# CARBayesST version 4.0: Spatio-Temporal Areal Unit Modelling in **R** with Conditional Autoregressive Priors

**Duncan Lee, Alastair Rushworth, Gary Napier and William Pettersson**
University of Glasgow

### Abstract

This is a vignette for the R package **CARBayesST** version 4.0, and is an updated version of a paper in the Journal of Statistical Software in 2018, Volume 84, Issue 9. The package has been developed to model spatio-temporal data relating to a set of non-overlapping areal units that are observed over multiple time periods. These data typically contain spatio-temporal autocorrelation, and this package provides a suite of models for capturing this autocorrelation via random effects that are assigned spatio-temporal extensions of conditional autoregressive (CAR) priors within a hierarchical Bayesian model. **CARBayesST** is the first dedicated R package for modelling spatio-temporal areal unit data in a Bayesian setting using Markov chain Monte carlo (MCMC) simulation. The software can fit a range of models with different spatio-temporal structures, including average spatial and temporal trends, and the identification of clusters of areal units that exhibit elevated values. This vignette outlines the class of models that the software can implement, before applying them to exemplar case studies. The new functionality in version 4.0 is that for most models you can now generate multiple MCMC chains in one model run using the `n.chains` argument, and the multiple chains can either be run in parallel (if `n.cores` > 1) for increased computational speed or in series (`n.cores=1`) if you do not want to use multiple cores.

*Keywords*: Bayesian inference, conditional autoregressive priors, 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 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 have become increasingly available in recent times, due to the creation of databases such as Scottish Statistics (`http://statistics.gov.scot/`), 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), identifying clusters of contiguous areal units that exhibit an elevated risk of disease compared with neighbouring areas (see 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).

However, a common challenge when modelling these data is spatio-temporal autocorrelation, whereby 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 due to unmeasured confounding, neighbourhood effects and grouping effects. A Bayesian inferential approach is typically taken to modelling these data, where the spatio-temporal autocorrelation 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 this autocorrelation. A range of different models have been proposed to date, and the most appropriate model depends on the goals of the analysis.

An array of freely available software can now implement purely spatial areal unit models, including BUGS (Lunn, Spiegelhalter, Thomas, and Best 2009) and the R packages **CARBayes** (Lee 2013) and **R-INLA** (Rue, Martino, and Chopin 2009). However, specialist software for spatio-temporal modelling with conditional autoregressive priors using Markov chain Monte Carlo (MCMC) simulation was not previously available, which has motivated the development of **CARBayesST** (Lee, Rushworth, and Napier 2018). Section 2 summarises the models that can be implemented for univariate spatio-temporal data, while Section 3 summarises the models available for multivariate spatio-temporal data. Section 4 provides an overview of the software and its functionality. 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 developments.

## 2. Spatio-temporal models for univariate areal unit data

This section outlines the class of univariate models that **CARBayesST** can implement, and in all cases inference is in a Bayesian setting using 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 $k = 1, \ldots, K$ non-overlapping areal units $\mathcal{S} = \{\mathcal{S}_1, \ldots, \mathcal{S}_K\}$, and data are recorded for each unit for $t = 1, \ldots, N$ 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 $\mathbf{Y} = (\mathbf{Y}_1, \ldots, \mathbf{Y}_N)_{K \times N}$, where $\mathbf{Y}_t = (Y_{1t}, \ldots, Y_{Kt})$ denotes the $K \times 1$ column vector of observations for all $K$ spatial units for time period $t$. Next, a matrix of known offsets is denoted by $\mathbf{O} = (\mathbf{O}_1, \ldots, \mathbf{O}_N)_{K \times N}$, where similarly $\mathbf{O}_t = (O_{1t}, \ldots, O_{Kt})$ denotes the $K \times 1$ column vector of offsets for time period $t$. Finally, $\mathbf{x}_{kt} = (x_{kt1}, \ldots, x_{ktp})$ 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 (e.g., squared) or spline basis functions (e.g., using `ns()`) of covariates to $\mathbf{x}_{kt}$. **CARBayesST** can fit the following generalised linear mixed model to these data.

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

where $\mu_{kt} = \mathbf{E}(Y_{kt})$ denotes the expectation of $Y_{kt}$. The vector of covariate regression parameters are denoted by $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$, and a multivariate Gaussian prior is assumed with mean $\boldsymbol{\mu}_\beta$ and diagonal covariance matrix $\boldsymbol{\Sigma}_\beta$ that can be chosen by the user. The $\psi_{kt}$ 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 $\boldsymbol{\psi} = (\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_N)$, where $\boldsymbol{\psi}_t = (\psi_{1t}, \ldots, \psi_{Kt})$. **CARBayesST** can fit a number of different spatio-temporal structures for $\psi_{kt}$, which are outlined in Section 2.2 below. The software can implement Equation (1) for binomial, Gaussian and Poisson data models, 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** - $Y_{kt} \sim \mathrm{N}(\mu_{kt}, \nu^2)$ 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 $(n_{kt}, \theta_{kt})$ 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 $\nu^2$ is the observation variance. An inverse-gamma$(a, b)$ prior is specified for the Gaussian variance $\nu^2$, 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

Spatial autocorrelation is controlled by a symmetric non-negative $K \times K$ neighbourhood or adjacency matrix $\mathbf{W} = (w_{kj})$, where $w_{kj}$ represents the spatial closeness between areal units $(\mathcal{S}_k, \mathcal{S}_j)$. Larger values represent spatial closeness between the two areas in question, where as smaller or zero values correspond to areas that are not spatially close. Most often $\mathbf{W}$ 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, $w_{kk} = 0$ for all areal units $k$. Under this binary specification the values of $(\psi_{kt}, \psi_{jt})$ for spatially adjacent areal units (where $w_{kj} = 1$) are partially 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 $|j - t| = 1$

| Model | Equation | Description |
|---|---|---|
| ST.CARlinear() | (2) | This model is similar to that proposed by Bernardinelli, Clayton, Pascutto, Montomoli, Ghislandi, and Songini (1995), and represents the spatio-temporal pattern in the mean response with spatially varying linear time trends. Allowable data models are binomial, Gaussian and Poisson. |
| 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. Allowable data models are binomial, Gaussian and Poisson. |
| ST.CARsepspatial() | (4) | This model is that proposed by Napier, Lee, Robertson, Lawson, and Pollock (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. Allowable data models are binomial and Poisson. |
| ST.CARar() | (5) | This model is based on that proposed by Rushworth, Lee, and Mitchell (2014), and represents the spatio-temporal pattern in the mean response with a single set of spatially and temporally autocorrelated random effects. These effects follow a multivariate autoregressive process of order 1 or 2. Allowable data models are binomial, Gaussian and Poisson. |
| ST.CARadaptive() | (8) | This model is that proposed by Rushworth, Lee, and Sarran (2017), and has the same spatio-temporal random effect structure as the ST.CARar() model, but with an adaptive spatial autocorrelation structure via simultaneously estimating $\mathbf{W}$. Allowable data models are binomial, Gaussian and Poisson. |
| ST.CARlocalised() | (10) | This model is that proposed by Lee and Lawson (2016), and has the same spatio-temporal random effect structure as the ST.CARar() model, with an additional piecewise constant intercept term. Allowable data models are binomial and Poisson. |
| ST.CARclustrends() | (13) | This model is that proposed by Napier, Lee, Robertson, and Lawson (2019), and contains a mixture model for clustering areas based on their temporal trends in disease risk, where the candidate trend functions have fixed parametric forms or constrained shapes. Allowable data models are binomial and Poisson. |

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

and $d_{tj} = 0$ otherwise.

**CARBayesST** can fit the models for $\boldsymbol{\psi}$ outlined in Table 1, and full details for each one are given below. 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 (**Y**) for these models, and are estimated using data augmentation (Tanner and Wong 1987). 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 autocorrelated linear time trends for each areal unit. Thus it is appropriate if the goal of the analysis is to estimate which areas exhibit increasing or decreasing (linear) trends in the response over time. The full model specification is given below.

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

where $\boldsymbol{\phi}_{-k} = (\phi_1, \ldots, \phi_{k-1}, \phi_{k+1}, \ldots, \phi_K)$ and $\boldsymbol{\delta}_{-k} = (\delta_1, \ldots, \delta_{k-1}, \delta_{k+1}, \ldots, \delta_K)$. Here $\bar{t} = (1/N)\sum_{t=1}^N t$ is the average (mean) time so that the linear temporal trend $t^* = (t - \bar{t})/N$ above 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 $\beta_1$ term comes from the covariate component $\mathbf{x}_{kt}^\top \boldsymbol{\beta}$ in (1). The random effects $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_K)$ and $\boldsymbol{\delta} = (\delta_1, \ldots, \delta_K)$ are modelled as spatially autocorrelated by the CAR prior proposed by Leroux, Lei, and Breslow (2000), and are mean centred, that is $\sum_{k=1}^K \phi_k = \sum_{k=1}^K \delta_k = 0$. Here $(\rho_{int}, \rho_{slo})$ 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 $\delta_k \sim \mathrm{N}(0, \tau_{slo}^2)$). Flat uniform priors on the unit interval are specified for the spatial dependence parameters $(\rho_{int}, \rho_{slo})$, 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 $\alpha$ respectively. The corresponding hyperparameters $(a, b, \mu_\alpha, \sigma_\alpha^2)$ can be chosen by the user, and the default values are $(a = 1, \ b = 0.01, \ \mu_\alpha = 0, \ \sigma_\alpha^2 = 100,000)$, which correspond to weakly informative prior distributions. Alternatively, the dependence parameters $(\rho_{int}, \rho_{slo})$ can be fixed at values in the unit interval $[0, 1]$ rather than being estimated, by specifying arguments $(\rho_{int}, \rho_{slo})$ in the `ST.CARlinear()` function. Finally, missing (`NA`) values are allowed in the response data **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. This model is appropriate if the goal is to estimate overall time trends and spatial patterns, and the model is given by.

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

Here the spatio-temporal autocorrelation is modelled by a common set of spatial random effects $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_K)$ and a common set of temporal random effects $\boldsymbol{\delta} = (\delta_1, \ldots, \delta_N)$, 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 $\boldsymbol{\gamma} = (\gamma_{11}, \ldots, \gamma_{KN})$, 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 $(\rho_S, \rho_T)$ and conjugate $(\tau_S^2, \tau_T^2, \tau_I^2)$ priors are specified for the remaining parameters, and the default hyperparameter specifications for the latter are $(a = 1, \ b = 0.01)$. Alternatively, the dependence parameters $(\rho_S, \rho_T)$ 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.

$$\psi_{kt} = \phi_{kt} + \delta_t \tag{4}$$

$$\phi_{kt}|\boldsymbol{\phi}_{-k,t}, \mathbf{W} \sim \mathrm{N}\left(\frac{\rho_S \sum_{j=1}^K w_{kj}\phi_{jt}}{\rho_S \sum_{j=1}^K w_{kj} + 1 - \rho_S}, \frac{\tau_t^2}{\rho_S \sum_{j=1}^K w_{kj} + 1 - \rho_S}\right) \quad t = 1,\ldots,N$$

$$\delta_t|\boldsymbol{\delta}_{-t}, \mathbf{D} \sim \mathrm{N}\left(\frac{\rho_T \sum_{j=1}^N d_{tj}\delta_j}{\rho_T \sum_{j=1}^N d_{tj} + 1 - \rho_T}, \frac{\tau_T^2}{\rho_T \sum_{j=1}^N d_{tj} + 1 - \rho_T}\right)$$

$$\tau_1^2, \ldots, \tau_N^2, \tau_T^2, \sim \mathrm{Inverse\text{-}Gamma}(a,b)$$

$$\rho_S, \rho_T \sim \mathrm{Uniform}(0,1),$$

where $\boldsymbol{\phi}_{-k,t} = (\phi_{1,t},\ldots,\phi_{k-1,t},\phi_{k+1,t},\ldots,\phi_{K,t})$. This model fits an overall temporal trend to the data $\boldsymbol{\delta} = (\delta_1,\ldots,\delta_N)$ that is common to all areal units, which is augmented with a separate (uncorrelated) spatial surface $\boldsymbol{\phi}_t = (\phi_{1t},\ldots,\phi_{Kt})$ 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 $\rho_S$ but a temporally-varying variance parameter $\tau_t^2$. Thus the collection $(\tau_1^2,\ldots,\tau_N^2)$ 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 $\rho_S$ to be constant in time as it is not orthogonal to $\tau_t^2$, thus if it varied then any changes in $\tau_t^2$ would not directly correspond to changes in the level of spatial variation over time. As with all other models the random effects are zero-mean centred, while flat and conjugate priors are specified for $(\rho_S, \rho_T)$ and $(\tau_T^2, \tau_1^2, \ldots, \tau_N^2)$ respectively. Alternatively, the dependence parameters $(\rho_S, \rho_T)$ can be fixed at values in the unit interval $[0,1]$ rather than being estimated in the model. Missing (`NA`) values are not allowed in the response data $\mathbf{Y}$ in this model.

`ST.CARar()`
There are two versions of this model, which are based on either a first (AR(1)) or a second (AR(2)) order temporal autoregressive process. The first order 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 autocorrelated precision matrix. The second model extends this by using a multivariate second order autoregressive process with a spatially autocorrelated precision matrix. These models are appropriate if one wishes to estimate the evolution of the spatial structure in the data over time. The model specifications in both cases are given below.

**AR(1) model**

$$\psi_{kt} = \phi_{kt} \tag{5}$$

$$\boldsymbol{\phi}_t|\boldsymbol{\phi}_{t-1} \sim \mathrm{N}\left(\rho_T \boldsymbol{\phi}_{t-1}, \tau^2 \mathbf{Q}(\mathbf{W},\rho_S)^{-1}\right) \quad t = 2,\ldots,N$$

$$\boldsymbol{\phi}_1 \sim \mathrm{N}\left(\mathbf{0}, \tau^2 \mathbf{Q}(\mathbf{W},\rho_S)^{-1}\right)$$

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

$$\rho_S, \rho_T \sim \mathrm{Uniform}(0,1).$$

**AR(2) model**

$$
\begin{aligned}
\psi_{kt} &= \phi_{kt} & (6) \\
\phi_t | \phi_{t-1}, \phi_{t-2} &\sim \mathrm{N}\left(\rho_{T_1}\phi_{t-1} + \rho_{T_2}\phi_{t-2}, \tau^2 \mathbf{Q}(\mathbf{W}, \rho_S)^{-1}\right) & t = 3, \ldots, N \\
\phi_1, \phi_2 &\sim \mathrm{N}\left(\mathbf{0}, \tau^2 \mathbf{Q}(\mathbf{W}, \rho_S)^{-1}\right) \\
\tau^2 &\sim \text{Inverse-Gamma}(a, b) \\
\rho_S &\sim \text{Uniform}(0, 1) \\
f(\rho_{T_1}, \rho_{T_2}) &\propto 1.
\end{aligned}
$$

Here $\phi_t = (\phi_{1t}, \ldots, \phi_{Kt})$ is the vector of random effects for time period $t$, which evolve over time via a multivariate first or second order autoregressive process with temporal autoregressive parameter(s) $\rho_T$ (AR(1) model) or $(\rho_{T_1}, \rho_{T_2})$ (AR(2) model). The choice between these two options is made by the argument `AR=1` or `AR=2` within the function. Thus temporal autocorrelation is induced via the mean 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

$$
\mathbf{Q}(\mathbf{W}, \rho_S) = \rho_S[\text{diag}(\mathbf{W1}) - \mathbf{W}] + (1 - \rho_S)\mathbf{I}, \tag{7}
$$

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 inverse-gamma priors are specified for $(\rho_S, \rho_T, \rho_{T_1}, \rho_{T_2})$ and $\tau^2$ respectively, with $(a = 1, b = 0.01)$ being the default hyperparameter values for the latter. The dependence parameters $(\rho_S, \rho_T)$ can be fixed at values in the unit interval $[0, 1]$ rather than being estimated in the model, while $(\rho_{T_1}, \rho_{T_2})$ can also be fixed if required. 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 first order `ST.CARar()` model, namely:

$$
\begin{aligned}
\psi_{kt} &= \phi_{kt} & (8) \\
\phi_t | \phi_{t-1} &\sim \mathrm{N}\left(\rho_T \phi_{t-1}, \tau^2 \mathbf{Q}(\mathbf{W}, \rho_S)^{-1}\right) & t = 2, \ldots, N \\
\phi_1 &\sim \mathrm{N}\left(\mathbf{0}, \tau^2 \mathbf{Q}(\mathbf{W}, \rho_S)^{-1}\right) \\
\tau^2 &\sim \text{Inverse-Gamma}(a, b) \\
\rho_S, \rho_T &\sim \text{Uniform}(0, 1).
\end{aligned}
$$

However, the random effects from `ST.CARar()` have a single level of spatial dependence that is controlled by $\rho_S$. All pairs of adjacent areal units (for which $w_{kj} = 1$) will have strongly autocorrelated random effects if $\rho_S$ is close to one, while no such spatial dependence will exist anywhere if $\rho_S$ is close to zero. However, real data may exhibit spatially varying dependencies, as two adjacent areal units may exhibit similar values suggesting a value of $\rho_S$ close to one, while another adjacent pair may exhibit very different values suggesting a value of $\rho_S$ close to zero.

This model allows for localised spatial autocorrelation by allowing spatially neighbouring random effects to be partially autocorrelated (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 at 1. For this model $\mathbf{W}$ must be a binary matrix. These adjacency parameters are collectively denoted by $\mathbf{w}^+ = \{w_{kj}|k \sim j\}$, where $k \sim j$ means areas $(k, j)$ are neighbours and $w_{kj} = 1$. Estimating $w_{kj} \in \mathbf{w}^+$ equal to zero means $(\phi_{kt}, \phi_{jt})$ are conditionally independent for all time periods $t$ given the remaining random effects, while estimating it close to one means they are autocorrelated. These adjacency parameters in $\mathbf{w}^+$ are each modelled on the unit interval, by assuming a multivariate Gaussian prior distribution on the transformed scale $\mathbf{v}^+ = \log\left(\mathbf{w}^+/(\mathbf{1} - \mathbf{w}^+)\right)$. This prior is a shrinkage model with a constant mean $\mu$ and a diagonal covariance matrix with variance parameter $\zeta^2$, and is given by

$$
\begin{aligned}
f(\mathbf{v}^+|\tau_w^2, \mu) &\propto \exp\left[-\frac{1}{2\tau_w^2}\left(\sum_{v_{ik} \in \mathbf{v}^+}(v_{ik} - \mu)^2\right)\right] \qquad (9)\\
\tau_w^2 &\sim \text{Inverse-Gamma}(a, b).
\end{aligned}
$$

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). Under small values of $\tau_w^2$ the elements of $\mathbf{v}^+$ are shrunk to $\mu$, 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 $\tau_w^2 \to 0$ the prior becomes the global smoothing model `ST.CARar()`, where as when $\tau_w^2$ 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 $\tau_w^2$ are $(a = 1, b = 0.01)$. It is possible to fix $\rho_S$ using the `rho` argument. For further details see Rushworth *et al.* (2017). Missing (`NA`) values are not allowed in the response data $\mathbf{Y}$ in this model.

`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 process. 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. This model captures any step-changes in the response via the mean function, and is given by

$$\begin{aligned}
\psi_{kt} &= \lambda_{Z_{kt}} + \phi_{kt} & &(10) \\
\phi_t|\phi_{t-1} &\sim \mathrm{N}\left(\rho_T\phi_{t-1}, \tau^2\mathbf{Q}(\mathbf{W})^-\right) & t&=2,\ldots,N \\
\phi_1 &\sim \mathrm{N}\left(\mathbf{0}, \tau^2\mathbf{Q}(\mathbf{W})^-\right) & & \\
\tau^2 &\sim \text{Inverse-Gamma}(a,b) & & \\
\rho_T &\sim \text{Uniform}(0,1),
\end{aligned}$$

where the '$^-$' in $\mathbf{Q}(\mathbf{W})^-$ denotes a generalised inverse. The random effects $\phi = (\phi_1,\ldots,\phi_T)$ are modelled by a simplification of the `ST.CARar()` model with a first order autoregressive process where $\rho_S = 1$, which corresponds to strong spatial dependence. Note, for this model the inverse $\mathbf{Q}(\mathbf{W})^{-1}$ does not exist as the precision matrix is singular. This simplification of $\rho_S = 1$ is made so that the random effects $\phi$ capture the globally smooth spatio-temporal autocorrelation in the data, allowing the other component $\lambda_{Z_{kt}}$ to capture localised clustering and step-changes. This second component is a piecewise constant intercept process $\lambda_{Z_{kt}}$. Spatially and temporally adjacent data points $(Y_{kt}, Y_{js})$ will be similar if they have the same intercept, that is if $\lambda_{Z_{kt}} = \lambda_{Z_{js}}$, but exhibit a step-change if they are have different intercepts, that is if $\lambda_{Z_{kt}} \neq \lambda_{Z_{js}}$. This piecewise constant intercept process comprises at most $G$ distinct levels, making this component a piecewise constant intercept term. The $G$ levels are ordered via the prior specification:

$$\lambda_j \sim \text{Uniform}(\lambda_{j-1}, \lambda_{j+1}) \quad \text{for } j=1,\ldots,G, \tag{11}$$

where $\lambda_0 = -\infty$ and $\lambda_{G+1} = \infty$. Here $Z_{kt} \in \{1,\ldots,G\}$ 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:

$$\begin{aligned}
f(Z_{kt}|Z_{k,t-1}) &= \frac{\exp(-\delta[(Z_{kt}-Z_{k,t-1})^2 + (Z_{kt}-G^*)^2])}{\sum_{r=1}^G \exp(-\delta[(r-Z_{k,t-1})^2 + (r-G^*)^2])} \quad \text{for } t=2,\ldots,N \quad (12) \\
f(Z_{k1}) &= \frac{\exp(-\delta(Z_{k1}-G^*)^2)}{\sum_{r=1}^G \exp(-\delta(r-G^*)^2)} \\
\delta &\sim \text{Uniform}(1,m).
\end{aligned}$$

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 $\delta$, 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). Missing (`NA`) values are not allowed in

the response data $\mathbf{Y}$ in this model.

**ST.CARclustrends()**

The model is that proposed by Napier *et al.* (2019) and represents the latent random effects with an overall spatial pattern, and a mixture of temporal trend functions with fixed parametric forms (e.g., linear, step-change) or constrained shapes (e.g., monotonically increasing). Note, that due to identifiability issues, covariates are not allowed in this model, but offsets (using the `offset()` function) are allowed. This model is appropriate if the goal is to identify clusters of areas that exhibit similar temporal trends. The model specification is given below.

$$\psi_{kt} = \phi_k + \sum_{s=1}^{S} \omega_{ks} f_s(t|\boldsymbol{\gamma}_s) \tag{13}$$

$$\phi_k|\boldsymbol{\phi}_{-k}, \mathbf{W}, \rho, \tau^2 \sim N\left(\frac{\rho \sum_{j=1}^{K} w_{kj}\phi_j}{\rho \sum_{j=1}^{K} w_{kj} + 1 - \rho}, \frac{\tau^2}{\rho \sum_{j=1}^{K} w_{kj} + 1 - \rho}\right)$$

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

$$\rho \sim \text{Uniform}(0, 1)$$

$$\boldsymbol{\omega}_k = (\omega_{k1}, \ldots, \omega_{kS}) \sim \text{Multinomial}(1; \boldsymbol{\lambda})$$

$$\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_S) \sim \text{Dirichlet}(\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_S)),$$

where $\boldsymbol{\phi}_{-k} = (\phi_1, \ldots, \phi_{k-1}, \phi_{k+1}, \ldots, \phi_K)$. This model fits an overall spatial pattern to the data $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_K)$ that is common to all time periods, which is modelled by the CAR prior proposed by Leroux *et al.* (2000). As with all other models the spatial random effects are zero-mean centred, while flat and conjugate priors are specified for $\rho$ and $\tau^2$ respectively with $(a = 1, b = 0.01)$ being the default hyperparameter values. The model clusters areas according to their temporal trends where the $S$ trends $(f_1(t|\boldsymbol{\gamma}_1), \ldots, f_S(t\boldsymbol{\gamma}_S))$ are user-specified and are given in Table 2. An area $k$ is assigned to one of the $S$ candidate trends via the binary indicator $\boldsymbol{\omega}_k = (\omega_{k1}, \ldots, \omega_{kS})$, where $\omega_{ks} = 1$ if area $k$ is assigned to trend $s$, and is zero otherwise. Region-wide probabilities are associated with each candidate trend via $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_S)$, and are assigned a weakly informative conjugate Dirichlet prior distribution ($\alpha_i = 1$ for $i = 1, \ldots, S$).

| Trend | | Function | R identifier |
|---|---|---|---|
| Constant | | $f(t) = 0$ | Constant |
| Linear | decreasing | $f(t|\gamma) = \gamma t$ | LD |
| | increasing | $f(t|\gamma) = \gamma t$ | LI |
| Known change point | peak | $f(t|\boldsymbol{\gamma}) = \gamma_1 t + \gamma_2(t - t^*)_+$ | CP |
| | trough | $f(t|\boldsymbol{\gamma}) = \gamma_1 t + \gamma_2(t - t^*)_+$ | CT |
| Monotonic cubic splines | decreasing | $f(t|\boldsymbol{\gamma}) = \gamma_0 t + \sum_{j=1}^{q} \gamma_j(t - t_j^*)_+^3$ | MD |
| | increasing | $f(t|\boldsymbol{\gamma}) = \gamma_0 t + \sum_{j=1}^{q} \gamma_j(t - t_j^*)_+^3$ | MI |

Table 2: **ST.CARclustrends()** temporal trend functions. The shapes of the trend functions come via the prior specifications of the $\gamma$'s. The known change point is denoted by $t^*$.

## 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 ($\beta$) and the random effects (e.g., $\phi$ etc) in the Gaussian data models, as well as the variance parameters (e.g., $\tau^2$ 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 are updated via the simple Gaussian random walk Metropolis algorithm. The regression parameters in the Poisson and binomial data models are updated in blocks using either 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 updates by specifying `MALA=FALSE` in the function call. 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 **W** and **D** are utilised via their triplet forms when updating the random effects within the algorithms, which increases the computational efficiency of the software. The new functionality in version 4.0 is that you can now generate multiple MCMC chains in one model run for most of the models. This can be done by setting the `n.chains` argument to be greater than 1, and the multiple chains can either be run in parallel (if `n.cores > 1`) for increased computational speed or in series (`n.cores=1`) if you do not want to use multiple cores. The models this cannot be done for are `ST.CARadaptive()` and `ST.CARclustrends()`, which is because these models were not written by the package maintainer and the authors of these models no longer work in this area.

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 for inference. Convergence can be assessed using many metrics, the simplest of which is by eye, by viewing trace-plots of the parameters that should be stationary and show random fluctuations around a single mean level. In addition, **CARBayesST** presents 2 different convergence diagnostics depending on how many parallel Markov chains have been generated (i.e., depending on the value of `n.chains`). If only 1 chain has been generated then the convergence diagnostic proposed by Geweke (1992) is provided for a sample of key parameters. This information is presented when applying the `print()` function to a fitted model object. The Geweke diagnostic is implemented using 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. In contrast, if more than 1 chain has been generated (i.e., if `n.chains` is greater than 1) then the convergence diagnostic presented is the Upper C.I. of the potential scale reduction factor (PSRF, Gelman, Carlin, Stern, Dunson, Vehtari, and Rubin 2013), and a a value less than 1.1 is indicative 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 vignette, and further details can be found in general texts on Bayesian modelling such as Gelman *et al.* (2013).

# 3. Spatio-temporal models for multivariate areal unit data

This section extends the previous one by outlining the class of multivariate spatio-temporal models that **CARBayesST** can implement. The study region and time frame comprising $K$ areal units for $N$ consecutive time periods is the same as in the previous section. The extension considered here is to model $J > 1$ outcome variables simultaneously, yielding a multivariate spatio-temporal data structure. Let $Y_{ktj}$ denote the outcome (response) variable for the $k$th areal unit ($k = 1, \ldots, K$) in the $t$th time period ($t = 1, \ldots, N$) for the $j$th outcome ($j = 1, \ldots, J$), while $O_{ktj}$ is a corresponding offset variable if required. Then letting $\mathbf{x}_{kt} = (x_{kt1}, \ldots, x_{ktp})$ be a vector of $p$ known covariates for areal unit $k$ and time period $t$ that is common to all responses $j = 1, \ldots, J$, **CARBayesST** can fit the following generalised linear mixed model to these data.

$$
\begin{aligned}
Y_{ktj}|\mu_{ktj} &\sim f(y_{ktj}|\mu_{ktj}, \nu_j^2) \quad \text{for } k = 1, \ldots, K, \ \ t = 1, \ldots, N, \ \ j = 1, \ldots, J \quad (14)\\
g(\mu_{ktj}) &= \mathbf{x}_{kt}^\top \boldsymbol{\beta}_j + O_{ktj} + \psi_{ktj}\\
\boldsymbol{\beta}_j &\sim \mathrm{N}(\boldsymbol{\mu}_\beta, \boldsymbol{\Sigma}_\beta),
\end{aligned}
$$

where again $\mu_{ktj} = \mathbf{E}(Y_{ktj})$ denotes the expectation of $Y_{ktj}$. The regression coefficients $\boldsymbol{\beta}_j$ vary by response variable $j$, and Gaussian priors are assumed for each vector of regression parameters $\boldsymbol{\beta}_j$ as before. The following data likelihood models are available:

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

- **Gaussian -** $Y_{ktj} \sim \mathrm{N}(\mu_{ktj}, \nu_j^2)$ and $\mu_{ktj} = \mathbf{x}_{kt}^\top \boldsymbol{\beta}_j + O_{ktj} + \psi_{ktj}$.

- **Poisson -** $Y_{ktj} \sim \text{Poisson}(\mu_{ktj})$ and $\ln(\mu_{ktj}) = \mathbf{x}_{kt}^\top \boldsymbol{\beta}_j + O_{ktj} + \psi_{ktj}$.

As before, in the binomial model $(n_{ktj}, \theta_{ktj})$ respectively denote the number of trials and the probability of success in each trial in area $k$, time period $t$ and outcome $j$, while in the Gaussian model $\nu_j^2$ is the outcome specific observation variance. An inverse-gamma$(a, b)$ prior is specified for each Gaussian variance $\nu_j^2$, and default hyperparameter values of ($a = 1$, $b = 0.01$) are specified by the software but can be changed by the user. Currently, **CARBayesST** can fit the following multivariate spatio-temporal random effects model.

`MVST.CARar()`
This model was proposed by Lee, Robertson, and Marques (2022), and represents the random effects $\{\psi_{ktj}\}$ with a single set of random effects $\{\phi_{ktj}\}$ that are modelled as jointly correlated over time, space and outcome, so that $\psi_{ktj} = \phi_{ktj}$. These random effects therefore induce (auto)correlations in time, space and between outcomes. The entire set of random effects are denoted by $\boldsymbol{\phi} = (\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_N)$, where $\boldsymbol{\phi}_t = (\boldsymbol{\phi}_{1t}, \ldots, \boldsymbol{\phi}_{Kt})$ denotes the set of $K \times J$ random effects at time $t$, while $\boldsymbol{\phi}_{kt} = (\phi_{kt1}, \ldots, \phi_{tkJ})$ denotes the subset of these effects at the $k$th areal unit for all $J$ outcomes. The general form of the model is given by

$$
\boldsymbol{\phi} \ \sim \ \mathrm{N}\left(\mathbf{0}, \left[\mathbf{D}(\boldsymbol{\alpha}) \otimes \mathbf{Q}(\mathbf{W}, \rho) \otimes \boldsymbol{\Sigma}^{-1}\right]^{-1}\right), \tag{15}
$$

where $\otimes$ denotes a Kronecker product. The joint precision matrix is given by $\mathbf{P}(\boldsymbol{\alpha}, \rho, \boldsymbol{\Sigma}) = \mathbf{D}(\boldsymbol{\alpha}) \otimes \mathbf{Q}(\mathbf{W}, \rho) \otimes \boldsymbol{\Sigma}^{-1}$, where $\mathbf{D}(\boldsymbol{\alpha})_{N \times N}$ controls the temporal autocorrelations, $\mathbf{Q}(\mathbf{W}, \rho)_{K \times K}$

controls the spatial autocorrelations and $\mathbf{\Sigma}_{J \times J}$ captures the between outcome correlations. The between outcome covariance matrix $\mathbf{\Sigma}$ is not assigned a specific structure, and the correlations are instead estimated from the data. We specify the marginally weakly-informative prior for $\mathbf{\Sigma}$ that was designed for covariance matrices by Huang and Wand (2013), which has the scale mixture Inverse-Wishart form:

$$
\begin{aligned}
\mathbf{\Sigma}|a_1, \ldots, a_J &\sim \text{Inverse-Wishart}(df = \nu + J - 1, \mathbf{\Omega} = 2\nu \text{diag}(1/a_1, \ldots, 1/a_J)) \quad (16) \\
a_j &\sim \text{Inverse-Wishart}(1/2, 1/A_j^2),
\end{aligned}
$$

where $J$ is the number of outcome variables being modelled. Here, $\text{diag}(1/a_1, \ldots, 1/a_J)$ denotes a diagonal matrix with diagonal entries $(1/a_1, \ldots, 1/a_J)$. Thus in the prior specification the user can specify the degrees of freedom $\nu$ via the scalar argument `prior.Sigma.df`, while the vector of scales $(A_1, \ldots, A_J)$ are specified by the vector argument `prior.Sigma.scale` in the function call. This prior is chosen because Huang and Wand (2013) show that it is marginally weakly-informative, in the sense that:

- The standard deviation of the $j$th vector of random effects $\sqrt{\mathbf{\Sigma}_{jj}}$ is modelled by a zero centred half-t$(\nu, A_j)$ prior distribution with degrees of freedom $\nu$ and scale $A_j$. The default value used by the software is $A_j = 100,000$, so that the prior is only weakly informative.

- The correlation parameters $\mathbf{\Sigma}_{ij}/\sqrt{\mathbf{\Sigma}_{ii}\mathbf{\Sigma}_{jj}}$, where $i \neq j$ are modelled with a non-informative Uniform$(-1, 1)$ prior as long as $\nu = 2$, which is the default value used by the software.

Spatial autocorrelation is modelled by the conditional autoregressive (CAR) prior proposed by Leroux *et al.* (2000) in common with the univariate models, which corresponds to the following spatial precision matrix

$$
\mathbf{Q}(\mathbf{W}, \rho) = \rho_S(\text{diag}[\mathbf{W1}] - \mathbf{W}) + (1 - \rho_S)\mathbf{I}. \quad (17)
$$

Thus $\rho_S$ is again a global spatial dependence parameter, with a value of 0 corresponding to spatial independence. In contrast, if $\rho_S = 1$ the model captures strong spatial autocorrelation and simplifies to the intrinsic CAR model proposed by Besag *et al.* (1991). We enforce a non-informative uniform prior on the unit interval for $\rho_S$, that is $\rho_S \sim \text{Uniform}(0, 1)$, which provides equal prior weight for all allowable values of $\rho_S$ and allows the data to play the dominant role in estimating its value. Temporal autocorrelation is modelled using either first order or second order autoregressive processes, which can be chosen by specifying `AR=1` or `AR=2` in the function call. The model for $\phi$ from (15) in each case is described below.

**A - First-order autoregressive process**

$$
\begin{aligned}
\psi_{ktj} &= \phi_{ktj} && (18)\\
\phi_t | \phi_{t-1} &\sim \mathrm{N}\left(\rho_T \phi_{t-1}, \left[\mathbf{Q}(\mathbf{W}, \rho_S) \otimes \boldsymbol{\Sigma}^{-1}\right]^{-1}\right) && \text{for } t = 2, \ldots, N\\
\phi_1 &\sim \mathrm{N}\left(\mathbf{0}, \left[\mathbf{Q}(\mathbf{W}, \rho_S) \otimes \boldsymbol{\Sigma}^{-1}\right]^{-1}\right)\\
f(\rho_T) &\propto 1.
\end{aligned}
$$

**B - Second-order autoregressive process**

$$
\begin{aligned}
\psi_{ktj} &= \phi_{ktj} && (19)\\
\phi_t | \phi_{t-1}, \phi_{t-2} &\sim \mathrm{N}\left(\rho_{T_1} \phi_{t-1} + \rho_{T_2} \phi_{t-2}, \left[\mathbf{Q}(\mathbf{W}, \rho_S) \otimes \boldsymbol{\Sigma}^{-1}\right]^{-1}\right) && \text{for } t = 3, \ldots, N\\
\phi_1, \phi_2 &\sim \mathrm{N}\left(\mathbf{0}, \left[\mathbf{Q}(\mathbf{W}, \rho_S) \otimes \boldsymbol{\Sigma}^{-1}\right]^{-1}\right)\\
f(\rho_{T_1}, \rho_{T_2}) &\propto 1.
\end{aligned}
$$

The prior distributons for $\rho_S$ and $\boldsymbol{\Sigma}$ in both models are outlined above.

### 3.1. Inference

All the multivariate models in this package are fitted in a Bayesian setting using Markov chain Monte Carlo simulation, using similar updating steps to those used for the univariate models. Multiple chains are allowed for all these models using the `n.chains` and `n.cores` arguments in the `MVST.CARar()` function call.

## 4. Loading and using the software

### 4.1. Loading the software

**CARBayesST** can be downloaded from the Comprehensive R Archive Network (CRAN, http://cran.r-project.org/) for Windows, Linux and macOS 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 2022), **coda**(Plummer, Best, Cowles, and Vines 2006), **dplyr** (Wickham and Francois 2015), **GGally** (Schloerke, Cook, Larmarange, Briatte, Marbach, Thoen, Elberg, and Crowley 2021), **ggplot2** (Wickham 2016), **gtools** (Warnes, Bolker, and Lumley 2018), **leaflet** (Cheng, Karambelkar, and Xie 2018), **matrixcalc** (Novomestky 2012), **sf** (Pebesma 2018), **spam** (Furrer and Sain 2010), **spdep** (Bivand, Pebesma, and Gómez-Rubio 2013), **stats**, **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 visualising the data, creating the neighbourhood matrix **W**, and plotting the results requires a number of other packages.

## 4.2. Using the software

The software can fit eight classes of model, and the main arguments common to all the models are as follows.

- `formula` - A formula for the covariate part of the model using the syntax of the `lm()` function. Offsets can be included here using the `offset()` function. For the univariate models the response and each covariate should be vectors of length $(KN) \times 1$, where $K$ is the number of spatial units and $N$ is the number of time periods. Each vector should be 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 locations for time 2 and so on. For the multivariate models the response and the offset (if included) should be matrices of dimension $(KN) \times J$, where $K$ is the number of spatial units, $N$ is the number of time periods and $J$ is the number of different variables. Each column of the response and offset matrices relates to a different outcome variable. The covariates should each be a $(KN) \times 1$ vector, and different regression parameters are estimated for each of the $J$ variables.

- `family` - One of either `"binomial"`, `"gaussian"` or `"poisson"`.

- `trials` - This is a vector (univariate models) or matrix (multivariate models) the same size and in the same order as the response containing the total number of trials for each area and time period. This is only needed if `family="binomial"`.

- `W` - A $K \times K$ symmetric and non-negative neighbourhood matrix whose row sums must all be positive.

- `burnin` - The number of MCMC samples to discard as the burn-in period.

- `n.sample` - The number of MCMC samples to generate.

- `thin` - The level of thinning to apply to the MCMC samples to reduce their temporal autocorrelation. Defaults to 1 (no thinning).

Additionally, most of the models (not `ST.CARadaptive()` or `ST.CARclustrends()`) have two additional important arguments:

- `n.chains` - The number of MCMC chains to run when fitting the model. Defaults to 1.

- `n.cores` - The number of computer cores to run the MCMC chains on. Must be less than or equal to n.chains. Defaults to 1.

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

- `coef()` - returns the estimated (posterior mean) regression coefficients.

- `fitted()` - returns the fitted values based on posterior means.

- `logLik()` - returns the estimated loglikelihood based on posterior means.

- `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), or "pearson", residuals from the model (based on posterior means).

If `n.chains` is greater than 1 and `n.cores` is greater than 1 then the software updates the user on the progress of each model run by writing it to the file *CARBayesSTprogress.txt* in the current working directory. In contrast, if either `n.chains` equals 1 or `n.cores` equals 1 then the progress of the current model run is written to the R console. However, using the `verbose=FALSE` option will disable this feature. Once run, each model returns a list object with the following components.

- `summary.results` - A summary table for selected parameters that is presented when using the `print()` function. The table includes the posterior mean (`Mean`) and 95% credible interval (`2.5%, 97.5%`), 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 either an `mcmc()` object (if `n.chains` equals 1) or an `mcmc.list()` object (if `n.chains` is greater than 1). The names of these objects correspond to the parameters defined in Section 2 of this vignette. For example, for `ST.CARlinear()` the (`tau2, rho`) elements of the list have elements 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 elements ordered as $(\tau_S^2, \tau_T^2, \tau_I^2)$ (the latter only if `interaction=TRUE`) and $(\rho_S^2, \rho_T^2)$ respectively. Finally, each model returns samples from the posterior distribution of the fitted values for each data point (`fitted`), and the missing values in the response variable (`Y`).

- `fitted.values` - For the univariate models a vector of fitted values based on posterior means for each area and time period in the same order as the data **Y**. For the multivariate models a matrix of fitted values where each column relates to a single outcome.

- `residuals` - For the univariate models a matrix of 2 types of residuals in the same order as the response. The 2 columns of this matrix correspond to "response" (raw), and "pearson", residuals. For the multivariate models a list of matrices corresponding to "response" (raw), and "pearson", residuals. For each matrix each column relates to a single outcome.

- `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), the Log Marginal Predictive Likelihood (LMPL, Congdon 2005) and the loglikelihood. The best fitting model is one that minimises the DIC and WAIC but maximises the LMPL. If the response data contain missing values, the DIC is computed based on only the observed data (see Celeux, Forbes, Robert, and Titterington (2006)).

- `accept` - The acceptance probabilities for the parameters in the MCMC sampling algorithm.

- `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 $\mathbf{Y}$.

- `formula` - The formula (as a text string) for the covariate and offset parts 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.

### 4.3. Estimating the neighbourhood matrix

The package also contains functionality (`W.estimate()`) to estimate an appropriate neighbourhood matrix $\mathbf{W}_E$ for the data being modelled, rather than relying on a purely geographical $\mathbf{W}$ constructed, for example, by the border sharing rule. Using a neighbourhood matrix estimated for the data at hand will likely lead to improved estimation of covariate effects, random effects and fitted values. It will also allow you to identify boundaries in the random effects surface that separate 2 geographically adjacent areas that have very different values. The matrix is estimated using a graph-based optimisation algorithm, and full details can be found in Lee, Meeks, and Pettersson (2021). In short, the algorithm uses a vector of spatial data observations and a binary baseline neighbourhood matrix $\mathbf{W}$, the latter being constructed using the border sharing rule. The estimated neighbourhood matrix is then constructed by removing neighbour relations (i.e. setting $w_{E_{ij}} = w_{E_{ji}} = 0$) if they are not appropriate for the data. Note, new edges not in the initial $\mathbf{W}$ matrix cannot be added when creating $\mathbf{W}_E$.

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

This 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

Scotland is split into 14 regional health boards, and this study focuses on the Greater Glasgow and Clyde health board containing the city of Glasgow and a population of around 1.2 million people. This health board is split into $K = 271$ Intermediate Zones (IZ), which are a key geography for the distribution of small-area statistics in Scotland. The available data span $N = 5$ years between 2007 and 2011 for each of the $K = 271$ IZs. 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 disease data relate to the observed and expected numbers of respiratory related hospital admissions in each IZ and year, which for the $k$th IZ and $t$th year are denoted by $(Y_{kt}, e_{kt})$ respectively. Here, the expected numbers of hospital admissions control for the varying population sizes and demographic structures (e.g., age and sex profile) in each IZ and year, and are computed via indirect standardisation. The Poisson model, $Y_{kt} \sim$ Poisson($e_{kt}\theta_{kt}$) is typically used to model these data, where $\theta_{kt}$ is the risk, relative to $e_{kt}$, of disease in IZ $k$ and year $t$. For example, a value of 1.2 corresponds to a 20% increased risk of disease. Operationally, $e_{kt}$ is included as an offset term in the model on the linear predictor (natural log) scale, that is $O_{kt} = \ln(e_{kt})$ in (1).

The pollution data comprise yearly average modelled concentrations of particulate matter less than 10 microns (PM$_{10}$), which are available on a 1 kilometre square grid resolution and are produced by the numerical model described in Ricardo-AEA (2015). These 1 kilometre gridded estimates are spatially misaligned with the irregularly shaped polygonal IZs at which the disease data are available, and thus simple averaging is used to produce IZ level PM$_{10}$ estimates. Finally, we have 2 potential confounders to include in the model, both of which are measures of socio-economic deprivation (poverty) which is a major factor influencing population health. The first is the average property price in each IZ and year (in hundreds of thousands), while the second is the proportion of the working age population who are in receipt of job seekers allowance (JSA), 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.IZ`. This object is an `sf` object from the **sf** package. These data can be loaded using the following commands.

```
R> library("CARBayesdata")
R> library("sf")
R> data("GGHB.IZ")
R> data("pollutionhealthdata")
```

These two data sets have the following structures.

```
R> class(GGHB.IZ)
```

```
[1] "sf"          "data.frame"
```

```
R> head(GGHB.IZ)
```

```
Simple feature collection with 6 features and 4 fields
Geometry type: MULTIPOLYGON
Dimension:      XY
Bounding box:   xmin: 253239 ymin: 669235 xmax: 263842 ymax: 672791.6
Projected CRS: OSGB36 / British National Grid
        IZ                          name                      geometry
1 S02000260                    Auchinairn MULTIPOLYGON (((260718 6698...
2 S02000261                  Woodhill East MULTIPOLYGON (((262047 6699...
3 S02000262                  Woodhill West MULTIPOLYGON (((261158.7 67...
4 S02000263                 Westerton East MULTIPOLYGON (((253771 6709...
5 S02000264 Bishopbriggs West and Cadder MULTIPOLYGON (((260183 6706...
6 S02000265                 Westerton West MULTIPOLYGON (((253252 6710...
   easting northing
1 261624.5 669657.4
2 262927.1 670027.8
3 262142.9 670428.0
4 254570.5 670593.8
5 261248.4 670928.0
6 253764.4 670982.6


R> class(pollutionhealthdata)


[1] "data.frame"


R> head(pollutionhealthdata)


        IZ 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 `GGHB.IZ` object contains a unique identifier for each IZ (`IZ`) and the polygon information (`geometry`), while `pollutionhealthdata` contains the data set. For the latter, the first column labelled `IZ` is the set of unique identifiers for each IZ, 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 IZ and $t$th year is computed as $\text{SMR}_{kt} = Y_{kt}/E_{kt}$. 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 using functionality of the **dplyr** package.

```
R> library(dplyr)
R> pollutionhealthdata <- pollutionhealthdata %>% mutate(
```
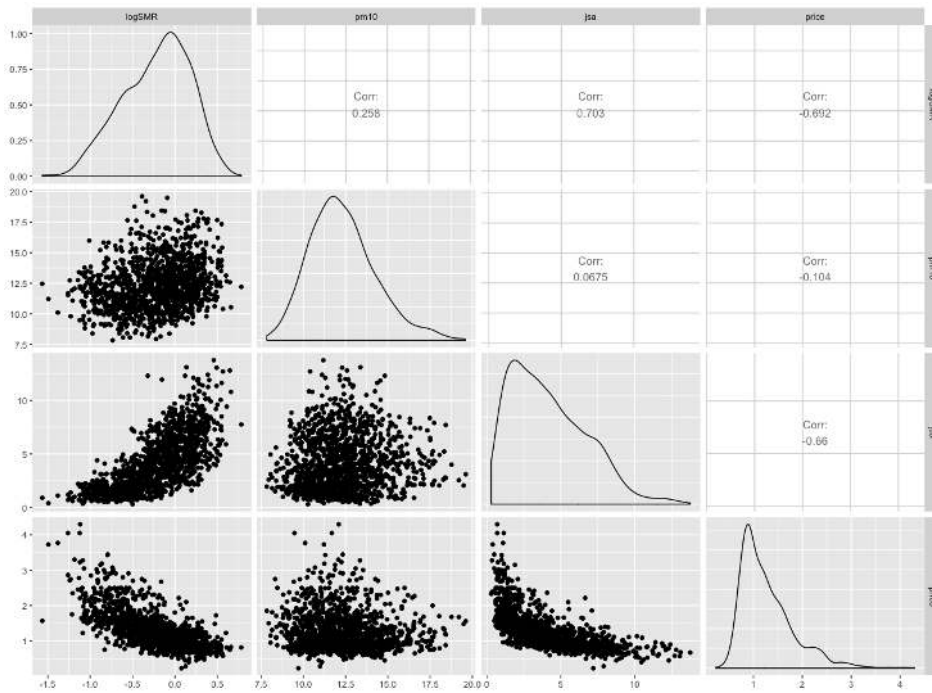
Figure 1: Scatterplot of the disease, pollution and covariate data.

```
+           SMR = pollutionhealthdata$observed / pollutionhealthdata$expected,
+           logSMR = log(pollutionhealthdata$observed / pollutionhealthdata$expected))
R>  head(pollutionhealthdata)
```

```
          IZ year observed   expected    pm10 jsa price       SMR      logSMR
1 S02000260 2007       97  98.24602 14.02699 2.25 1.150 0.9873174 -0.01276372
2 S02000261 2007       15  45.26085 13.30402 0.60 1.640 0.3314122 -1.10439222
3 S02000262 2007       49  92.36517 13.30402 0.95 1.750 0.5305030 -0.63392964
4 S02000263 2007       44  72.55324 14.00985 0.35 2.385 0.6064512 -0.50013097
5 S02000264 2007       68 125.41904 14.08074 0.80 1.645 0.5421824 -0.61215277
6 S02000265 2007       24  55.04868 14.08884 1.25 1.760 0.4359777 -0.83016413
```

Then we visualise the relationships in the data with a scatterplot matrix using functionality from the **GGally** package.

```
R>  library(GGally)
R>  ggpairs(pollutionhealthdata, columns=c(9, 5:7))
```

The pairs plot shown in Figure 1 shows respectively positive and negative relationships between the natural log of SMR and the two deprivation covariates `jsa` and `price`, in both 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(SMR) and $PM_{10}$, while the only relationship that exists between the covariates is a negative nonlinear one between `jsa` 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 final line adds the aggregated averages to the `GGHB.IZ` sf object.

```
R> group_IZ <- group_by(pollutionhealthdata, IZ)
R> SMR.av <- summarise(group_IZ, SMR.mean = mean(SMR))
R> GGHB.IZ$SMR <- SMR.av$SMR.mean
R> head(GGHB.IZ)

Simple feature collection with 6 features and 5 fields
Geometry type: MULTIPOLYGON
Dimension:     XY
Bounding box:  xmin: 253239 ymin: 669235 xmax: 263842 ymax: 672791.6
Projected CRS: OSGB36 / British National Grid
        IZ                        name                      geometry
1 S02000260                  Auchinairn MULTIPOLYGON (((260718 6698...
2 S02000261               Woodhill East MULTIPOLYGON (((262047 6699...
3 S02000262               Woodhill West MULTIPOLYGON (((261158.7 67...
4 S02000263              Westerton East MULTIPOLYGON (((253771 6709...
5 S02000264 Bishopbriggs West and Cadder MULTIPOLYGON (((260183 6706...
6 S02000265              Westerton West MULTIPOLYGON (((253252 6710...
   easting northing       SMR
1 261624.5 669657.4 0.9924849
2 262927.1 670027.8 0.3863466
3 262142.9 670428.0 0.5012227
4 254570.5 670593.8 0.4716168
5 261248.4 670928.0 0.4294583
6 253764.4 670982.6 0.4128424
```

A spatial map of the aggregated `SMR` variable can be overlaid on an OpenStreetMap using the functionality of the **leaflet** package. However, first the `GGHB.IZ` object needs to have its coordinate reference system changed to longitude and latitude as this is what the **leaflet** package requires, which can be done using the following R code.

```
R> GGHB.IZ <- st_transform(x=GGHB.IZ, crs='+proj=longlat +datum=WGS84 +no_defs')
```

Then a map of `SMR` can be drawn using the following code.

```
R> library(leaflet)
R> colours <- colorNumeric(palette = "YlOrRd", domain = GGHB.IZ$SMR)
R> leaflet(data=GGHB.IZ) %>%
+     addTiles() %>%
+     addPolygons(fillColor = ~colours(GGHB.IZ$SMR),
+                 color="grey", weight=1,
+                 fillOpacity = 0.7) %>%
+     addLegend(pal = colours, values = GGHB.IZ$SMR,
+               opacity = 1, title="SMR") %>%
+     addScaleBar(position="bottomleft")
```
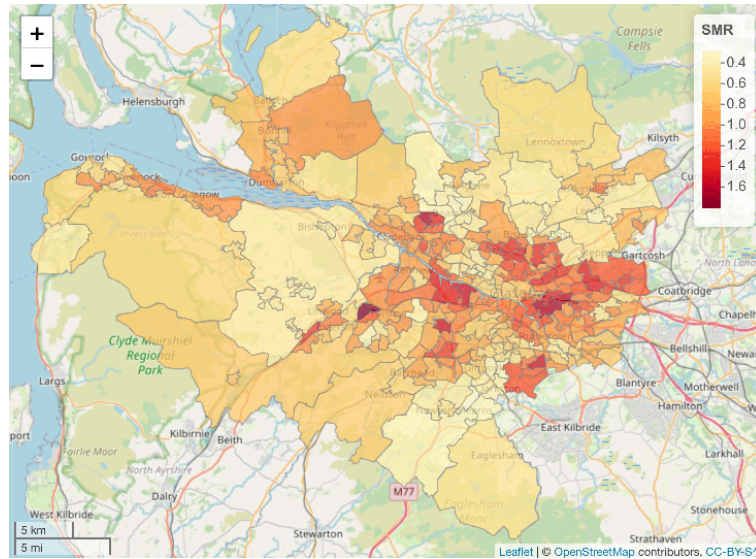
Figure 2: Map showing the average SMR over all five years from 2007 to 2011.

The map is shown in Figure 2, where the yellow shaded areas are low risk (SMR<1) while the red 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 Greenock area in the far west of the region on the lower bank of the river Clyde. The spatio-temporal models we fit to these data require the neighbourhood matrix **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 `sf` object GGHB.IZ using functionality from the **spdep** package as follows.

```
R>  library("spdep")
R>  W.nb <- poly2nb(GGHB.IZ, row.names = GGHB.IZ$IZ)
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 residual spatial autocorrelation

The spatio-temporal models in **CARBayesST** allow for any remaining spatio-temporal auto-correlation in the disease data after the effects of the known covariates have been accounted for. Therefore, we assess the presence of spatial autocorrelation in 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>  model1 <- glm(formula = formula, family = "quasipoisson",
+      data = pollutionhealthdata)
```

```
R>  resid.glm <- residuals(model1)
R>  summary(model1)$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(model1)$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 (the over dispersion parameter is around 4.4). To quantify the presence of spatial autocorrelation in the residuals from this model we 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 positive spatial autocorrelation, 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 = 9968, p-value = 0.0033
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 IZ, 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 allow for this residual autocorrelation by applying the `ST.CARar()` model with a first order temporal autoregressive structure to the data, details of which are given in Section 2.

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. Here we fit the model with 3 independent Markov chains (`n.chains=3`) that are run in parallel (with `n.cores=3`).

```
R>  library("CARBayesST")
R>  chains <- ST.CARar(formula = formula, family = "poisson",
+       data = pollutionhealthdata, W = W, burnin = 20000, n.sample = 220000,
+       thin = 100, AR=1, n.chains=3, n.cores=3)
```

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 `W` is also defined above. The `ST.CARar()` model is run for three parallel Markov chains, each of which are run for 220,000 MCMC samples with the first 20,000 samples removed as the burn-in period. The samples are then thinned by 100 to reduce the autocorrelation in the Markov chains, resulting in 6,000 samples for inference. A summary of the model run can be visualised using the `print()` function developed for **CARBayesST** as shown below.

```
R>  print(chains)


#################
#### Model fitted
#################
Likelihood model - Poisson (log link function)
Latent structure model - Autoregressive order 1 CAR model
Regression equation - observed ~ offset(log(expected)) + jsa + price + pm10



#################
#### MCMC details
#################
Total number of post burnin and thinned MCMC samples generated - 6000
Number of MCMC chains used - 3
Length of the burnin period used for each chain - 20000
Amount of thinning used - 100


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

              Mean     2.5%    97.5% n.effective PSRF (upper 95% CI)
(Intercept) -0.6570 -0.8239 -0.4925       3228.7                   1
jsa          0.0673  0.0570  0.0773       3050.5                   1
price       -0.1944 -0.2357 -0.1538       3735.2                   1
pm10         0.0334  0.0217  0.0445       3239.0                   1
```

```
tau2            0.0588  0.0496  0.0691        5726.1                       1
rho.S           0.5677  0.4079  0.7297        4965.0                       1
rho.T           0.7560  0.6944  0.8157        6135.5                       1


DIC =  10387.77          p.d =  767.4694        LMPL =  -5332.763
```

The output from the `print()` function is split into three sections, the first part (`Model fitted`) describes the model that was fitted, the second part (`MCMC details`) summarises the number of MCMC samples generated, while the third part (`Results`) presents a summary of the numerical results. This summary includes parameter estimates (column `Mean`), 95% credible intervals (the columns headed `2.5%` and `97.5%`) and convergence diagnostics (columns `n.effective` and `PSRF (upper 95% CI)`) for certain parameters, as well as overall model fit measures such as the DIC. The model object `chains` is a list, and details of its elements are described in Section 3 of this vignette. A list object containing the MCMC samples for each individual parameter and the fitted values are stored in `chains$samples`, and each element of this list corresponds to a different group of parameters and is stored as a `mcmc.list` object from the **coda** package. Applying the `summary()` function to this object yields:

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

```
       Length Class      Mode
beta   3      mcmc.list  list
phi    3      mcmc.list  list
rho    3      mcmc.list  list
tau2   3      mcmc.list  list
fitted 3      mcmc.list  list
Y      1      -none-     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 would also been an `mcmc.list` object containing the MCMC samples for the $m$ missing values. Before making inference from the model you have to ensure the Markov chains appear to have converged, and the multiple chain diagnostic, the potential scale reduction factor (Gelman *et al.* 2013) is presented in the above summary object. As all values are less than 1.1 it suggests the Markov chains have converged. To be more sure of convergence another method is a traceplot comparing the results from the multiple chains, which for the regression parameters (`beta`) can be produced as follows using functionality from the **coda** package.

```
R>  library(coda)
R>  plot(chains$samples$beta)
```

The plot is shown in Figure 3 and superficially shows good mixing between and convergence of the chains, as they all have very similar means and show little trend from left to right. 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 $\epsilon$ unit increase in a covariate with regression parameter $\beta_s$ is given by the transformation $\exp(\epsilon\beta_s)$, and a relative
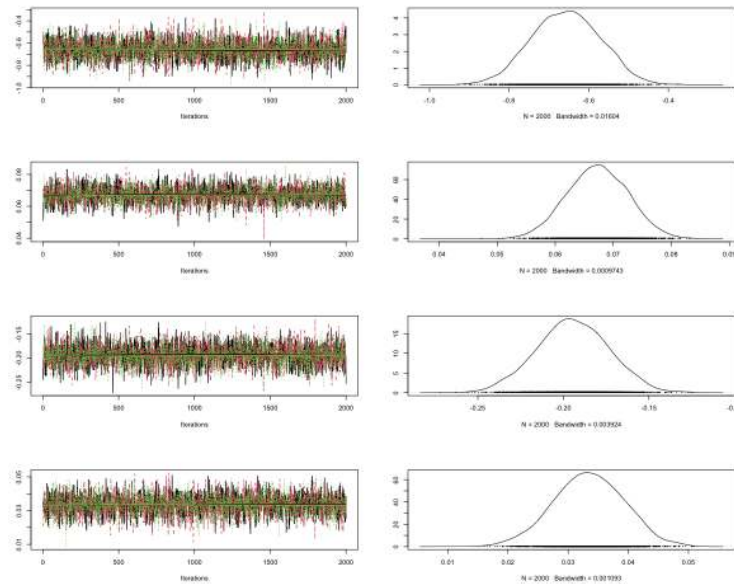
Figure 3: Convergence of the Markov chains.

risk of 1.02 corresponds to a 2% increased risk if the covariate increased by $\epsilon$. The code below computes the posterior mean and 95% credible intervals for the relative risks associated with a one unit increase in each covariate, which are all realistic increases given the variation observed in the data in Figure 1.

```
R>  beta.mcmcsummary <- summary(chains$samples$beta)
R>  beta.summary <- cbind(beta.mcmcsummary$statistics[2:4 ,1],
+          beta.mcmcsummary$quantiles[2:4 ,c(1, 5)])
R>  RR.summary <- exp(beta.summary)
R>  round(RR.summary, 3)


           2.5% 97.5%
[1,] 1.070 1.059 1.080
[2,] 0.823 0.790 0.857
[3,] 1.034 1.022 1.046
```

The output above shows that the posterior mean and 95% credible interval for the relative risk of a $1 \mu g m^{-3}$ increase in $PM_{10}$ is 1.034 (1.022, 1.046), suggesting that such an increase corresponds to 3.4% additional hospital admissions. The corresponding relative risk for a one percent increase in JSA is 1.070 (1.059, 1.080), while for a one hundred thousand pounds increase in property price (the units for the property price data were in hundreds of thousands) the risk is 0.823 (0.790, 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 encompassed 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 Zones 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 the number of property sales $Y_{kt}$ in each IZ (indexed by $k$) and year (indexed by $t$). Additionally, we have the total number of properties $n_{kt}$ in each IZ 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 $\theta_{kt}$ 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 $\theta_{kt}$ 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.IZ`). These data can be loaded using the following commands.

```
R>  library("CARBayesdata")
R>  library("sf")
R>  data("GGHB.IZ")
R>  data("salesdata")
R>  head(salesdata)


          IZ year sales stock
1 S02000260 2003   122  2002
2 S02000261 2003    54   908
3 S02000262 2003    83  1693
4 S02000263 2003    65  1198
5 S02000264 2003   124  2305
6 S02000265 2003    37  1109
```

The `data.frame salesdata` contains 4 columns, the Intermediate Zone code (`IZ`), 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 the raw rate of property sales as a proportion of the total number of properties using boxplots via the **ggplot2** package as shown below.

```
R>  salesdata <- salesdata %>% mutate(salesprop = salesdata$sales / salesdata$stock)
R>  library(ggplot2)
R>  ggplot(salesdata, aes(x = factor(year), y = salesprop)) +
+     geom_boxplot(fill="red", alpha=0.7) +
```
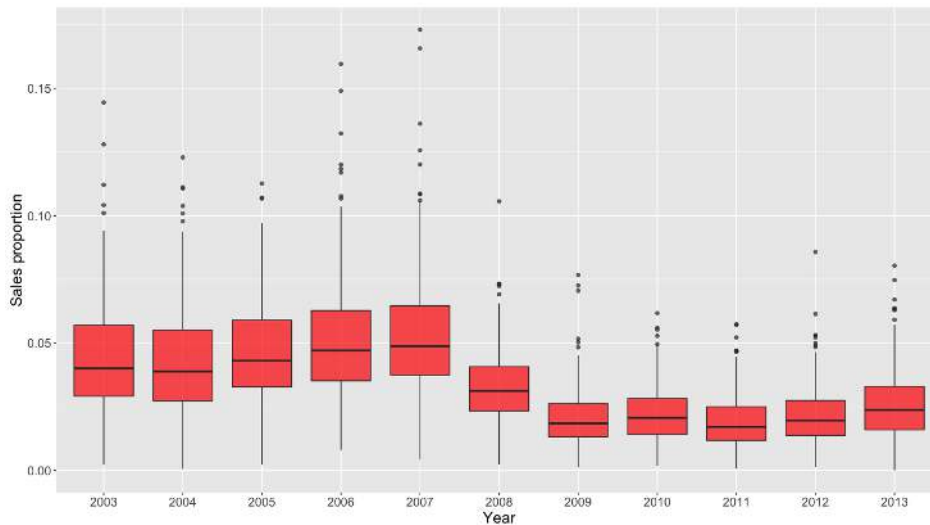
Figure 4: Boxplots showing the temporal trend in the raw rate of property sales as a proportion of the total number of properties between 2003 and 2013.

```
+       scale_x_discrete(name = "Year") +
+       scale_y_continuous(name = "Sales proportion") +
+       theme(text=element_text(size=16), plot.title=element_text(size=18, face="bold"))
```

This produces the boxplot shown in Figure 4, 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 computed using the code below.

```
R>  library(dplyr)
R>  group_IZ <- group_by(salesdata, IZ)
R>  salesprop <- summarise(group_IZ, salesproprtion.mean = mean(salesprop))
R>  GGHB.IZ$sales <- salesprop$salesproprtion.mean
R>  head(GGHB.IZ)


Simple feature collection with 6 features and 5 fields
Geometry type: MULTIPOLYGON
Dimension:     XY
Bounding box:  xmin: 253239 ymin: 669235 xmax: 263842 ymax: 672791.6
Projected CRS: OSGB36 / British National Grid
        IZ                     name                       geometry
1 S02000260               Auchinairn MULTIPOLYGON (((260718 6698...
2 S02000261            Woodhill East MULTIPOLYGON (((262047 6699...
```

```
3 S02000262               Woodhill West MULTIPOLYGON (((261158.7 67...
4 S02000263               Westerton East MULTIPOLYGON (((253771 6709...
5 S02000264 Bishopbriggs West and Cadder MULTIPOLYGON (((260183 6706...
6 S02000265               Westerton West MULTIPOLYGON (((253252 6710...
  easting northing      sales
1 261624.5 669657.4 0.03878125
2 262927.1 670027.8 0.04259660
3 262142.9 670428.0 0.03015482
4 254570.5 670593.8 0.05281445
5 261248.4 670928.0 0.03430728
6 253764.4 670982.6 0.03240008
```

This variable can be mapped using the code below, which first changes the coordinate reference system to longitude and latitude, and the result is displayed in Figure 5.

```
R> GGHB.IZ <- st_transform(x=GGHB.IZ, crs='+proj=longlat +datum=WGS84 +no_defs')
R> library(leaflet)
R> colours <- colorNumeric(palette = "YlOrRd", domain = GGHB.IZ$sales)
R> map1 <- leaflet(data=GGHB.IZ) %>%
+    addTiles() %>%
+    addPolygons(fillColor = ~colours(sales), color="grey", weight=1,
+               fillOpacity = 0.7) %>%
+    addLegend(pal = colours, values = GGHB.IZ$sales, opacity = 1,
+              title="Sales") %>%
+    addScaleBar(position="bottomleft")
R> map1
```

The map shows a largely similar pattern to that seen for respiratory disease risk in Figure 2, with areas that exhibit relatively high sales rates largely being the same ones that exhibit relatively low disease risk. Figures 4 and 5 highlight the change in temporal dynamics and the spatial structure in property sales in Glasgow. Therefore we now apply the ST.CARsepspatial() model from **CARBayesST** and proposed by Napier *et al.* (2016) to more formally quantify these features. This model is chosen because it allows the level of spatial variation to change each year, a feature of the data that is illustrated by Figure 4.

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

Before fitting the model we need to create the neighbourhood matrix using the following code:

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

Then the model can be fitted using the code below, where inference this time is based on 2,000 post burn-in and thinned MCMC samples from 1 Markov chain.
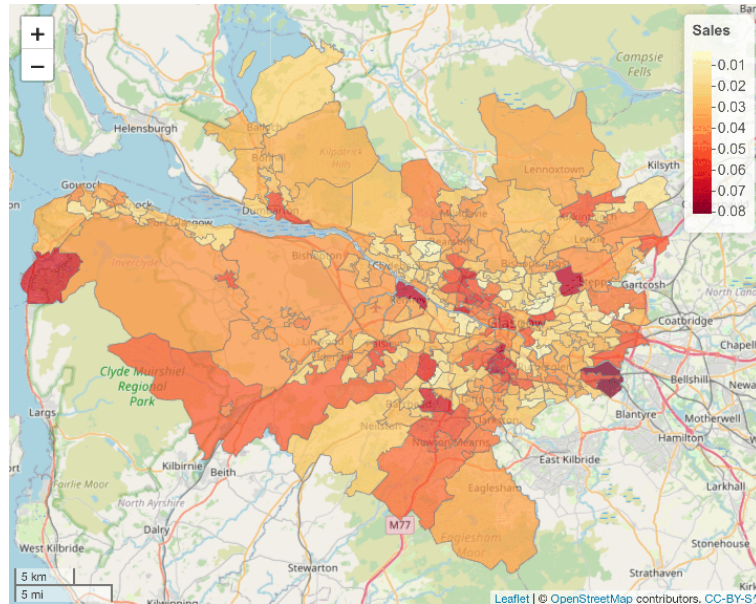
Figure 5: Map showing the average (between 2003 to 2013) raw rate of property sales as a proportion of the total number of properties.

```
R> library("CARBayesST")
R> formula <- sales ~ offset(log(stock))
R> chain1 <- ST.CARsepspatial(formula = formula, family = "poisson",
+         data = salesdata, W = W, burnin = 20000, n.sample = 220000,
+         thin = 100)
```

Convergence of the Markov chain can be assessed using traceplots and the Geweke diagnostic outlined in the previous section, which are not shown for this example for brevity. Also not shown for brevity is a summary of the fitted model, which as before can be obtained using the `print()` function. The model 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 $\beta_1$, a space-time effect $\phi_{kt}$ with a time period specific variance, and a region-wide temporal trend $\delta_t$. The mean and standard deviation of $\{\theta_{kt}\}$ over space for each year is computed by the following code, which produces the posterior mean and a 95% credible interval for each quantity for each year.

```
R> trend.mean <- data.frame(Year=2003:2013, array(NA, c(11, 3)))
R> colnames(trend.mean) <- c("Year", "Mean", "LCI", "UCI")
R> trend.sd <- data.frame(Year=2003:2013, array(NA, c(11, 3)))
R> colnames(trend.sd) <- c("Year", "Mean", "LCI", "UCI")
R>     for(i in 1:11)
+       {
+       posterior <- exp(chain1$samples$phi[ , ((i-1) * 271 + 1):(i * 271)] +
+               matrix(rep(chain1$samples$beta + chain1$samples$delta[ , i], 271),
```

```
+                     ncol = 271, byrow = FALSE))
+          trend.mean[i, 2:4] <- c(mean(apply(posterior, 1, mean)),
+                             quantile(apply(posterior, 1, mean), c(0.025, 0.975)))
+          trend.sd[i, 2:4] <- c(mean(apply(posterior, 1, sd)),
+                             quantile(apply(posterior, 1, sd), c(0.025, 0.975)))
+      }
```

The 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 6.

```
R>  meanplot <- ggplot(aes(x = factor(year), y = salesprop),
+              data=salesdata) +
+          geom_jitter(color="blue") +
+          scale_x_discrete(name = "Year") +
+          scale_y_continuous(name = "Average sales rate") +
+          geom_line(data=trend.mean, mapping=aes(x=factor(Year), y=Mean,
+              group=1), colour="red", lwd=1) +
+          geom_line(data=trend.mean, mapping=aes(x=factor(Year), y=LCI,
+              group=1)) +
+          geom_line(data=trend.mean, mapping=aes(x=factor(Year), y=UCI,
+              group=1)) +
+          ggtitle("(a) - Average sales rate") +
+          theme(text=element_text(size=16), plot.title=element_text(size=18,
+              face="bold"))

R>  sdplot <- ggplot() +
+          scale_x_discrete(name = "Year") +
+          scale_y_continuous(name = "Spatial standard deviation") +
+          geom_line(data=trend.sd, mapping=aes(x=factor(Year), y=Mean,
+              group=1), colour="red", lwd=1) +
+          geom_line(data=trend.sd, mapping=aes(x=factor(Year), y=LCI,
+              group=1)) +
+          geom_line(data=trend.sd, mapping=aes(x=factor(Year), y=UCI,
+              group=1)) +
+          ggtitle("(b) - Variation in the sales rate") +
+          theme(text=element_text(size=16), plot.title=element_text(size=18,
+              face="bold"))

R>  library(gridExtra)
R>  grid.arrange(meanplot, sdplot, nrow=2, ncol=1)
```

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 the rates of property sales show similar underlying trends, with maximum values just before the global financial crisis in 2007, and then 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 around 5.0% (0.05) in 2007 to 3% (0.03) in 2009. The spatial standard
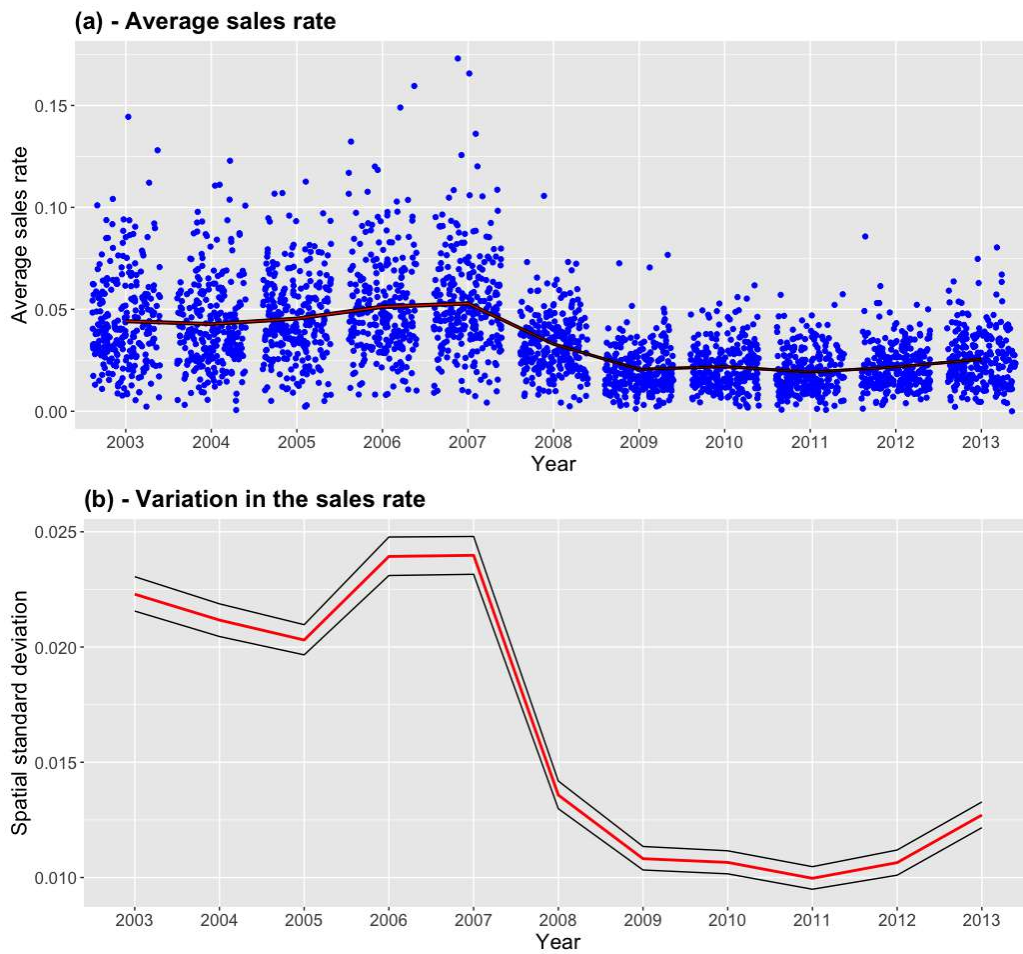
Figure 6: Posterior mean (red) and 95% credible interval (black) for the temporal trend in: (a) region-wide average property sales rates; and (b) spatial standard deviation in property sales rates. In panel (a) the blue dots are the raw sales proportions for each area and year (jittered in the x direction to improve the presentation).

deviation also dropped from 0.022 to 0.01 over the same three-year period, suggesting that the global financial crisis had the effect of reducing the disparity in sales rates in different regions of Greater Glasgow.

# 7. Discussion

**CARBayesST** is the first software package dedicated to fitting spatio-temporal CAR type models to areal unit data, with inference in a Bayesian setting using MCMC simulation. Future development of the software will focus on extending the number of spatio-temporal models that can be implemented, giving the user an even wider set of modelling tools for both univariate and multivariate spatio-temporal data.

# 8. Acknowledgements

# References

Anderson C, Lee D, Dean N (2014). "Identifying Clusters in Bayesian Disease Mapping." *Biostatistics*, **15**, 457–469.

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, 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 RS, Pebesma E, Gómez-Rubio V (2013). *Applied spatial data analysis with R, Second edition.* Springer, NY. URL https://asdar-book.org/.

Celeux G, Forbes F, Robert C, Titterington D (2006). "Deviance Information Criteria for Missing Data Models." *Bayesian Analysis*, **1**, 651–674.

Cheng J, Karambelkar B, Xie Y (2018). *leaflet: Create Interactive Web Maps with the JavaScript 'Leaflet' Library.* R package version 2.0.1, URL https://CRAN.R-project.org/package=leaflet.

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.

Eddelbuettel D, Francois R (2011). "**Rcpp**: Seamless R and C++ Integration." *Journal of Statistical Software*, **40**, 8.

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.

Huang A, Wand M (2013). "Simple Marginally Noninformative Prior Distributions for Covariance Matrices." *Bayesian Analysis*, **8**, 439–452.

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 (2022). **CARBayesdata**: *Data Used in the Vignettes Accompanying the **CARBayes** and **CARBayesST** Packages*. R package version 3.0.

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, Meeks K, Pettersson W (2021). "Improved inference for areal unit count data using graph-based optimisation." *Statistics and Computing*, **31**, 51.

Lee D, Minton J, Pryce G (2015). "Bayesian Inference for the Dissimilarity Index in the Presence of Spatial Autocorrelation." *Spatial Statistics*, **11**, 81–95.

Lee D, Robertson C, Marques D (2022). "Quantifying the small-area spatio-temporal dynamics of the Covid-19 pandemic in Scotland during a period with limited testing capacity." *Spatial Statistics*, **to appear**, https://doi.org/10.1016/j.spasta.2021.100508.

Lee D, Rushworth A, Napier G (2018). "Spatio-Temporal Areal Unit Modeling in R with Conditional Autoregressive Priors Using the CARBayesST Package." *Journal of Statistical Software, Articles*, **84**(9), 1–39. doi:10.18637/jss.v084.i09.

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.

Lunn D, Spiegelhalter D, Thomas A, Best N (2009). "The BUGS Project: Evolution, Critique and Future Directions ." *Statistics in Medicine*, **28**, 3049–3082.

Moran P (1950). "Notes on Continuous Stochastic Phenomena." *Biometrika*, **37**, 17–23.

Napier G, Lee D, Robertson C, Lawson A (2019). "A Bayesian space–time model for clustering areal units based on their disease trends." *Biostatistics*, **20**, 681–697.

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.

Pebesma E (2018). "Simple Features for R: Standardized Support for Spatial Vector Data." *The R Journal*, **10**, 439–446.

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/.

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.

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.

Schloerke B, Cook D, Larmarange J, Briatte F, Marbach M, Thoen E, Elberg A, Crowley J (2021). *GGally: Extension to 'ggplot2'*. R package version 2.1.2, URL https://CRAN.R-project.org/package=GGally.

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.

Tanner M, Wong W (1987). "The Calculation of Posterior Distributions by Data Augmentation." *Journal of the American Statistical Association*, **82**, 528–540.

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.

Warnes GR, Bolker B, Lumley T (2018). *gtools: Various R Programming Tools.* R package version 3.8.1, URL https://CRAN.R-project.org/package=gtools.

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 (2016). *ggplot2: Elegant Graphics for Data Analysis.* Springer-Verlag New York. ISBN 978-3-319-24277-4. URL https://ggplot2.tidyverse.org.

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
University Place University of Glasgow
Glasgow
G12 8SQ, Scotland
E-mail: Duncan.Lee@glasgow.ac.uk
URL: http://www.gla.ac.uk/schools/mathematicsstatistics/staff/duncanlee/