

Spatio-Temporal Areal Unit Modelling in R with Conditional Autoregressive Priors Using the CARBayesST Package

Duncan Lee
University of Glasgow

Alastair Rushworth
University of Strathclyde

Gary Napier
University of Glasgow

Abstract

Spatial data relating to non-overlapping areal units are prevalent in fields such as economics, environmental science, epidemiology and social science, and a large suite of modelling tools have been developed for analysing these data. Many utilise conditional autoregressive (CAR) priors to capture the spatial autocorrelation inherent in these data, and software such as **CARBayes** and **R-INLA** have been developed to make these models easily accessible to others. Such spatial data are typically available for multiple time periods, and the development of methodology for capturing temporally changing spatial dynamics is the focus of much current research. A sizeable proportion of this literature has focused on extending CAR priors to the spatio-temporal domain, and this article presents the **R** package **CARBayesST**, which is the first dedicated software for spatio-temporal areal unit modelling with conditional autoregressive priors. The software can fit a range of models focused on different aspects of space-time modelling, including estimation of overall space and time trends, and the identification of clusters of areal units that exhibit elevated values. This paper outlines the class of models that the software can implement, before applying them to simulated and two real examples, the latter in the fields of epidemiology and housing market analysis.

Keywords: Bayesian inference, conditional autoregressive priors, **R** package, spatio-temporal areal unit modelling.

1. Introduction

Areal unit data are a type of spatial data where the observations relate to a set of K contiguous but non-overlapping areal units, such as electoral wards or census tracts. Each observation relates to an entire areal unit, and thus is typically a summary measure such as an average, proportion, or total, for the entire unit. Examples include the total yield in sectors in an agricultural field trial (Besag and Higdon 1999), the proportion of people who are Catholic in lower super output areas in Northern Ireland (Lee, Minton, and Pryce 2015), the average score on SAT college entrance exams across US states (Wall 2004), and the total number of cases of chronic obstructive pulmonary disease from populations living in counties in Georgia, USA (Choi and Lawson 2011). Areal unit data such as these have become increasingly available in recent times, due to the creation of databases such as Scottish Statistics (<http://statistics.gov.scot/>), Health and Social Care Information Centre Indicator Portal (<http://www.hscic.gov.uk/indicatorportal>), and cancer registries such as

the Surveillance Epidemiology and End Results programme (<http://seer.cancer.gov>).

These databases provide data on a set of K areal units for N consecutive time periods, yielding a rectangular array of $K \times N$ spatio-temporal observations. The motivations for modelling these data are varied, and include estimating the effect of a risk factor on a response (see Wakefield 2007 and Lee, Ferguson, and Mitchell 2009), identifying clusters of contiguous areal units that exhibit an elevated risk of disease compared with neighbouring areas (see Charras-Garrido, Abrial, and de Goer 2012 and Anderson, Lee, and Dean 2014), and quantifying the level of segregation in a city between two or more different groups (see Lee *et al.* 2015). As a result different space-time structures have been proposed for modelling spatio-temporal data, which depend on the underlying question(s) of interest and the goals of the analysis.

However, a common challenge when modelling these data is that of spatio-temporal autocorrelation, namely that observations from geographically close areal units and temporally close time periods tend to have more similar values than units and time periods that are further apart. Temporal autocorrelation occurs because the data relate to largely the same populations in consecutive time periods, while spatial autocorrelation can arise for a number of reasons. The first is unmeasured confounding, which occurs when a spatially patterned risk factor for the response variable is not included in a regression model, and hence its omission induces spatial structure into the residuals. Other causes of spatial autocorrelation include neighbourhood effects, where the behaviours of individuals in an areal unit are influenced by individuals in adjacent units, and grouping effects where groups of people with similar behaviours choose to live close together.

Predominantly, a Bayesian approach is taken to modelling these data in the statistical community, where the spatio-temporal structure is modelled via sets of autocorrelated random effects. Conditional autoregressive (CAR, Besag, York, and Mollié 1991) priors and spatio-temporal extensions thereof are typically assigned to these random effects to capture the autocorrelation, which are special cases of a Gaussian Markov Random Field (GMRF). A range of models have been proposed with different space-time structures, which can be used to answer different questions of interest about the data. For example, Knorr-Held (2000) proposed an ANOVA style decomposition of the data into overall spatial and temporal trends and a space-time interaction, which allows common patterns such as the region wide average temporal trend to be estimated. In contrast, Li, Best, Hansell, Ahmed, and Richardson (2012) developed a model for identifying areas that exhibited unusual trends that were different from the overall region wide trend, while Lee and Lawson (2016) presented a model for identifying spatio-temporal clustering in the data.

An array of freely available software can now implement purely spatial areal unit models, ranging from general purpose statistical modelling software such as BUGS (Lunn, Spiegelhalter, Thomas, and Best 2009) and **R-INLA** (Rue, Martino, and Chopin 2009), to specialist spatial modelling packages in the statistical software R (R Core Team 2016) such as **CARBayes** (Lee 2013), **spatcounts** (Schabenberger 2009) and **spdep** (Bivand and Piras 2015). However, due to the flexibility of general purpose software, implementing spatial models, in say BUGS, requires a degree of programming that is non-trivial for applied researchers. Specialist software for

spatio-temporal modelling is much less well developed, with examples for geostatistical data including **spTimer** (Bakar and Sahu 2015) and **spBayes** (Finley, Banerjee, and Gelfand 2015). For areal unit data the **surveillance** (Paul and Meyer 2016) package models epidemic data, the **plm** (Croissant and Millo 2008) and **splm** (Millo and Piras 2012) packages model panel data, while the **nlme** (Pinheiro, Bates, DebRoy, Sarkar, and R Core Team 2015) and **lme4** (Bates, Mächler, Bolker, and Walker 2015) packages have functionality to model spatial and temporal random effects structures. However, software to fit a range of spatio-temporal areal unit models with CAR type autocorrelation structures is not available, which has motivated us to develop the R package **CARBayesST**.

CARBayesST can fit models with different spatio-temporal structures, which allows the user to answer different questions about their data within a common software environment. A number of models can be implemented, including: (i) a spatially varying linear time trends model (similar to Bernardinelli, Clayton, Pascutto, Montomoli, Ghislandi, and Songini 1995); (ii) a spatial and temporal main effects and interaction model (similar to Knorr-Held 2000); (iii) a spatially autocorrelated autoregressive time series model (Rushworth, Lee, and Mitchell 2014); (iv) a model with a common temporal trend but varying spatial surfaces (similar to Napier, Lee, Robertson, Lawson, and Pollock 2016); (v) a spatially adaptive smoothing model for localised spatial smoothing (Rushworth, Lee, and Sarrao 2017); and (vi) a spatio-temporal clustering model (Lee and Lawson 2016). The software has the same syntax and feel as the R package **CARBayes** for spatial areal unit modelling, and retains all of its easy-to-use features. These include specifying the spatial adjacency information via a single matrix (unlike BUGS that requires 3 separate list objects), fitting models via a one-line function call, and compatibility with **CARBayes** that allows it to share the latter’s model summary functionality for interpreting the results. The models available in this software can be fitted to binomial, Gaussian (in some cases) or Poisson data, and Section 2 summarises the models that can be implemented. Section 3 provides an overview of the software and its functionality, while Section 4 presents simulation results to illustrate the correctness of the **CARBayesST** implementation of one of the models. Sections 5 and 6 present two worked examples illustrating how to use the software for epidemiological and housing market research, while Section 7 concludes with a summary of planned future development for this package.

2. Spatio-temporal models for areal unit data

This section outlines the class of Bayesian hierarchical models that **CARBayesST** can implement, and in all cases inference is based on Markov chain Monte Carlo (MCMC) simulation. The first part of this section outlines the general hierarchical model that can be fitted, while the second describes the range of space-time random effects structures that are available.

2.1. General Bayesian hierarchical model

The study region comprises a set of $k = 1, \dots, K$ non-overlapping areal units $\mathcal{S} = \{\mathcal{S}_1, \dots, \mathcal{S}_K\}$, and data are recorded for each unit for $t = 1, \dots, N$ consecutive time periods. Thus data are available for a $K \times N$ rectangular array with K rows (spatial units) and N columns (time periods). The response data are denoted by $\mathbf{Y} = (\mathbf{Y}_1, \dots, \mathbf{Y}_N)_{K \times N}$, where $\mathbf{Y}_t = (Y_{1t}, \dots, Y_{Kt})$ denotes the $K \times 1$ column vector of observations for all K spatial units for time period

t . Next, a vector of known offsets are denoted by $\mathbf{O} = (\mathbf{O}_1, \dots, \mathbf{O}_N)_{K \times N}$, where similarly $\mathbf{O}_t = (O_{1t}, \dots, O_{Kt})$ denotes the $K \times 1$ column vector of offsets for time period t . Finally, $\mathbf{x}_{kt} = (x_{kt1}, \dots, x_{ktp})$ is a vector of p known covariates for areal unit k and time period t , and can include factors or continuous variables and a column of ones for the intercept term. Note, non-linear covariate-response relationships can be included by adding transformations of covariates (e.g., squared) or spline basis functions (e.g., using `ns()`) to the covariate vector. **CARBayesST** can fit a generalised linear mixed model to these data, whose general form is:

$$\begin{aligned} Y_{kt} | \mu_{kt} &\sim f(y_{kt} | \mu_{kt}, \nu^2) \quad \text{for } k = 1, \dots, K, \quad t = 1, \dots, N, \\ g(\mu_{kt}) &= \mathbf{x}_{kt}^\top \boldsymbol{\beta} + O_{kt} + \psi_{kt}, \\ \boldsymbol{\beta} &\sim N(\boldsymbol{\mu}_\beta, \Sigma_\beta). \end{aligned} \quad (1)$$

The vector of covariate regression parameters are denoted by $\boldsymbol{\beta} = (\beta_1, \dots, \beta_p)$, and a multivariate Gaussian prior is assumed with mean $\boldsymbol{\mu}_\beta$ and diagonal variance matrix Σ_β that can be chosen by the user. The ψ_{kt} term is a latent component for areal unit k and time period t encompassing one or more sets of spatio-temporally autocorrelated random effects, and the complete set are denoted by $\boldsymbol{\psi} = (\boldsymbol{\psi}_1, \dots, \boldsymbol{\psi}_N)$, where $\boldsymbol{\psi}_t = (\psi_{1t}, \dots, \psi_{Kt})$. **CARBayesST** can fit a number of different spatio-temporal structures for ψ_{kt} , all of which are outlined in Section 2.2 below. The software can implement model (1) for binomial, Gaussian and Poisson data, and the exact specifications of each are given below:

- **Binomial** - $Y_{kt} \sim \text{Binomial}(n_{kt}, \theta_{kt})$ and $\ln(\theta_{kt}/(1 - \theta_{kt})) = \mathbf{x}_{kt}^\top \boldsymbol{\beta} + O_{kt} + \psi_{kt}$.
- **Gaussian** - $Y_{kt} \sim N(\mu_{kt}, \nu^2)$ and $\mu_{kt} = \mathbf{x}_{kt}^\top \boldsymbol{\beta} + O_{kt} + \psi_{kt}$.
- **Poisson** - $Y_{kt} \sim \text{Poisson}(\mu_{kt})$ and $\ln(\mu_{kt}) = \mathbf{x}_{kt}^\top \boldsymbol{\beta} + O_{kt} + \psi_{kt}$.

In the binomial model (n_{kt}, θ_{kt}) respectively denote the number of trials and the probability of success in each trial in area k and time period t , while in the Gaussian model ν^2 is the observation variance. An inverse-gamma(a, b) prior is specified for ν^2 , and default values of ($a = 1, b = 0.01$) are specified by the software but can be changed by the user.

2.2. Spatio-temporal random effects models

All models implementable in this package induce spatio-temporal autocorrelation into the response data \mathbf{Y} via the latent random effects $\boldsymbol{\psi}$, using CAR type prior distributions and spatio-temporal extensions thereof. Spatial autocorrelation is controlled by a symmetric non-negative $K \times K$ neighbourhood, weight or adjacency matrix $\mathbf{W} = (w_{kj})$, where w_{kj} represents the spatial closeness between areal units ($\mathcal{S}_k, \mathcal{S}_j$). Larger valued elements represent spatial closeness between the two areas in question, where as smaller or zero values correspond to areas that are not spatially close. Typically, \mathbf{W} is assumed to be binary, where $w_{kj} = 1$ if areal units ($\mathcal{S}_k, \mathcal{S}_j$) share a common border (i.e., are spatially close) and is zero otherwise. Additionally, $w_{kk} = 0$. Under this binary specification the values of (ψ_{kt}, ψ_{jt}) for spatially adjacent areal units (where $w_{kj} = 1$) are spatially autocorrelated, where as values for non-neighbouring areal units (where $w_{kj} = 0$) are conditionally independent given the remaining $\{\psi_{it}\}$ values. This binary specification of \mathbf{W} based on sharing a common border is the most

commonly used for areal data, but the only requirement by **CARBayesST** is for \mathbf{W} to be symmetric, non-negative, and for each row sum to be greater than zero. Similarly, the model **ST.CARanova()** defined below uses a binary $N \times N$ temporal neighbourhood matrix $\mathbf{D} = (d_{tj})$, where $d_{tj} = 1$ if $|j - t| = 1$ and $d_{tj} = 0$ otherwise.

CARBayesST can fit the models for ψ outlined in Table 1, and full details for each one are given below. This set of models allows users to fit different spatio-temporal structures to their data and thus examine different underlying hypotheses. Out of these models **ST.CARlinear()**, **ST.CARanova()** and **ST.CARar()** are the simplest in terms of parsimony, and thus missing (NA) values are allowed in the response data (\mathbf{Y}) for these models. Missing values are not allowed in the remaining models as they have more complex forms, and exploratory simulated-based testing showed that missing values could not be well recovered in these cases.

ST.CARlinear()

This model is a modification of that proposed by [Bernardinelli et al. \(1995\)](#), and estimates separate but correlated linear time trends for each areal unit. Thus it is appropriate if the goal of the analysis is to estimate which areas are exhibiting increasing or decreasing (linear) trends in the response over time. The full model specification is given below.

$$\begin{aligned}
 \psi_{kt} &= \beta_1 + \phi_k + (\alpha + \delta_k) \frac{(t - \bar{t})}{N}, & (2) \\
 \phi_k | \phi_{-k}, \mathbf{W} &\sim \text{N} \left(\frac{\rho_{int} \sum_{j=1}^K w_{kj} \phi_j}{\rho_{int} \sum_{j=1}^K w_{kj} + 1 - \rho_{int}}, \frac{\tau_{int}^2}{\rho_{int} \sum_{j=1}^K w_{kj} + 1 - \rho_{int}} \right), \\
 \delta_k | \delta_{-k}, \mathbf{W} &\sim \text{N} \left(\frac{\rho_{slo} \sum_{j=1}^K w_{kj} \delta_j}{\rho_{slo} \sum_{j=1}^K w_{kj} + 1 - \rho_{slo}}, \frac{\tau_{slo}^2}{\rho_{slo} \sum_{j=1}^K w_{kj} + 1 - \rho_{slo}} \right), \\
 \tau_{int}^2, \tau_{slo}^2 &\sim \text{Inverse-Gamma}(a, b), \\
 \rho_{int}, \rho_{slo} &\sim \text{Uniform}(0, 1), \\
 \alpha &\sim \text{N}(\mu_\alpha, \sigma_\alpha^2).
 \end{aligned}$$

Here $\bar{t} = (1/N) \sum_{t=1}^N t$, and thus the modified linear temporal trend covariate is $t^* = (t - \bar{t})/N$ and runs over a centred unit interval. Thus areal unit k has its own linear time trend, with a spatially varying intercept $\beta_1 + \phi_k$ and a spatially varying slope $\alpha + \delta_k$. Note, the β_1 term comes from the covariate component $\mathbf{x}_{kt}^\top \boldsymbol{\beta}$ in (1). Each set of random effects $\boldsymbol{\phi} = (\phi_1, \dots, \phi_K)$ and $\boldsymbol{\delta} = (\delta_1, \dots, \delta_K)$ are modelled as spatially autocorrelated by the CAR prior proposed by [Leroux et al. \(2000\)](#), and are mean centred, that is $\sum_{j=1}^K \phi_j = \sum_{j=1}^K \delta_j = 0$. Here (ρ_{int}, ρ_{slo}) are spatial dependence parameters, with values of one corresponding to strong spatial smoothness that is equivalent to the intrinsic CAR prior proposed by [Besag et al. \(1991\)](#), while values of zero correspond to independence (for example if $\rho_{slo} = 0$ then $\delta_k \sim \text{N}(0, \tau_{slo}^2)$). Flat uniform priors on the unit interval are specified for the spatial dependence parameters (ρ_{int}, ρ_{slo}) , while conjugate inverse-gamma and Gaussian priors are specified for the random effects variances $(\tau_{int}^2, \tau_{slo}^2)$ and the overall slope parameter α respectively. The corresponding hyperparameters $(a, b, \mu_\alpha, \sigma_\alpha^2)$ can be chosen by the user, and the default values specified by the software are $(a = 1, b = 0.01, \mu_\alpha = 0, \sigma_\alpha^2 = 1000)$, which correspond to weakly informative prior distributions. Alternatively, the dependence parameters (ρ_{int}, ρ_{slo}) can be fixed at values in

Model	Equation	Description
ST.CARlinear()	(2)	This model is similar to that proposed by Bernardinelli <i>et al.</i> (1995) , and represents the spatio-temporal pattern in the mean response with spatially varying linear time trends. The slope and intercept parameters for each area's temporal trend are spatially autocorrelated, by assigning each the conditional autoregressive prior proposed by Leroux, Lei, and Breslow (2000) .
ST.CARanova()	(3)	This model is similar to that proposed by Knorr-Held (2000) , and represents the spatio-temporal pattern in the mean response with an ANOVA style decomposition into overall spatial and temporal main effects and a space-time interaction. The spatial and temporal main effects are modelled by the conditional autoregressive prior proposed by Leroux <i>et al.</i> (2000) , while the interactions are assumed to be independent.
ST.CARsepspatial()	(4)	This model is that proposed by Napier <i>et al.</i> (2016) , and represents the spatio-temporal pattern in the mean response with an overall temporal effect and separate independent spatial effects for each time period. Each time-period's spatial effect as well as the overall temporal effect are modelled by the conditional autoregressive prior proposed by Leroux <i>et al.</i> (2000) .
ST.CARar()	(5)	This model is that proposed by Rushworth <i>et al.</i> (2014) , and represents the spatio-temporal pattern in the mean response with a single set of spatially and temporally autocorrelated random effects. The effects follow a multivariate autoregressive process of order 1, where spatial autocorrelation is induced via the precision matrix based on the conditional autoregressive prior proposed by Leroux <i>et al.</i> (2000) .
ST.CARadaptive()	(7)	This model is that proposed by Rushworth <i>et al.</i> (2017) , and has the same spatio-temporal random effect structure as the ST.CARar() model. Additionally, the spatially autocorrelated precision matrix based on the adjacency matrix \mathbf{W} is estimated in the model, by means of treating those elements of \mathbf{W} that correspond to spatially adjacent areas as random quantities on the unit interval. This allows adaptive levels of spatial smoothness in the random effects surface.
ST.CARlocalised()	(9)	This model is that proposed by Lee and Lawson (2016) , and has the same spatio-temporal random effect structure as the ST.CARar() model. The difference is that it additionally has a piecewise constant intercept term with a maximum of G different levels, which can capture step-changes in disease risk between two spatially adjacent areas.

Table 1: Summary of the models available in the **CARBayesST** package together with the equation numbers defining them mathematically in this paper.

the unit interval $[0, 1]$ rather than being estimated in the model, by specifying arguments to the `ST.CARlinear()` function. For example, using the arguments `fix.rho.slo=TRUE`, `rho.slo=1` sets $\rho_{slo} = 1$ when fitting the model. Finally, missing (NA) values are allowed in the response data \mathbf{Y} for this model.

ST.CARanova()

The model is a modification of that proposed by Knorr-Held (2000), and decomposes the spatio-temporal variation in the data into 3 components, an overall spatial effect common to all time periods, an overall temporal trend common to all spatial units, and a set of independent space-time interactions. Thus this model is appropriate if the goal is to estimate overall time trends and spatial patterns. The model specification is given below.

$$\begin{aligned}
 \psi_{kt} &= \phi_k + \delta_t + \gamma_{kt}, & (3) \\
 \phi_k | \phi_{-k}, \mathbf{W} &\sim \text{N} \left(\frac{\rho_S \sum_{j=1}^K w_{kj} \phi_j}{\rho_S \sum_{j=1}^K w_{kj} + 1 - \rho_S}, \frac{\tau_S^2}{\rho_S \sum_{j=1}^K w_{kj} + 1 - \rho_S} \right), \\
 \delta_t | \delta_{-t}, \mathbf{D} &\sim \text{N} \left(\frac{\rho_T \sum_{j=1}^N d_{tj} \delta_j}{\rho_T \sum_{j=1}^N d_{tj} + 1 - \rho_T}, \frac{\tau_T^2}{\rho_T \sum_{j=1}^N d_{tj} + 1 - \rho_T} \right), \\
 \gamma_{kt} &\sim \text{N}(0, \tau_I^2), \\
 \tau_S^2, \tau_T^2, \tau_I^2 &\sim \text{Inverse-Gamma}(a, b), \\
 \rho_S, \rho_T &\sim \text{Uniform}(0, 1).
 \end{aligned}$$

Here the spatio-temporal autocorrelation is modelled by a common set of spatial random effects $\phi = (\phi_1, \dots, \phi_K)$ and a common set of temporal random effects $\delta = (\delta_1, \dots, \delta_N)$, and both are modelled by the CAR prior proposed by Leroux *et al.* (2000). Additionally, the model can incorporate an optional set of independent space-time interactions $\gamma = (\gamma_{11}, \dots, \gamma_{KN})$, which can be specified by the argument `interaction=TRUE` (the default) in the function call. All sets of random effects are mean centred. Fixed uniform (ρ_S, ρ_T) or conjugate ($\tau_S^2, \tau_T^2, \tau_I^2$) priors are specified for the remaining parameters, and the default specifications for the latter are ($a = 1, b = 0.01$). Alternatively, in common with the `ST.CARlinear()` function the dependence parameters (ρ_S, ρ_T) can be fixed at values in the unit interval $[0, 1]$ rather than being estimated in the model, for full details see the help file for this function. Finally, missing (NA) values are allowed in the response data \mathbf{Y} for this model.

ST.CARsepspatial()

The model is a generalisation of that proposed by Napier *et al.* (2016), and represents the data by two components, an overall temporal trend, and separate spatial surfaces for each time period that share a common spatial dependence parameter but have different spatial variances. This model is appropriate if the goal is to estimate both a common overall temporal trend and the extent to which the spatial variation in the response has changed over time. The model specification is given below.

$$\psi_{kt} = \phi_{kt} + \delta_t, \quad (4)$$

$$\begin{aligned}
\phi_{kt} | \phi_{-kt}, \mathbf{W} &\sim \text{N} \left(\frac{\rho_S \sum_{j=1}^K w_{kj} \phi_{jt}}{\rho_S \sum_{j=1}^K w_{kj} + 1 - \rho_S}, \frac{\tau_t^2}{\rho_S \sum_{j=1}^K w_{kj} + 1 - \rho_S} \right), \\
\delta_t | \delta_{-t}, \mathbf{D} &\sim \text{N} \left(\frac{\rho_T \sum_{j=1}^N d_{tj} \delta_j}{\rho_T \sum_{j=1}^N d_{tj} + 1 - \rho_T}, \frac{\tau_T^2}{\rho_T \sum_{j=1}^N d_{tj} + 1 - \rho_T} \right), \\
\tau_1^2, \dots, \tau_N^2, \tau_T^2 &\sim \text{Inverse-Gamma}(a, b), \\
\rho_S, \rho_T &\sim \text{Uniform}(0, 1),
\end{aligned}$$

where $\phi_{-k,t} = (\phi_{1,t}, \dots, \phi_{k-1,t}, \phi_{k+1,t}, \dots, \phi_{K,t})$. This model fits an overall temporal trend to the data $\boldsymbol{\delta} = (\delta_1, \dots, \delta_N)$ that is common to all areal units, which is augmented with a separate (uncorrelated) spatial surface $\boldsymbol{\phi}_t = (\phi_{1t}, \dots, \phi_{Kt})$ at each time period t . The overall temporal trend and each spatial surface are modelled by the CAR prior proposed by Leroux *et al.* (2000), and the latter have a common spatial dependence parameter ρ_S but a temporally-varying variance parameter τ_t^2 . Thus the collection $(\tau_1^2, \dots, \tau_N^2)$ allows one to examine the extent to which the magnitude of the spatial variation in the data has changed over time. Note that here we fix ρ_S to be constant in time as it is not orthogonal to τ_t^2 , thus if it varied then any changes in τ_t^2 would not directly correspond to changes in spatial variance over time. As with all other models the random effects are zero-mean centred, while flat and conjugate priors are specified for (ρ_S, ρ_T) and $(\tau_T^2, \tau_1^2, \dots, \tau_N^2)$ respectively with $(a = 1, b = 0.01)$ being the default values. Alternatively, in common with the `ST.CARlinear()` function, the dependence parameters (ρ_S, ρ_T) can be fixed at values in the unit interval $[0, 1]$ rather than being estimated in the model.

ST.CARar()

The model is that proposed by Rushworth *et al.* (2014), and represents the spatio-temporal structure with a multivariate first order autoregressive process with a spatially correlated precision matrix. This model is appropriate if one wishes to estimate the evolution of the spatial response surface over time without forcing it to be the same for each time period. The model specification is given below.

$$\begin{aligned}
\psi_{kt} &= \phi_{kt}, \\
\phi_t | \phi_{t-1} &\sim \text{N}(\rho_T \phi_{t-1}, \tau^2 \mathbf{Q}(\mathbf{W}, \rho_S)^{-1}) \quad t = 2, \dots, N, \\
\phi_1 &\sim \text{N}(\mathbf{0}, \tau^2 \mathbf{Q}(\mathbf{W}, \rho_S)^{-1}), \\
\tau^2 &\sim \text{Inverse-Gamma}(a, b), \\
\rho_S, \rho_T &\sim \text{Uniform}(0, 1).
\end{aligned} \tag{5}$$

In this model $\boldsymbol{\phi}_t = (\phi_{1t}, \dots, \phi_{Kt})$ is the vector of random effects for time period t , which evolve over time via a multivariate first order autoregressive process with temporal autoregressive parameter ρ_T . The temporal autocorrelation is thus induced via the mean $\rho_T \boldsymbol{\phi}_{t-1}$, while spatial autocorrelation is induced by the variance $\tau^2 \mathbf{Q}(\mathbf{W}, \rho_S)^{-1}$. The corresponding precision matrix $\mathbf{Q}(\mathbf{W}, \rho_S)$ was proposed by Leroux *et al.* (2000) and corresponds to the CAR models used in the other models above. The algebraic form of this matrix is given by

$$\mathbf{Q}(\mathbf{W}, \rho_S) = \rho_S [\text{diag}(\mathbf{W}\mathbf{1}) - \mathbf{W}] + (1 - \rho_S) \mathbf{I}, \tag{6}$$

where $\mathbf{1}$ is the $K \times 1$ vector of ones while \mathbf{I} is the $K \times K$ identity matrix. In common with all other models the random effects are zero-mean centred, while flat and conjugate priors are specified for (ρ_S, ρ_T) and τ^2 respectively, with $(a = 1, b = 0.01)$ being the default values for the latter. In common with the `ST.CARanova()` function the dependence parameters (ρ_S, ρ_T) can be fixed at values in the unit interval $[0, 1]$ rather than being estimated in the model. Finally, missing (NA) values are allowed in the response data \mathbf{Y} for this model.

`ST.CARadaptive()`

The model is that proposed by Rushworth *et al.* (2017), and is an extension of `ST.CARar()` proposed by Rushworth *et al.* (2014) to allow for spatially adaptive smoothing. It is appropriate if one believes that the residual spatial autocorrelation in the response after accounting for the covariates is consistent over time but has a localised structure. That is, it is strong in some parts of the study region but weak in others. The model has the same autoregressive random effects structure as the previous model `ST.CARar()`, namely:

$$\begin{aligned} \psi_{kt} &= \phi_{kt}, \\ \phi_t | \phi_{t-1} &\sim N(\rho_T \phi_{t-1}, \tau^2 \mathbf{Q}(\mathbf{W}, \rho_S)^{-1}) \quad t = 2, \dots, N, \\ \phi_1 &\sim N(\mathbf{0}, \tau^2 \mathbf{Q}(\mathbf{W}, \rho_S)^{-1}), \\ \tau^2 &\sim \text{Inverse-Gamma}(a, b), \\ \rho_S, \rho_T &\sim \text{Uniform}(0, 1). \end{aligned} \tag{7}$$

However, the random effects from `ST.CARar()` have a single level of spatial dependence that is controlled by ρ_S . All pairs of adjacent areal units will have strongly autocorrelated random effects if ρ_S is close to one, while no such spatial dependence will exist anywhere if ρ_S is close to zero. However, real data may exhibit spatially varying dependences, as two adjacent areal units may exhibit similar values suggesting a value of ρ_S close to one, while another adjacent pair may exhibit very different values suggesting a value of ρ_S close to zero.

This model allows for localised spatial autocorrelation by allowing spatially neighbouring random effects to be correlated (inducing smoothness) or conditionally independent (no smoothing), which is achieved by modelling the non-zero elements of the neighbourhood matrix \mathbf{W} as unknown parameters, rather than assuming they are fixed constants. These adjacency parameters are collectively denoted by $\mathbf{w}^+ = \{w_{kj} | k \sim j\}$, where $k \sim j$ means areas (k, j) share a common border. Estimating $w_{kj} \in \mathbf{w}^+$ as equal to zero means (ϕ_{kt}, ϕ_{jt}) are conditionally independent for all time periods t given the remaining random effects, while estimating it close to one means they are correlated. The adjacency parameters in \mathbf{w}^+ are each modelled on the unit interval, by assuming a multivariate Gaussian prior distribution on the transformed scale $\mathbf{v}^+ = \log(\mathbf{w}^+ / (\mathbf{1} - \mathbf{w}^+))$. This prior is a shrinkage model with a constant mean μ and a diagonal variance matrix with variance parameter ζ^2 , and is given by

$$f(\mathbf{v}^+ | \tau_w^2, \mu) \propto \exp \left[-\frac{1}{2\tau_w^2} \left(\sum_{v_{ik} \in \mathbf{v}^+} (v_{ik} - \mu)^2 \right) \right], \tag{8}$$

$$\tau_w^2 \sim \text{Inverse-Gamma}(a, b).$$

The prior distribution for \mathbf{v}^+ assumes that the degree of smoothing between pairs of adjacent random effects is not spatially dependent, which results from the work of [Rushworth *et al.* \(2017\)](#) that shows poor estimation performance when \mathbf{v}^+ (and hence \mathbf{w}^+) is assumed to be spatially autocorrelated. Under small values of τ_w^2 the elements of \mathbf{v}^+ are shrunk to μ , and here we follow the work of [Rushworth *et al.* \(2017\)](#) and fix $\mu = 15$ because it avoids numerical issues when transforming between \mathbf{v}^+ and \mathbf{w}^+ and implies a prior preference for values of w_{kj} close to 1. That is as $\tau_w^2 \rightarrow 0$ the prior becomes the global smoothing model `ST.CARar()`, where as when τ_w^2 increases both small and large values in \mathbf{w}^+ are supported by the prior. As with the other models the default values for the inverse-gamma prior for τ_w^2 are $(a = 1, b = 0.01)$. Alternatively, it is possible to fix ρ_S using the `rhofix` argument, e.g., `rhofix=1` fixes $\rho_S = 1$, so that globally the spatial correlation is strong and is altered locally by the estimates of \mathbf{w}^+ . For further details see [Rushworth *et al.* \(2017\)](#).

ST.CARlocalised()

The model was proposed by [Lee and Lawson \(2016\)](#), and augments the smooth spatio-temporal variation in `ST.CARar()` with a piecewise constant intercept component. This model is appropriate when the aim of the analysis is to identify clusters of areas that exhibit elevated (or reduced) values of the response compared with their geographical and temporal neighbours. Thus this model is similar to `ST.CARadaptive()`, in that both relax the restrictive assumption that if two areas are close together then their estimated random effects must be similar. This model captures any step-changes in the response via the mean function, whereas `ST.CARadaptive()` captures them via the correlation structure (via \mathbf{W}). Model `ST.CARlocalised()` is given by

$$\begin{aligned} \psi_{kt} &= \lambda_{Z_{kt}} + \phi_{kt}, & (9) \\ \phi_t | \phi_{t-1} &\sim \text{N}(\rho_T \phi_{t-1}, \tau^2 \mathbf{Q}(\mathbf{W})^-) & t = 2, \dots, N, \\ \phi_1 &\sim \text{N}(\mathbf{0}, \tau^2 \mathbf{Q}(\mathbf{W})^-), \\ \tau^2 &\sim \text{Inverse-Gamma}(a, b), \\ \rho_T &\sim \text{Uniform}(0, 1), \end{aligned}$$

where the ‘ $-$ ’ in $\mathbf{Q}(\mathbf{W})^-$ denotes a generalised inverse. The random effects $\phi = (\phi_1, \dots, \phi_T)$ are modelled by a simplification of the `ST.CARar()` model with $\rho_S = 1$, which corresponds to the intrinsic CAR model proposed by [Besag *et al.* \(1991\)](#). Note, for this model the inverse $\mathbf{Q}(\mathbf{W})^{-1}$ does not exist as the precision matrix is singular. This simplification is made so that the random effects capture the globally smooth spatio-temporal autocorrelation in the data, allowing the other component to capture localised clustering and step-changes. This second component is a piecewise constant clustering or intercept component $\lambda_{Z_{kt}}$. Thus spatially and temporally adjacent data points (Y_{kt}, Y_{js}) will be similar (autocorrelated) if they are in the same cluster or intercept, that is if $\lambda_{Z_{kt}} = \lambda_{Z_{js}}$, but exhibit a step-change if they are estimated to be in different clusters, that is if $\lambda_{Z_{kt}} \neq \lambda_{Z_{js}}$. The piecewise constant intercept or clustering component comprises at most G distinct levels, making this component a piecewise constant intercept term. The G levels are ordered via the prior specification:

$$\lambda_j \sim \text{Uniform}(\lambda_{j-1}, \lambda_{j+1}) \quad \text{for } j = 1, \dots, G, \quad (10)$$

where $\lambda_0 = -\infty$ and $\lambda_{G+1} = \infty$. Here $Z_{kt} \in \{1, \dots, G\}$ and controls the assignment of the (k, t) th data point to one of the G intercept levels. A penalty based approach is used to model Z_{kt} , where G is chosen larger than necessary and a penalty prior is used to shrink it to the middle intercept level. This middle level is $G^* = (G + 1)/2$ if G is odd and $G^* = G/2$ if G is even, and this penalty ensures that Z_{kt} is only in the extreme low and high risk classes if supported by the data. Thus G is the maximum number of distinct intercept terms allowed in the model, and is not the actual number of intercept terms estimated in the model. The allocation prior is independent across areal units but correlated in time, and is given by:

$$\begin{aligned} f(Z_{kt}|Z_{k,t-1}) &= \frac{\exp(-\delta[(Z_{kt} - Z_{k,t-1})^2 + (Z_{kt} - G^*)^2])}{\sum_{r=1}^G \exp(-\delta[(r - Z_{k,t-1})^2 + (r - G^*)^2])} \quad \text{for } t = 2, \dots, N, \quad (11) \\ f(Z_{k1}) &= \frac{\exp(-\delta(Z_{k1} - G^*)^2)}{\sum_{r=1}^G \exp(-\delta(r - G^*)^2)}, \\ \delta &\sim \text{Uniform}(1, m). \end{aligned}$$

Temporal autocorrelation is induced by the $(Z_{kt} - Z_{k,t-1})^2$ component of the penalty, while the $(Z_{kt} - G^*)^2$ component penalises class indicators Z_{kt} towards the middle risk class G^* . The size of this penalty and hence the amount of smoothing that is imparted on \mathbf{Z} is controlled by δ , which is assigned a uniform prior. The default value is $m = 10$, and full details of this model can be found in [Lee and Lawson \(2016\)](#).

2.3. Inference

All models in this package are fitted in a Bayesian setting using Markov chain Monte Carlo simulation. All parameters whose full conditional distributions have a closed form distribution are Gibbs sampled, which includes the regression parameters (β) and the random effects (e.g. ϕ etc) in the Gaussian data models, as well as the variance parameters (e.g. τ^2 etc) in all models. The remaining parameters are updated using Metropolis or Metropolis-Hastings steps, and the random effects in the binomial and Poisson data models can be updated via the simple Gaussian random walk Metropolis algorithm or the Metropolis Adjusted Langevin Algorithm (MALA, [Roberts and Rosenthal 1998](#)). The default is to use MALA, but the user can choose simple random walk Metropolis steps by specifying the `MALA=FALSE` argument in the function call. The regression parameters are updated in blocks of size 10 utilising MALA updates, although if only a single covariate is incorporated in the model or only an intercept term is included then simple random walks are used as they were found to perform better. The remaining parameters utilise simple Gaussian random walk Metropolis updates. The simple random walk Metropolis updates are automatically tuned in the algorithms to have acceptance rates of between 40% - 50% for scalar parameter updates, and between 20% - 40% for vector parameters. The MALA updates are also automatically tuned in the software to have acceptance rates between 40% - 50%. The overall functions that implement the MCMC algorithms are written in R, while the computationally intensive updating steps are written as computationally efficient C++ routines using the R package **Rcpp** ([Eddelbuettel and Francois](#)

2011). Additionally, the sparsity of the neighbourhood matrices \mathbf{W} and \mathbf{D} are utilised via their triplet forms when updating the random effects within the algorithms, which increases the computational efficiency of the software. Finally, we note that the software runs only on one core, and leaves possibilities of parallelization by the user.

As a note of caution, all conclusions from MCMC-based inference in Bayesian models are only valid if the samples generated are well behaved, that is they are realisations from the target posterior distribution. Thus one of the challenges of fitting Bayesian models using any software is determining when the Markov chains have converged, and as a result how many samples to discard as the burn-in period and then how many more to generate on which to base inference. Convergence can be assessed using many metrics, the simplest of which is by eye by viewing trace-plots of the parameters, which should be stationary and show random fluctuations around a single mean level (see Figure 1 for an example of Markov chains showing no evidence against convergence). In addition to this visual check **CARBayesST** presents the convergence diagnostic proposed by Geweke (1992) for sample parameters when applying the `print()` function to a fitted model object, which uses the `geweke.diag()` function from the **coda** package. This statistic is in the form of a Z-score, and values between (-1.96, 1.96) are suggestive of convergence. A full discussion of how many samples to generate, burn-in lengths and whether or not to thin the Markov chains are beyond the scope of this paper, and further details can be found in general texts on Bayesian modelling such as Robert and Casello (2010) and Gelman, Carlin, Stern, Dunson, Vehtari, and Rubin (2013).

3. Loading and using the software

3.1. Loading the software

CARBayesST is a package for the R (R Core Team 2016) statistical software and can be downloaded from the Comprehensive R Archive Network (CRAN, <http://cran.r-project.org/>) for Windows, Linux and Apple platforms. The package requires R ($\geq 3.0.0$) and depends on packages **MASS** (Venables and Ripley 2002) and **Rcpp** ($\geq 0.11.5$). Additionally, it imports functionality from the **CARBayesdata** (Lee 2016), **coda** (Plummer, Best, Cowles, and Vines 2006), **dplyr** (Wickham and Francois 2015), **matrixcalc** (Novomestky 2012), **sp** (Bivand, Pebesma, and Gomez-Rubio 2013), **spam** (Furrer and Sain 2010), **spdep**, **stats**, **testthat** (Wickham 2011), **truncdist** (Novomestky and Nadarajah 2012), **truncnorm** (Trautmann, Steuer, Mersmann, and Bornkamp 2014) and **utils** packages. Once installed it can be loaded using the command `library("CARBayesST")`.

The packages listed above are automatically loaded for use in **CARBayesST** by the above call, but a complete spatial analysis beginning with reading in and formatting shapefiles and data, creating the neighbourhood matrix \mathbf{W} , and plotting the results requires a number of other packages. Thus the worked examples in this paper utilise functionality from the following packages: **CARBayes**, **CARBayesdata**, **dplyr**, **maptools** (Bivand and Lewin-Koh 2015), **MASS**, **sp** and **spdep**.

3.2. Using the software

The software can fit six models: `ST.CARlinear()`, `ST.CARanova()`, `ST.CARsepspatial()`, `ST.CARar()`, `ST.CARadaptive()` `ST.CARlocalised()`, and full details of the arguments required for each function are given in the help-files. However, the main arguments common to all the functions that are required for a baseline analysis (for example using default priors) are as follows.

- **formula** - A formula for the covariate part of the model using the same syntax used in the `lm()` function. Offsets can be included here using the `offset()` function. The response and each covariate should be vectors of length $KT \times 1$, where each vector is ordered so that the first K data points are the set of all K spatial locations at time 1, the next K are the set of spatial points for time 2 and so on.
- **family** - The likelihood model which must be one of ‘binomial’, ‘Gaussian’ or ‘Poisson’.
- **trials** - This is only needed if `family=‘binomial’`, and is a vector the same length and in the same order as the response containing the total number of trials for each area and time period.
- **W** - A $K \times K$ symmetric and non-negative neighbourhood matrix whose row sums must all be positive. Typically a binary specification is used, where the kj th element w_{kj} equals one if areas $(\mathcal{S}_j, \mathcal{S}_k)$ are spatially close (e.g., share a common border) and is zero otherwise. This matrix can be specified by hand or created from a shapefile and data frame using functionality from the **CARBayes** and **spdep** packages.
- **burn-in** - The number of MCMC samples to discard as the burn-in period.
- **n.sample** - The number of MCMC samples to generate.

When a model has been fitted in **CARBayesST**, the software provides the following summary extractor functions:

- `coef()` - returns the estimated (posterior median) regression coefficients.
- `fitted()` - returns the fitted values based on posterior medians.
- `logLik()` - returns the estimated loglikelihood based on posterior medians.
- `model.matrix()` - returns the design matrix of covariates.
- `print()` - prints a summary of the fitted model to the screen, including both parameter summaries and convergence diagnostics for the MCMC run.
- `residuals()` - returns either the ‘response’ (raw), ‘pearson’, or ‘deviance’ residuals from the model (based on posterior medians).

Additionally, the **CARBayes** functions `summarise.samples()` and `summarise.lincomb()` can be applied to **CARBayesST** models to summarise the results. The software updates the user on its progress to the R console, which allows the user to monitor the function’s progress. However, using the `verbose=FALSE` option will disable this feature. Once run, each model in this package returns a list object with the following components.

- **summary.results** - A summary table of selected parameters that is presented when using the `print()` function. The table includes the posterior median (`Median`) and 95% credible interval (`2.5%`, `97.5%`), the number of samples generated (`n.sample`), the acceptance rate for the Markov chain (`% accept`), the effective number of independent samples using the `effectiveSize()` function from the `coda` package (`n.effective`), and the convergence diagnostic proposed by Geweke (1992) and implemented in the `coda` package (`Geweke.diag`). This diagnostic takes the form of a Z-score, so that convergence is suggested by the statistic being within the range (-1.96, 1.96).
- **samples** - A list containing the MCMC samples from the model, where each element in the list is a matrix. The names of these matrix objects correspond to the parameters defined in Section 2 of this paper, and each column contains the set of samples for a single parameter. For example, for `ST.CARlinear()` the (`tau2`, `rho`) elements of the list have columns ordered as $(\tau_{int}^2, \tau_{slo}^2)$ and $(\rho_{int}^2, \rho_{slo}^2)$ respectively. Similarly, for `ST.CARanova()` the (`tau2`, `rho`) elements of the list have columns ordered as $(\tau_S^2, \tau_T^2, \tau_I^2)$ (the latter only if `interaction=TRUE`) and (ρ_S^2, ρ_T^2) respectively. Finally, each model returns samples from the posterior distribution of the fitted values for each data point (`fitted`).
- **fitted.values** - A vector of fitted values based on posterior medians for each area and time period in the same order as the data `Y`.
- **residuals** - A matrix of 3 types of residuals in the same order as the response. The 3 columns of this matrix correspond to “response” (raw), “pearson”, and “deviance” residuals.
- **modelfit** - Model fit criteria including the Deviance Information Criterion (DIC, Spiegelhalter, Best, Carlin, and Van der Linde 2002) and its corresponding estimated effective number of parameters (p.d), the Watanabe-Akaike Information Criterion (WAIC, Watanabe 2010) and its corresponding estimated number of effective parameters (p.w), and the Log Marginal Predictive Likelihood (LMPL, Congdon 2005). Also included is the loglikelihood. The best fitting model is one that minimises the DIC and WAIC but maximises the LMPL.
- **accept** - The acceptance probabilities for the parameters.
- **localised.structure** - This element is NULL except for the models `ST.CARadaptive()` and `ST.CARlocalised()`. For `ST.CARadaptive()` this element is a list with 2 $K \times K$ matrices, `Wmn` and `W99`, which summarise the estimated adjacency relationships. `Wmn` contains the posterior median for each w_{kj} element estimated in the model for adjacent areal units, while `W99` contains indicator variables for whether $\mathbb{P}(w_{jk} < 0.5 | \mathbf{Y}) > 0.99$. For both matrices, elements corresponding to non-adjacent pairs of areas have NA values. For `ST.CARlocalised()` this element is a vector of length $KT \times 1$, and gives the posterior median class (Z_{kt} value) that each data point is assigned to. This vector is in the same order as the data `Y`.
- **formula** - The formula (as a text string) for the covariate and offset part of the model.
- **model** - A text string describing the model that has been fitted.
- **X** - The design matrix of covariates inherited from the `formula` argument.

The remainder of this paper illustrates the **CARBayesST** software via a small simulation study to illustrate the correctness of the MCMC algorithms, as well as two worked examples, the latter of which utilise spatio-temporal data to answer important questions in public health and the housing market.

4. Simulation exercises

This section is split into 3 parts. The first illustrates how to use the software to fit a model, the second presents a short simulation study to illustrate the correctness of the **CARBayesST** implementation of a model, while the third describes a comparison of run times for various data sizes. All three exercises are based on the `ST.CARanova()` model, but similar studies could be done for the other models. We note in passing that the correctness of the **CARBayesST** implementations of the `ST.CARsepspatial()` (Napier *et al.* 2016), `ST.CARar()` (Rushworth *et al.* 2014), `ST.CARadaptive()` (Rushworth *et al.* 2017), and `ST.CARlocalised()` (Lee and Lawson 2016) models have been assessed in the accompanying papers where the models were developed. Here we generate data from a binomial logistic model, thus the model comprises the data likelihood $Y_{kt} \sim \text{Binomial}(n_{kt} = 50, \theta_{kt})$ and $\ln(\theta_{kt}/(1 - \theta_{kt})) = \beta_1 + x_{kt}\beta_2 + \psi_{kt}$, which is combined with equation (3), yielding parameters $(\beta_{2 \times 1}, \phi_{K \times 1}, \delta_{N \times 1}, \gamma_{KN \times 1}, \rho_S, \rho_T, \tau_S^2, \tau_T^2, \tau_I^2)$. Generation of the data is described below, and in what follows we fix $\beta = (0, 0.1)$, $\rho_S = \rho_T = 0.8$, $\tau_S^2 = \tau_T^2 = \tau_I^2 = 0.01$.

4.1. Generating data and fitting a model

Consider a spatial region comprising $K = 400$ areal units on a regular 20×20 grid and $N = 20$ consecutive time periods. Such a grid can be constructed from the code

```
R> n.space <- 20
R> N <- 20
R> x.easting <- 1:n.space
R> x.northing <- 1:n.space
R> Grid <- expand.grid(x.easting, x.northing)
R> K <- nrow(Grid)
R> N.all <- N * K
```

A binary 400×400 spatial neighbourhood matrix **W** can be constructed for this region based on spatial adjacency (rook, in chess) using the code

```
R> distance <- as.matrix(dist(Grid))
R> W <- array(0, c(K,K))
R> W[distance==1] <-1
```

Similarly, a binary 20×20 temporal neighbourhood matrix **D** can be constructed using the code

```
R> distance <- as.matrix(dist(1:N))
R> D <- array(0, c(N,N))
R> D[distance==1] <-1
```

From \mathbf{W} the precision matrix can be computed for the multivariate Gaussian representation of the spatial random effects ϕ from (6) as follows:

```
R> Q.W <- 0.8 * (diag(apply(W, 2, sum)) - W) + 0.2 * diag(rep(1,K))
```

where $\rho_S = 0.8$. This matrix can then be inverted and a sample of random effects ϕ generated (assuming $\tau_S^2 = 0.01$) using the code

```
R> Q.W.inv <- solve(Q.W)
R> library("MASS")
R> phi <- mvrnorm(n = 1, mu = rep(0, K), Sigma = (0.01 * Q.W.inv))
R> phi.long <- rep(phi, N)
```

Here the last line repeats the spatial random effects N times, as the `ST.CARanova()` model assumes that there is a single set of spatial random effects for all time periods. The temporal random effects under the `ST.CARanova()` model have the same functional form but depend on \mathbf{D} rather than \mathbf{W} , and thus a realisation can be generated analogously using the code

```
R> Q.D <- 0.8 * (diag(apply(D, 2, sum)) - D) + 0.2 * diag(rep(1, N))
R> Q.D.inv <- solve(Q.D)
R> delta <- mvrnorm(n = 1, mu = rep(0, N), Sigma = (0.01 * Q.D.inv))
R> delta.long <- kronecker(delta, rep(1, K))
```

Again, the final line repeats the temporal random effects for each spatial unit. Next, we generate space-time interactions and a covariate \mathbf{x} , both of which are generated independently from Gaussian distributions.

```
R> x <- rnorm(n = N.all, mean = 0, sd = 1)
R> gamma <- rnorm(n = N.all, mean = 0, sd = sqrt(0.01))
```

Finally, we set the intercept term $\beta_1 = 0$, the regression coefficient $\beta_2 = 0.1$, and the number of trials for the binomial likelihood in each area and time period being $n_{kt} = 50$. Then we generate the response variable via the code below. Here `LP` denotes the linear predictor, which contains an intercept term, a covariate and three sets of random effects (spatial, temporal, and interactions).

```
R> beta1 <- 0
R> beta2 <- 0.1
R> n <- rep(50, N.all)
R> LP <- beta1 + beta2 * x + phi.long + delta.long + gamma
R> theta.true <- exp(LP) / (1 + exp(LP))
R> Y <- rbinom(n = N.all, size = n, prob = theta.true)
```

The `ST.CARanova()` model can then be applied to these data using the following code.

```
R> library("CARBayesST")
R> model <- ST.CARanova(formula = Y~x, family = "binomial", trials = n,
+   W = W, burn-in = 20000, n.sample = 120000, thin = 10)
```

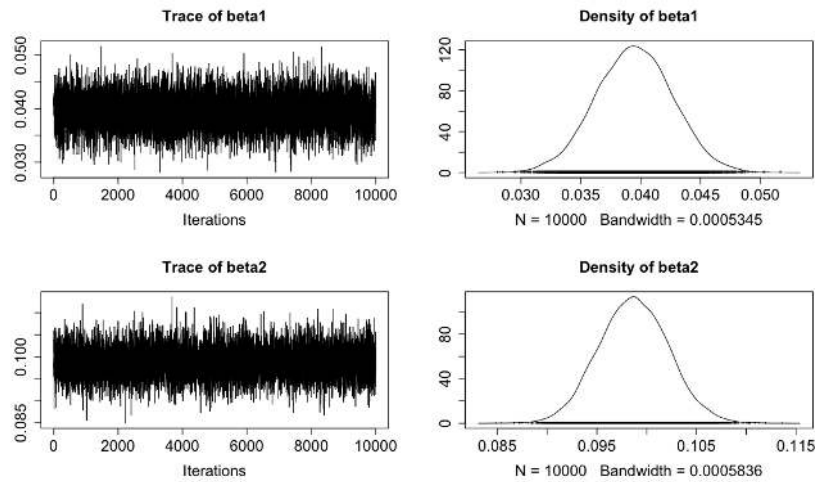



Figure 1: Posterior distributions of the regression parameters (*the true values are* $\beta_1 = 0, \beta_2 = 0.1$). The left panel contains trace-plots while the right panel are density estimates.

In the code above inference is based on 10,000 MCMC samples, which were generated from a single Markov chain that was run for 120,000 iterations with a 10,000 burn-in period and subsequently thinned by 10 to reduce the autocorrelation of the Markov chain. The model object is a `list` containing elements such as the posterior samples for all parameters, fitted values and residuals, and model fit criteria, and further details are given in Section 3. The posterior samples are available in the `samples` element of the list object `model`, which is itself a list of `mcmc` objects (from the `coda` package) for each set of parameters. Trace-plots of the parameters for β can be produced using the code below, and the result is shown in Figure 1.

```
R> colnames(model$samples$beta) <- c("beta1", "beta2")
R> plot(model$samples$beta)
```

The figures show no evidence against convergence, and that the posterior distributions for both parameters are centred close to their true values. A summary of the fitted model can be obtained using the `print()` function as follows.

```
R> print(model)

#####
#### Model fitted
#####
Likelihood model - binomial (logit link function)
Latent structure model - spatial and temporal main effects and an interaction
Regression equation - Y ~ x

#####
#### Results
#####
```

Posterior quantities for selected parameters and DIC

	Median	2.5%	97.5%	n.sample	% accept	n.effective	Geweke.diag
(Intercept)	0.0395	0.0332	0.0456	10000	35.0	8417.5	1.0
x	0.0987	0.0919	0.1056	10000	35.0	7816.7	-1.1
tau2.S	0.0097	0.0072	0.0130	10000	100.0	4774.6	0.0
tau2.T	0.0055	0.0031	0.0110	10000	100.0	10000.0	-1.2
tau2.I	0.0124	0.0096	0.0153	10000	100.0	287.0	-1.2
rho.S	0.7131	0.4577	0.9102	10000	44.1	4761.8	2.0
rho.T	0.6599	0.2049	0.9441	10000	65.3	8927.1	-0.5

DIC = 44079.5 p.d = 1225.449 LMPL = -20882.66

The Summary is presented in two parts, the first of which describes the model that has been fit. The second summarises the results, and includes the posterior median (`Median`) and 95% credible intervals (2.5%, 97.5%) for selected parameters (not the random effects), the convergence diagnostic proposed by Geweke (1992) (`Geweke.diag`) as a Z-score and the effective number of independent samples (`n.effective`). Also displayed are the actual number of samples kept from the MCMC run (`n.sample`) as well as the acceptance rate for each parameter (`% accept`). Note, parameters that have an acceptance rate of 100% have been Gibbs sampled due to their full conditional distributions being a standard distribution. Finally, the DIC and LMPL overall model fit criteria are displayed, which allows models with different space-time structures to be compared.

4.2. Small simulation study

This section illustrates the correctness of the `CARBayesST` implementation of the `ST.CARanova()` model, by simulating 100 data sets using the code presented above and summarising the bias and 95% coverage probabilities of the estimated model parameters. However, we note that this simulation study does not provide evidence against errors in the implementation, and that also we have not validated `CARBayesST` against another software implementation. The results of this simulation study are presented in Table 2, which shows bias and 95% coverage probabilities (mean square error is not presented as we are not seeking to compare two different models) for $(\beta_1, \beta_2, \rho_S, \rho_T, \tau_S^2, \tau_T^2, \tau_I^2, \phi, \delta, \gamma)$ as well as for the fitted values. For the random effects and fitted values all results are averaged over both the 100 simulated data sets and over all the elements (either `K`, `N` or `N.all`) in each simulated data set. The results show that overall the `CARBayesST` implementation of the `ST.CARanova()` model produces largely unbiased parameter estimates, with all parameters except the dependence parameters (ρ_S, ρ_T) having negligible biases. The largest bias in absolute size is -0.2062 for ρ_T , which is not surprising because it is a temporal dependence parameter estimated from data on only 20 time points. Additionally, the table shows that the coverage probabilities for all the parameters are close to the nominal 0.95 levels, suggesting that the 95% credible intervals are the correct width.

4.3. Timing and data sizes

The final part of this section presents some timings for fitting the `ST.CARanova()` model

Parameter	Bias	Coverage probability
β_0	-0.000202	0.940
β_1	0.000867	0.970
ρ_S	-0.093660	0.910
ρ_T	-0.206200	0.940
τ_S^2	-0.000471	0.970
τ_T^2	-0.000697	0.970
τ_I^2	-5.52×10^{-5}	0.950
ϕ	4.29×10^{-7}	0.950
δ	-1.56×10^{-7}	0.948
γ	-1.09×10^{-7}	0.947
Fitted values	-0.002513	0.948

Table 2: Summary of the simulation study undertaken to assess the bias and 95% coverage probabilities of the parameter estimates from the `ST.CARanova()` model. All results are based on 100 simulated data sets generated as outlined above.

K	N	N.all	Timing in seconds (h:m.s)
$K = 10 \times 10 = 100$	N=10	1,000	142 (2.21)
$K = 10 \times 10 = 100$	N=20	2,000	246 (4.06)
$K = 20 \times 20 = 400$	N=10	4,000	467 (4.47)
$K = 20 \times 20 = 400$	N=20	8,000	895 (14.55)
$K = 30 \times 30 = 900$	N=20	18,000	2247 (37.27)
$K = 30 \times 30 = 900$	N=30	27,000	3235 (53.55)
$K = 40 \times 40 = 1600$	N=30	48,000	5699 (1:34.59)
$K = 40 \times 40 = 1600$	N=40	64,000	7620 (2:07.00)
$K = 50 \times 50 = 2500$	N=40	100,000	13509 (3:45.09)

Table 3: Summary of the time taken to run the `ST.CARanova()` model in seconds (in hours minutes and second in brackets) on a regular grid with different square grid sizes (K), numbers of time periods (N) and total number of data points ($N.all$).

to data sets of various sizes. All data sets are generated on a regular lattice with a single covariate and interaction random effects present as illustrated in Section 4.1. The results are presented in Table 3, and relate to fitting the model for a total of 120,000 iterations, with a burn-in period of 20,000 and thinning the resulting Markov chains by 10. The timings were carried out on an Apple iMac computer with a 3.5 GHZ Intel Core i7 processor and 32GB 1600 MHz DDR3 memory. The table shows that the example run times range between just over 2 minutes for 1,000 data points to around 3 hours and 45 minutes for 100,000 data points, which shows the increased computational effort required as the number of data points increases.

5. Example 1 - Quantifying the effect of air pollution on human health

This first example is an ecological regression problem, whose aim is to estimate the effect that

air pollution concentrations have on respiratory disease risk.

5.1. Data and exploratory analysis

For the purposes of delivering health care Scotland is split into 14 health boards, and this study focuses on the Greater Glasgow and Clyde health board, which contains the city of Glasgow and has a population of around 1.2 million people during the 2007 to 2011 study period. This health board is split into $K = 271$ intermediate geographies (IG), which are a key geography for the distribution of small-area statistics in Scotland and contain populations of between 2,468 and 9,517 people. The aim of this study is to quantify the impact of particulate matter air pollution concentrations on respiratory ill health, and we have yearly data for $N = 5$ years (2007 to 2011) for the $K = 271$ IGs. The disease and covariate data are freely available from Scottish Statistics (<http://statistics.gov.scot/>), while the particulate matter pollution concentrations are available from the Department for the Environment, Food and Rural Affairs (DEFRA, <https://uk-air.defra.gov.uk/data/pcm-data>).

The respiratory disease data are population level counts of the numbers of admissions to hospital in each IG and year with a primary diagnosis of respiratory disease, which corresponds to the International Classification of Disease tenth revision (ICD-10) codes J00-J99 and R09.1. However, the observed numbers of admissions in an IG and year depends on the size and demographic structure (e.g., age and sex profile) of the population living there, which is adjusted for using indirect standardisation. This involves computing the number of admissions that would be expected in each IG and year if national age and sex specific admissions rates applied. The observed and expected numbers of respiratory hospital admissions in the k th IG and t th year are denoted by (Y_{kt}, E_{kt}) respectively, and the Poisson model, $Y_{kt} \sim \text{Poisson}(E_{kt}R_{kt})$ is typically used to model these data. Here R_{kt} is the risk, relative to E_{kt} , of disease in IG k and year t , and a value of 1.2 corresponds to a 20% increased risk of disease. Operationally, the E_{kt} is included as an offset term in the model on the natural log scale, that is $O_{kt} = \ln(E_{kt})$ in (1).

The pollution data we utilise are yearly average modelled concentrations of particulate matter less than 10 microns (PM_{10}), which come from both anthropogenic (e.g., particles in car exhaust fumes) and natural (e.g., sea salt) sources. These data are estimates on a 1 kilometre square grid produced by a numerical simulation model, and full details can be found in [Ricardo-AEA \(2015\)](#). These 1 kilometre square estimates are spatially misaligned with the irregularly shaped polygonal IGs at which the disease and covariate data are available, and thus simple averaging is used to produce IG level PM_{10} estimates. Specifically, the median value of PM_{10} over the set of 1 kilometre grid squares having centroids lying within each IG was computed, and if an IG was too small to contain a grid square centroid then the nearest grid square was used as the concentration.

Finally, the data set contains 2 potential confounders that will be included in the model, both of which are proxy measures of socio-economic deprivation (poverty). The main confounder in spatio-temporal air pollution and health studies is smoking rates, but such smoking data are unavailable. However, smoking rates are strongly linked to socio-economic deprivation, and thus existing studies such as [Haining, Li, Maheswaran, Blangiardo, Law, Best, and Richardson \(2010\)](#) control for smoking effects using deprivation based proxy measures. Here we have

two measures of socio-economic deprivation, the average property price in each IG and year (in hundreds of thousands), and the proportion of the working age population who are in receipt of job seekers allowance (JSA), the latter being a benefit paid to individuals who are unemployed and seeking employment.

These data are available in the **CARBayesdata** package in the object `pollutionhealthdata`, and the package also contains the spatial polygon information for the Greater Glasgow and Clyde health board study region in the object `GGHB.IG` as a `SpatialPolygonsDataFrame` object. These data can be loaded using the following commands.

```
R> library("CARBayesdata")
R> library("sp")
R> data("GGHB.IG")
R> data("pollutionhealthdata")
```

The structure of `pollutionhealthdata` is shown below

```
R> head(pollutionhealthdata)
```

	IG	year	observed	expected	pm10	jsa	price
1	S02000260	2007	97	98.24602	14.02699	2.25	1.150
2	S02000261	2007	15	45.26085	13.30402	0.60	1.640
3	S02000262	2007	49	92.36517	13.30402	0.95	1.750
4	S02000263	2007	44	72.55324	14.00985	0.35	2.385
5	S02000264	2007	68	125.41904	14.08074	0.80	1.645
6	S02000265	2007	24	55.04868	14.08884	1.25	1.760

The first column labelled `IG` is the set of unique identifiers for each IG, while `observed` and `expected` are respectively the observed (e.g., Y_{kt}) and expected (e.g., E_{kt}) numbers of hospital admissions due to respiratory disease. An exploratory measure of disease risk is the standardised morbidity ratio (SMR), which for the k th IG and t th year is computed as $SMR_{kt} = Y_{kt}/E_{kt}$. However, due to the natural log link function in the Poisson model, the covariates are related in the model to the natural log of the SMR. Therefore the code below adds the SMR and the natural log of the SMR to the data set and produces a `pairs()` plot showing the relationship between the variables.

```
R> pollutionhealthdata$SMR <- pollutionhealthdata$observed
+ / pollutionhealthdata$expected
R> pollutionhealthdata$logSMR <- log(pollutionhealthdata$SMR)
R> par(pty="s", cex.axis=1.5, cex.lab=1.5)
R> pairs(pollutionhealthdata[, c(9, 5:7)], pch = 19, cex = 0.5,
+ lower.panel=NULL, panel=panel.smooth,
+ labels = c("ln(SMR)", "PM10", "JSA", "Price (*100,000)"))
```

The pairs plot shown in Figure 2 shows respectively positive and negative relationships between the natural log of SMR and the two deprivation covariates `jsa` and `price`, in both

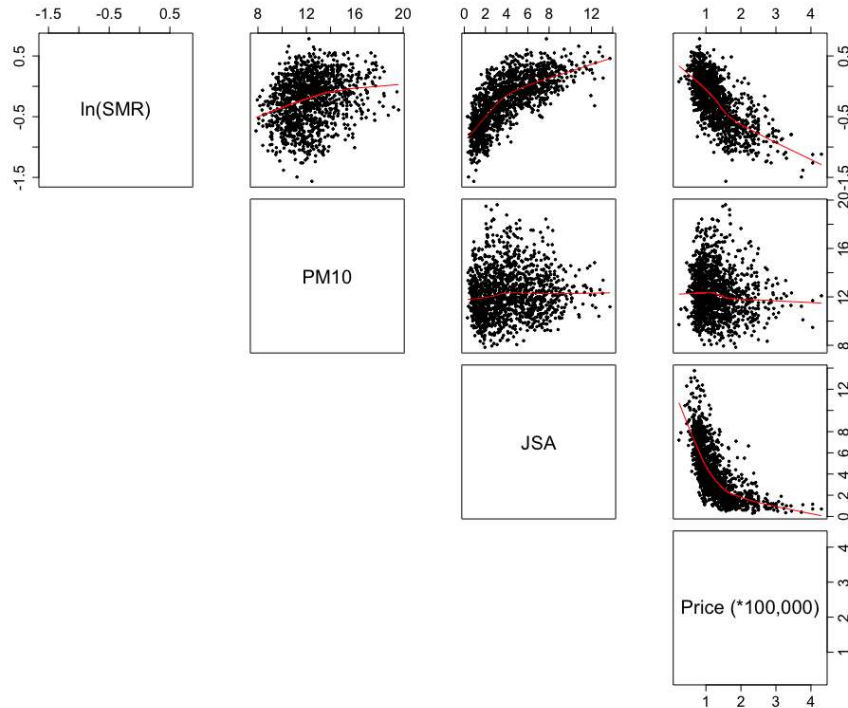


Figure 2: Scatterplot of the disease, pollution and covariate data.

cases suggesting that increasing levels of poverty are related to an increased risk of respiratory hospitalisation. There also appears to be a weak positive relationship between $\log(\text{SMR})$ and PM_{10} , while the only relationship that exists between the covariates is a negative non-linear one between `jса` and `price`. Next, it is of interest to visualise the average spatial pattern in the SMR over all five years, and the data can be appropriately aggregated using the `summarise()` function from the `dplyr` package using the code below. The aggregation is done by the second line, while the final line adds the aggregated averages to the `GGHB.IG SpatialPolygonsDataFrame` object.

```
R> library("dplyr")
R> SMR.av <- summarise(group_by(pollutionhealthdata,IG), SMR.mean =
+   mean(SMR))
R> GGHB.IG@data$SMR <- SMR.av$SMR.mean
```

Finally, the `splot()` function from the `sp` package can be used to draw a map of the average SMR over time using the code below:

```
R> l1 = list("SpatialPolygonsRescale", layout.north.arrow(),
+   offset = c(220000,647000), scale = 4000)
R> l2 = list("SpatialPolygonsRescale", layout.scale.bar(), offset =
+   c(225000, 647000), scale = 10000, fill = c("transparent","black"))
R> l3 = list("sp.text", c(225000,649000), "0")
```

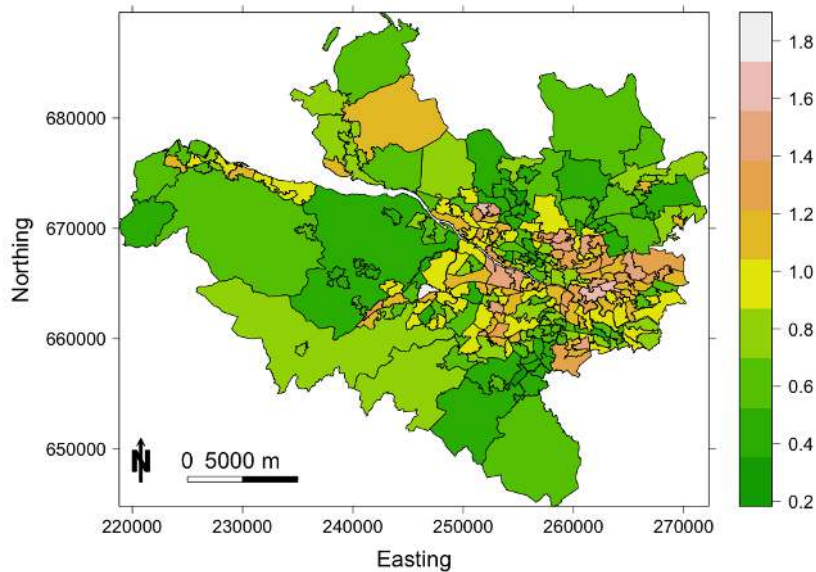


Figure 3: Map showing the average SMR over all five years from 2007 to 2011.

```
R> l4 = list("sp.text", c(230000,649000), "5000 m")
R> breakpoints <- seq(min(SMR.av$SMR.mean)-0.1, max(SMR.av$SMR.mean)+0.1,
+   length.out = 11)
R> spplot(GGHB.IG, "SMR", sp.layout = list(l1, l2, l3, l4),
+   xlab = "Easting", ylab = "Northing",
+   scales = list(draw = TRUE), at = breakpoints,
+   col.regions = terrain.colors(n = length(breakpoints)-1),
+   par.settings=list(fontsize=list(text=20)))
```

The first four lines in the code create a north-arrow and scale-bar to add to the map, the fifth line creates the breakpoints for the colour scheme into 10 equally sized intervals, while the last line draws the map. The resulting map is shown in Figure 3, where the green shaded areas are low risk ($SMR < 1$) while the orange to silver areas exhibit elevated risks ($SMR > 1$). The map shows that the main high-risk areas are in the east-end of Glasgow in the east of the study region and the Port Glasgow area in the far west of the region on the lower bank of the river Clyde (the white line running north west to south east). The analysis that follows requires us to compute the neighbourhood matrix \mathbf{W} and a `listw` object variant of the same spatial information, the latter being used in a hypothesis test for spatial autocorrelation. Both of these quantities can be computed from the `SpatialPolygonsDataFrame` object using functionality from the `spdep` package, and code to achieve this is shown below.

```
R> library("spdep")
R> W.nb <- poly2nb(GGHB.IG, row.names = SMR.av$IG)
R> W.list <- nb2listw(W.nb, style = "B")
R> W <- nb2mat(W.nb, style = "B")
```

Here W is a binary $K \times K$ neighbourhood matrix computed based on sharing a common border, and `W.list` is the `listw` object variant of this spatial information.

5.2. Assessing the presence of spatial autocorrelation

The spatio-temporal models in **CARBayesST** allow for spatio-temporal autocorrelation via random effects, which capture the remaining autocorrelation in the disease data after the effects of the known covariates have been accounted for. Therefore, we assess the presence of spatial autocorrelation in our data set by first computing the residuals from a simple overdispersed Poisson log-linear model that incorporates the covariate effects. This model is fitted using the code:

```
R> formula <- observed ~ offset(log(expected)) + jsa + price + pm10
R> modell <- glm(formula = formula, family = "quasipoisson",
+ data = pollutionhealthdata)
R> resid.glm <- residuals(modell)
R> summary(modell)$coefficients
```

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-0.59752496	0.054333524	-10.99735	5.287385e-27
jsa	0.06041994	0.003231475	18.69732	1.467196e-69
price	-0.28293191	0.018292049	-15.46748	8.225472e-50
pm10	0.04174701	0.003282156	12.71938	4.344434e-35

```
R> summary(modell)$dispersion
```

```
[1] 4.399561
```

The results show significant effects of all three covariates on disease risk, as well as substantial overdispersion with respect to the Poisson equal mean and variance assumption (over dispersion parameter equal to around 4.40). To quantify the presence of spatial autocorrelation in the residuals from this model we can compute Moran's I statistic (Moran 1950) and conduct a permutation test for each year of data separately. The permutation test has the null hypothesis of no spatial autocorrelation and an alternative hypothesis of spatial autocorrelation (either positive or negative), and is conducted using the `moran.mc()` function from the **spdep** package. The test can be implemented for the first year of residuals (2007) using the code below.

```
R> moran.mc(x = resid.glm[1:271], listw = W.list, nsim = 10000)
```

Monte-Carlo simulation of Moran I

```
data: resid.glm[1:271]
weights: W.list
number of simulations + 1: 10001
```

```
statistic = 0.10358, observed rank = 9971, p-value = 0.003
alternative hypothesis: greater
```


The estimated Moran's I statistic is 0.10358 and the p-value is less than 0.05, suggesting strong evidence of unexplained spatial autocorrelation in the residuals from 2007 after accounting for the covariate effects. Similar results were obtained for the other years and are not shown for brevity. We note that residual temporal autocorrelation could be assessed similarly for each IG, for example by computing the lag-1 autocorrelation coefficient, but with only 5 time points the resulting estimates would not be reliable. These results show that the assumption of independence is not valid for these data, and that spatio-temporal autocorrelation should be allowed for when estimating the covariate effects.

5.3. Spatio-temporal modelling with CARBayesST

We illustrate model fitting in **CARBayesST** by applying the `ST.CARar()` model to the data, details of which are given in Section 2. This model has previously been used to account for spatio-temporal autocorrelation in an air pollution and health study, for details see [Rushworth *et al.* \(2014\)](#). The model can be fitted with the following one-line function call, and we note that all data vectors (response, offset and covariates) have to be ordered so that the first K data points relate to all spatial units at time 1, the next K data points to all spatial units at time 2 and so on.

```
R> library("CARBayesST")
R> model2 <- ST.CARar(formula = formula, family = "poisson",
+   data = pollutionhealthdata, W = W, burn-in = 20000, n.sample = 220000,
+   thin = 10)
```

In the above code the covariate and offset component defined by `formula` is the same as for the simple Poisson log-linear model fitted earlier, and the neighbourhood matrix is binary and defined by whether or not two areas share a common border. The `ST.CARar()` model is run for 220,000 MCMC samples, the first 20,000 of which are removed by the burn-in period. The samples are then thinned by 10 to reduce the autocorrelation of the Markov chain, resulting in 20,000 samples for inference. A summary of the model results can be visualised using the `print()` function developed for **CARBayesST**, which gives a very similar summary to that produced in the **CARBayes** package.

```
R> print(model2)

#####
#### Model fitted
#####
Likelihood model - Poisson (log link function)
Latent structure model - Autoregressive CAR model
Regression equation - observed ~ offset(log(expected)) + jsa + price + pm10

#####
#### Results
#####
Posterior quantities for selected parameters and DIC
```

	Median	2.5%	97.5%	n.sample	% accept	n.effective	Geweke.diag
(Intercept)	-0.6604	-0.8376	-0.4826	20000	45.2	1084.6	1.5
jsa	0.0652	0.0548	0.0759	20000	45.2	960.6	2.1
price	-0.1958	-0.2376	-0.1544	20000	45.2	1970.9	0.4
pm10	0.0346	0.0221	0.0464	20000	45.2	1105.5	-2.7
tau2	0.0584	0.0493	0.0688	20000	100.0	5545.2	1.6
rho.S	0.5573	0.4011	0.7197	20000	43.7	3286.8	1.8
rho.T	0.7574	0.6955	0.8163	20000	100.0	11143.9	-1.4

DIC = 10397.92 p.d = 773.2666 LMPL = -4533.099

The output from the `print()` function shows that all three covariates exhibit relationships with disease risk, as none of the 95% credible intervals contain zero. Furthermore, the spatial (`rho.S`) and temporal (`rho.T`) dependence parameters exhibit relatively high values in the unit interval, suggesting that both spatial and temporal autocorrelation are present in these data after adjusting for the covariate effects. The model object `model2` is a list, and details of its elements are described in Section 3 of this paper. A list object containing the MCMC samples for each individual parameter and the fitted values are stored in `model2$samples`, and each element of this list corresponds to a different group of parameters and is stored as a `mcmc` object from the `coda` package. Applying the `summary()` function to this object yields:

```
R> summary(model2$samples)
```

	Length	Class	Mode
beta	16000	mcmc	numeric
phi	5420000	mcmc	numeric
rho	8000	mcmc	numeric
tau2	4000	mcmc	numeric
fitted	5420000	mcmc	numeric
Y	1	mcmc	logical

Here the `Y` object is `NA` as there are no missing Y_{kt} observations in this data set. If there had been say m missing values, then the `Y` component of the list would have contained m columns, with each one containing posterior predictive samples for one of the missing observations. The key interest in this analysis is the effects of the covariates on disease risk, which for Poisson models are typically presented as relative risks. The relative risk for an ϵ unit increase in a covariate with regression parameter β_s is given by the transformation $\exp(\epsilon\beta_s)$, and a relative risk of 1.02 corresponds to a 2% increased risk if the covariate increased by ϵ . The code below draws the posterior relative risk distributions for a one unit increase in each covariate, which are all realistic increases given the variation observed in the data in Figure 2.

```
R> colnames(model2$samples$beta) <- c("Intercept", "JSA", "Price", "PM10")
R> plot(exp(model2$samples$beta[ , -1]))
```

These distributions are displayed in Figure 4, where the left side shows trace-plots and the right side shows density estimates. Posterior medians and 95% credible intervals for the relative risks can be computed using the `summarise.samples()` function from the `CARBayes` package using the code below:

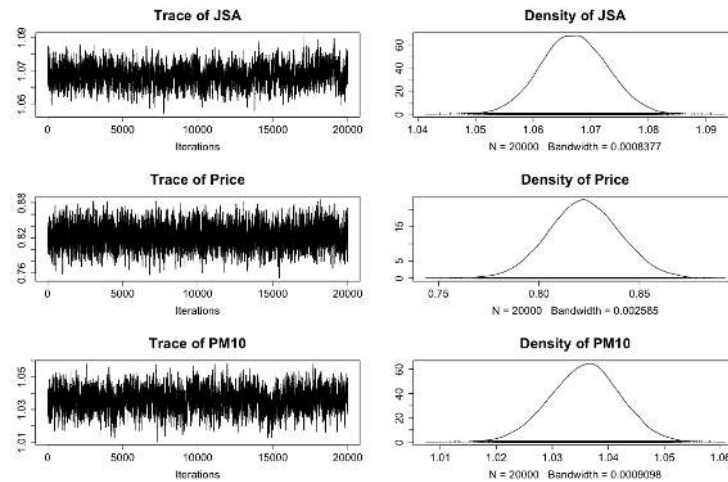


Figure 4: Posterior distributions for the covariate effects.

```
library("CARBayes")
R> parameter.summary <- summarise.samples(exp(model2$samples$beta[ , -1]),
+   quantiles = c(0.5, 0.025, 0.975))
R> round(parameter.summary$quantiles, 3)
```

```
      0.5 0.025 0.975
[1,] 1.067 1.056 1.079
[2,] 0.822 0.789 0.857
[3,] 1.035 1.022 1.047
```

The output above shows that the posterior median and 95% credible interval for the relative risk of a $1\mu\text{gm}^{-3}$ increase in PM_{10} is 1.035 (1.022, 1.047), suggesting that such an increase corresponds to 3.5% additional hospital admissions. The corresponding relative risk for a one percent increase in JSA is 1.067 (1.055, 1.079), while for a one hundred thousand pound increase in property price (the units for the property price data were in hundreds of thousands) the risk is 0.822 (0.789, 0.857). Thus, we find that increased air pollution concentrations are related, at this ecological level, to increased respiratory hospitalisation, while decreased socio-economic deprivation, as measured by both property price and JSA, is related to decreased risks of hospital admission.

6. Example 2 - Monitoring the changing state of the housing market

This second example focuses on the state of the housing market, specifically property sales, and aims to quantify its changing trend over time in an era that encompasses the global financial crisis that began in late 2007.

6.1. Data and exploratory analysis

The study region is the same as for the first example, namely the set of $K = 271$ intermediate geographies that make up the Greater Glasgow and Clyde health board. The data

also come from the same source (Scottish Statistics, <http://statistics.gov.scot/>), and include yearly observations of house sales from 2003 to 2013 inclusive. The response variable is the number of property sales Y_{kt} in each IG (indexed by k) and year (indexed by t), and we additionally have the total number of properties n_{kt} in each IG and year that will be used in the model as the offset term. We use the following Poisson log-linear model for these data, $Y_{kt} \sim \text{Poisson}(n_{kt}\theta_{kt})$, where θ_{kt} is the rate of property sales as a proportion of the total number of properties. We note that we have not used a binomial model here as a single property could sell more than once in a year, meaning that each property does not constitute a Bernoulli trial. Thus θ_{kt} is not strictly the proportion of properties that sell in a year, but is on approximately the same scale for interpretation purposes.

These data are available in the **CARBayesdata** package in the object `salesdata`, as is the spatial polygon information for the Greater Glasgow and Clyde health board study region (in the object `GGHB.IG`). These data can be loaded using the following commands.

```
R> library("CARBayesdata")
R> library("sp")
R> data("GGHB.IG")
R> data("salesdata")
```

The `data.frame` `salesdata` contain 4 columns, the intermediate geography code (IG), the year the data relate to (`year`), the number of property sales (`sales`, Y_{kt}) and the total number of properties (`stock`, n_{kt}). We visualise the temporal trend in these data using the code below, where the first line creates the raw rate of property sales as a proportion of the total number of properties.

```
R> salesdata$salesprop <- salesdata$sales / salesdata$stock
R> boxplot(salesdata$salesprop ~ salesdata$year, range = 0, xlab = "Year",
+         ylab = "Property sales rate",
+         col = "darkseagreen", border = "navy")
```

This produces the boxplot shown in Figure 5, where the global financial crisis began in 2007. The plot shows a clear step-change in property sales between 2007 and 2008, as sales were increasing up to and including 2007, before markedly decreasing in subsequent years. Sales in the last year of 2013 show slight evidence of increasing relative to the previous 4 years, possibly suggesting the beginning of an upturn in the market. Also there appears to be a change in the level of spatial variation from year to year, with larger amounts of spatial variation observed before the global financial crisis. The spatial pattern in the average (over time) rate of property sales as a proportion of the total number of properties is shown in Figure 6, which was created using the code below.

```
R> library("dplyr")
R> salesprop.av <- summarise(group_by(salesdata, IG),
+   salesprop.mean = mean(salesprop))
R> GGHB.IG@data$sales <- salesprop.av$salesprop.mean
R> l1 = list("SpatialPolygonsRescale", layout.north.arrow(),
```

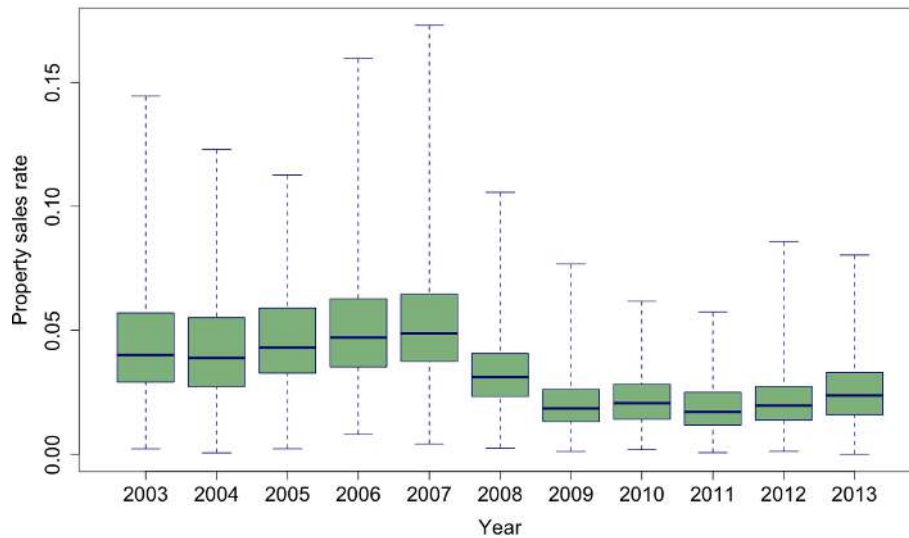


Figure 5: Boxplots showing the temporal trend in the raw rate of property sales as a proportion of the total number of properties between 2003 and 2013.

```

+   offset = c(220000,647000), scale = 4000)
R> l2 = list("SpatialPolygonsRescale", layout.scale.bar(),
+   offset = c(225000,647000), scale = 10000,
+   fill = c("transparent","black"))
R> l3 = list("sp.text", c(225000,649000), "0")
R> l4 = list("sp.text", c(230000,649000), "5000 m")
R> breakpoints <- c(0, quantile(salesprop.av$salesprop.mean,
+   seq(0.1, 0.9, 0.1)), 0.1)
R> spplot(GGHB.IG, "sales", sp.layout=list(l1, l2, l3, l4),
+   xlab="Easting", ylab="Northing",
+   scales=list(draw = TRUE), at=breakpoints,
+   col.regions=terrain.colors(n=length(breakpoints)-1),
+   par.settings=list(fontsize=list(text=20)))

```

The first 3 lines of code create the IG specific temporal averages using the **dplyr** package and add it to the GGHB.IG `SpatialPolygonsDataFrame` object, both of which are very similar to code used in the first example. The remaining lines of code produce the map shown in Figure 6, again using code very similar to that in Example 1. The map and its colour key shows that the property sales data are skewed to the right, as the colour key chosen has unequal groups. Initially an equally-spaced colour scheme was created, but that showed little colour variation, hence the use of the unequal quantile based colour key. Secondly, the map shows a largely similar pattern to that seen for respiratory disease risk in Figure 3, with areas that exhibit relative high sales rates largely being the same as those that exhibit relatively low disease risk. Figures 5 and 6 highlight the change in temporal dynamics and the spatial structure in property sales in Glasgow, and we now apply the `ST.CARsepspatial()` model from **CARBayesST** to more formally quantify these features.

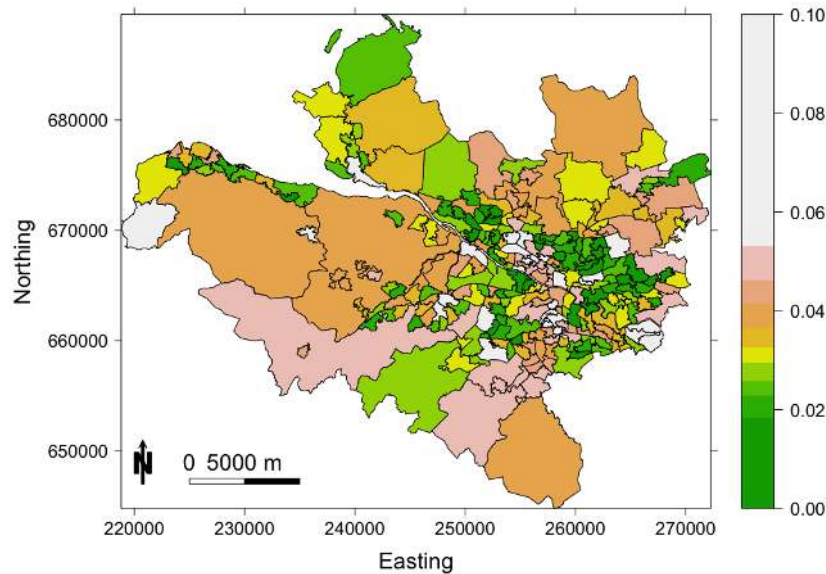


Figure 6: Map showing the average (between 2003 to 2013) raw rate of property sales as a proportion of the total number of properties.

6.2. Quantifying the changing temporal trends and spatial patterns in sales rates

The extent to which the region-wide average level of sales and its spatial variation and spatial structure changes over time can be assessed by applying the model proposed by [Napier *et al.* \(2016\)](#) to the data, which can be implemented using the `ST.CARsepspatial()` function. Before fitting this model we need to create the neighbourhood matrix using the following code:

```
R> library("spdep")
R> W.nb <- poly2nb(GGHB.IG, row.names = salesprop.av$salesprop.mean)
R> W <- nb2mat(W.nb, style = "B")
```

Then the model can be fitted using the code below, where inference is again based on 20,000 post burn-in and thinned MCMC samples.

```
R> library("CARBayesST")
R> formula <- sales ~ offset(log(stock))
R> model1 <- ST.CARsepspatial(formula = formula, family = "poisson",
+   data = salesdata, W = W, burn-in = 20000, n.sample = 220000, thin = 10)
```

A summary of the model fit can be obtained using the `print()` function, and the output is similar to that shown in example 1 and is not shown for brevity. The model fitted represents the estimated rate of property sales by

$$\theta_{kt} = \exp(\beta_1 + \phi_{kt} + \delta_t),$$

which is the sum of an overall intercept term β_1 , a space-time effect ϕ_{kt} with a time period specific variance, and a region-wide temporal trend δ_t . The mean and standard deviation of $\{\theta_{kt}\}$ over space for each year is computed by the following code, which produces the posterior median and a 95% credible interval for each quantity for each year.

```
R> trend.mean <- array(NA, c(11, 3))
R> trend.sd <- array(NA, c(11, 3))
R> for(i in 1:11)
+   {
+     posterior <- exp(model1$samples$phi[ , ((i-1) * 271 + 1):(i * 271)] +
+       matrix(rep(model1$samples$beta + model1$samples$delta[ , i], 271),
+         ncol = 271, byrow = FALSE))
+     trend.mean[i, ] <- quantile(apply(posterior, 1, mean),
+       c(0.5, 0.025, 0.975))
+     trend.sd[i, ] <- quantile(apply(posterior, 1, sd), c(0.5, 0.025, 0.975))
+   }
```

These temporal trends in the average rate of property sales and its level of spatial variation can be plotted by the following code, and the result is displayed in Figure 7.

```
R> par(mfrow=c(2, 1))
R> plot(jitter(salesdata$year), salesdata$salesprop, pch=19, cex=0.2,
+   col="blue", xlab="Year", main="(a)", ylab="Average sales rate",
+   ylim=c(0, 0.11), cex.axis=1.5, cex.lab=1.5, cex.main=1.5)
R> lines(2003:2013, trend.mean[ ,1], col="red", type="l")
R> lines(2003:2013, trend.mean[ ,2])
R> lines(2003:2013, trend.mean[ ,3])
R> plot(2003:2013, trend.sd[ ,1], col = "red", type = "l", xlab = "Year",
+   main = "(b)", ylab = "Spatial standard deviation", ylim = c(0, 0.06),
+   cex.axis=1.5, cex.lab=1.5, cex.main=1.5)
R> lines(2003:2013, trend.sd[ ,2])
R> lines(2003:2013, trend.sd[ ,3])
```

The figure shows that both the region-wide average (panel (a)) and the level of spatial variation (as measured by the spatial standard deviation, panel (b)) in property sales rates show the same underlying trend, with maximum values just before the global financial crisis in 2007, and then sharp decreases afterwards. This provides some empirical evidence that the global financial crisis negatively affected the housing market in Greater Glasgow, with average sales rates dropping from just under 6.0% in 2007 to 3.3% in 2008. The spatial standard deviation also dropped from 0.050 to 0.036 over the same two-year period, suggesting that the global financial crisis had the effect of reducing the disparity in sales rates in different regions of Greater Glasgow. We note that when measuring the spatial standard deviation we have not simply presented the posterior distribution of τ_t^2 , because this relates to the linear predictor scale and thus the results change after the inverse logit transformation to the $\{\theta_{kt}\}$ scale due to a non-constant mean level (due to the δ_t term).

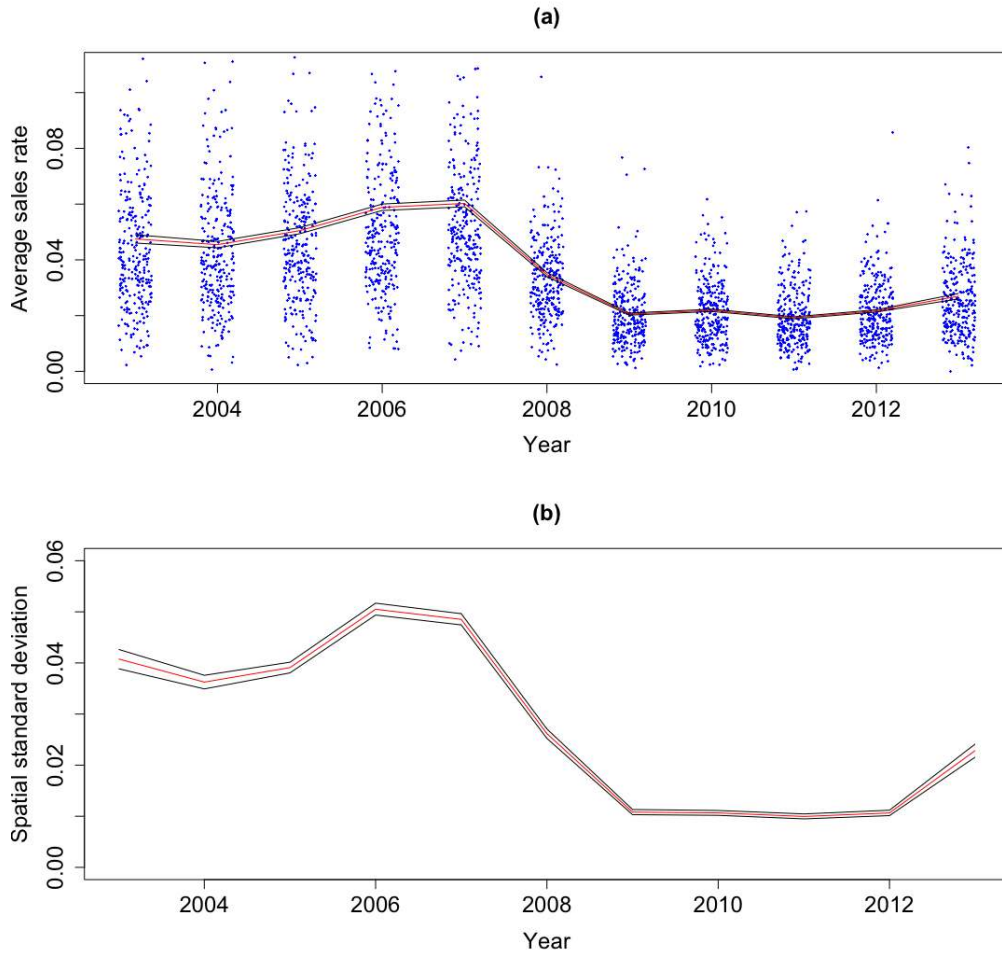


Figure 7: Posterior median (red) and 95% credible interval (black) for the temporal trend in: (a) region-wide average property sales rates; and (b) spatial standard deviation in property sales rates. In panel (a) the blue dots are the raw sales proportions for each area and year (jittered in the x direction to improve the presentation).

The posterior median sales rate $\{\theta_{kt}\}$ is computed and then plotted for the 6 odd numbered years using the code below, where the colour scale is the same as for Figure 6. The first 3 lines create a `data.frame` object of estimated sales rates, while the fourth line adds this sales rate to the `SpatialPolygonsDataFrame` object. Finally, the last line plots the estimated sales rates for odd numbered years.

```
R> rate.est <- matrix(model1$fitted.values / salesdata$stock, nrow = nrow(W),
+   byrow = FALSE)
R> rate.est <- as.data.frame(rate.est)
R> colnames(rate.est) <- c("r2003", "r2004", "r2005", "r2006", "r2007",
+   "r2008", "r2009", "r2010", "r2011", "r2012", "r2013")
R> GGHB.IG@data <- data.frame(GGHB.IG@data, rate.est)
R> splot(GGHB.IG, c("r2011", "r2013", "r2007", "r2009", "r2003", "r2005"),
+   names.attr = c("Rate 2011", "Rate 2013", "Rate 2007", "Rate 2009",
+   "Rate 2003", "Rate 2005"), sp.layout = list(11, 12, 13, 14),
+   xlab = "Easting", ylab = "Northing",
+   scales = list(draw = TRUE), at = breakpoints,
+   col.regions = terrain.colors(n = length(breakpoints - 1)),
+   par.settings=list(fontsize=list(text=15))
```

The maps are displayed in Figure 8, and show the clear changing spatial pattern in sales rates over time. The spatial rates for 2003 to 2007 are largely consistent, but a clear step-change is evident between 2007 and 2009, which incorporates the start of the global financial crisis. The figure shows that the downturn in sales rates continues into 2011 but that the property market is beginning an upturn by 2013. So in conclusion, Figures 7 and 8 show that the global financial crisis in 2007 resulted in a downturn in both the region-wide rate of sales and the level of spatial variation in sales across Glasgow, but that areas of high sales, such as the west-end of Glasgow (the thin strip of orange shaded areas north of the river in 2009), always remained higher than other parts of the study region.

7. Discussion

This paper has presented the **CARBayesST** software, which is the first software package dedicated to fitting spatio-temporal CAR type models to areal unit data. This paper has outlined the class of models that can be implemented by the software together with the Bayesian inferential framework used to fit these models. The key advantage of this package, compared to say implementing the models in BUGS, is its ease of use, which includes fitting models with a one-line function call, specifying the spatial adjacency information via a single matrix, and the ability to fit multiple models addressing different questions about the data in a common software environment. The paper has presented a small simulation study to illustrate the correctness of the software, as well as two fully worked examples based on quantifying the effect of air pollution on human health and the changing nature of the housing market.

Future development of the software will be in two main directions. First, as the literature on spatio-temporal modelling advances we aim to increase the number of spatio-temporal models that can be implemented, giving the user an even wider set of modelling tools. Second,

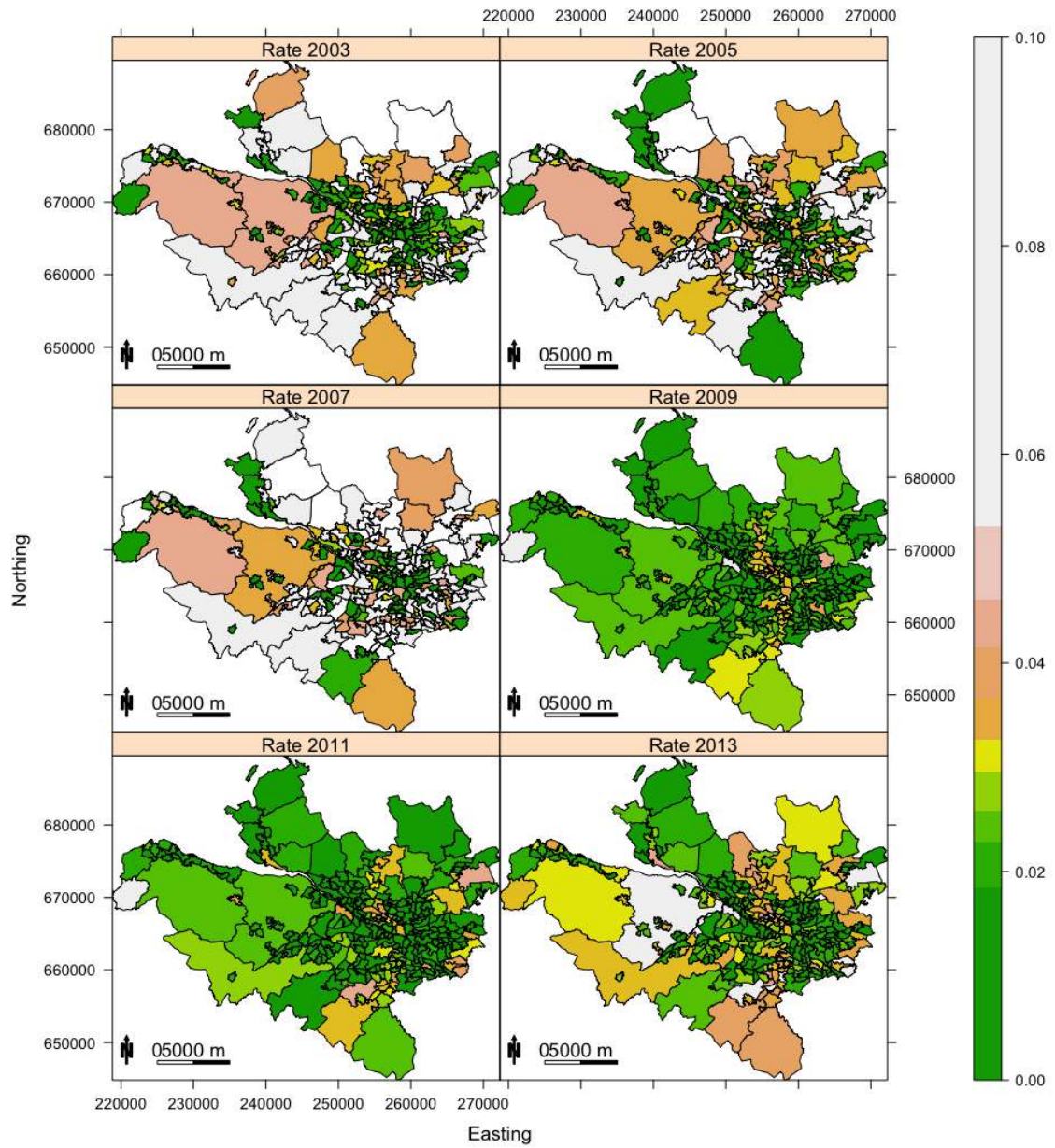


Figure 8: Maps showing the changing spatial evolution of the posterior median estimated sales rates $\{\theta_{kt}\}$ for odd numbered years.

with the rapid increase in the availability of small-area data, we aim to develop a suite of multivariate space-time models (MVST). The development of MVST methodology for areal unit data is in its infancy, and the ability to jointly examine the spatio-temporal patterns in multiple response variables simultaneously allows one to address questions that cannot be addressed by single variable models. For example, in a public health context it allows one to estimate overall and disease specific spatio-temporal patterns in disease risk, allowing one to see which areas repeatedly signal at high risk for all diseases and which exhibit elevated risks for only one disease. In the housing context of example 2 a MVST approach would allow one to extend the analysis carried out by different property types, e.g., flats, terraced houses, etc, which would allow more insight to be gained about the exact nature of the housing market.

8. Acknowledgements

The Development of the **CARBayesST** software and this paper were supported by the UK Engineering and Physical Sciences Research Council (EPSRC, grant EP/J017442/1) and the UK Medical Research Council (MRC, grant MR/L022184/1). The data and shapefiles used in this paper were provided by the Scottish Government via their Scottish Statistics website (<http://statistics.gov.scot/>).

References

- Anderson C, Lee D, Dean N (2014). “Identifying Clusters in Bayesian Disease Mapping.” *Biostatistics*, **15**, 457–469.
- Bakar K, Sahu S (2015). “spTimer: Spatio-Temporal Bayesian Modeling Using R.” *Journal of Statistical Software*, **63**(1). URL <https://www.jstatsoft.org/index.php/jss/article/view/v063i15>.
- Bates D, Mächler M, Bolker B, Walker S (2015). “Fitting Linear Mixed-Effects Models Using lme4.” *Journal of Statistical Software*, **67**(1). URL <https://www.jstatsoft.org/index.php/jss/article/view/v067i01>.
- Bernardinelli L, Clayton D, Pascutto C, Montomoli C, Ghislandi M, Songini M (1995). “Bayesian Analysis of Space-Time Variation in Disease Risk.” *Statistics in Medicine*, **14**, 2433–2443.
- Besag J, Higdon D (1999). “Bayesian Analysis of Agricultural Field Experiments.” *Journal of the Royal Statistical Society B*, **61**, 691–746.
- Besag J, York J, Mollié A (1991). “Bayesian Image Restoration with Two Applications in Spatial Statistics.” *Annals of the Institute of Statistics and Mathematics*, **43**, 1–59.
- Bivand R, Lewin-Koh N (2015). *maptools: Tools for Reading and Handling Spatial Objects*. R package version 0.8-36, URL <https://CRAN.R-project.org/package=maptools>.
- Bivand R, Pebesma E, Gomez-Rubio V (2013). *Applied Spatial Data Analysis with R*. 2nd edition. Springer-Verlag, New York.

- Bivand R, Piras G (2015). “Comparing Implementations of Estimation Methods for Spatial Econometrics.” *Journal of Statistical Software*, **63**(18). URL <http://www.jstatsoft.org/v63/i18/>.
- Charras-Garrido M, Abrial D, de Goer J (2012). “Classification Method for Disease Risk Mapping Based on Discrete Hidden Markov Random Fields.” *Biostatistics*, **13**, 241–255.
- Choi J, Lawson AB (2011). “Evaluation of Bayesian Spatial-Temporal Latent Models in Small Area Health Data.” *Environmetrics*, **22**, 1008–1022.
- Congdon P (2005). *Bayesian Models for Categorical Data*. 1st edition. John Wiley & Sons.
- Croissant Y, Millo G (2008). “Panel Data Econometrics in R: The plm Package.” *Journal of Statistical Software*, **27**(1). URL <https://www.jstatsoft.org/index.php/jss/article/view/v027i02>.
- Eddelbuettel D, Francois R (2011). “Rcpp: Seamless R and C++ Integration.” *Journal of Statistical Software*, **40**, 8.
- Finley A, Banerjee S, Gelfand A (2015). “spBayes for Large Univariate and Multivariate Point-Referenced Spatio-Temporal Data Models.” *Journal of Statistical Software*, **63**(1). URL <https://www.jstatsoft.org/index.php/jss/article/view/v063i13>.
- Furrer R, Sain SR (2010). “spam: A Sparse Matrix R Package with Emphasis on MCMC Methods for Gaussian Markov Random Fields.” *Journal of Statistical Software*, **36**(10), 1–25. URL <http://www.jstatsoft.org/v36/i10/>.
- Gelman A, Carlin J, Stern H, Dunson D, Vehtari A, Rubin D (2013). *Bayesian Data Analysis*. 3rd edition. Chapman & Hall / CRC.
- Geweke J (1992). “Evaluating the Accuracy of Sampling-Based Approaches to the Calculation of Posterior Moments.” In *Bayesian Statistics*, pp. 169–193. University Press.
- Haining R, Li G, Maheswaran R, Blangiardo M, Law J, Best N, Richardson S (2010). “Inference from Ecological Models: Estimating the Relative Risk of Stroke from Air Pollution Exposure Using Small Area Data.” *Spatial and Spatio-Temporal Epidemiology*, **1**, 123–131, DOI:10.1016/j.sste.2010.03.006.
- Knorr-Held L (2000). “Bayesian Modelling of Inseparable Space-Time Variation in Disease Risk.” *Statistics in Medicine*, **19**, 2555–2567.
- Lee D (2013). “CARBayes: An R Package for Bayesian Spatial Modelling with Conditional Autoregressive Priors.” *Journal of Statistical Software*, **55**, 13.
- Lee D (2016). *CARBayesdata: Data Used in the Vignettes Accompanying the CARBayes and CARBayesST Packages*. R package version 2.0.
- Lee D, Ferguson C, Mitchell R (2009). “Air Pollution and Health in Scotland: A Multicity Study.” *Biostatistics*, **10**, 409–423.
- Lee D, Lawson A (2016). “Quantifying the Spatial Inequality and Temporal Trends in Maternal Smoking Rates in Glasgow.” *Annals of Applied Statistics*, **10**, 1427–1446.

- Lee D, Minton J, Pryce G (2015). “Bayesian Inference for the Dissimilarity Index in the Presence of Spatial Autocorrelation.” *Spatial Statistics*, **11**, 81–95.
- Leroux BG, Lei X, Breslow N (2000). *Statistical Models in Epidemiology, the Environment, and Clinical Trials*, chapter Estimation of Disease Rates in Small Areas: A new Mixed Model for Spatial Dependence, pp. 179–191. Springer-Verlag, New York. URL http://dx.doi.org/10.1007/978-1-4612-1284-3_4.
- Li G, Best N, Hansell A, Ahmed I, Richardson S (2012). “BaySTDetect: Detecting Unusual Temporal Patterns in Small Area Data via Bayesian Model Choice.” *Biostatistics*, **13**, 695–710.
- Lunn D, Spiegelhalter D, Thomas A, Best N (2009). “The BUGS Project: Evolution, Critique and Future Directions .” *Statistics in Medicine*, **28**, 3049–3082.
- Millo G, Piras G (2012). “splm: Spatial Panel Data Models in R.” *Journal of Statistical Software*, **47**(1), 1–38. URL <http://www.jstatsoft.org/v47/i01/>.
- Moran P (1950). “Notes on Continuous Stochastic Phenomena.” *Biometrika*, **37**, 17–23.
- Napier G, Lee D, Robertson C, Lawson A, Pollock K (2016). “A Model to Estimate the Impact of Changes in MMR Vaccination Uptake on Inequalities in Measles Susceptibility in Scotland.” *Statistical Methods in Medical Research*, **25**, 1185–1200.
- Novomestky F (2012). *matrixcalc: Collection of Functions for Matrix Calculations*. R package version 1.0-3, URL <https://CRAN.R-project.org/package=matrixcalc>.
- Novomestky F, Nadarajah S (2012). *truncdist: Truncated Random Variables*. R package version 1.0-1, URL <https://CRAN.R-project.org/package=truncdist>.
- Paul M, Meyer S (2016). “hhh4: An Endemic-Epidemic Modelling Framework for Infectious Disease Counts, <http://CRAN.R-project.org/package=surveillance>.”
- Pinheiro J, Bates D, DebRoy S, Sarkar D, R Core Team (2015). *nlme: Linear and Nonlinear Mixed Effects Models*. R package version 3.1-121, URL <http://CRAN.R-project.org/package=nlme>.
- Plummer M, Best N, Cowles K, Vines K (2006). “CODA: Convergence Diagnosis and Output Analysis for MCMC.” *R News*, **6**(1), 7–11. URL <http://CRAN.R-project.org/doc/Rnews/>.
- R Core Team (2016). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL <http://www.R-project.org/>.
- Ricardo-AEA (2015). “Technical Report on UK Supplementary Assessment under the Air Quality Directive (2008/50/EC), the Air Quality Framework Directive (96/62/EC) and Fourth Daughter Directive (2004/107/EC) for 2013.” <https://uk-air.defra.gov.uk/data/pcm-data>.
- Robert C, Casello G (2010). *Introducing Monte Carlo Methods with R*. 1st edition. Springer.
- Roberts G, Rosenthal J (1998). “Optimal scaling of discrete approximations to the Langevin diffusions.” *Journal of the Royal Statistical Society B*, **60**, 255–268.

- Rue H, Martino S, Chopin N (2009). “Approximate Bayesian Inference for Latent Gaussian Models Using Integrated Nested Laplace Approximations (with discussion).” *Journal of the Royal Statistical Society B*, **71**, 319–392.
- Rushworth A, Lee D, Mitchell R (2014). “A Spatio-Temporal Model for Estimating the Long-Term Effects of Air Pollution on Respiratory Hospital Admissions in Greater London.” *Spatial and Spatio-temporal Epidemiology*, **10**, 29–38.
- Rushworth A, Lee D, Sarran C (2017). “An Adaptive Spatio-Temporal Smoothing Model for Estimating Trends and Step Changes in Disease Risk.” *Journal of the Royal Statistical Society C*, **66**, 141–157.
- Schabenberger H (2009). *spatcounts: Spatial Count Regression*. R package version 1.1, URL <http://CRAN.R-project.org/package=spatcounts>.
- Spiegelhalter D, Best N, Carlin B, Van der Linde A (2002). “Bayesian Measures of Model Complexity and Fit.” *Journal of the Royal Statistical Society B*, **64**, 583–639.
- Trautmann H, Steuer D, Mersmann O, Bornkamp B (2014). *truncnorm: Truncated normal distribution*. R package version 1.0-7, URL <https://CRAN.R-project.org/package=truncnorm>.
- Venables WN, Ripley BD (2002). *Modern Applied Statistics with S*. Fourth edition. Springer-Verlag, New York. URL <http://www.stats.ox.ac.uk/pub/MASS4>.
- Wakefield J (2007). “Disease Mapping and Spatial Regression with Count Data.” *Biostatistics*, **8**, 158–183.
- Wall M (2004). “A Close Look at the Spatial Structure Implied by the CAR and SAR Models.” *Journal of Statistical Planning and Inference*, **121**, 311–324.
- Watanabe S (2010). “Asymptotic Equivalence of the Bayes Cross-Validation and Widely Applicable Information Criterion in Singular Learning Theory.” *Journal of Machine Learning Research*, **11**, 3571–3594.
- Wickham H (2011). “testthat: Get Started with Testing.” *The R Journal*, **3**, 5–10. URL http://journal.r-project.org/archive/2011-1/RJournal_2011-1_Wickham.pdf.
- Wickham H, Francois R (2015). *dplyr: A Grammar of Data Manipulation*. R package version 0.4.3, URL <https://CRAN.R-project.org/package=dplyr>.

Affiliation:

Duncan Lee
School of Mathematics and Statistics
15 University Gardens
University of Glasgow
Glasgow
G12 8QQ, Scotland

E-mail: Duncan.Lee@glasgow.ac.uk

URL: <http://www.gla.ac.uk/schools/mathematicsstatistics/staff/duncanlee/>

Alastair Rushworth
Mathematics and Statistics
26 Richmond Street
University of Strathclyde
Glasgow
G1 1XH, Scotland

E-mail: alastair.rushworth@strath.ac.uk

URL: <https://www.strath.ac.uk/staff/rushworthalastairmr/>

Gary Napier
School of Mathematics and Statistics
15 University Gardens
University of Glasgow
Glasgow
G12 8QQ, Scotland

E-mail: Gary.Napier@glasgow.ac.uk

URL: <http://www.gla.ac.uk/schools/mathematicsstatistics/staff/garynapier/>