Cluster A is the affluent West End

of the city which is surrounded on all sides by more deprived areas. Cluster B includes a number of prosperous areas to the north of the city, including Milngavie, Milton of Campsie and Lennoxtown, which have much lower risks than neighbouring areas such as Kirkintilloch and the east end of the city, which are less affluent. The clusters appear to be based around grounds of socio-economic deprivation, which is well known to be linked with disease risk. The high risk areas in Figure 6.3 are generally areas with high levels of deprivation, while the lower risk areas are more affluent.

The bottom panel of Figure 6.3 displays the cluster configuration selected by the fixed effects model in Chapter 5. As you would expect given the nature of the clustering algorithm, there are a number of similarities between the cluster structures selected under the two modelling approaches. The fixed effects model in the bottom panel has 15 more clusters, but these additional clusters are formed by splitting the clusters shown in the top panel. The differences between the two approaches in terms of estimated disease risks are very slight, with the same areas being identified as high and low risk in both plots. The areas to the south and west appear to have slightly higher estimated risks under the random effects approach (top panel), and there appear to also be some higher risks observed to the east of the city, but overall it appears that both models are estimating similar disease risk patterns.

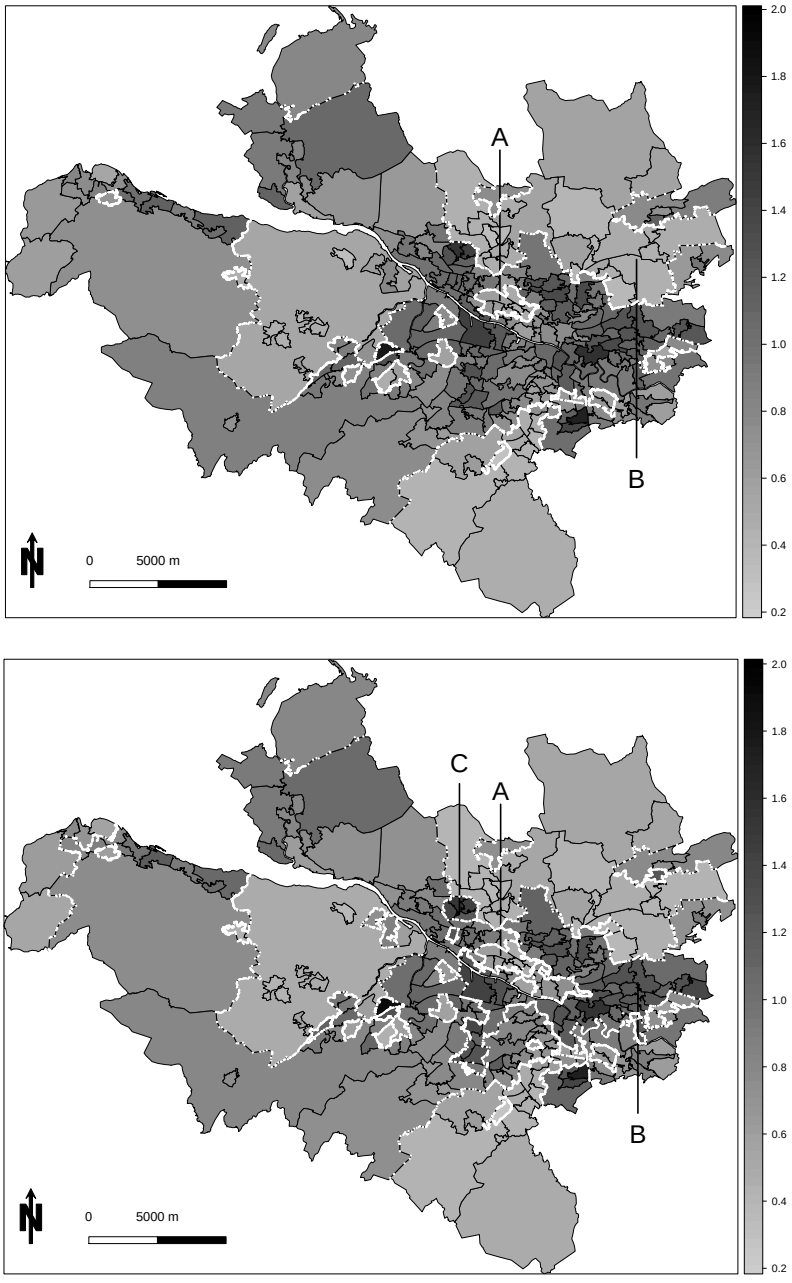

Figure 6.3: The top panel displays the estimated risk surface (grey-scale) from the random effects model with 18 clusters (white dots), while the bottom plot displays the estimated risk surface (grey-scale) from the fixed effects model with 33 clusters (white dots).

6.4.3 Sensitivity Analyses

The model outlined above uses a Gamma(0.001,0.001) prior for the precision parameter τ in the CAR model. To assess the effect of the choice of prior on the model fit and the number of clusters selected, we compared our choice of hyperparameters with two alternative choices - Gamma(0.1,0.1) and Gamma(1,0.0005). The two-stage model was applied to the real data using both of these alternative prior distributions, and in both cases 18 clusters were selected, with disease risk estimates very similar to those obtained in the original study. Therefore it does not appear that changing the prior for τ has a substantial effect on the ability of the model to identify the correct cluster structure or estimate the disease risk for each area.

6.5 Discussion

Here we have proposed statistical methodology which simultaneously estimates the spatial pattern in disease risk and identifies clusters of areas exhibiting high (and low) risk. This method involves a combination of spatial agglomerative hierarchical clustering techniques and an extended conditional autoregressive model, with inference based on Markov chain Monte Carlo simulation. The clustering techniques introduced in Chapter 4 are applied to disease risk data prior to our study period, allowing us to elicit candidate cluster structures for the study data. These candidate structures have a natural ordering in terms of the number of clusters, which allows them to be considered as a univariate parameter in our Bayesian hierarchical model.

This model estimates disease risk directly via the random effects, allowing for correlation between neighbouring areal units which are in the same cluster while not enforcing correlation for areas in different clusters. This approach differs from that used in Chapter 5, where the cluster structure was fixed when estimating the remaining model parameters. Unlike in the previous chapter, here we are able to produce a credible interval for the number of clusters and can identify other potential alternative cluster structures which are supported by the data.

The simulation study in Section 6.3 shows that our model generally outperforms the BYM model with the posterior classification step, with improved performances for both risk estimation and cluster identification. This is unsurprising, since our model attempts to estimate the cluster structure in the data, while the BYM approach estimates a smooth risk surface and then attempts to identify clusters in this smooth surface. Our model also performs well in certain cases when compared with the fixed effects approach proposed in Chapter 5. In terms of identifying the correct number of clusters, our model produces the correct median cluster number and the lowest standard deviation in the cases with $C = 0.5$ and $C = 1$, though the fixed effects model performs better in the case where $C = 0$. This model does not, however, perform as well as the fixed effects model in terms of estimating risk in the cases where $C = 0.5$ and $C = 1$. This is because the fixed effects model has extra parameters in the mean model, while our approach accounts for clusters in the correlation structure of the random effects. However, in the case where $C = 0$, our model performs the best of the three.

It seems clear that each of the methods developed within Chapters 5 and 6 have advantages over the other in certain sets of circumstances, but that both methods are preferable to the existing approaches used for cluster identification. The approach introduced here has the advantage of allowing us to quantify uncertainty in the selected cluster structure, and allows estimation within a single model rather than requiring comparison of multiple models. However, this model does not have additional parameters to control the means of the clusters, and that means that the estimation of risk is slightly poorer than the approach introduced in Chapter 5. Both methods obtain similar disease risk patterns when applied to the Glasgow respiratory admission data, suggesting that the differences between the methods in terms of estimation are not substantial enough to affect the overall conclusions about disease risk.

The decision about which of these two methods to use should be based on the aims of the analysis. This random effects approach is preferable if the primary aim is to identify the cluster structure, with the estimation of disease risk being a secondary consideration. An example of such an application would be a health board wishing to partition their region into a set of high and low risk clusters in order to focus their resources most appropriately; here the identification of the clusters is key, whilst in terms of risk estimation it is sufficient to simply know which areas have high and low disease risk. However, the fixed effects approach introduced in the previous chapter is likely to be preferable if the primary aim of the analysis is to estimate the disease risk, and the cluster identification is simply a mechanism to ensure that the

estimated spatial autocorrelation of the risk surface is as accurate as possible.

There is scope to extend these modelling approaches into the spatio-temporal domain, thus allowing us to identify changes in the risk surface over time. This would allow the government to identify whether a particular health intervention has had the desired effect in terms of reducing disease risk in a high-risk cluster. It would also allow for identification of clusters where the disease risk has increased over time, thus allowing health officials to investigate the possible causes for any such deterioration in health. Such an approach will be explored in Chapter 7.

Chapter 7

Identifying changes in the spatial structure over time: a spatio-temporal approach

7.1 Introduction

The Bayesian models developed in Chapters 5 and 6 are designed to identify clusters of areal units exhibiting similar disease risk at a specific point in time, but in many cases disease risk data will be available over a range of different time points. There is increasing interest in using spatio-temporal models to estimate how the disease risk pattern develops over time and to estimate the improvement or deterioration in the level of disease risk within an areal unit. Section 3.6 provided an overview of the existing spatio-temporal disease mapping literature, but there are few models which seek to identify

clusters in this spatio-temporal setting.

Here, we propose a new modelling approach for spatio-temporal clustering in the disease risk pattern. There are two separate clustering parameters within the model; the first is based on the average risk (intercept) and the second is based on the change in disease risk over time (slope), thus allowing for easy identification of groups of areal units which share similar characteristics in terms of their average risk and the evolution of that risk across the study period. Unlike the two-stage models in Chapters 5 and 6, this is a single stage approach which identifies the clusters and estimates disease risk within a single Bayesian model. Rather than determining a set of potential cluster structures in advance, the approach proposed here allows each areal unit to be assigned to a cluster within the model. The Bayesian model proposed here extends the Bernardinelli model (Bernardinelli et al. (1995)) by allowing different intercept and slope terms for each cluster. The model estimates disease risk via four parameters, a pair to estimate the intercept and a pair to estimate the slope. Each pair consists of a set of cluster-specific fixed effect terms and a set of spatially correlated random effects which follow a conditional autoregressive model. The fixed effects mean that areal units within the same intercept cluster will have similar intercept values, but the random effects allow for some variation within a cluster. The same is true for the slope fixed and random effects.

The remainder of this chapter is organised as follows. Section 7.2 outlines the spatio-temporal clustering model proposed here and how it differs from

the models proposed in Chapters 5 and 6. Section 7.3 uses simulated data to test the efficacy of this model and to compare it to an existing approach for spatio-temporal modelling. The model is applied to respiratory hospital admission data for the Greater Glasgow and Clyde Health Board area in Section 7.4 in order to estimate the change in the spatial pattern of disease risk over time. Finally, Section 7.5 discusses the advantages of this approach compared to existing spatio-temporal modelling methodology.

7.2 Methodology

7.2.1 Proposed model

We propose a Poisson generalised linear model to estimate the pattern of disease risk across the entire study period and identify clusters of areal units which are similar in terms of average disease risk and the rate of change in disease risk over time. This approach assumes that change in disease risk levels over time for an areal unit can be described by a linear relationship. There is assumed to be a unique linear relationship for each areal unit, and the modelling approach seeks to estimate disease risk by estimating the intercept and slope for each areal unit. Such an approach was first proposed by Bernardinelli (Bernardinelli et al. (1995)), who suggested the following model:

$$\begin{aligned} Y_{it} &\sim \text{Poisson}(\mu_{it}) \qquad i = 1, \ldots, n, \quad t = 1, \ldots, T \\ \log(\mu_{it}) &= (\alpha + \phi_i) + (\beta + \delta_i)(t - \bar{t}), \end{aligned} \tag{7.1}$$

where Y_{it} is the observed disease risk for area i and time point t, α is a global intercept term common to all areas, ϕ_i is the area-specific effect intercept, β is a slope effect common to all areas and δ_i represents an area-specific slope. Here, $\bar{t}$ is the mean of the time points, and is used to centre time to ensure that the intercept term represents the average risk over the time period. The random effect terms $\boldsymbol{\phi}$ and $\boldsymbol{\delta}$ can be modelled by the intrinsic CAR prior given in (2.5).

Here we extend this model to allow for the identification of clusters in both the intercept and slope. Rather than global effects, α and β, our model allows for two sets of cluster-specific fixed effects, $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_{N_C})$ and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_{N_D})$, where N_C and N_D are the number of clusters for the intercept and slope respectively. Two areal units in the same cluster will have the same fixed effect term, while two areal units in different clusters will have different fixed effects, which means that areal units in the same cluster are more likely to have similar risk values.

The areal units are partitioned into a set of intercept clusters, where all areas within a cluster are believed to have similar levels of average disease risk. The areal units are also partitioned into a separate set of slope clusters, where all areas within a cluster are believed to have similar rates of change

of disease risk over time. Areal units are allocated to the intercept and slope clusters independently; it is possible for two areal units to lie in the same intercept cluster, but in different slope clusters, and vice versa. For example, consider areal unit $\mathcal{A}_i$ where the level of disease risk is high on average and is increasing over time, and areal unit $\mathcal{A}_j$ where the level of disease risk is high on average but is decreasing over time. Both areal units may lie in the same intercept cluster on account of having high average disease risk, but would be in different slope clusters due to one having an increasing risk and the other having a decreasing risk.

Although the slope and intercept clusters are allocated separately, it is straightforward to combine them to identify groups of areal units which are similar in terms of both slope and intercept. Unlike the approaches in Chapters 5 and 6, this approach does not enforce spatially contiguous clusters, in order to avoid producing an overly large number of clusters. The approach in Chapter 5 identified 33 clusters, at just one time point, and if these clusters were further subdivided as a result of different changes in risk over time for different areal units within a cluster, then excessive numbers of clusters could be identified. However, if necessary, it is straightforward to carry out a post-hoc approach to partition the existing clusters into a set of spatially contiguous clusters.

The intercept and the slope are each estimated by two separate parameters in the model; each has a set of cluster-specific fixed effect terms and a set of spatially correlated random effects. The fixed effect means that areal units

within the same intercept cluster will have similar intercepts, but the random effects allow for some variation within a cluster. The same is true for the slope clusters. The proposed model is as follows:

$$\begin{aligned} Y_{it}|E_{it}, R_{it} &\sim \text{Poisson}(E_{it}R_{it}) \qquad i = 1, \ldots, n, \quad t = 1, \ldots, T \\ \ln(R_{it}) &= \alpha_{C_i} + \phi_i + [\beta_{D_i} + \delta_i](t - \bar{t}) \end{aligned} \tag{7.2}$$

This model is of a similar form to the Bernardinelli model (7.1), but allows for a set of cluster-specific fixed effects $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_{N_C})$ and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_{N_D})$ where N_C and N_D are the number of clusters for the intercept and slope respectively. Again, $\bar{t}$ is the mean of the time points, used to ensure that the intercept term represents the average risk over the time period.

Here, the random effect terms $\boldsymbol{\phi}$ and $\boldsymbol{\delta}$ are modelled using the Leroux CAR prior (see Section 2.4.2) as follows:

$$\begin{aligned} \phi_i|\boldsymbol{\phi_{-i}} &\sim \text{N}\left(\frac{\rho\sum_{j=1}^n w_{ij}\phi_j}{\rho\sum_{j=1}^n w_{ij} + 1 - \rho}, \frac{1}{\tau(\rho\sum_{j=1}^n w_{ij} + 1 - \rho)}\right) \qquad i = 1, \ldots, n \\ \delta_i|\boldsymbol{\delta_{-i}} &\sim \text{N}\left(\frac{\lambda\sum_{j=1}^n w_{ij}\delta_j}{\lambda\sum_{j=1}^n w_{ij} + 1 - \lambda}, \frac{1}{\sigma(\lambda\sum_{j=1}^n w_{ij} + 1 - \lambda)}\right) \qquad i = 1, \ldots, n \end{aligned}$$

Note that the joint distributions for $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_n)$ and $\boldsymbol{\delta} = (\delta_1, \ldots, \delta_n)$ corresponding to the above conditional distributions are $\boldsymbol{\phi} \sim \text{N}(\mathbf{0}, \frac{Q_\rho^{-1}}{\tau})$ where $Q_\rho = \rho[\text{diag}(W\mathbf{1}) - W] + (1 - \rho)I$ and $\boldsymbol{\delta} \sim \text{N}(\mathbf{0}, \frac{Q_\lambda^{-1}}{\sigma})$ where $Q_\lambda = \lambda[\text{diag}(W\mathbf{1}) - W] + (1 - \lambda)I$.

The fixed effects, $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are modelled as follows:

$$\begin{aligned} \alpha_j &\sim \text{Uniform}(\alpha_{j-1}, \alpha_{j+1}) \qquad j = 1, \ldots, N_C \\ \beta_j &\sim \text{Uniform}(\beta_{j-1}, \beta_{j+1}) \qquad j = 1, \ldots, N_D \end{aligned}$$

where N_C and N_D are the maximum number of clusters for α and β respectively. Here $\alpha_0 = \beta_0 - \infty$ and $\alpha_{N_C+1} = \beta_{N_D+1} = \infty$. In order to avoid the label switching problem (Stephens (2000)), the values of $\boldsymbol{\alpha}$ are ordered, with $\alpha_{j-1} \leq \alpha_j \leq \alpha_{j+1}$ so that a move from cluster j to cluster $j+1$ will always represent an increase in intercept value.

Initially, equal prior probabilities, $\text{P}(C_i = c) = \frac{1}{N_C}$ were considered for assigning areal units to clusters, but it was considered preferable to give additional weight to the central clusters to ensure that areal units only move to extreme high or low risk clusters if their risk level is substantially different to the mean. This is achieved via an exponential decay function as follows:

$$\begin{aligned} \text{P}(C_i = c) &= \frac{\exp(-\theta_C(c - \bar{C})^2)}{\sum_{j=1}^{N_C} \exp(-\theta_C(j - \bar{C})^2)} \qquad c = 1, \ldots, N_C \\ \text{P}(D_i = d) &= \frac{\exp(-\theta_D(d - \bar{D})^2)}{\sum_{j=1}^{N_D} \exp(-\theta_D(j - \bar{D})^2)} \qquad d = 1, \ldots, N_D \end{aligned}$$

where $\bar{C} = \frac{N_C+1}{2}$ when N_C is odd, and $\bar{C} = \frac{N_C}{2}$ when N_C is even, and likewise $\bar{D} = \frac{N_D+1}{2}$ when N_D is odd and $\bar{D} = \frac{N_D}{2}$ when N_D is even. The values of N_C and N_D are chosen in advance to reflect prior beliefs about the number of different intercept and slope levels expected within the study region. These are

simply the maximum number of clusters permitted in each direction, since it is possible for a cluster to be empty (i.e contain no areal units). These cluster membership probabilities each have an extra parameter (θ_C and θ_D) to control the level of weighting towards the central clusters, with larger values meaning higher weighting is assigned to the central clusters.

The hyperparameters of this model are outlined as follows:

$$\begin{aligned} \tau, \sigma &\sim \text{Gamma}(\gamma, \psi) \\ \rho, \lambda &\sim \text{Uniform}(0, 1) \\ \theta_C, \theta_D &\sim \text{Uniform}(1, 100). \end{aligned}$$

Here, τ and σ are the precision hyperparameters for the intercept and slope random effects respectively, and within this thesis we set $\gamma = 0.01$ and $\psi = 0.01$. The hyperparameters ρ and λ control the level of spatial autocorrelation within the intercept and slope random effects. As discussed above, θ_C and θ_D control the level of weighting towards the central clusters. The lower bound of the Uniform distribution for these terms is chosen to be 1 rather than 0 based on our prior belief that extra weight should be given to central clusters; a value of $\theta_C = 1$ would correspond to a standard exponential decay, while a value of $\theta_C = 0$ would assign equal probability to each cluster.

7.2.2 Inference via McMC

Inference for this model is carried out using an McMC algorithm, using a combination of Gibbs sampling and Metropolis-Hastings steps. The algorithm produces posterior distributions for each of the model parameters, and the full conditionals for the parameters of this McMC algorithm are as follows:

$\boldsymbol{\alpha}$ - cluster specific intercept term

The cluster-specific fixed effects are updated in order, starting with α_1 and finishing with α_{N_C}. The full conditional for α_j is:

$$\begin{aligned} f(\alpha_j|\boldsymbol{Y}) &\propto \prod_{i:C_i=j}\prod_{t=1}^{T}\text{Poisson}(Y_{it}|\alpha_j)\text{Uniform}(\alpha_{j-1},\alpha_{j+1}) \\ &\propto \prod_{i:C_i=j}\prod_{t=1}^{T}\left(E_{it}\exp\left(\alpha_j+\phi_i+[\beta_{D_i}+\delta_i](t-\bar{t})\right)\right)^{Y_{it}}\times \\ &\quad \exp\left(-E_{it}\exp\left(\alpha_j+\phi_i+[\beta_{D_i}+\delta_i](t-\bar{t})\right)\right) \end{aligned}$$

A Metropolis algorithm is used to update this parameter, with a proposal α_j^* drawn from a truncated normal distribution $\alpha_j^* \sim \text{N}(\alpha_j^{(m)}, v_\alpha)$, with $\alpha_{j-1}^{(m+1)} \leq \alpha_j^* \leq \alpha_{j+1}^{(m)}$ where $\alpha_j^{(m)}$ is the current state of the chain. The acceptance probability of a move from $\alpha_j^{(m)}$ to α_j^* is given by $\min\left(1, \frac{f(\alpha_j^*|\boldsymbol{Y})}{f(\alpha_j^{(m)}|\boldsymbol{Y})}\right)$. The proposal variance v_α can be altered within the algorithm to maintain an acceptance rate between 40% and 80%.

$\boldsymbol{\beta}$ - cluster specific slope term

The cluster-specific fixed effects are updated in order, starting with β_1 and finishing with β_{N_D}. The full conditional for β_j is:

$$\begin{aligned} f(\beta_j|\boldsymbol{Y}) &\propto \prod_{i:D_i=j}\prod_{t=1}^{T} \text{Poisson}(Y_{it}|\beta_j)\text{Uniform}(\beta_{j-1},\beta_{j+1}) \\ &\propto \prod_{i:D_i=j}\prod_{t=1}^{T} \left(E_{it}\exp\left(\alpha_{C_i}+\phi_i+[\beta_j+\delta_i](t-\bar{t})\right)\right)^{Y_{it}} \times \\ &\quad \exp\left(-E_{it}\exp\left(\alpha_{C_i}+\phi_i+[\beta_j+\delta_i](t-\bar{t})\right)\right) \end{aligned}$$

A Metropolis algorithm is used to update this parameter, with a proposal β_j^* drawn from a truncated normal distribution $\beta_j^* \sim \text{N}(\beta_j^{(m)}, v_\beta)$, with $\beta_{j-1}^{(m+1)} \leq \beta_j^* \leq \beta_{j+1}^{(m)}$. The acceptance probability of a move from $\beta_j^{(m)}$ to β_j^* is given by $\min\left(1, \frac{f(\beta_j^*|\boldsymbol{Y})}{f(\beta_j^{(m)}|\boldsymbol{Y})}\right)$. The proposal variance v_β can be altered within the algorithm to maintain an acceptance rate between 40% and 80%.

$\boldsymbol{\phi}$ - intercept random effects

Each of the intercept random effects, ϕ_i is updated in turn. The full conditional for ϕ_i is:

$$\begin{aligned} f(\phi_i|\boldsymbol{Y}) &\propto \prod_{t=1}^{T} \text{Poisson}(Y_{it}|\phi_i) \times \text{N}\left(\frac{\rho\sum_{j=1}^{n} w_{ij}\phi_j}{\rho\sum_{j=1}^{n} w_{ij} + 1 - \rho}, \frac{1}{\tau(\rho\sum_{j=1}^{n} w_{ij} + 1 - \rho)}\right) \\ &\propto \prod_{t=1}^{T} \left(E_{it} \exp\left(\alpha_{C_i} + \phi_i + [\beta_{D_i} + \delta_i](t - \bar{t})\right)\right)^{Y_{it}} \times \\ &\quad \exp\left(-E_{it} \exp\left(\alpha_{C_i} + \phi_i + [\beta_{D_i} + \delta_i](t - \bar{t})\right)\right) \times \\ &\quad \exp\left(-\frac{1}{2}\tau\left(\phi_i - \frac{\rho\sum_{j=1}^{n} w_{ij}\phi_j}{\rho\sum_{j=1}^{n} w_{ij} + 1 - \rho}\right)^2\right) \end{aligned}$$

A Metropolis algorithm is used to update the blocks, with a proposal ϕ_i^* drawn from the distribution $\phi_i^* \sim \text{N}(\phi_i^{(m)}, v_\phi)$. The acceptance probability of a move from $\phi_i^{(m)}$ to ϕ_i^* is given by $\min\left(1, \frac{f(\phi_i^*|\boldsymbol{\phi_{-i}}, \boldsymbol{Y})}{f(\phi_i^{(m)}|\boldsymbol{\phi_{-i}}, \boldsymbol{Y})}\right)$. The proposal variance v_ϕ can be altered within the algorithm to maintain an acceptance rate between 40% and 80%.

$\boldsymbol{\delta}$ - slope random effects

Each of the slope random effects, δ_i is updated in turn. The full conditional for δ_i is:

$$\begin{aligned} f(\delta_i|\boldsymbol{Y}) &\propto \prod_{t=1}^{T} \text{Poisson}(Y_{it}|\delta_i) \times \text{N}\left(\frac{\lambda \sum_{j=1}^{n} w_{ij}\delta_j}{\lambda \sum_{j=1}^{n} w_{ij} + 1 - \lambda}, \frac{1}{\sigma(\lambda \sum_{j=1}^{n} w_{ij} + 1 - \lambda)}\right) \\ &\propto \prod_{t=s_1}^{s_T} \left(E_{it} \exp\left(\alpha_{C_i} + \phi_i + [\beta_{D_i} + \delta_i](t - \bar{t})\right)\right)^{Y_{it}} \times \\ &\quad \exp\left(-E_{it} \exp\left(\alpha_{C_i} + \phi_i + [\beta_{D_i} + \delta_i](t - \bar{t})\right)\right) \times \\ &\quad \exp\left(-\frac{1}{2}\sigma\left(\delta_i - \frac{\lambda \sum_{j=1}^{n} w_{ij}\delta_j}{\lambda \sum_{j=1}^{n} w_{ij} + 1 - \lambda}\right)^2\right) \end{aligned}$$

A Metropolis algorithm is used to update the blocks, with a proposal δ_i^* drawn from the distribution $\delta_i^* \sim \text{N}(\delta_i^{(m)}, v_\delta)$. The acceptance probability of a move from $\delta_i^{(m)}$ to δ_i^* is given by $\min\left(1, \frac{f(\delta_i^*|\boldsymbol{\delta_{-i}}, \boldsymbol{Y})}{f(\delta_i^{(m)}|\boldsymbol{\delta_{-i}}, \boldsymbol{Y})}\right)$. The proposal variance v_δ can be altered within the algorithm to maintain an acceptance rate between 40% and 80%.

C_i - intercept cluster indicator

The indicator C_i determines which intercept cluster areal unit i belongs to. This parameter is updated for each areal unit in turn, and the full conditional for C_i is:

$$\begin{aligned} f(C_i|\boldsymbol{Y},\boldsymbol{\alpha},\theta_C) &\propto \prod_{t=1}^{T} \text{Poisson}(Y_{it}|\alpha_{C_i}) \times P(C_i|\theta_C) \\ &\propto \prod_{t=1}^{T} \left(E_{it} \exp\left(\alpha_{C_i} + \phi_i + [\beta_{D_i} + \delta_i](t-\bar{t})\right)\right)^{Y_{it}} \times \\ &\quad \exp\left(- E_{it} \exp\left(\alpha_{C_i} + \phi_i + [\beta_{D_i} + \delta_i](t-\bar{t})\right)\right) \times \\ &\quad \frac{\exp(-\theta_C(C_i - \bar{C})^2)}{\sum_{j=1}^{N_C} \exp(-\theta_C(j-\bar{C})^2)} \end{aligned}$$

A Metropolis algorithm is used to update this parameter, with a proposal C_i^* drawn from the set $1, \ldots, C_i - 1, C_i + 1, \ldots, N_C$ with probability $\frac{1}{N_C - 1}$ assigned to each. This is equivalent to randomly allocating the areal unit to another cluster. The acceptance probability of a move from $C_i^{(m)}$ to C_i^* is given by $\min\left(1, \frac{f(C_i^*|\boldsymbol{Y},\boldsymbol{\alpha},\theta_C)}{f(C_i^{(m)}|\boldsymbol{Y},\boldsymbol{\alpha},\theta_C)}\right)$.

D_i - slope cluster indicator

The indicator D_i determines which slope cluster areal unit i belongs to. This parameter is updated for each areal unit in turn, and the full conditional for D_i is:

$$\begin{aligned} f(D_i|\boldsymbol{Y},\boldsymbol{\beta},\theta_D) &\propto \prod_{t=1}^{T} \text{Poisson}(Y_{it}|\beta_{D_i}) \times P(D_i|\theta_D) \\ &\propto \prod_{t=1}^{T} \left(E_{it} \exp\left(\alpha_{C_i} + \phi_i + [\beta_{D_i} + \delta_i](t-\bar{t})\right)\right)^{Y_{it}} \times \\ &\quad \left(- E_{it} \exp\left(\alpha_{C_i} + \phi_i + [\beta_{D_i} + \delta_i](t-\bar{t})\right)\right) \times \\ &\quad \frac{\exp(-\theta_D(D_i - \bar{D})^2)}{\sum_{j=1}^{N_D} \exp(-\theta_D(j-\bar{D})^2)} \end{aligned}$$

A Metropolis algorithm is used to update this parameter, with a proposal D_i^* drawn from the set $1, \ldots, D_i - 1, D_i + 1, \ldots, N_D$ with probability $\frac{1}{N_D - 1}$ assigned to each. This is equivalent to randomly allocating the areal unit to another cluster. The acceptance probability of a move from $D_i^{(m)}$ to D_i^* is given by $\min\left(1, \frac{f(D_i^*|\boldsymbol{Y},\boldsymbol{\beta},\theta_D)}{f(D_i^{(m)}|\boldsymbol{Y},\boldsymbol{\beta},\theta_D)}\right)$.

τ - precision hyperparameter for intercept random effects

The full conditional for τ is:

$$\begin{aligned} f(\tau|\boldsymbol{\phi}) &\propto \mathrm{N}\left(\boldsymbol{\phi}\middle|\mathbf{0}, \frac{Q_\rho^{-1}}{\tau}\right) \mathrm{Gamma}(\tau|\gamma,\psi) \\ &\propto |\tau|^{\frac{n}{2}} \exp\left(-\frac{1}{2}(\boldsymbol{\phi}^T Q_\rho \boldsymbol{\phi})\tau\right) \times \tau^{\gamma-1}\exp(-\psi\tau) \\ &\propto |\tau|^{(\frac{n+1}{2}+\gamma)-1} \exp\left(-(\frac{1}{2}\boldsymbol{\phi}^T Q_\rho \boldsymbol{\phi} + \psi)\tau\right) \\ &\sim \mathrm{Gamma}\left(\frac{n+1}{2}+\gamma, \frac{1}{2}\boldsymbol{\phi}^T Q_\rho \boldsymbol{\phi} + \psi\right) \end{aligned}$$

This full conditional distribution can be sampled from using Gibbs sampling. To update τ we simply draw from the posterior Gamma distribution.

σ - precision hyperparameter for slope random effects

The full conditional for σ is:

$$\begin{aligned} f(\sigma|\boldsymbol{\delta}) &\propto \mathrm{N}\left(\boldsymbol{\delta}\middle|\mathbf{0}, \frac{Q_\lambda^{-1}}{\sigma}\right) \mathrm{Gamma}(\sigma|\gamma,\psi) \\ &\propto |\sigma|^{\frac{n}{2}} \exp\left(-\frac{1}{2}(\boldsymbol{\delta}^T Q_\lambda \boldsymbol{\delta})\sigma\right) \times \sigma^{\gamma-1}\exp(-\psi\sigma) \\ &\propto |\sigma|^{(\frac{n+1}{2}+\gamma)-1} \exp\left(-(\frac{1}{2}\boldsymbol{\delta}^T Q_\lambda \boldsymbol{\delta} + \psi)\sigma\right) \\ &\sim \mathrm{Gamma}\left(\frac{n+1}{2}+\gamma, \frac{1}{2}\boldsymbol{\delta}^T Q_\lambda \boldsymbol{\delta} + \psi\right) \end{aligned}$$

This full conditional distribution can be sampled from using Gibbs sampling. To update σ we simply draw from the posterior Gamma distribution.

ρ - hyperparameter to control spatial autocorrelation in intercept random effects

The full conditional for ρ is:

$$\begin{aligned} f(\rho|\boldsymbol{\phi},\tau) &\propto \mathrm{N}\left(\boldsymbol{\phi}\middle|\mathbf{0},\frac{Q_\rho^{-1}}{\tau}\right)\times \mathrm{Uniform}(\rho|0,1) \\ &\propto |Q_\rho|^{\frac{1}{2}}\exp\left(-\frac{1}{2}(\boldsymbol{\phi}^T Q_\rho \boldsymbol{\phi})\tau\right)\times I_{[\rho\in[0,1]]} \end{aligned}$$

A Metropolis algorithm is used to update this parameter, with a proposal ρ^* drawn from a truncated normal distribution $\rho^* \sim \mathrm{N}(\rho^{(m)},\, v_\rho)$, with $0 \leq \rho^* \leq 1$. The acceptance probability of a move from $\rho^{(m)}$ to ρ^* is given by $\min\left(1, \frac{f(\rho^*|\boldsymbol{\phi},\tau)}{f(\rho^{(m)}|\boldsymbol{\phi},\tau)}\right)$. The proposal variance v_ρ can be altered within the algorithm to maintain an acceptance rate between 40% and 80%.

λ - hyperparameter to control spatial autocorrelation in slope random effects

The full conditional for λ is:

$$\begin{aligned} f(\lambda|\boldsymbol{\delta},\sigma) &\propto \mathrm{N}\left(\boldsymbol{\delta}\middle|\mathbf{0},\frac{Q_\lambda^{-1}}{\sigma}\right)\times \mathrm{Uniform}(\lambda|0,1) \\ &\propto |Q_\lambda|^{\frac{1}{2}}\exp\left(-\frac{1}{2}(\boldsymbol{\delta}^T Q_\lambda \boldsymbol{\delta})\sigma\right)\times I_{[\lambda\in[0,1]]} \end{aligned}$$

A Metropolis algorithm is used to update this parameter, with a proposal λ^* drawn from a truncated normal distribution $\lambda^* \sim \mathrm{N}(\lambda^{(m)},\, v_\lambda)$, with $0 \leq \lambda^* \leq 1$. The acceptance probability of a move from $\lambda^{(m)}$ to λ^* is given by $\min\left(1, \frac{f(\lambda^*|\boldsymbol{\delta},\sigma)}{f(\lambda^{(m)}|\boldsymbol{\delta},\sigma)}\right)$. The proposal variance v_λ can be altered within the algorithm to maintain an acceptance rate between 40% and 80%.

θ_C - parameter for controlling the spatial weights for intercept clusters

The full conditional for θ_C is:

$$\begin{aligned} f(\theta_C|\boldsymbol{C}) &\propto \prod_{i=1}^{n} P(C_i|\theta_C) \times \text{Uniform}(\theta_C|1,100) \\ &\propto \prod_{i=1}^{n} \frac{\exp(-\theta_C(C_i-\bar{C})^2)}{\sum_{j=1}^{N_C}\exp(-\theta_C(j-\bar{C})^2)} \times I_{[\theta_C\in[1,100]]} \end{aligned}$$

A Metropolis algorithm is used to update this parameter, with a proposal θ_C^* drawn from a truncated normal distribution $\theta_C^* \sim \text{N}(\theta_C^{(m)}, v_{\theta_C})$, with $1 \leq \theta_C^* \leq 100$. The acceptance probability of a move from $\theta_C^{(m)}$ to θ_C^* is given by $\min\left(1, \frac{f(\theta_C^*|\boldsymbol{C})}{f(\theta_C^{(m)}|\boldsymbol{C})}\right)$. The proposal variance v_{θ_C} can be altered within the algorithm to maintain an acceptance rate between 40% and 80%.

θ_D - parameter for controlling the spatial weights for slope clusters

The full conditional for θ_D is:

$$\begin{aligned} f(\theta_D|\boldsymbol{D}) &\propto \prod_{i=1}^{n} P(D_i|\theta_D) \times \text{Uniform}(\theta_D|1,100) \\ &\propto \prod_{i=1}^{n} \frac{\exp(-\theta_D(D_i-\bar{D})^2)}{\sum_{j=1}^{N_D}\exp(-\theta_D(j-\bar{D})^2)} \times I_{[\theta_D\in[1,100]]} \end{aligned}$$

A Metropolis algorithm is used to update this parameter, with a proposal θ_D^* drawn from a truncated normal distribution $\theta_D^* \sim \text{N}(\theta_D^{(m)}, v_{\theta_D})$, with $1 \leq \theta_D^* \leq 100$. The acceptance probability of a move from $\theta_D^{(m)}$ to θ_D^* is given

by $\min\left(1, \frac{f\left(\theta_D^*|\boldsymbol{D}\right)}{f\left(\theta_D^{(m)}|\boldsymbol{D}\right)}\right)$. The proposal variance v_{θ_D} can be altered within the algorithm to maintain an acceptance rate between 40% and 80%.

7.3 Simulation study

7.3.1 Aim

A simulation study was conducted to establish the efficacy of the Bayesian spatio-temporal clustering model outlined in the previous section. The template for the study was the set of 271 Intermediate Geographies comprising the Greater Glasgow and Clyde Health Board, which is the study region for the motivating application presented in Section 7.4. A study was conducted comparing the model proposed here with the Bernardinelli model outlined in Section 2.5.1.

7.3.2 Data Generation

In order to match the application presented in Section 7.4, clustered disease data were generated based on the Greater Glasgow and Clyde Health Board region over ten time points. Each areal unit in the study region was assigned to an intercept cluster and a slope cluster. Intercept clusters and slope clusters were each separately simulated in a similar manner to the simulation template outlined in Chapters 4, 5 and 6, with both the intercept and slope templates having three clusters with values -1, 0 and 1 respectively. These

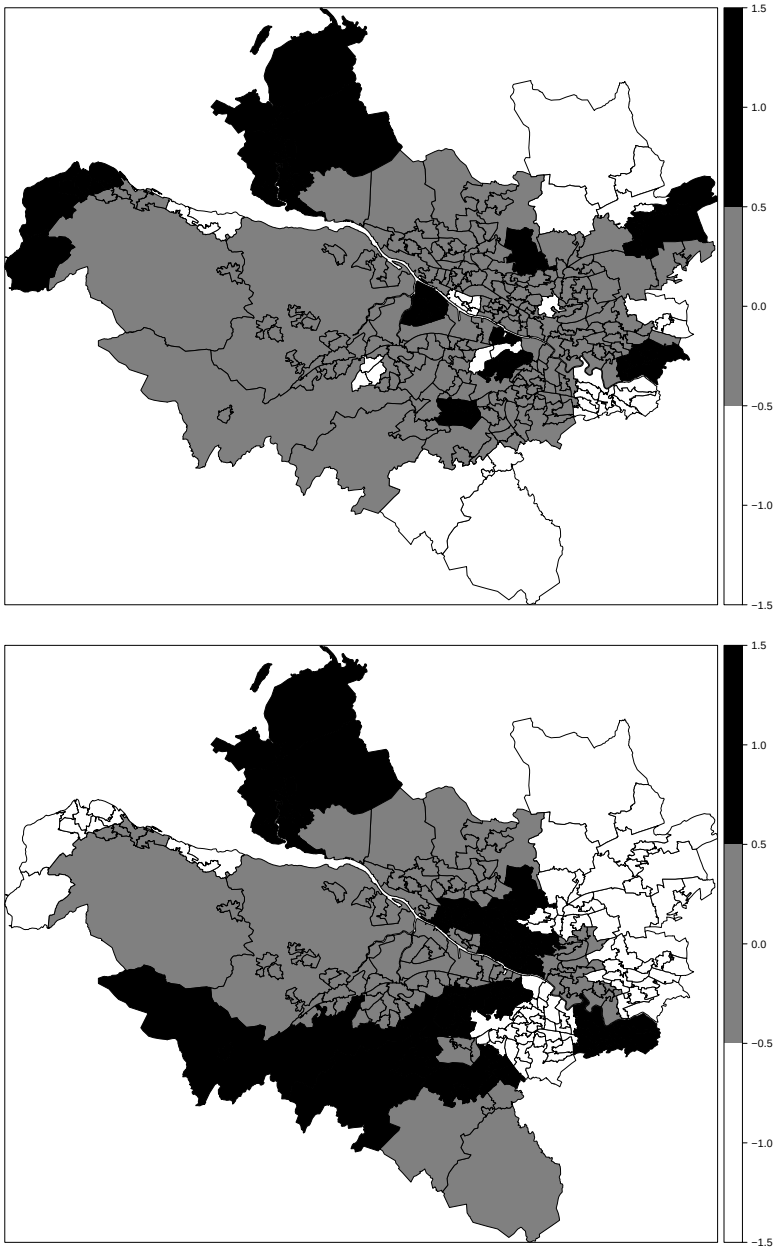

Figure 7.1: Plots of the simulated intercept and slope clusters. The top panel shows the set of intercept clusters, while the bottom panel shows the set of slope clusters.

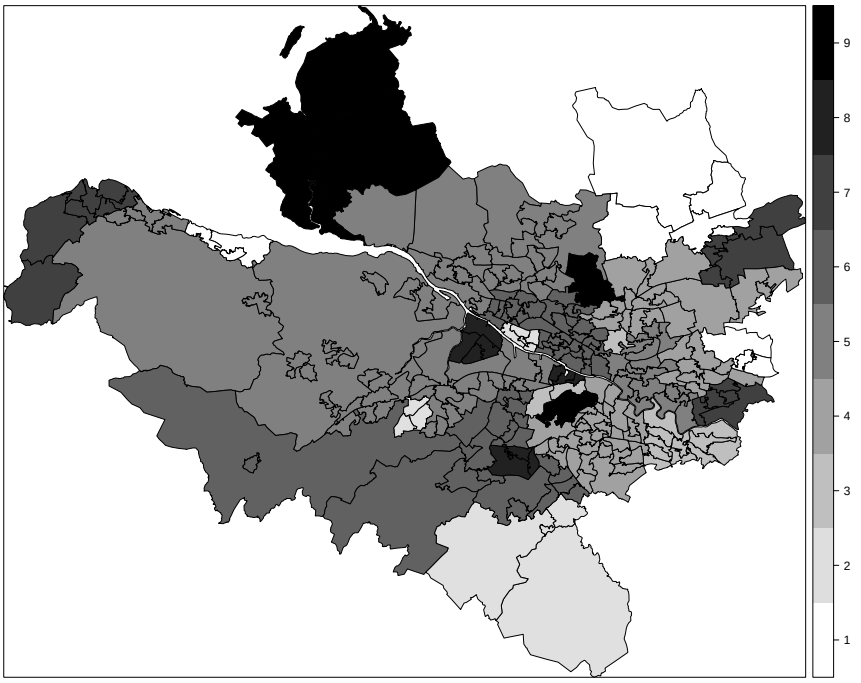

Figure 7.2: Plot of the set of combined intercept/slope clusters.

templates are displayed in Figure 7.1, with the intercept clusters in the top panel and the slope clusters in the bottom panel. Note that these clusters are not spatially contiguous. These cluster templates have been designed so that every possible combination of intercept and slope clusters has been accounted for, meaning that there are nine possible intercept/slope clusters. These nine combined clusters are shown in Figure 7.2. A set of intercept means, $\boldsymbol{\mu_C} = (\mu_{C_1}, \ldots, \mu_{C_n})$ is constructed by multiplying the intercept cluster values by a constant Z, where larger values of Z represent larger differences between the clusters. Likewise, a set of slope means, $\boldsymbol{\mu_D} = (\mu_{D_1}, \ldots, \mu_{D_n})$ is constructed by multiplying the slope cluster values by the same constant Z.

Disease data were generated for ten time points (T=10) under this template under the following model, which is similar to that in Section 4.4.

$$\begin{aligned} Y_{it}|E_{it}, R_{it} &\sim \text{Poisson}(E_{it}R_{it}) \qquad i = 1, \ldots, n, \quad t = 1, \ldots, T \\ \ln(R_{it}) &= \phi_i + \delta_i(t - \bar{t}) \\ \phi_i &\sim \text{N}(\mu_{C_i}, Q^{-1}) \\ \delta_i &\sim \text{N}(\mu_{D_i}, Q^{-1}) \end{aligned} \tag{7.3}$$

The random effects, $\boldsymbol{\phi}$ and $\boldsymbol{\delta}$ were generated from multivariate Gaussian distributions with a common spatially correlated precision matrix, given by $Q = (\text{diag}(W\mathbf{1}) - W) + \epsilon I$, which corresponds to the intrinsic CAR model with a small $\epsilon = 0.001$ added to ensure that the precision matrix is diagonally dominant and hence invertible. Here $W\mathbf{1}$ is a vector containing the number of neighbours for each areal unit and I_n is an $n \times n$ identity matrix. Clustered disease data were obtained by specifying a piecewise constant mean function for $\boldsymbol{\phi}$ and $\boldsymbol{\delta}$, which follows the templates shown in Figure 7.1. The values in Figure 7.1 are multiplied by a constant Z for both for intercept and slope, where larger values of Z represent larger differences between the clusters, which should thus be easier to identify. In this study, three scenarios are considered. Scenario one sets $Z = 1$ and corresponds to a case where there are large differences between the clusters, scenario two has $Z = 0.5$ and corresponds to a more difficult case where there are smaller differences and $Z = 0$ corresponds to a spatially smooth risk surface with no change over time where one would hope to identify a single cluster covering the entire study region. In this example, both the slope and intercept have been

multiplied by a common constant Z, but this does not have to be the case, and two separate constants, Z_1 for intercept and Z_2 for slope could be used.

Two hundred datasets were generated for each of the three data generation approaches ($Z = 0, 0.5, 1$). In this simulated example, we know the number of intercept and slope clusters, but in practice our model would be applied to data where the true number of clusters is not known. Therefore, we wish to test our model under different values of N_C and N_D to investigate how reliant the model is on the user's prior choice of the number of clusters. In each scenario, the maximum number of clusters for both intercept and slope were set to be the same value, M. As discussed in Section 7.2.1, our model may produce empty clusters, so selecting a value of M which is larger than the true number of clusters does not mean that the correct cluster structure cannot be estimated. Three scenarios were compared in this case, $M = 3, 5, 7$. Note that $M = 4, 6$ were also tested, and produced similar results, but these have been excluded for brevity. For each data generation approach and clustering scenario, our model was compared to the Bernardinelli model (Bernardinelli et al. (1995)). The Bernardinelli model does not enforce clustering on the data, so a post-hoc classification method based on mixture models (Fraley and Raftery (2002)) was used to estimate a cluster structure under this model, with a maximum of $N_C \times N_D$ clusters permitted in each case.

7.3.3 Results

The results of the study are outlined in Tables 7.1, 7.2 and 7.3, and summarised in Figures 7.3, 7.4 and 7.5, which display a comparison of the relative performances of our approach and the Bernardinelli model using three different metrics. The accuracy of the risk surfaces estimated by each approach is quantified by their root mean square error (RMSE), while the correctness of the estimated cluster structures is quantified by both the number of clusters identified and the Rand Index between the true and estimated cluster structures. The latter is a measure of the similarity between two cluster structures and lies in the interval $[0, 1]$. It is computed as the proportion of pairs of areal units classified either in the same or in different clusters by both methods, that is the proportion of pairwise agreements between the two methods. A value of 1 indicates complete agreement between the two cluster configurations and a value of 0 indicates that no pair of areal units are classified in the same way under both configurations. For more information on the Rand Index, see Section 2.6.

Figure 7.3 shows boxplots of the number of combined slope-intercept clusters estimated under each approach in the 200 simulated data sets, where the true values of 1 (when $Z = 0$) and 9 (when $Z = 0.5, 1$) are represented by dashed lines. The top panel shows that when $M = 3$, our approach performs better than the Bernardinelli model for all three values of Z. When $Z = 0$, both models obtain a median of 1 cluster, but our approach has a standard deviation of 0.26 compared to 0.71 for the Bernardinelli model. For $Z = 0.5$,

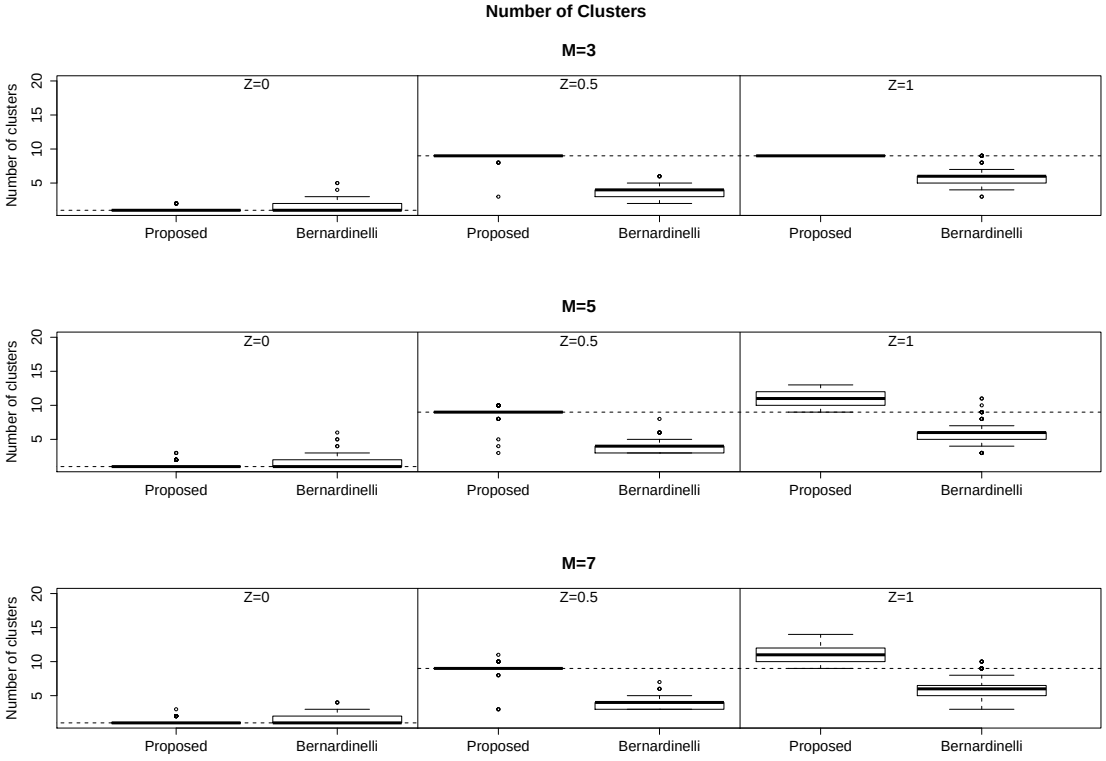

Figure 7.3: Summary of the number of clusters obtained under each approach in the simulation study. The top, middle and bottom panels display boxplots for $M = 3, M = 5$ and $M = 7$ respectively. The results relate to $Z = 0$ (left panels), $Z = 0.5$ (middle panels) and $Z = 1$ (right panels). The dashed lines represent the true number of clusters in each case.

	Mean Difference	Proposed Model	Bernardinelli
M = 3	$Z = 0$	1 (0.264)	1 (0.713)
	$Z = 0.5$	9 (0.446)	4 (0.728)
	$Z = 1$	9 (0.000)	6 (1.166)
M = 5	$Z = 0$	1 (0.393)	1 (0.808)
	$Z = 0.5$	9 (0.750)	4 (0.815)
	$Z = 1$	11 (1.343)	6 (1.375)
M = 7	$Z = 0$	1 (0.317)	1 (0.666)
	$Z = 0.5$	9 (0.795)	4 (0.701)
	$Z = 1$	11 (1.434)	6 (1.559)

Table 7.1: Number of clusters obtained under each simulation approach.

our model obtains a median of 9 clusters, while the Bernardinelli model underestimates the number of clusters, with a median of 4, and likewise for $Z = 1$, our approach obtains a median of 9, while the Bernardinelli model has a median of 6.

The middle and bottom panels, where $M = 5$ and $M = 7$ respectively, show similar results, with our model outperforming the Bernardinelli model for each value of Z. It is clear that our model is better than the Bernardinelli approach at estimating the correct number of clusters under each scenario tested, but we must also compare the Rand index values to ensure that the clusters being identified are close to the true clusters. Our model obtains very similar results under the three different values of M. The only major

difference is that for $M = 5$ and $M = 7$ our model slightly overestimates the number of clusters for $Z = 1$, with a median of 11 clusters in both cases. It is likely that this overestimation is either a result of the model partitioning a true high or low intercept or slope cluster into more than one group, or by the model placing a single outlier in a cluster on its own. The effect of this overestimation of clusters can be identified by comparing the RMSE for each value of M. It should be noted that it would be impossible to overestimate the number of clusters when $M = 3$, because the maximum number of clusters allowed by the model is 9.

Figure 7.4 displays the Rand index values obtained under each approach. The top panel shows that when $M = 3$, our model performs better than the Bernardinelli approach in each case. For $Z = 0$, both models have a median Rand index of 1, but our approach has a much lower standard deviation of 0.005 compared to 0.20 for the Bernardinelli approach. The Bernardinelli model performs very poorly in some cases under $Z = 0$, with a Rand index as low as 0.22 obtained in one case; this makes the model unreliable for clustering, because it produces false positives. If this approach was applied to data and clustering was identified, then the possibility of a false positive would lead to doubt over the veracity of the results. For $Z = 0.5$, our model obtains a median Rand index of 0.76, while the Bernardinelli model obtains a median of 0.63. For $Z = 1$, both models have very high Rand index values; the median for our model is 0.91 compared to 0.92 for the Bernardinelli model. However, our model performs more consistently, with a standard deviation of 0.06 compared with 0.16 for the Bernardinelli model.

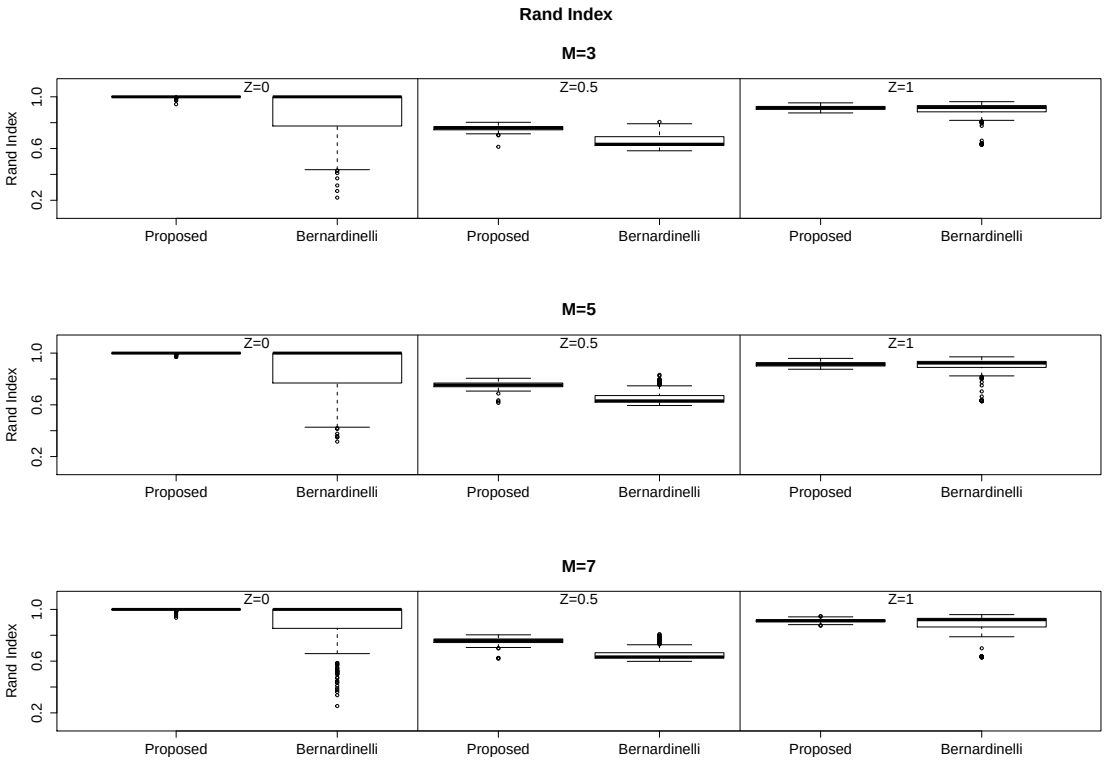

Figure 7.4: Summary of the Rand index obtained under each approach in the simulation study. The top, middle and bottom panels display boxplots for $M = 3, M = 5$ and $M = 7$ respectively. The results relate to $Z = 0$ (left panels), $Z = 0.5$ (middle panels) and $Z = 1$ (right panels).

	Mean Difference	Proposed Model	Bernardinelli
M = 3	$Z = 0$	1 (0.005)	1 (0.202)
	$Z = 0.5$	0.758 (0.021)	0.633 (0.060)
	$Z = 1$	0.914 (0.016)	0.917 (0.092)
M = 5	$Z = 0$	1 (0.005)	1 (0.196)
	$Z = 0.5$	0.753 (0.025)	0.631 (0.060)
	$Z = 1$	0.913 (0.017)	0.923 (0.089)
M = 7	$Z = 0$	1 (0.007)	1 (0.193)
	$Z = 0.5$	0.759 (0.025)	0.633 (0.052)
	$Z = 1$	0.913 (0.013)	0.920 (0.101)

Table 7.2: Rand index scores obtained under each simulation approach.

Similar results are obtained when $M = 5$ and $M = 7$, which indicates that our approach is preferable to the Bernardinelli model in each case investigated here. The results obtained for our model are consistent across all three values of M. When $Z = 0$ our model obtains medians of 1 in each case and standard deviations of 0.006, 0.005 and 0.007 for $M = 3, M = 5$ and $M = 7$ respectively. For $Z = 0.5$ we obtain medians of 0.75, 0.76 and 0.75 and standard deviations of 0.021, 0.025 and 0.025 for $M = 3, M = 5$ and $M = 7$ respectively, and for $Z = 1$ medians of 0.91 are obtained for each case, with medians of 0.016, 0.017 and 0.013 for $M = 3, M = 5$ and $M = 7$ respectively. These results indicate that the choice of the maximum number of clusters allowed does not affect the accuracy of the clusters identified within the model.

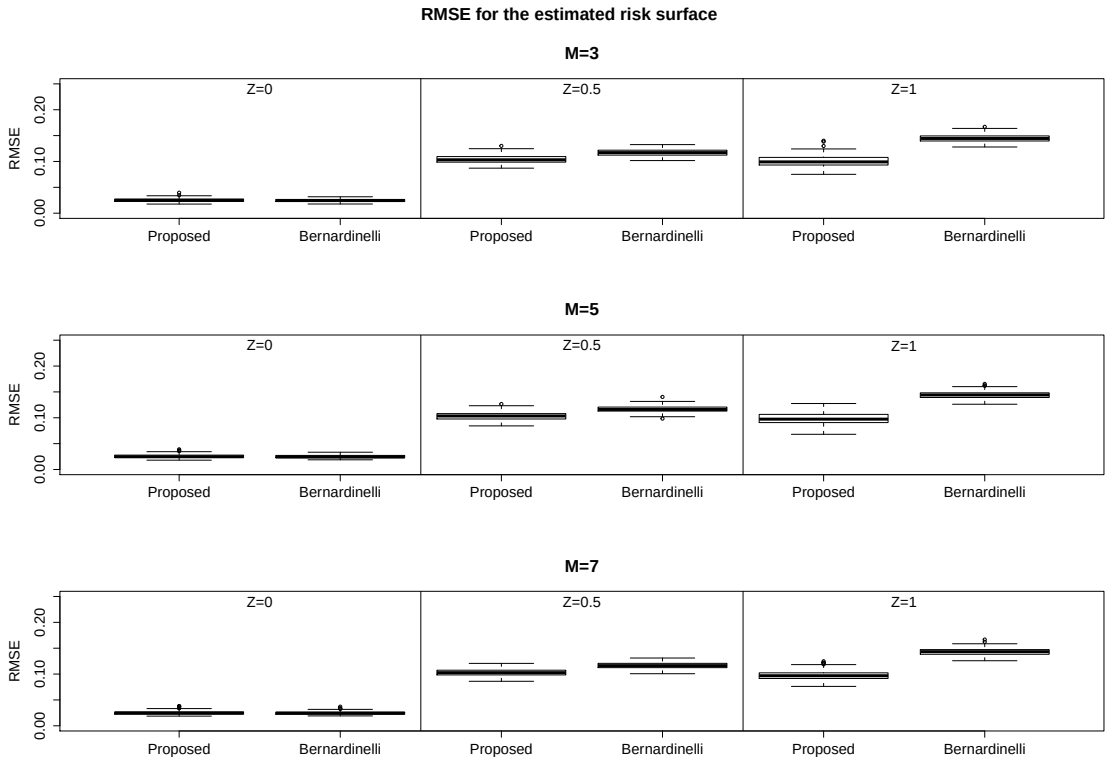

Figure 7.5: Summary of the RMSE for the estimated risk surface obtained under each approach in the simulation study. The top, middle and bottom panels display boxplots for $M = 3, M = 5$ and $M = 7$ respectively. The results relate to $Z = 0$ (left panels), $Z = 0.5$ (middle panels) and $Z = 1$ (right panels).

	Mean Difference	Proposed Model	Bernardinelli
M = 3	$Z = 0$	0.025 (0.003)	0.024 (0.003)
	$Z = 0.5$	0.103 (0.008)	0.117 (0.006)
	$Z = 1$	0.099 (0.011)	0.144 (0.007)
M = 5	$Z = 0$	0.025 (0.004)	0.024 (0.003)
	$Z = 0.5$	0.103 (0.008)	0.116 (0.006)
	$Z = 1$	0.097 (0.011)	0.144 (0.007)
M = 7	$Z = 0$	0.025 (0.007)	0.024 (0.193)
	$Z = 0.5$	0.103 (0.025)	0.116 (0.052)
	$Z = 1$	0.097 (0.013)	0.143 (0.101)

Table 7.3: RMSE obtained under each simulation approach.

Figure 7.5 displays boxplots of the root mean square error of the estimated risk surface obtained under each approach in the simulation study. The top panel shows that when $M = 3$, our model provides more accurate risk estimates than the Bernardinelli approach. For $Z = 0$, both approaches have very similar results; our approach has a median of 0.025 compared to 0.024 for the Bernardinelli model, while both approaches have a standard deviation of 0.003. When $Z = 0.5$, our approach has a median RMSE of 0.103 compared with 0.117 for the Bernardinelli model, and for $Z = 1$ a median of 0.099 is obtained for our proposed model compared with 0.144 for the Bernardinelli model. We can see that both models produce similarly accurate risk estimates when the risk surface is smooth and constant over time ($Z = 0$), but our model performs better in cases where clusters exist. This is unsurprising, since our model allows for different fixed effects for each cluster,

while the Bernardinelli approach has two fixed effects (one for intercept and one for slope) which are common to all areas across different clusters.

The results obtained for our model are consistent across all three values of M. For $Z = 0$, the median RMSE is 0.025 in all three cases, with standard deviations of 0.003, 0.004 and 0.004 for $M = 3, M = 5$ and $M = 7$ respectively. When $Z = 0.5$, a median RMSE of 0.103 is obtained for all three values of M, with standard deviations of 0.008, 0.008 and 0.007 for $M = 3, M = 5$ and $M = 7$ respectively, and for $Z = 1$, medians of 0.099, 0.097 and 0.097, and standard deviations of 0.011, 0.011 and 0.009 are obtained for $M = 3, M = 5$ and $M = 7$ respectively. These results indicate that the choice of the maximum number of clusters allowed does not affect the accuracy of the model's risk estimates.

7.4 Application to real data

This section continues the analysis of the respiratory hospitalisation risk data presented in Chapters 5 and 6. Here, we use our spatio-temporal clustering model to analyse the rate of change in the disease risk pattern over a ten year period between 2002 and 2011 in the Greater Glasgow and Clyde Health Board region. As in the previous chapter, the study region is the Greater Glasgow and Clyde Health Board area, and we use the respiratory admission data introduced in Section 4.5. Figure 4.4 contains a map of Glasgow and the surrounding areas, with pins in the map to identify each location which

is mentioned in this thesis. Table 4.2 provides a key for this map, with the numbers in the table corresponding to those in the pins in Figure 4.4.

The response data, $\boldsymbol{Y} = (\boldsymbol{Y_1}, \ldots, \boldsymbol{Y_n})$, are based on the data from 2002-2011, where $\boldsymbol{Y_i} = (Y_{i1}, \ldots, Y_{it})$ and Y_{it} represents the number hospital admissions with a primary diagnosis of respiratory disease in areal unit i in year t. The expected values, $\boldsymbol{E} = (\boldsymbol{E_1}, \ldots, \boldsymbol{E_n})$, are the expected hospital admission numbers for each areal unit and year. Values of $N_C = N_D = 5$ were chosen in order to allow for a possible distinction between high (and low) risk and extremely high (and low) risk intercept clusters, and also for different levels of increasing and decreasing clusters in each direction.

The standardised incidence ratios (SIRs) for 2002 and 2011, the first and last years of the study period, are displayed in Figure 7.6. We can see that for most areas, the SIR has not changed dramatically over this ten year period. However, there are some areas where the disease risk appears to have increased, such as rural Dunbartonshire to the far north of the map. There are also a number of areas where the disease risk appears to have decreased over the study period, such as Wemyss Bay to the far west. In addition, simple linear models were fitted for each areal unit in turn in order to provide an estimate of the potential patterns in slope and intercept. The top panel of Figure 7.7 displays the linear model intercepts for each areal unit, which can be interpreted as the average disease risk over the study period, and this follows a similar pattern to the SIR plots. The bottom panel of Figure 7.7 displays the linear model slopes for each areal unit. Here, we can identify

Figure 7.6: Plot of SIR values in the first and last years of the study period. The top panel shows the SIR in 2002, while the bottom panel shows the SIR in 2011.

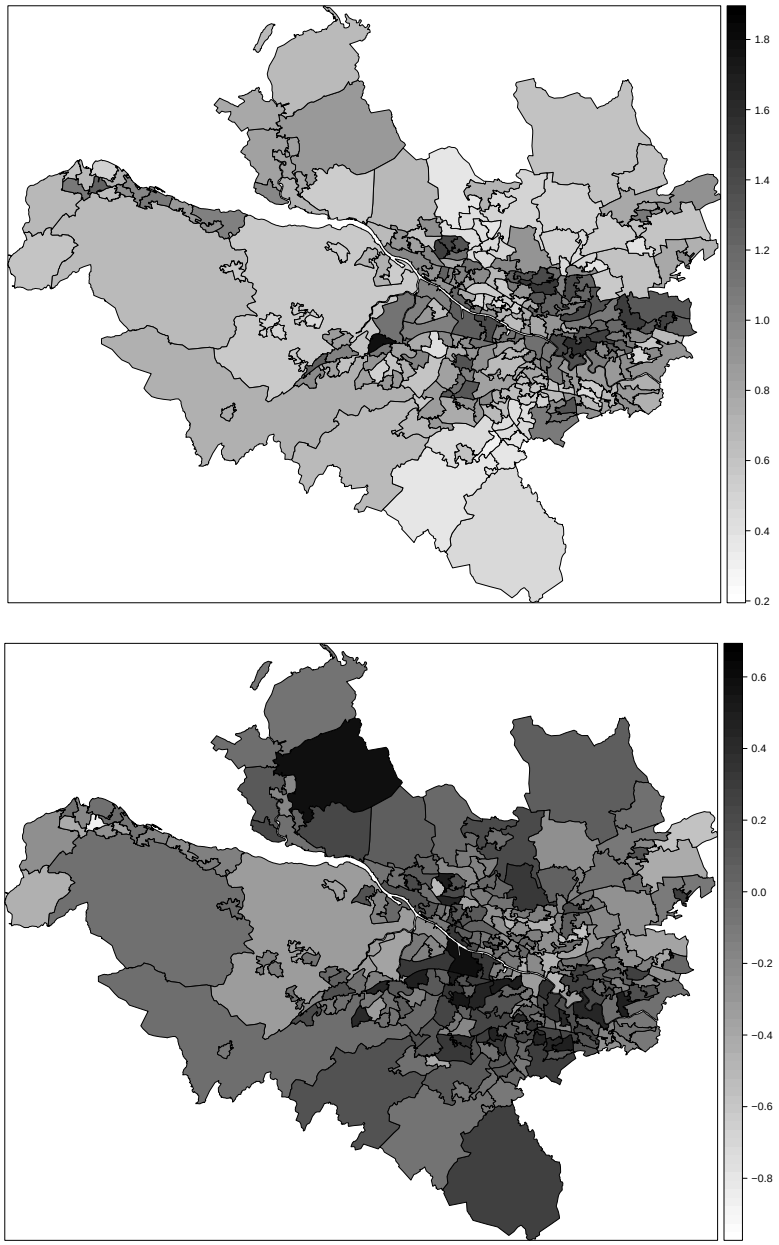

Figure 7.7: Plot of intercepts and slopes for a simple linear model for each areal unit. The top panel shows the linear model intercepts, while the bottom panel displays the fitted slopes.

areas where the disease risk appears to be increasing over the study period, such as rural Dunbartonshire in the north, and Govan in the centre of the map, to the south of the river. The disease risk appears to be decreasing in a number of areal units to the east of the city, including Stepps to the extreme east of the map.

The model was fitted to the data, and five intercept clusters and three slope clusters were identified. Not every combination of intercept and slope clusters was obtained, overall there were 14 different intercept-slope cluster pairings. Figure 7.8 graphically displays the risk values estimated by the model for each areal unit within each possible cluster combination, and shows that there are no areal units in Cluster 3. Each column represents a different intercept cluster, with the intercept term increasing as you move from left to right. Each row represents a different slope cluster; the top row contains the cluster with increasing risk, the middle row contains the cluster with little or no change, and the bottom row contains the cluster with decreasing risk. Here we can see that the model ensures that areal units in the same cluster have similar risks over the study period, but does still allow for some variation in risk levels within a cluster.

The top panel of Figure 7.9 displays the estimated intercept values for each areal unit (i.e the estimated overall average risk), given by $\{\alpha_{C_i} + \phi_i\}$, with high intercept values corresponding to high average risk. Many of the areas with high intercepts are areal units which were identified as having a high disease risk in Chapters 5 and 6, which is what would be expected given that

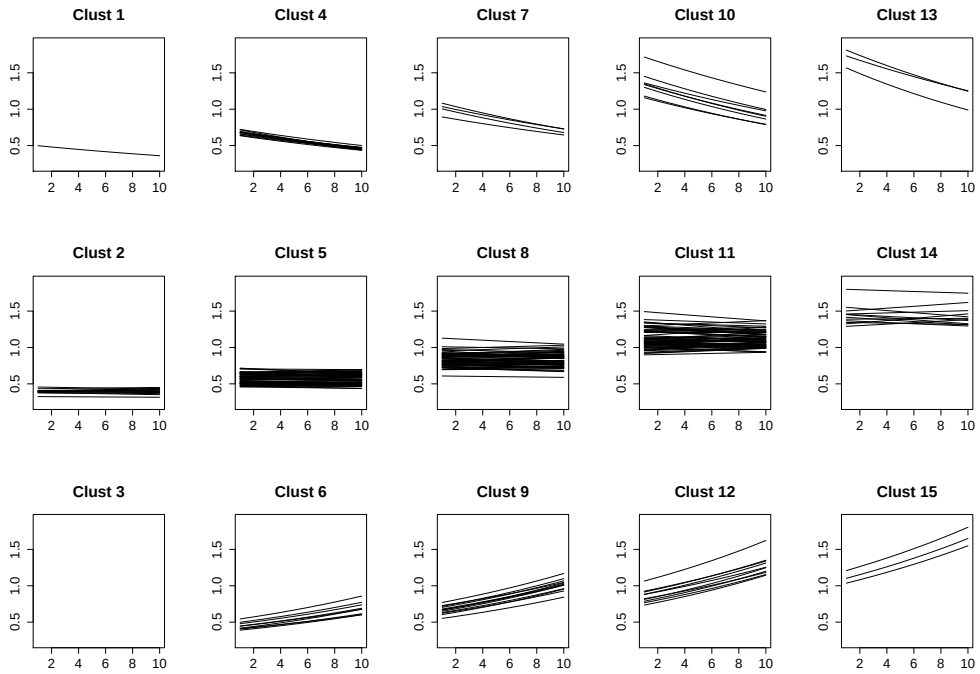

Figure 7.8: Plot of the estimated risks from the model for each intercept and slope cluster. The intercept clusters are represented by the columns, with the intercept increasing from left to right. The slope clusters are represented by the rows, with the top row containing areas of increasing risk, the middle row containing areas where there was little or no change in risk, and the bottom row containing areas with decreasing risk. The number of lines in each plot corresponds to the number of areal units in that combination of intercept and slope clusters.

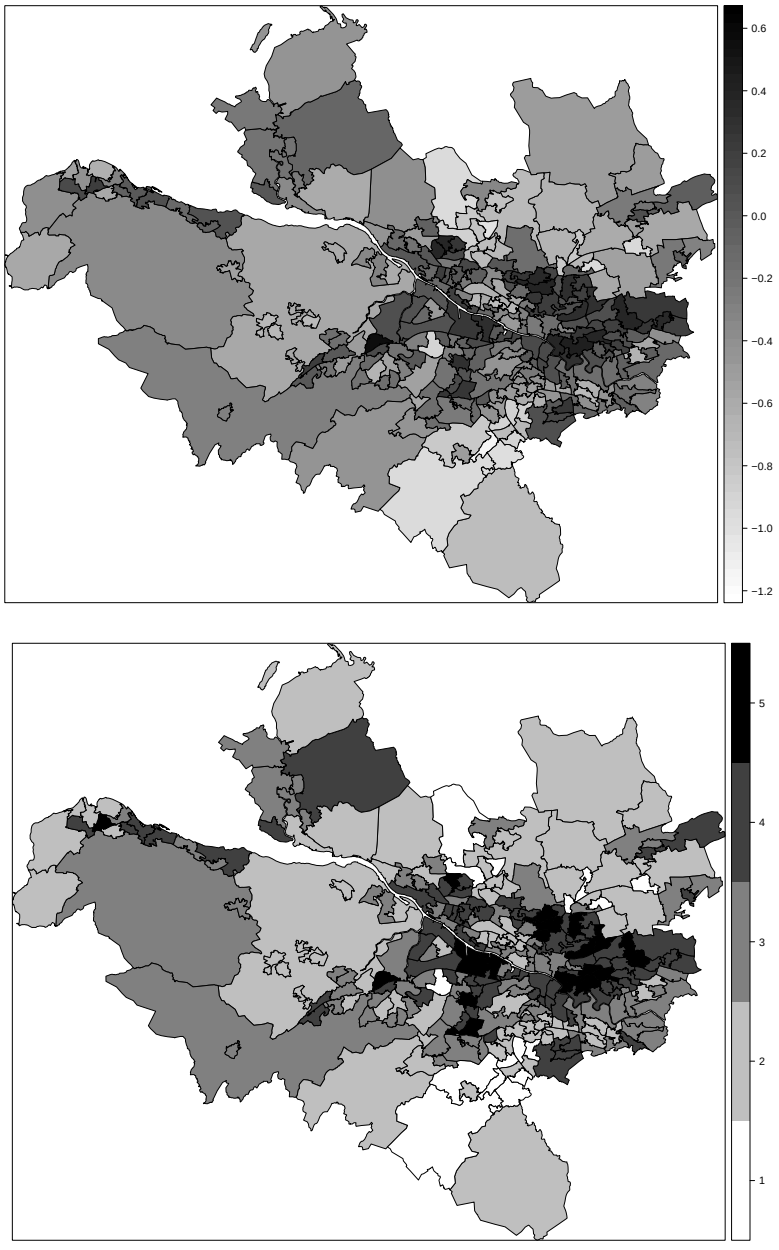

Figure 7.9: The top panel shows the estimated model intercept values ($\alpha_{C_i} + \phi_i$), while the bottom panel displays the estimated intercept clusters.

a large intercept corresponds to high average risk. The model identifies a number of high risk areas in the east end of the city, which is known to be one of the most deprived parts of Glasgow. Further west, Drumry, another area with high deprivation, is again picked out as being at much higher risk than neighbouring areas to the north such as Milngavie. The affluent West End area of the city is once more picked out as being at low risk. The bottom panel of Figure 7.9 displays the estimated intercept clusters from the model, with the dark shading corresponding to higher risk clusters. The very high risk cluster in black picks out a number of deprived areas, including Govan in the centre of the map just south of the river, Drumry to the north west of the city and Easterhouse to the east. In contrast, the very low risk cluster in white picks out some of the most affluent areas of the city, including Bearsden to the north and Newton Mearns to the south. A comparison of the two maps in Figure 7.9 shows that the areal units within a cluster have similar risk due to the cluster specific $\boldsymbol{\alpha}$ term, but that the random effects, $\boldsymbol{\phi}$ still allow for differences between areal units in the same cluster.

The top panel of Figure 7.10 displays the estimated slope values for each areal unit, given by $\{\beta_{D_i} + \delta_i\}$, with positive slopes corresponding to increasing risk and negative slopes corresponding to decreasing risk. The model identifies a number of areas with large increases in disease risk over the study period, including rural areas of Dunbartonshire to the extreme north west and Eaglesham to the extreme south east. It also picks out a few areas such Stepps to the north east and Wemyss Bay to the extreme west as having a substantial decrease in disease risk over time. The bottom panel of Fig-

ure 7.10 displays the three slope clusters identified by the model, with the black cluster corresponding to an increase in disease risk, the grey cluster corresponding to little or no change and the white cluster corresponding to a decrease in risk. The majority of areal units lie in the grey cluster, which suggests that the level of disease risk has remained fairly stationary for most of the study region. This should not be surprising given the reasonably short period of time being studied; most areas will not have undergone any sort of substantial changes in that period. The main interest lies in the 34 areas which have exhibited an increase and the 27 areas which have exhibited a decrease in disease risk over the study period. Further investigation could be carried out by health authorities to ascertain potential causes for these changes, either in terms of physical changes to the environment or a difference in population behaviour over the study period.

In Chapters 5 and 6, the fitted values of the model were displayed graphically on a single plot, however this is not possible for this model given that there are ten different time points, each of which has its own set of fitted values. In order to compare the evolution of disease risk over the study period, Figure 7.11 displays the fitted risks from the model for 2002 and 2011, the first and last years of the study period. The two maps are fairly similar, which again suggests that there has been little change in the risk pattern across the study period. A notable difference between the two maps lies in the east end of the city, where it appears that the disease risk may have decreased slightly across the ten year period. The rural areas of Renfrewshire to the south east of the map also appear to have lower risk in 2011 than in 2002. However, the

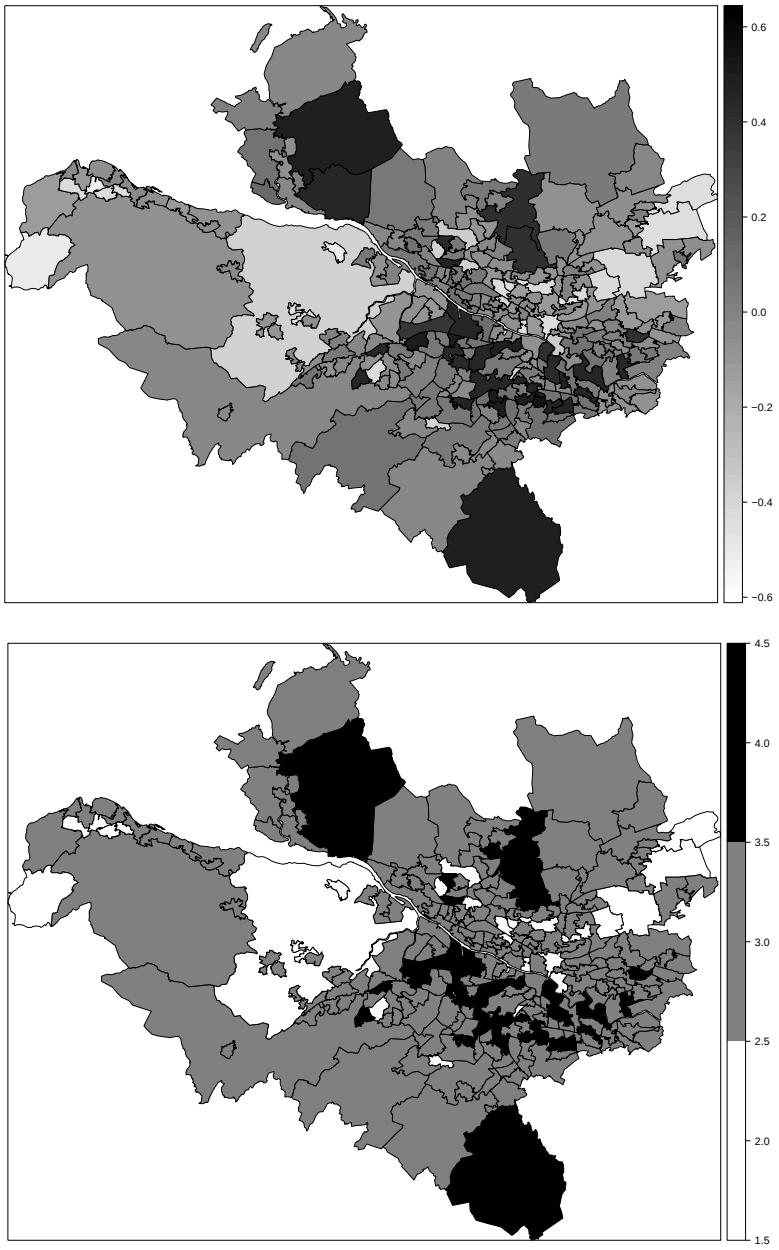

Figure 7.10: The top panel shows the estimated model slope values ($\beta_{D_i} + \delta_i$), while the bottom panel displays the estimated slope clusters.

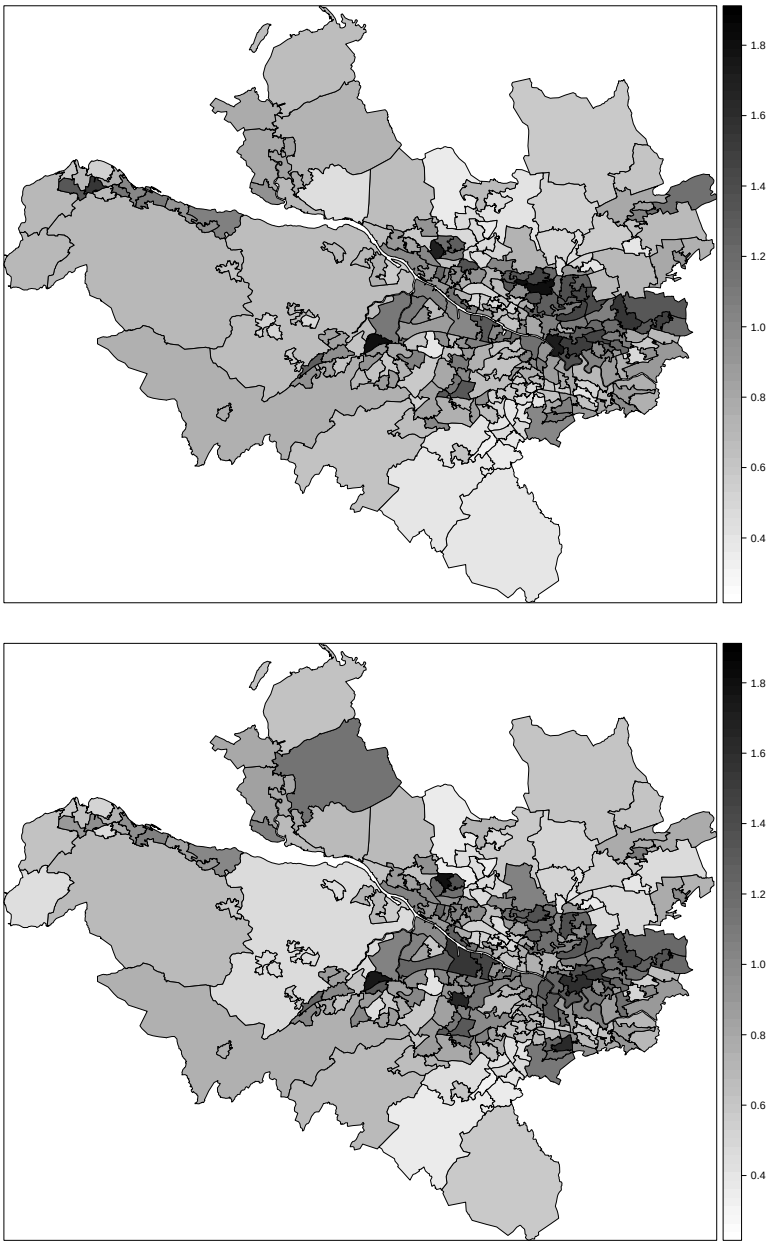

Figure 7.11: The top panel shows the estimated respiratory disease risks in 2002, while the bottom panel shows the estimated respiratory disease risks in 2011.

risk in Dunbartonshire to the extreme north east of the map appears to be increasing, as was identified by the slope clusters. Again, it may be of interest to carry out an investigation into the potential causes of these differences.

It is straightforward to combine the intercept and slope clusters within this model to group together areal units which have similar characteristics. This may improve efficiency within medical applications by allowing the same service to be delivered to all areas within the cluster. Combined intercept-slope clusters can easily be obtained within this model, and Figure 7.12 displays the combined clusters for respiratory disease risk in Glasgow. The top panel displays the clusters on the map while the bottom panel provides a reminder of the visual representation of the slope and intercept for each cluster, previously outlined in Figure 7.8. The most concerning cluster for health authorities will be cluster 15 (the black cluster), which contains areal units which have a very high disease risk which is increasing over time. This cluster contains areas such as Drumry to the north of the city and Govan to the south which are known to have high levels of deprivation. An investigation into what these areal units have in common may lead to the identification of possible risk factors for respiratory disease. The set of clusters in the top panel of Figure 7.12 also illustrates why spatial contiguity is not enforced in this approach; further partitioning these 14 clusters to enforce spatial contiguity would lead to a large number of very small clusters being identified, which is not suitable for the aims of the clustering approach. However, it should be noted that there is still a high degree of spatial contiguity in these clusters, particularly within Clusters 5 and 8, which are coloured dark green and or-

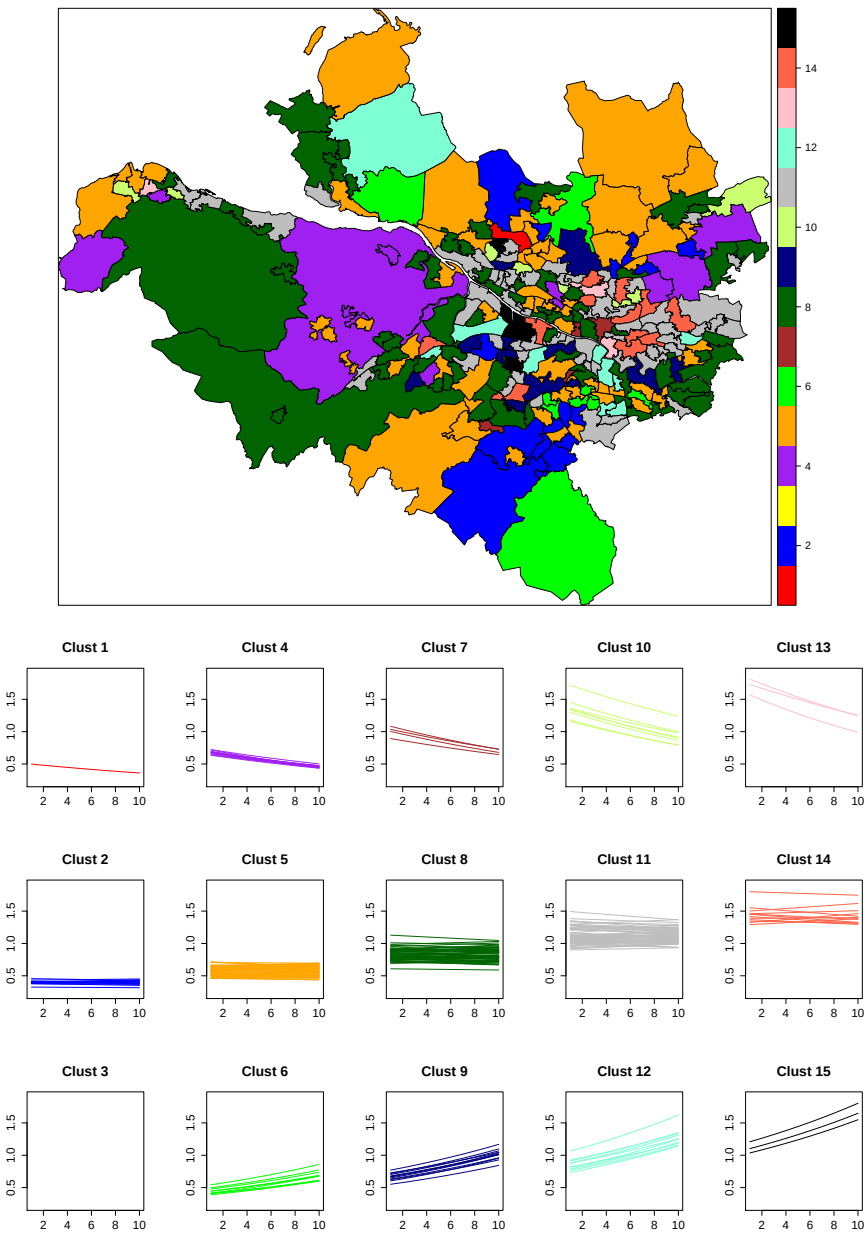

Figure 7.12: The top panel displays the combined intercept-slope clusters, while the bottom panel provides a visual representation of the characteristics of each cluster. The number of lines in each plot in the bottom panel corresponds to the number of areal units in that combination of intercept and slope clusters.

ange respectively. This should not be unexpected, because it is still the case in most parts of the region that nearby areas will share similar characteristics.

7.5 Discussion

Here we have proposed a Bayesian spatio-temporal model which estimates the disease risk pattern over multiple time points and also identifies clusters of areas which have a similar disease risk characteristics over the study period. There are two separate clustering parameters within the model; the first is based on the average risk (intercept) and the second is based on the change in disease risk over time (slope). Unlike in the previous chapters, spatially contiguity is not forced on the clusters because doing so may lead to excessive numbers of clusters. The model proposed here extends the Bernardinelli model (Bernardinelli et al. (1995)) by allowing different intercept and slope terms for each cluster. The model estimates disease risk via four parameters, a pair to estimate the intercept and a pair to estimate the slope. Each pair consists of a set of cluster-specific fixed effect terms and a set of spatially correlated random effects which follow a conditional autoregressive model. The fixed effects ensure that areal units within the same cluster will have similar intercept levels, but the random effects allow for some variation in intercept levels within a cluster.

The simulation study presented in Section 7.3 showed that our model outperforms the Bernardinelli model with post-hoc clustering on across a variety

of simulation scenarios. Our model was more accurate than the Bernardinelli model in terms of estimating the correct number of clusters, and also identified more accurate clusters as measured by the Rand index. The risk estimates from our model were also more accurate than those obtained from the Bernardinelli model. This improved estimation is a result of the additional fixed effect terms within our model; the Bernardinelli model has two fixed effects (one for intercept and one for slope) which are common to all areas across different clusters while our model allows for different fixed effects for each cluster. The simulation study also showed that the performance of our model is not affected by the choice of N_C and N_D, the maximum number of clusters for intercept and slope respectively. Based on this result, it is our recommendation that the values of N_C and N_D are chosen to be a slightly larger than the number of clusters expected for intercept and slope respectively.

It is straightforward to combine model clusters to produce intercept-slope clusters containing areal units which have similar levels of average disease risk and similar changes in risk over time. This allows health authorities to identify groups of areal units with similar risk values across the entire study period, which has two important uses. Firstly, the clusters can be used to determine policy across the region; similar levels of resources can be allocated to areal units in the same cluster. Secondly, there may be interest in identifying factors which may be causing increased disease risk; for example the areal units in a cluster with increasing disease risk could be compared to identify possible common changes in these areas which could have caused

the increase in risk.

This model provides an extension of the spatial methodology introduced in Chapters 5 and 6 into the spatio-temporal domain. Both of those approaches modelled disease risk and identified clusters via a two-stage approach, but here we can do both simultaneously in a single model. As shown in the simulation study, this method represents an improvement on the Bernardinelli model (Bernardinelli et al. (1995)) and is more straightforward to implement than existing spatio-temporal clustering models such as Knorr-Held and Rasser (2000), which requires complex reversible-jump McMC algorithms to identify the clusters. The existing approaches outlined in Section 3.5 all assume that the disease risk is constant within a cluster, but the approach proposed here allows disease risk to vary within a cluster via the random effects. Such variation is likely to exist in real datasets, and therefore the approach proposed here represents a more realistic alternative to the existing models.

This model currently has two separate clustering terms, one set for the intercept and the other for the slope, although it is straightforward to combine these. Nonetheless, it may be of interest to extend the model to include a single slope-intercept cluster term which can partition the areal units based on both characteristics rather than separately. This could be implemented within a similar modelling structure by allowing each intercept-slope cluster to have its own separate intercept and slope fixed effects. This would mean that the intercept-slope interactions were taken into account when estimat-

ing disease risk instead of forming these clusters by a post-hoc combination of intercept and slope clusters as is the case here. A possible challenge in such an approach would be avoiding overparameterisation as a result of the increased number of fixed effects. This model could also be extended by developing a reversible-jump McMC algorithm to allow the number of clusters to be shaped by the data, rather than relying on a user-defined maximum. This would allow the possibility of an additional cluster being formed, or two clusters being joined together, at each stage of the McMC algorithm. It would also be of interest to extend this model to allow for a non-linear trend over time, which would enable the approach to be applied to data where there is a more complex temporal trend.

Chapter 8

Conclusion

This thesis focused on identifying spatial patterns in disease data, an issue which has important public health implications. Such approaches generally partition the study region into a set of non-overlapping areal units, and then estimate the disease risk for the population living in each of these areal units. These disease risks can then be presented on a colour-coded disease map in order to provide a visual representation of the spatial risk pattern. Disease mapping approaches are most commonly based on conditional autoregressive (CAR) models, which assume spatial autocorrelation between pairs of adjacent areal units. The majority of CAR models are based on a constant level of spatial smoothness, but there are many cases where this would not be sensible. As a result, there has been recent interest in developing CAR models which allow for discontinuities in the spatial risk pattern, some of which has focused on partitioning the study region into clusters. The main aim of this thesis was to develop new spatial and spatio-temporal methodology

which can simultaneously identify disease clusters and estimate disease risk. .

The existing approaches to tackle these challenges were introduced in Chapters 2 and 3 of this thesis. Chapter 2 introduced statistical methods which are utilised in our new methodology, including Bayesian inference, generalised linear models, clustering and CAR models. An introduction to disease mapping and a critique of existing disease mapping approaches was then outlined in Chapter 3. Chapter 4 outlined a new spatial agglomerative hierarchical clustering algorithm which is used to identify a set of potential cluster structures for partitioning the areal units into spatially contiguous clusters. This algorithm formed the first step of a two stage Bayesian modelling approach, and two different approaches for Stage 2 of the model were introduced in Chapters 5 and 6. Chapter 7 proposed a spatio-temporal modelling approach for identifying clusters based on both the disease risk level and the changes in disease risk over time. Each of these new modelling approaches were tested on simulated data and then applied to respiratory hospital admission data for the Greater Glasgow and Clyde Health Board area.

8.1 Clustering Algorithm

The new spatial hierarchical agglomerative clustering algorithm outlined in Chapter 4 produces a set of n cluster structures, and unlike regular hierarchical clustering, these structures partition the study region into *spatially contiguous* clusters with similar disease risk. These structures are produced

based on disease risk data from a set of time periods prior to the study period, thus allowing the observed data from the study period to be used to estimate risk in the Poisson log-linear model. Such prior data are available for most disease mapping examples, but if such data do not exist then an alternative set of data would be necessary to estimate the cluster structure. A sensible approach in this case would be to use covariate data which has strong correlation with disease risk. The agglomerative nature of the clustering algorithm provides a natural ordering of these cluster structures, with the first structure containing one cluster and the last structure containing n clusters. The algorithm was tested on a set of simulated data with a known cluster structure, and was successful in identifying either the correct structure or a structure very close to the true structure, which indicates that the sets of clusterings obtained will contain sensible cluster structures for the data. The set of cluster structures obtained from this algorithm can then be compared using a Poisson log-linear model, and two such models were introduced in Chapters 5 and 6.

8.2 Fixed Effect Model

Chapter 5 outlined a Bayesian hierarchical model for selecting the optimal cluster structure from the set obtained from the spatial hierarchical agglomerative clustering algorithm. This model represents disease risk by fusing together a spatially smooth intrinsic CAR model and a piecewise constant cluster model. Different mean risk levels are assigned to each cluster via a fixed effect term and the risk in each area is therefore based on a combination

of this cluster-specific mean term and an area-specific random effect which allows correlation between neighbouring areal units within the same cluster. This approach allows disease risk to evolve smoothly within a cluster whilst having a disjoint jump between clusters. This model is applied with each of the potential cluster structures in turn, and the cluster structure which produces the lowest Deviance Information Criterion (DIC) is selected as being the optimal cluster structure. A simulation study was carried out to compare this model to the BYM model (Besag et al. (1991)) with a posterior classification step, and this proposed model performed better in terms of both risk estimation and cluster identification. The model was then applied to respiratory hospital admission data for the Greater Glasgow and Clyde Health Board area and was able to identify a number of high risk disease clusters. This model always identifies a single optimal cluster structure for the data, something which is not necessarily the case for existing approaches which may select different cluster structures at each iteration of an McMC algorithm. This has the advantage of making our approach straightforward to implement and understandable for non-specialist users, but it does also have the disadvantage of not being able quantify the uncertainty surrounding the choice of cluster structure.

8.3 Random Effect Model

An alternative modelling approach which does allow for uncertainty in the choice of cluster structure was introduced in Chapter 6. This approach accounts for spatial disease risk clusters by modelling the correlation structure

in the random effects rather than via mean level fixed effects as was the case in Chapter 5. Here, we take advantage of the natural ordering present in the set of cluster structures identified by the spatial hierarchical agglomerative clustering algorithm, by considering the number of clusters as a univariate parameter within a Bayesian hierarchical model. This model estimates the cluster structure directly via the random effects by only allowing correlation between neighbouring values when both areas lie in the same cluster. If two adjacent areal units are not in the same cluster then no spatial autocorrelation is enforced between the random effects and the estimated risks in these areas are not smoothed towards each other. A simulation study was carried out to compare this model to the BYM model with a posterior classification step, and to the Bayesian model introduced in Chapter 5. The random effects model outperformed the BYM approach in terms of both risk estimation and cluster identification, and performs well in certain cases compared to the fixed effects model. In terms of identifying the correct number of clusters, the random effects model performs best in cases where there are true clusters present, while the fixed effects model performs better in the case where there is a completely spatially smooth surface. The random effects model does not, however, perform as well as the fixed effects approach in terms of estimating risk in the cases where true clusters are present; this is because the fixed effects model has extra parameters which can account for the differences in mean, while the random effects approach accounts for clusters in the correlation structure of the random effects.

8.4 Comparison of Spatial Models

Each of the two methods has advantages over the other in certain sets of circumstances, and both methods are preferable to the existing approaches used for cluster identification. The random effects approach has the advantage of allowing us to quantify uncertainty in the selected cluster structure, and allows estimation within a single model rather than requiring comparison of multiple models. However, the fixed effects model has additional parameters which can control the means of the clusters, and can therefore often provide better estimation of risk. The choice between these two methods should be made based on the purpose of the disease mapping study. The fixed effects approach is likely to perform better if the estimation of the disease risk is the key aim, with the clustering only being introduced to account for the correct spatial autocorrelation surface. However, the random effects approach is likely to be more appropriate if the identification of the cluster structure is the main aim of the analysis; for example this may apply to a health authority who would like to pick out clusters of high risk areas which require further investment. Both methods obtain similar disease risk patterns when applied to the Glasgow respiratory admission data, suggesting that the differences between the methods in terms of estimation are not substantial enough to affect the overall conclusions about disease risk.

8.5 Spatio-temporal Model

Chapter 7 introduced a spatio-temporal modelling approach for identifying changes in the spatial pattern of disease risk over time. A novel spatio-temporal Bayesian modelling approach is proposed to partition the areal units into clusters based on both their average risk (intercept) and the change in their disease risk over time (slope). This model estimates disease risk via four parameters, a pair to estimate the intercept and a pair to estimate the slope. Each pair consists of a set of cluster-specific fixed effect terms and a set of spatially correlated random effects which follow a conditional autoregressive model. This approach allows both the intercept and slope to evolve smoothly within a cluster, whilst allowing for a disjoint jump between clusters. Although the model contains two separate clustering terms, it is straightforward to combine these together in order to produce a set of clusters based on intercept and slope together. The simulation study presented in Section 7.3 showed that our model outperforms the Bernardinelli model (Bernardinelli et al. (1995)) with post-hoc clustering. Our model performed better than the Bernardinelli model in terms of estimating the correct number of clusters, and also identified more accurate clusters as measured by the Rand index. The risk estimates from our model were also better than those obtained from the Bernardinelli model. This improved estimation is a result of the additional fixed effect terms within our model; the Bernardinelli model has two fixed effects (one for intercept and one for slope) which are common to all areas across different clusters while our model allows for different fixed effects for each cluster.

8.6 Applications to Greater Glasgow and Clyde respiratory hospital admission data

Both of the spatial Bayesian models were applied to respiratory hospital admission data for the Greater Glasgow and Clyde Health Board area for 2011 in order to identify possible clusters in the level of respiratory disease within the area. The fixed effects model in Chapter 5 identified a final cluster structure containing 33 clusters, while the random effects model in Chapter 6 favoured a configuration with 18 clusters. However, it should be noted that due to the agglomerative nature of the clustering algorithm, these structures are not too dissimilar, with the 18 clusters identified in the random effects approach being formed by combining some of the 33 clusters identified in the fixed effects approach. Both approaches picked out a low risk cluster in the West End of Glasgow, one of the more affluent areas of the city. Similarly, a low risk cluster was identified to the north east of the city, containing areas such as Bearsden, Milngavie and Lennoxtown which are also prosperous parts of the city. The two approaches also identified a high risk cluster containing Easterhouse in the east and Springburn and Summerston in the north of the city, which are amongst the most deprived neighbourhoods of Glasgow. Under the random effects approach (which has fewer clusters), this cluster extends further east to include other deprived areas such as Drumchapel and Maryhill, while this region is included in three additional high risk clusters under the fixed effects model. Many of the differences between the models were of the same form; the fixed effects approach estimated a number of high and moderately high risk clusters to the south and east of the city,

while the random effects approach identified a single high risk cluster which extends to the very east of the study region. These sets of clusters suggest that the fixed effects model is slightly overestimating the number of clusters, and that the risk surface can adequately be described by the smaller number of clusters identified in the random effects approach. The smaller number of clusters identified by random effects approach may be more appealing to health authorities because it may be easier to focus in on clusters which need specific attention. However, it should be noted that the fixed effects approach also has some appealing features; it is more likely to identify small or even singleton clusters which may be of particular interest, such as the cluster containing Drumchapel in the Glasgow example. It may therefore be prudent to apply both approaches to the data in order to provide a comparison of potential cluster structures.

Both approaches produced very similar disease risk surfaces, with the same areas being identified as high and low risk in both plots. The areas to the south and west appear to have slightly higher estimated risks under the random effects approach, and there appear to also be some higher risks observed to the east of the city, but overall it appears that both models are estimating similar disease risk patterns. The main driver of this pattern of disease risk is socio-economic deprivation, which is well known to have a large effect on population health. The high-risk areas typically exhibit high levels of socio-economic deprivation, where as low-risk areas are more affluent. Deprivation could be accounted for by including a covariate in the regression model, but although it would allow the spatial pattern in respiratory disease risk to be

explained, the spatial extent of the high-risk clusters based on this covariate information could not be identified with this approach.

The spatio-temporal model proposed in Chapter 7 was applied to annual respiratory hospital admission data for the Greater Glasgow and Clyde Health Board area for the ten years from 2002 to 2011 in order to identify changes in the spatial disease risk pattern over time. The final model identified five intercept clusters and three slope clusters which were combined to make 14 different intercept-slope clusters. Unlike in the previous chapters, the clusters obtained were not spatially contiguous, and so clusters can contain areal units which are far apart geographically but exhibit similar disease risks over the entire study period. A number of areal units are identified as having a high intercept, which corresponds to a high level of average disease risk, and unsurprisingly many of these are the same areal units which were identified as having high risks in 2011 in Chapters 5 and 6, such as Drumchapel and Easterhouse. The slope clusters suggest that while the majority of areal units have similar disease risks across the study period, there were a small number of areas where changes were identified. The estimated disease risk in Eaglesham to the southeast and rural Dunbartonshire to the north west has increased over the ten years between 2002 and 2011, while areas such as Stepps and Wemyss Bay have experienced a decrease in estimated risk over the study period. It would be of interest to investigate the reasons for these changes, which could be as a result of changes to environmental changes in these areas or a change in population behaviour over the study period.

8.7 Summary

The methodology proposed within this thesis enhances the existing disease mapping literature by providing new approaches for clustering in both spatial and spatio-temporal data. The novel clustering approach allows the study region to be partitioned into sensible spatially contiguous clusters, and the two proposed spatial Bayesian models allow for improved estimation of disease risk levels as well as being able to identify the optimal cluster structure for the data. The proposed spatio-temporal Bayesian model provides an improved method of modelling the change in the spatial structure over time, and is able to identify clusters which have similar disease risk levels and similar rates of change over time. Both spatial clustering models have the drawback of requiring a set of prior data, so there may be interest in develop a single stage clustering model along similar lines to the spatio-temporal model proposed here. There is also scope to extend the spatio-temporal model by identifying a single set of clusters in space and time rather than having separate clustering terms for the model intercept and slope. Such an approach could be implemented within a similar modelling structure by allowing each intercept-slope cluster to have its own separate intercept and slope fixed effects. This form of model would allow the intercept-slope interactions to be taken into account when estimating disease risk instead of forming these clusters by a post-hoc combination of intercept and slope clusters as is the case here. Under such an approach, it would be imperative that the maximum number of clusters was set appropriately to avoid overparameterisation as a result of the increased number of fixed effects. Another extension to our model would be to use a reversible-jump McMC algorithm to remove the

requirement for a user selected maximum cluster number. Such an approach would be more computationally intensive, but would allow the number of clusters to be shaped by the data, with the possibility of an additional cluster being formed, or two clusters being joined together, at each stage of the McMC algorithm. There is also scope for extending this model to allow for a non-linear trend over time, enabling applications with data which contains a more complex temporal trend.

Bibliography

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. *Proceedings of the Second International Symposium on Information Theory*, 267–281.

Audit Scotland (2012). Health inequalities in Scotland. Technical report, The Scottish Government.

Bayes, T. (1764). An essay toward solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society of London 53*, 370–418.

Bernardinelli, L., D. Clayton, C. Pascutto, C. Montomoli, M. Ghislandi, and M. Songini (1995). Bayesian analysis of space-time variation in disease risk. *Statistics in Medicine 14*, 2433–2443.

Besag, J., J. York, and A. Mollié (1991). Bayesian image restoration, with two applications in spatial statistics. *Annals of the Institute of Statistical Mathematics 43*, 1–20.

Bohning, D. (2003). Empirical Bayes estimators and non-parametric mixture models for space and timespace disease mapping and surveillance. *Environmetrics 14*, 431–451.

Breslow, N, E. and G. Clayton, D (1993). Approximate inference in generalised linear mixed models. *Journal of the American Statistical Association 88*, 9–25.

Calinski, T. and J. Harabasz (1974). A dendrite method for cluster analysis. *Communications in Statistics Theory and Methods 3*, 1–27.

Celeux, G. and G. Govaert (1995). Gaussian parsimonious clustering models. *Pattern Recognition 28*, 781–793.

Charras-Garrido, M., D. Abrial, and J. de Goer (2012). Classification method for disease risk mapping based on discrete hidden Markov random fields. *Biostatistics 13*, 241–255.

Charras-Garrido, M., L. Azizi, F. Forbes, S. Doyle, N. Peyrard, and D. Abrial (2013). On the difficulty to delimit disease risk hot spots. *Journal of Applied Earth Observation and Geoinformation 22*, 99–105.

Chib, S. (1993). Bayes regression with autoregressive errors : A Gibbs sampling approach. *Journal of Econometrics 58*, 275–294.

Congdon, P. and H. Southall (2005). Trends in inequality in infant mortality in the north of England, 1921 to 1973, and their association with urban and social structure. *Journal of the Royal Statistical Society: Series A 168*, 679–700.

de Boor, C. (1972). On calculation with B-splines. *Journal of Approximation Theory 6*, 50–62.

Dempster, A., N. Laird, and D. Rubin (1977). Maximum likelihood from

incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B 39*, 1–38.

Eddelbuettel, D. and R. François (2011). Rcpp: Seamless R and C++ integration. *Journal of Statistical Software 40*, 1–18.

Eilers, P. and B. Marx (1996). Flexible smoothing with b-splines and penalties. *Statistical Science 11*, 89–121.

Fraley, C. and A. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association 97*, 611–631.

Gelman, A., J. Hwang, and A. Vehtari (2014). Understanding predictive information criteria for Bayesian models. *Statistics and Computing 24*, 997–1016.

Gelman, A. and D. Rubin (1992). Inference from iterative simulation using multiple sequences. *Statistical Science 7*, 457–511.

Geman, S. and D. Geman (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence 6*, 721–741.

Green, P. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika 82*, 711–732.

Green, P. and S. Richardson (2002). Hidden Markov models and disease mapping. *Journal of the American Statistical Association 97*, 1055–1070.

Hastie, T., R. Tibshirani, and J. Friedman (2001). *The Elements of Statistical Learning*, Chapter 14.3. Springer New York Inc.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika 57*, 97–109.

Jeffreys, H. (1946). An invariant form for the prior probability in estimation problems. *Proceedings of the Royal Society of London: Series A 186*, 453–461.

Knorr-Held, L. (2000). Bayesian modelling of inseparable space-time variation in disease risk. *Statistics in Medicine 19*, 2555–2567.

Knorr-Held, L. and G. Rasser (2000). Bayesian detection of clusters and discontinuities in disease maps. *Biometrics 56*, 13–21.

Kulldorff, M. (1997). A Spatial Scan Statistic. *Communications in Statistics 26*, 1481–1496.

Lee, D. and R. Mitchell (2012). Boundary detection in disease mapping studies. *Biostatistics 13*, 415–426.

Lee, D. and R. Mitchell (2013). Locally adaptive spatial smoothing using conditional autoregressive models. *Journal of the Royal Statistical Society Series C 62*, 593–608.

Lee, D., A. Rushworth, and S. Sahu (2014). A Bayesian localised conditional auto-regressive model for estimating the health effects of air pollution. *Biometrics 70*, 419–429.

Leroux, B., X. Lei, and N. Breslow (1999). *Estimation of disease rates in small areas: A new mixed model for spatial dependence*, Chapter Statistical Models in Epidemiology, the Environment and Clinical Trials, Halloran, M and Berry, D (eds), pp. 135–178. Springer-Verlag, New York.

Li, P., S. Banerjee, and A. McBean (2011). Mining boundary effects in areally referenced spatial data using the Bayesian information criterion. *Geoinformatica 15*, 435–454.

Link, W. and M. Eaton (2012). On thinning of chains in MCMC. *Methods in Ecology and Evolution 3*, 112–115.

Lu, H., C. Reilly, S. Banerjee, and B. Carlin (2007). Bayesian areal wombling via adjacency modelling. *Environmental and Ecological Statistics 14*, 433–452.

MacNab, Y. and B. Dean, C (2001). Autoregressive spatial smoothing and temporal spline smoothing for mapping rates. *Biometrics 57*, 949–956.

MacNab, Y., P. Farrell, P. Gustafson, and S. Wen (2004). Estimation in bayesian disease mapping. *Biometrics 60*, 865–873.

McLachlan, G. and D. Peel (2004). *Finite Mixture Models*. Wiley.

Metropolis, N., W. Rosenbluth, A, N. Rosenbluth, M, H. Teller, A, and E. Teller (1953). Equations of state calculations by fast computing machines. *Journal of Chemical Physics 21*, 1087–1092.

Mitchell, R. and D. Lee (2014). Is there really a wrong side of the tracks in urban areas and does it matter for spatial analysis? *Annals of the Association of American Geographers 104*, 432–443.

Moran, P. (1950). Notes on continuous stochastic phenomena. *Biometrika 37*, 17–23.

Nelder, J, A. and W. M. Wedderburn, R (1972). Generalized linear models. *Journal of the Royal Statistical Society, Series A 135*, 370–384.

Palm, T. (1890). The geographical distribution and etiology of rickets. *Practitioner 45*, 270–342.

R Development Core Team (2008). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. ISBN 3-900051-07-0.

Rand, W. (1971). Objective Criteria for the Evaluation of Clustering Methods. *Journal of the American Statistical Association 66*, 846–850.

Rousseeuw, P. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Computational and Applied Mathematics 20*, 53–65.

Rue, H., S. Martino, and N. Chopin (2009). Approximate Bayesian inference for latent Gaussian models using integrated nested Laplace approximations (with discussion). *Journal of the Royal Statistical Society, Series B 71*, 319–392.

Schrödle, B. and L. Held (2010). Spatio-temporal disease mapping using INLA. *Environmetrics 22*, 725–734.

Schrödle, B., L. Held, R. Riebler, and J. Danuser (2011). Using integrated nested Laplace approximations for the evaluation of veterinary surveillance data from Switzerland: a case study. *Journal of the Royal Statistical Society series C 60*, 261–279.

Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics 6*, 461–464.

Snow, J. (1855). *On the Mode of Communication of Cholera* (Second Edition ed.). John Churchill,.

Spiegelhalter, D., N. Best, B. Carlin, and A. Van der Linde (2002). Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society series B 64*, 583639.

Stephens, M. (2000). Dealing with label switching in mixture models. *Journal of the Royal Statistical Society: Series B 62*, 795–809.

Tibshirani, R., G. Walter, and T. Hastie (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society, Series B 63*, 411–423.

Ugarte, M, D., A. Adin, T. Goicoa, and F. Militino, A (2014). On fitting spatio-temporal disease mapping models using approximate Bayesian inference. *Statistical Methods in Medical Research 23*, 507–530.

Ugarte, M, D., T. Goicoa, and F. Militino, A (2010). Spatio-temporal modelling of mortality risks using penalized splines. *Environmetrics 21*, 270–289.

Ugarte, M, D., F. Militino, A, and T. Goicoa (2008). Prediction error estimators in empirical bayes disease mapping. *Environmetrics 19*, 287–300.

Wakefield, J. and A. Kim (2013). A Bayesian model for cluster detection. *Biostatistics 14*, 752–765.

Waller, L., B. Carlin, H. Xia, and A. Gelfand (1997). Hierarchical spatiotemporal mapping of disease rates. *Journal of the American Statistical Association 92*, 607–617.

World Health Organization (1993). *ICD-10 : International Statistical classification of diseases and related health problems* (10th revision ed.). World Health Organization, Geneva.

Wu, F, Y. (1982). The Potts model. *Review of Modern Physics 54*, 235–268.

Appendix A

Computer Code for Models

This section contains the R computer code used to carry out the analysis of the Greater Glasgow and Clyde respiratory admissions data.

A.1 Spatial Hierarchical Agglomerative Clustering Algorithm

This function takes in a set of data (*data*) and a neighbourhood matrix (W), and applies the spatial agglomerative hierarchical clustering model outlined in Chapter 4. The output consists of a series of updated neighbourhood matrices at each stage of the algorithm (*W.list*) and a list of the cluster structure at each stage of the algorithm (*cluster.store*).

```
euclid.cluster.func <- function(data, W){
```

```
n.prior <- ncol(data)

###Creating the storage matrices###
cluster.store <- matrix(rep(0,nrow(data)*nrow(data)),
                  ncol=nrow(data))
cluster.store[1,] <- 1:nrow(data)
cluster.size <- rep(1,nrow(data))
initial.W <- W

#############Centroid Linkage###############
update.data <- data

###Loop##
for (i in 1:(nrow(data)-1)){

  ###Count cluster size###
  for (j in 1:nrow(data)){
    cluster.size[j] <- sum(cluster.store[i,]==j)
  }

  ###Induce cluster structure in dissimilarity matrix###
  clusts <- cluster.store[i,][cluster.size>1]
  if(length(clusts)>0){
    for (j in 1:length(clusts)){
      num.row <- sum(cluster.store[i,]==clusts[j])
      update.data[cluster.store[i,]==clusts[j],] <-
      matrix(rep(apply(as.matrix(data[cluster.store[i,]
      ==clusts[j],]),2,mean),num.row),nrow=num.row, byrow=T)
    }
  }
  sim.mat <- as.matrix(dist(update.data, diag=TRUE, upper=TRUE))
```

```
sim.mat[lower.tri(sim.mat)] <- max(sim.mat)+1
sim.mat[W==0] <- max(sim.mat)
diag(sim.mat) <- max(sim.mat)

###Ensure points in same cluster aren't compared###
clust.mat <- as.matrix(dist(cbind(cluster.store[i,],
cluster.store[i,]), method="maximum", diag=TRUE, upper=TRUE))
sim.mat[clust.mat==0] <- max(sim.mat)

###Find most similar###
similar <- which(sim.mat==min(sim.mat), arr.ind=TRUE)
choice <- sample(x=1:nrow(similar), size=1)
join.1 <- cluster.store[i,similar[choice,1]]
join.2 <- cluster.store[i,similar[choice,2]]

###Store results###
cluster.store[(i+1),] <- cluster.store[i,]
cluster.store[(i+1),][cluster.store[(i+1),]==join.1] <-
        min(join.1,join.2)
cluster.store[(i+1),][cluster.store[(i+1),]==join.2] <-
        min(join.1,join.2)

###Update the W matrix ###
new.W.row <- apply((W[cluster.store[(i+1),]
        ==min(join.1,join.2),]),2,sum)
num.replaced <- nrow(W[cluster.store[(i+1),]
        ==min(join.1,join.2),])
W[cluster.store[(i+1),]==min(join.1,join.2),] <- matrix
(rep(new.W.row,num.replaced),nrow=num.replaced, byrow=TRUE)
W[,cluster.store[(i+1),]==min(join.1,join.2)] <- matrix
(rep(new.W.row,num.replaced),ncol=num.replaced, byrow=FALSE)
```

```
    W[W>0] <- 1
  }

  W.list <- vector("list", nrow(data))
  for (i in 1:nrow(data)){
    W.list[[i]] <- matrix(data = NA, nrow = nrow(data),
        ncol = nrow(data), byrow= FALSE, dimnames = NULL)
  }

    for(i in 1:nrow(data)){
    clust.mat <- as.matrix(dist(cbind(cluster.store[i,],
    cluster.store[i,]),method="maximum", diag=TRUE, upper=TRUE))
    W.list[[i]] <- initial.W
    W.list[[i]][clust.mat>0] <- 0
  }

  result <- list(W.list=W.list, cluster.store=cluster.store)
  return(result)
}
```

A.2 Fixed Effect Model

This function takes in a set of observed data (Y) and expected data (E), a neighbourhood matrix (*W.contiguity*) and the maximum number of clusters permitted (*max.cluster*), and applies the fixed effects model outlined in Chapter 5. The output is a list of DIC values for the INLA models (*dic.list*).

```
fixed.func <- function(Y, E, W.contiguity, max.cluster){

source("euclidean.r")  #clustering function outlined in previous section

###Set Data###
prior.Y <- Y[,-1]
prior.E <- E[,-1]
prior.SIR <- prior.Y / prior.E
log.prior.SIR <- log(prior.SIR)
Y.real <- Y[,1]
E.real <- E[,1]
SIR.real <- as.data.frame(Y.real/E.real)
rownames(SIR.real) <- Y[,1]

###Set up initial values###
prior.num <- ncol(prior.Y)
dic.list <- rep(NA, 100)
best.clust.store <- 0
best.model.fitted <- 0
n <- nrow(Y)
beta <- rep(0,n)
tau2.true <- 0.001
```

```
###Clustering###
clust.select <- euclid.cluster.func(log.prior.SIR, W.contiguity)

## Create Generic 0 C matrix
C <- diag(apply(W.contiguity,2,sum)) - W.contiguity

#### Fit a sequence of models and choose the best by minimising DIC.

## Store the DIC values
dic.list <- rep(NA, max.cluster)

## Fit a model with a single cluster
data.temp <- data.frame(Y.real=Y.real, offset=log(E.real), region=1:n)
formula <- Y.real ~ offset(offset) +  f(region, model="generic0", Cmatrix = C,
    constr=TRUE, hyper=list(theta=list(prior="loggamma", param=c(1,1))))
model  =  inla(formula, family="poisson", data=data.temp,
          control.results=list(return.marginals.predictor=TRUE),
          control.fixed=list(mean=0, mean.intercept=0, prec=0.001,
                             prec.intercept=0.001),
          control.compute=list(dic=TRUE, mlik=TRUE, cpo=TRUE),
          control.predictor=list(compute=TRUE),
          control.inla=list(strategy = "simplified.laplace", npoints = 21))

dic.list[1] <- model$dic$dic

## Fit separate models with between 2 and the max number of clusters.
for(i in 2:max.cluster)
   {
   j <- n+1-i
   factor.clust <- cluster.prior$cluster.store[j,]
   data.temp <- data.frame(Y.real=Y.real, offset=log(E.real),
```

```
                          region=1:n, factor.clust=factor.clust)
   formula <- Y.real ~ factor(factor.clust) + offset(offset) +
                f(region, model="generic0", Cmatrix = C, constr=TRUE,
                hyper=list(theta=list(prior="loggamma", param=c(1,1))))
   model  =  inla(formula, family="poisson", data=data.temp,
        control.results=list(return.marginals.predictor=TRUE),
        control.fixed=list(mean=0, mean.intercept=0, prec=0.001,
            prec.intercept=0.001),
        control.compute=list(dic=TRUE, mlik=TRUE, cpo=TRUE),
        control.predictor=list(compute=TRUE),
        control.inla=list(strategy = "simplified.laplace", npoints = 21))

dic.list[i] <- model$dic$dic
    }
results <- list(dic.list=dic.list)
}
```

A.3 Random Effect Model

This function performs the McMC inference for the random effect model. The function takes in a set of observed data (*data*), a set of expected values (*E.i*), the original neighbourhood matrix (*W*), the list of neighbourhood matrices from the clustering function (*W.list*) and a set of parameter starting values. The function outputs a set of vectors, each containing the full set of McMC draws for one of the model parameters.

```
update.W <- function(data, E.i, W, W.list ,b.start, phi.start, tau2.start,
        W.start.num, theta.start, var.theta, block.size.b, block.size.phi,
        n.rep=10000, b.prior.var=10,prop.var.b=0.01,tau2.prior.shape=0.001,
        tau2.prior.scale=0.001){

n <- nrow(data)
p <- ncol(data)-1
original.W <- W
y <- data[,1]
x <- data[,-1]

###Create stores###
b.store <- matrix(rep(0,(p+1)*n.rep),ncol=p+1)
phi.store <- matrix(rep(0,(n+1)*n.rep),ncol=n+1)
tau2.store <- rep(0,n.rep)
W.store <- rep(0,n.rep)
theta.store <- rep(0,n.rep)

###Set initial values based on function input###
tau2 <- tau2.start
b <- b.start
```

```
phi <- phi.start
W <- as.matrix(W.list[[W.start.num]])
W.num <- W.start.num
theta <- theta.start

###Store info about W.list
max.cluster.num <- length(W.list)

###Standardise the covariates (if applicable)###
w <- x*0
if(ncol(w)>0){
for (i in 1:p){
w[,i] <- (x[,i]-mean(x[,i]))/sd(x[,i])
}
}

###Append an intercept column###
z <- cbind(rep(1,n),w)
z <- as.matrix(z)

###Create an acceptance parameter###
accept.b <- c(0,0)
accept.phi <- c(0,0)
accept.W <- c(0,0)
accept.theta <- c(0,0)

###Create an parameter which will allow for calibration of the variance##
check.accept.b <- c(0,0)

###Calculate initial Q and its determinant###
Q <- -W
```

```
diag(Q) <- as.numeric(apply(W, 1, sum)) + 0.001
det.Q <- as.numeric(determinant(Q,logarithm = TRUE)$modulus)

##################
##Start the loop##
##################
for (i in 1:n.rep){

if(floor(i/100)==i/100){print(i)}

###Randomly allocate the first break and use to calculate block size for b
first.break.b <- sample(1:block.size.b,1)
n.block.b <- ceiling((p+1-first.break.b)/block.size.b)+1

##Create vectors containing the start and end points of each block for b
begin.b <- c(1,seq(from=first.break.b+1, by=block.size.b, length=n.block.b-1))
final.b <- begin.b
final.b[1] <- first.break.b
if(n.block.b>2){
final.b[2:(n.block.b-1)] <- final.b[2:(n.block.b-1)]+block.size.b-1
}
final.b[n.block.b] <- p+1

##Set initial R.i
R.i <- exp(z%*%b+phi[1:n])

##Update for b
for (j in 1:n.block.b)
{
proposal.b <- b
proposal.b[begin.b[j]:final.b[j]] <- rnorm(n=final.b[j]-begin.b[j]+1,
```

```
            mean=b[begin.b[j]:final.b[j]], sd=sqrt(prop.var.b))
prop.R.i <- exp(z%*%proposal.b+phi[1:n])
full.prop.b <- sum(-E.i*prop.R.i+y*log(E.i*prop.R.i))-
        sum(proposal.b^2/(2*b.prior.var))
full.b <- sum(-E.i*R.i+y*log(E.i*R.i)) -
            sum(b^2/(2*b.prior.var))
ratio.b <- exp(full.prop.b - full.b)
if(runif(1,0,1) < ratio.b)
{
b <- proposal.b
R.i <- prop.R.i
accept.b[1] <- accept.b[1]+1
check.accept.b[1] <- check.accept.b[1]+1
}
accept.b[2] <- accept.b[2]+1
check.accept.b[2] <- check.accept.b[2]+1
}
b.store[i,] <- b

###Randomly allocate the first break and use to calculate block size for phi
first.break.phi <- sample(1:block.size.phi,1)
n.block.phi <- ceiling((n-first.break.phi)/block.size.phi)+1

##Create vectors containing the start and end points of each block for phi
begin.phi <- c(1,seq(from=first.break.phi+1, by=block.size.phi,
                length=n.block.phi-1))
final.phi <- begin.phi
final.phi[1] <- first.break.phi
if(n.block.phi>2){
final.phi[2:(n.block.phi-1)] <- final.phi[2:(n.block.phi-1)]+block.size.phi-1
}else
```

```
{
}
final.phi[n.block.phi] <- n

###Now update phi in blocks
Q.temp <- Q / tau2
for (j in 1:n.block.phi)
{
q.rsrs <- Q.temp[begin.phi[j]:final.phi[j],begin.phi[j]:final.phi[j]]
q.rsrs.inv <- solve(q.rsrs)
q.rs.minus.rs <- Q.temp[begin.phi[j]:final.phi[j],-(begin.phi[j]:final.phi[j])]
proposal.phi <- phi
proposal.phi[begin.phi[j]:final.phi[j]] <- mvrnorm(n=1,
            mu=phi[begin.phi[j]:final.phi[j]], Sigma=q.rsrs.inv)

    prop.R.i <- exp(z[begin.phi[j]:final.phi[j],]%*%as.matrix(b)+
        proposal.phi[begin.phi[j]:final.phi[j]])
R.i <- exp(z[begin.phi[j]:final.phi[j],]%*%as.matrix(b)+
        phi[begin.phi[j]:final.phi[j]])
prop.mean.term <- proposal.phi[begin.phi[j]:final.phi[j]] + q.rsrs.inv
        %*%(q.rs.minus.rs%*%proposal.phi[-(begin.phi[j]:final.phi[j])])
full.prop.phi <- sum(-E.i[begin.phi[j]:final.phi[j]]*prop.R.i +
        y[begin.phi[j]:final.phi[j]]*log(E.i[begin.phi[j]:final.phi[j]]*
        prop.R.i))- (0.5)*t(prop.mean.term)%*%q.rsrs%*%prop.mean.term
mean.term <- phi[begin.phi[j]:final.phi[j]] +
         q.rsrs.inv%*%(q.rs.minus.rs%*%phi[-(begin.phi[j]:final.phi[j])])
full.phi <- sum(-E.i[begin.phi[j]:final.phi[j]]*R.i +
        y[begin.phi[j]:final.phi[j]]*log(E.i[begin.phi[j]:final.phi[j]]*R.i))
        - (0.5)*t(mean.term)%*%q.rsrs%*%mean.term

    ratio.phi <- exp(full.prop.phi - full.phi)
```

```
if(runif(1,0,1) < ratio.phi)
{
phi <- proposal.phi
accept.phi[1] <- accept.phi[1]+1
}
   accept.phi[2] <- accept.phi[2]+1
}

phi.star.mean <- sum(W[272,1:271]*phi[1:271])/(sum(W[272,1:271])+0.001)
phi.star.var <- tau2/(sum(W[272,1:271])+0.001)
phi[272] <- rnorm(1, phi.star.mean, phi.star.var)
phi[1:271] <- phi[1:271] - mean(phi[1:271])
phi.store[i,] <- phi

##Updating tau2
tau2.shape <- (n+1)/2 + tau2.prior.shape
tau2.scale <- 0.5*t(phi)%*%Q%*%phi + tau2.prior.scale
tau2 <- rinvgamma(1,tau2.shape,tau2.scale)
tau2.store[i] <- tau2

##Updating W
weights <- exp(-(1:n)*theta)/sum(exp(-(1:n)*theta))
poss.nums <- (W.num-2):(W.num+2)
valid.nums <- poss.nums[poss.nums>0 & poss.nums
        <=max.cluster.num & poss.nums!=W.num]
choice <- sample(x=1:length(valid.nums), size=1)
proposal.num <- valid.nums[choice]
proposal.W <- W.list[[proposal.num]]
prop.Q <- -proposal.W
diag(prop.Q) <- as.numeric(apply(proposal.W, 1, sum)) + 0.001
prop.det.Q <- as.numeric(determinant(prop.Q,logarithm = TRUE)$modulus)
```

```
full.prop.W <- log(weights[proposal.num]) +
    0.5*(prop.det.Q-(t(phi)%*%prop.Q%*%phi)/tau2)
full.W <- log(weights[W.num]) + 0.5*
    (det.Q-(t(phi)%*%Q%*%phi)/tau2)

###Calculating reverse probability for Metropolis-Hastings###
back.poss.nums <- (proposal.num-2):(proposal.num+2)
back.valid.nums <- back.poss.nums[back.poss.nums>0 &
        back.poss.nums<=max.cluster.num & back.poss.nums!=proposal.num]
W.to.W.star <- 1/length(valid.nums)
W.star.to.W <- 1/length(back.valid.nums)
ratio.W <- exp(full.prop.W - full.W + log(W.star.to.W) - log(W.to.W.star))
if(runif(1,0,1) < ratio.W)
{
W <- proposal.W
    W.num <- proposal.num
accept.W[1] <- accept.W[1]+1
Q <- prop.Q
det.Q <- prop.det.Q
}
accept.W[2] <- accept.W[2]+1


W.store[i] <- W.num

###Update theta
prop.theta <- rnorm(1,theta,var.theta)
while(prop.theta<0|prop.theta>1){
  prop.theta <- rnorm(1,theta,var.theta)
}
```

```
full.prop.theta <- log(exp(-W.num*prop.theta)/sum(exp(-(1:271)*prop.theta)))
full.theta <- log(exp(-W.num*theta)/sum(exp(-(1:271)*theta)))
ratio.theta <- exp(full.prop.theta - full.theta)
  if(runif(1,0,1) < ratio.theta)
  {
    theta <- prop.theta
    accept.theta[1] <- accept.theta[1]+1
  }

accept.theta[2] <- accept.theta[2]+1
theta.store[i] <- theta

##Calibrating acceptance rate

if(floor(i/100)==i/100)
{
accept.ratio.b <- check.accept.b[1]/check.accept.b[2]
if(accept.ratio.b < 0.4)
{
prop.var.b <- prop.var.b/2
} else if(accept.ratio.b > 0.8)
{
prop.var.b <- prop.var.b*2
}
check.accept.b <- c(0,0)
}

##Undo the standardisation
if(p>0){
for (i in 2:(p+1)){
```

```
b.store[,i] <- b.store[,i]/sd(x[,i-1])
}
}

}

##Return the results and acceptance rate.

result <- list(b.store=b.store,phi.store=phi.store,tau2.store=tau2.store,
            W.store=W.store,theta.store=theta.store,accept.b=accept.b,
            accept.phi=accept.phi,accept.W=accept.W,accept.theta=accept.theta)
return(result)
}
```

A.4 Spatio-Temporal Model

This function performs the McMC inference for the spatio-temporal model. The function takes in a set of observed data (Y), a set of expected values (E), the original neighbourhood matrix (W), the maximum number of intercept and slope clusters allowed ($num.C$ and $num.D$) and a set of parameter starting values. The function outputs a set of vectors, each containing the full set of McMC draws for one of the model parameters. A number of the parameter updates are written in C++ using the "rcpp" function in R, and the code for these is listed separately from the main function.

A.4.1 Main R Function

```
MCMCfunc <- function(Y, E, W, n.rep, num.C, num.D, time, rho, tau, theta.C,
                     lambda, sigma, theta.D, normal.prior.var,
                     gamma.prior.scale, gamma.prior.shape){

n <- ncol(W)
n.time <- ncol(Y)

## Set initial parameter values
slope <- rep(NA, nrow(Y))
intercept <- rep(NA, nrow(Y))
     for(i in 1:n)
     {
     mod <- glm(Y[i, ]~offset(log(E[i ,])) + time, family="poisson")
     intercept[i] <- mod$coefficients[1]
     slope[i] <- mod$coefficients[2]
```

```
     }

kmean <- kmeans(x=intercept, centers=num.C, nstart=1000)
alpha.temp <- kmean$centers
alpha.order <- order(alpha.temp)
cluster.temp <- kmean$cluster
alpha <- sort(alpha.temp)
C <- rep(NA, nrow(Y))
     for(i in 1:nrow(Y))
     {
      C[i] <- which(alpha.temp[cluster.temp[i]]==alpha)
     }

kmean <- kmeans(x=slope, centers=num.D, nstart=1000)
beta.temp <- kmean$centers
beta.order <- order(beta.temp)
cluster.temp <- kmean$cluster
beta <- sort(beta.temp)
D <- rep(NA, nrow(Y))
     for(i in 1:nrow(Y))
     {
      D[i] <- which(beta.temp[cluster.temp[i]]==beta)
     }

phi <- rep(0, nrow(Y))
delta <- rep(0, nrow(Y))

###Set initial parameters
CC <- C
alpha.list <- alpha[CC]
C.bar <- floor((num.C+1)/2)
```

```
centres.C <- ((1:num.C)-C.bar)^2
DD <- D
beta.list <- beta[DD]
D.bar <- floor((num.D+1)/2)
centres.D <- ((1:num.D)-D.bar)^2
logE <- log(E)

###Setting W as a double
n.neighbours <- as.numeric(apply(W, 1, sum))
W.duplet <- c(NA, NA)
     for(i in 1:n)
     {
          for(j in 1:n)
          {
               if(W[i,j]==1)
               {
               W.duplet <- rbind(W.duplet, c(i,j))
               }else{}
          }
     }
W.duplet <- W.duplet[-1, ]
n.duplet <- nrow(W.duplet)

## Create the list object
Wlist <- as.list(rep(NA,n))
     for(i in 1:n)
     {
     Wlist[[i]] <- which(W[i, ]==1)
     }

###Create stores###
```

```
alpha.store <- matrix(nrow=n.rep, ncol=num.C)
phi.store <- matrix(nrow=n.rep,ncol=n)
tau.store <- rep(0,n.rep)
rho.store <- rep(0,n.rep)
C.store <- matrix(nrow=n.rep, ncol=n)
theta.C.store <- rep(0,n.rep)
beta.store <- matrix(nrow=n.rep, ncol=num.D)
delta.store <- matrix(nrow=n.rep,ncol=n)
sigma.store <- rep(0,n.rep)
lambda.store <- rep(0,n.rep)
D.store <- matrix(nrow=n.rep, ncol=n)
theta.D.store <- rep(0,n.rep)

###Create acceptance stores###
accept.alpha <- c(0,0)
accept.phi <- c(0,0)
accept.rho <- c(0,0)
accept.beta <- c(0,0)
accept.delta <- c(0,0)
accept.lambda <- c(0,0)
accept.C <- c(0,0)
accept.D <- c(0,0)
accept.theta.C <- c(0,0)
accept.theta.D <- c(0,0)
accept.alpha.all <- c(0,0)
accept.phi.all <- c(0,0)
accept.rho.all <- c(0,0)
accept.beta.all <- c(0,0)
accept.delta.all <- c(0,0)
accept.lambda.all <- c(0,0)
```

```
alphapropvar <- 0.1
betapropvar <- 0.1
deltapropvar <- 0.1
phipropvar <- 0.1
rhopropvar <- 0.1
lambdapropvar <- 0.1
thetapropvar <- 0.05

## Create the set of determinants
Wstar <- diag(n.neighbours) - W
Wstar.eigen <- eigen(Wstar)
Wstar.val <- Wstar.eigen$values
Q.rho <- rho*Wstar + (1 - rho)*diag(1, n, n)
det.Q.rho <-  0.5 * sum(log((rho * Wstar.val + (1-rho))))
Q.lambda <- lambda*Wstar + (1 - lambda)*diag(1, n, n)
det.Q.lambda <-  0.5 * sum(log((lambda * Wstar.val + (1-lambda))))

###Start the loop###
for(i in 1:n.rep){

if(floor(i/100)==i/100){print(i)}

###Update C###
if(num.C>1){
prop.C <- rep(0,n)
nums <- 1:num.C
for(j in 1:n){
  options <- nums[-CC[j]]
  prop.C[j] <- sample(x=options,size=1)
}
prop.alpha <- alpha[prop.C]
```

```
prob1 <- theta.C * (CC - C.bar)^2 - theta.C * (prop.C - C.bar)^2
offset <- E*exp(matrix(rep(phi,n.time),nrow=n)+
        (matrix(rep(beta.list,n.time),nrow=n)+matrix(rep(delta,n.time),
        nrow=n))*matrix(rep(time,n),nrow=n, byrow=TRUE))
test=Cupdate(Y, offset, prob1, CC, prop.C, alpha.list, prop.alpha, n.time, n)
CC <- test[[1]]
accept.C[1] <- accept.C[1] + test[[2]]
accept.C[2] <- accept.C[2] + n
}else{
  accept.C[1] <- accept.C[1] + n
  accept.C[2] <- accept.C[2] + n
}
alpha.list <- alpha[CC]
C.store[i,] <- CC

###Update D###
if(num.D > 1){
prop.D <- rep(0,n)
nums <- 1:num.D
for(j in 1:n){
  options <- nums[-DD[j]]
  prop.D[j] <- sample(x=options,size=1)
}
prop.beta <- beta[prop.D]
prob1 <- theta.D * (DD - D.bar)^2 - theta.D * (prop.D - D.bar)^2
offset <- E*exp(matrix(rep(alpha.list,n.time),nrow=n)+
        matrix(rep(phi,n.time),nrow=n)+(matrix(rep(delta,n.time),nrow=n))
        *matrix(rep(time,n),nrow=n, byrow=TRUE))
test=Dupdate(Y, offset, prob1, DD, prop.D, beta.list, prop.beta, n.time, n, time)
DD <- test[[1]]
accept.D[1] <- accept.D[1] + test[[2]]
```

```
accept.D[2] <- accept.D[2] + n
}else{
  accept.D[1] <- accept.D[1] + n
  accept.D[2] <- accept.D[2] + n
}
beta.list <- beta[DD]
D.store[i,] <- DD

###Update alpha###
proposal <- c(-1000, alpha, 1000)
for(j in 1:num.C)
{
  proposal[(j+1)] <- rtrunc(n=1, spec="norm", a=proposal[j], b=proposal[(j+2)],
                     mean=proposal[(j+1)], sd=alphapropvar)
}
prop.alpha <- proposal[2:(num.C+1)]
prop.alpha.list <- prop.alpha[CC]
offset <- E*exp(matrix(rep(phi,n.time),nrow=n)+
    (matrix(rep(beta.list,n.time),nrow=n)+matrix(rep(delta,n.time),nrow=n))
    *matrix(rep(time,n),nrow=n, byrow=TRUE))
test=clustalphaupdate(Y, offset, normal.prior.var, alpha,
    prop.alpha, alpha.list, prop.alpha.list, n.time, n)
alpha <- test[[1]]
accept.alpha[1] <- accept.alpha[1]+test[[2]]
accept.alpha[2] <- accept.alpha[2]+1
alpha.store[i,] <- alpha

###Update beta###
proposal <- c(-1000, beta, 1000)
for(j in 1:num.D)
{
```

```
  proposal[(j+1)] <- rtrunc(n=1, spec="norm", a=proposal[j],
            b=proposal[(j+2)], mean=proposal[(j+1)], sd=betapropvar)
}
prop.beta <- proposal[2:(num.D+1)]
prop.beta.list <- prop.beta[DD]
offset <- E*exp(matrix(rep(alpha.list,n.time),nrow=n)+
    matrix(rep(phi,n.time),nrow=n)+(matrix(rep(delta,n.time),nrow=n))*
    matrix(rep(time,n),nrow=n, byrow=TRUE))
test=clustbetaupdate(Y, offset, normal.prior.var, beta, prop.beta,
            beta.list, prop.beta.list, n.time, n, time)
beta <- test[[1]]
accept.beta[1] <- accept.beta[1]+test[[2]]
accept.beta[2] <- accept.beta[2]+1
beta.store[i,] <- beta

###Update phi###
offset <- E*exp(matrix(rep(alpha.list,n.time),nrow=n)+
    (matrix(rep(beta.list,n.time),nrow=n)+matrix(rep(delta,n.time),nrow=n))
    *matrix(rep(time,n),nrow=n, byrow=TRUE))
test = clustpoissoncarupdate(Y, offset, Wlist, n.neighbours,
                                phi, rho, tau, phipropvar, n)
phi <- test[[1]]
accept.phi[1] <- accept.phi[1]+test[[2]]
accept.phi[2] <- accept.phi[2]+n
for(j in 1:num.C){
  phi[which(CC==j)] <- phi[which(CC==j)] - mean(phi[which(CC==j)])
}
phi.store[i,] <- phi

###Update delta###
offset <- E*exp(matrix(rep(alpha.list,n.time),nrow=n)+
```

```
    matrix(rep(phi,n.time),nrow=n) + matrix(rep(beta.list,n.time),nrow=n)
    *matrix(rep(time,n),nrow=n, byrow=TRUE))
test = clustpoissoncarupdate2(Y, offset, Wlist, n.neighbours, time,
                      delta, lambda, sigma, deltapropvar, n)
delta <- test[[1]]
accept.delta[1] <- accept.delta[1]+test[[2]]
accept.delta[2] <- accept.delta[2]+n
for(j in 1:num.D){
  delta[which(DD==j)] <- delta[which(DD==j)] - mean(delta[which(DD==j)])
}
delta.store[i,] <- delta

##Update tau2###
tau.shape <- n/2 + gamma.prior.shape
tau.scale <- 0.5*t(phi)%*%Q.rho%*%phi + gamma.prior.scale
tau <- rinvgamma(1,tau.shape,tau.scale)
tau.store[i] <- tau

##Update sigma###
sigma.shape <- n/2 + gamma.prior.shape
sigma.scale <- 0.5*t(delta)%*%Q.lambda%*%delta + gamma.prior.scale
sigma <- rinvgamma(1,sigma.shape,sigma.scale)
sigma.store[i] <- sigma

###Update rho###
prop.rho <-  rtrunc(n=1, spec="norm", a=0, b=1, mean=rho, sd=rhopropvar)
prop.Q.rho <- prop.rho*Wstar + (1 - prop.rho)*diag(1, n, n)
prop.det.Q.rho <-  0.5 * sum(log((prop.rho * Wstar.val + (1-prop.rho))))
full.rho <- det.Q.rho - 0.5*(t(phi)%*%Q.rho%*%phi)/tau
full.prop.rho <- prop.det.Q.rho - 0.5*(t(phi)%*%prop.Q.rho%*%phi)/tau
ratio.rho <- exp(full.prop.rho - full.rho)
```

```
if(runif(1,0,1) < ratio.rho)
{
  rho <- prop.rho
  Q.rho <- prop.Q.rho
  det.Q.rho <- prop.det.Q.rho
  accept.rho[1] <- accept.rho[1]+1
}
accept.rho[2] <- accept.rho[2]+1
rho.store[i] <- rho

###Update lambda###
prop.lambda <-  rtrunc(n=1, spec="norm", a=0, b=1, mean=lambda, sd=lambdapropvar)
prop.Q.lambda <- prop.lambda*Wstar + (1 - prop.lambda)*diag(1, n, n)
prop.det.Q.lambda <-  0.5 * sum(log((prop.lambda * Wstar.val + (1-prop.lambda))))
full.lambda <- det.Q.lambda - 0.5*(t(delta)%*%Q.lambda%*%delta)/tau
full.prop.lambda <- prop.det.Q.lambda - 0.5*(t(delta)%*%prop.Q.lambda%*%delta)/tau
ratio.lambda <- exp(full.prop.lambda - full.lambda)
if(runif(1,0,1) < ratio.lambda)
{
  lambda <- prop.lambda
  Q.lambda <- prop.Q.lambda
  det.Q.lambda <- prop.det.Q.lambda
  accept.lambda[1] <- accept.lambda[1]+1
}
accept.lambda[2] <- accept.lambda[2]+1
lambda.store[i] <- lambda

###Update theta.C###
prop.theta.C <- rtrunc(n=1, spec="norm", a=1, b=100, mean=theta.C, sd=thetapropvar)
prob1 <- sum((CC-C.bar)^2) * (theta.C - prop.theta.C)
prob2 <- n*log(sum(exp(-theta.C * centres.C))) -
```

```
        n*log(sum(exp(-prop.theta.C * centres.C)))
ratio.theta.C <- exp(prob1 + prob2)
if(runif(1,0,1) < ratio.theta.C)
{
  theta.C <- prop.theta.C
  accept.theta.C[1] <- accept.theta.C[1]+1
}
accept.theta.C[2] <- accept.theta.C[2]+1
theta.C.store[i] <- theta.C

###Update theta.D###
prop.theta.D <- rtrunc(n=1, spec="norm", a=1, b=100, mean=theta.D, sd=thetapropvar)
prob1 <- sum((DD-D.bar)^2) * (theta.D - prop.theta.D)
prob2 <- n*log(sum(exp(-theta.D * centres.D))) -
        n*log(sum(exp(-prop.theta.D * centres.D)))
ratio.theta.D <- exp(prob1 + prob2)
if(runif(1,0,1) < ratio.theta.D)
{
  theta.D <- prop.theta.D
  accept.theta.D[1] <- accept.theta.D[1]+1
}
accept.theta.D[2] <- accept.theta.D[2]+1
theta.D.store[i] <- theta.D
}

###Return the results and acceptance rates###
result <- list(alpha.store=alpha.store, phi.store=phi.store, tau.store=tau.store,
               rho.store=rho.store, C.store=C.store, theta.C.store=theta.C.store,
               beta.store=beta.store, delta.store=delta.store,
               sigma.store=sigma.store, lambda.store=lambda.store, D.store=D.store,
               theta.D.store=theta.D.store, accept.alpha=
```

```
              accept.alpha.all[1]/accept.alpha.all[2], accept.phi=
              accept.phi.all[1]/accept.phi.all[2], accept.rho=
              accept.rho.all[1]/accept.rho.all[2], accept.beta=
              accept.beta.all[1]/accept.beta.all[2], accept.delta=
              accept.delta.all[1]/accept.delta.all[2], accept.lambda=
              accept.lambda.all[1]/accept.lambda.all[2], accept.C=
              accept.C[1]/accept.C[2],accept.D=accept.D[1]/accept.D[2],
              accept.theta.C=accept.theta.C[1]/accept.theta.C[2],
              accept.theta.D=accept.theta.D[1]/accept.theta.D[2])

return(result)
}
```

A.4.2 C++ Functions

```
// [[Rcpp::export]]
List Cupdate(NumericMatrix Y, NumericMatrix offset, NumericVector prob1,
             NumericVector C, NumericVector propC, NumericVector alpha,
             NumericVector propalpha, const int ntime, const int n)
{
double logaccept=0, accept, accepted=0;
double prob2, prob3;

//Update each C value in turn
for(int j = 0; j < n; j++){

   //Compute the acceptance probability
   prob2 = sum(Y( j, _) * (propalpha[j]-alpha[j]));
   prob3 = sum(offset( j, _) * (exp(alpha[j])-exp(propalpha[j])));
   logaccept = prob1[j]+prob2+prob3;
```

```
   //Accept or not
   accept = exp(logaccept);
   if(runif(1)[0] <= accept)
          {
          C[j] = propC[j];
          accepted = accepted + 1;
          }
          else
          {
          }
}
List out(3);
out[0] = C;
out[1] = accepted;
out[2] = logaccept;
return out;
}

// [[Rcpp::export]]
List Dupdate(NumericMatrix Y, NumericMatrix offset, NumericVector prob1,
             NumericVector D, NumericVector propD, NumericVector beta,
             NumericVector propbeta, const int ntime, const int n,
             NumericVector time)
{
double logaccept=0, accept, accepted=0;
double prob2, prob3;

//Update each D value in turn
for(int j = 0; j < n; j++){
```

```
   //Compute the acceptance probability
   prob2 = sum(Y( j, _) * time * (propbeta[j]-beta[j]));
   prob3 = sum(offset( j, _) * (exp(beta[j]*time)-exp(propbeta[j]*time)));
   logaccept = prob1[j]+prob2+prob3;

   //Accept or not
   accept = exp(logaccept);
   if(runif(1)[0] <= accept)
          {
          D[j] = propD[j];
          accepted = accepted + 1;
          }
          else
          {
          }
}
List out(2);
out[0] = D;
out[1] = accepted;
return out;
}

// [[Rcpp::export]]
List clustalphaupdate(NumericMatrix Y, NumericMatrix offset,
        double normalpriorvar, NumericVector alpha, NumericVector propalpha,
        NumericVector alphalist, NumericVector propalphalist,
        const int ntime, const int n)
{

//Compute the acceptance probability
double logaccept=0, accept, accepted=0;
```

```
double prob1, prob2;

for(int t = 0; t < ntime; t++){
     prob1 = sum(Y( _, t) * (propalphalist-alphalist));
     prob2 = sum(offset( _, t) * (exp(alphalist)-exp(propalphalist)));
     logaccept = logaccept + prob1+prob2;
}
accept = exp(logaccept);

//Accept or not
   if(runif(1)[0] <= accept)
          {
          alpha = propalpha;
          accepted = 1;
          }
          else
          {
          }
List out(2);
out[0] = alpha;
out[1] = accepted;
return out;
}

// [[Rcpp::export]]
List clustbetaupdate(NumericMatrix Y, NumericMatrix offset,
            double normalpriorvar, NumericVector beta, NumericVector propbeta,
            NumericVector betalist, NumericVector propbetalist,
            const int ntime, const int n, NumericVector time)
{
```

```
//Compute the acceptance probability
double logaccept=0, accept, accepted=0;
double prob1, prob2;

for(int t = 0; t < ntime; t++){
     prob1 = sum(Y( _, t) * time[t] * (propbetalist-betalist));
     prob2 = sum(offset( _, t) * (exp(betalist*time[t])-
                    exp(propbetalist*time[t])));
     logaccept = logaccept + prob1+prob2;
}
accept = exp(logaccept);

//Accept or not
   if(runif(1)[0] <= accept)
          {
          beta = propbeta;
          accepted = 1;
          }
          else
          {
          }
List out(2);
out[0] = beta;
out[1] = accepted;
return out;
}

// [[Rcpp::export]]
List clustpoissoncarupdate(NumericMatrix Y, NumericMatrix offset,
                     List Wlist,NumericVector nneighbours,
                     NumericVector phi, double rho, double tau2,
```

```
                    double phipropvar, const int n)
{
// Update the spatially correlated random effects
//Create new objects
double logaccept=0, accepted=0;
double acceptance, sumphi;
double oldpriorbit, newpriorbit;
double priordenom, priormean, priorvar;
double propphi;
double lik1, lik2;
NumericVector phinew(n);

//  Update each random effect in turn
phinew = phi;
     for(int j = 0; j < n; j++)
     {
     // calculate prior mean and variance
     IntegerVector neighbourvec = Wlist[j];
     int m = neighbourvec.size();
     sumphi = 0;
          for(int l = 0; l < m; l++) sumphi += phinew[(neighbourvec[l]-1)];
      priordenom = (nneighbours[j] * rho + (1-rho));
      priorvar = tau2 / priordenom;
      priormean = rho * sumphi / priordenom;

      // propose a value
      propphi = rnorm(1, phinew[j], sqrt(priorvar*phipropvar))[0];

      // Accept or reject it
      newpriorbit = (0.5/priorvar) * pow((propphi - priormean), 2);
      oldpriorbit = (0.5/priorvar) * pow((phinew[j] - priormean), 2);
```

```
      lik1 = sum(offset(j,_) * (exp(phinew[j]) - exp(propphi)));
      lik2 = sum(Y(j,_) * (propphi - phinew[j]));
      logaccept = lik1 + lik2 + oldpriorbit - newpriorbit;
      acceptance = exp(logaccept);
          if(runif(1)[0] <= acceptance)
          {
          phinew[j] = propphi;
          accepted = accepted + 1;
          }
          else
          {
          }
    }
List out(2);
out[0] = phinew;
out[1] = accepted;
return out;
}

// [[Rcpp::export]]
List clustpoissoncarupdate2(NumericMatrix Y, NumericMatrix offset,
        List Wlistm, NumericVector nneighbours, NumericVector time,
        NumericVector delta, double lambda, double sigma,
        double deltapropvar, const int n)
{
// Update the spatially correlated random effects
//Create new objects
double logaccept=0, accepted=0;
double acceptance, sumdelta;
double oldpriorbit, newpriorbit;
double priordenom, priormean, priorvar;
```

```
double propdelta;
double lik1, lik2;
NumericVector deltanew(n);

//  Update each random effect in turn
deltanew = delta;
     for(int j = 0; j < n; j++)
     {
     // calculate prior mean and variance
     IntegerVector neighbourvec = Wlist[j];
     int m = neighbourvec.size();
     sumdelta = 0;
          for(int l = 0; l < m; l++) sumdelta += deltanew[(neighbourvec[l]-1)];
      priordenom = (nneighbours[j] * lambda + (1-lambda));
      priorvar = sigma / priordenom;
      priormean = lambda * sumdelta / priordenom;

      // propose a value
      propdelta = rnorm(1, deltanew[j], sqrt(priorvar*deltapropvar))[0];

      // Accept or reject it
      newpriorbit = (0.5/priorvar) * pow((propdelta - priormean), 2);
      oldpriorbit = (0.5/priorvar) * pow((deltanew[j] - priormean), 2);
      lik1 = sum(offset(j,_) * (exp(deltanew[j] * time)
                - exp(propdelta * time)));
      lik2 = sum(Y(j,_) * time * (propdelta - deltanew[j]));
      logaccept = lik1 + lik2 + oldpriorbit - newpriorbit;
      acceptance = exp(logaccept);
          if(runif(1)[0] <= acceptance)
          {
          deltanew[j] = propdelta;
```

```
          accepted = accepted + 1;
          }
          else
          {
          }
    }
List out(2);
out[0] = deltanew;
out[1] = accepted;
return out;
}
```

Appendix B

Computational Times

The analysis of the Greater Glasgow and Clyde respiratory admissions data was all carried out on a Samsung RV520 laptop with an Intel Core i3-2330M CPU 2.20 GHz processor and 4GB of RAM. The computational time for the analysis of the dataset under each of the three models proposed here (fixed effect, random effect and spatio-temporal) and the two existing models used for comparison (BYM, Bernardinelli) are outlined in Table B.1.

The BYM approach is substantially faster than the Fixed Effects and Random Effects models proposed here, however this is to be expected because the models proposed here are more complex. The simulation studies in Chapters 5 and 6 show that the methods proposed here perform better than the BYM model in almost every scenario. The Fixed Effects model is slower than the Random Effects model, although this is to be expected given that 100 separate models were fitted using INLA under this approach.

The Bernardinelli model is faster than the more complex Spatio-Temporal model proposed here, but again the simulation study in Chapter 7 showed that the Spatio-Temporal model performed better. Despite being more complex, the Spatio-Temporal model is fastest of our three models due to the use of the C++ code to update parameters within the model. The Random Effects model could be speeded up by using similar C++ functions to update the model parameters, and this will be addressed in future applications of the model. The Random Effects model is less complex than the Spatio-Temporal model, and it therefore seems likely that such an adaptation would make it faster than the Spatio-Temporal model.

Model	Inference	Clustering	Elapsed Time
Fixed Effects	INLA	Before Model	1144.52s
BYM	INLA	After Model	10.12s
Random Effects	McMC	Before Model	663.95s
Spatio-Temporal	McMC (with C++)	Within Model	380.19s
Bernardinelli	McMC (with C++)	After Model	182.46s

Table B.1: Comparison of computational Times for the analysis of the Greater Glasgow and Clyde Data under the three modelling approaches proposed here and two existing methods.